On new Classes of Multistage Interconnection Networks

Julio C. D. de Melo† and Roy M. Jenevein‡

RESUMO

Neste trabalho é proposto um modelo generalizado para redes de interconexão multinível que extende as classes de redes existentes, como as SW- e CC-banyans. Partindo de esquemas para identificação de nodos entre estágios adjacentes, são estabelecidas fórmulas de conexão que permitem descrever um número extremamente elevado de novas subclasses de redes, expandindo o leque de topologias alternativas. Para mostrar a variedade e a vantagem de algumas subclasses sobre outras, a distância média entre nodos da rede é calculada para redes de determinado tamanho.

ABSTRACT

In this work a new generalized model for Multistage Interconnection Networks (MINs) is proposed, extending the classes of existing MINs, such as SW- and CC-banyans. By introducing a labelling scheme for nodes in the network, a connection formula is established that makes it possible to describe an enormous number of new subclasses, extending substantially the choice of alternative topologies. To give a flavor of the great variety and advantage of some subclasses, their average distance is computed for networks of given size.

†Departamento de Engenharia Eletrônica
Escola de Engenharia da UFMG
Rua Espírito Santo, 35
30160 - Belo Horizonte, MG

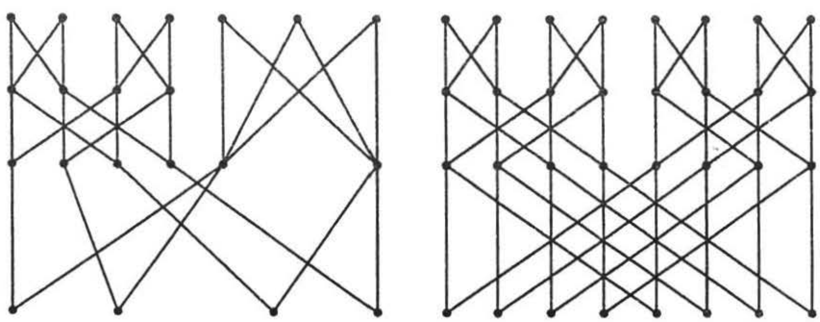
‡Department of Computer Science
The University of Texas at Austin
Austin, TX 78712 - EUA

1 Introduction

The need for ever increasing performance has pushed computers that use the conventional von Neumann architecture to its limits. A natural extension to the uniprocessor architecture is to add one or more processors to share the task of solving a single problem. By doing so, it is anticipated that the time that is taken to solve it in this multiple processor (or multiprocessor) system will be lower than in the single processor system. As expected, though, the task of connecting two or more processors to work in cooperation is not trivial, and many design issues are involved in building such a multiprocessor. Various interconnection networks (ICNs) have been suggested for use in multiprocessors, some within a particular context or with some application in mind. In this work, we will be dealing with the class of networks called *Multistage Interconnection Networks*, or MINs, of which many different subclasses exist. These networks are characterized by the fact that nodes are laid out in stages, within which there are no interconnections, and connections exist only between nodes at adjacent stages.

A number of MINs have been proposed, among which we can cite baseline [15], delta [12], omega [8], indirect-binary cube [13], and flip [1]. These networks have been shown to be topologically equivalent to each other [15], and consequently, results obtained for one can be readily applied to the others. Banyan networks [6, 9, 14] constitute another large class of the many proposed topologies, and reports on their fault-tolerance properties, resource allocation algorithms and performance evaluation can be found in the literature [5, 7, 11].

Conceptually, a *banyan* network is defined in terms of a directed graph representation, which is a Hasse diagram of a partial ordering with a unique path between every base and apex. A *base* is defined as a node of in-degree 0 and an *apex* is defined as a node with out-degree 0. We assume the convention that levels are numbered from base to apex, with level 0 corresponding to the base nodes, and level l corresponding to the apex nodes. A regular banyan is characterized by the fact that all of its nodes (with the exceptions mentioned above) have the same out-degree (called *spread* in banyan terminology) and the same in-degree (called *fanout*). These parameters are represented by the letters s and f, respectively. Two examples of banyan networks are shown in Figure 1. The left network is irregular and is restricted to special applications, such as directly mapping an algorithm into a network. The right network, being regular, is very appropriate for use in interconnection networks, both because data routing algorithms can be easily specified, and because of its excellent properties for data manipulation due to the embedded tree structures.

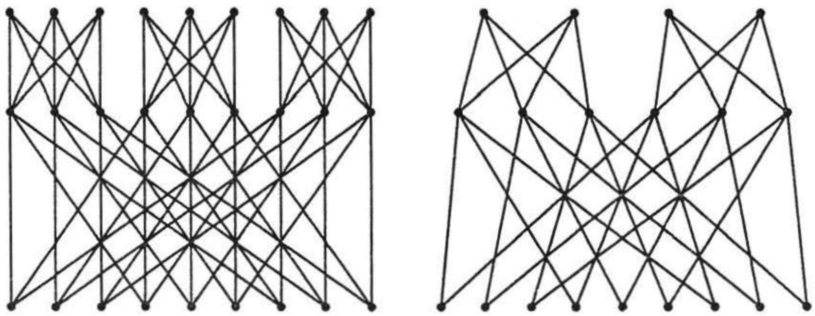


(a) Irregular (b) Regular

Figure 1: Examples of banyan networks.

Regular banyans can be further classified into different networks, according to the connection scheme applied to the nodes. Two of them have been reported in the literature, the SW-banyan and the CC-banyan. The former is defined as a recursive expansion of a crossbar structure (which itself can be thought of as a one-level SW-banyan), by interconnecting s^{l-1} such crossbars and f identical $l-1$ SW structures, all with the same fanout and spread.

SW-banyans can also be further divided into *rectangular* and *non-rectangular* SW-banyans, the only difference being that for the former the spread and fanout are the same. SW-banyans are described by a set of numbers in the format (s, f, l) for non-rectangular banyans, and by (s, l) or (f, l) for rectangular banyans. For $f = 2$, this latter class has been shown to be isomorphic to the networks described at the beginning of this section, and as such, the results presented here can also be applied to those networks. Examples of rectangular and non-rectangular banyan networks are shown in Figure 2.



(a) Rectangular (b) Non-rectangular

Figure 2: Examples of rectangular and non-rectangular banyan networks

The SK-banyan formulation represents a unification and not a fragmentation of the networks issue, because the SW-banyan (and all of its isomorphic counterparts) is a member of the class of SK-banyans. This unification is even more evident when it is proved in [4] that another class of banyan networks, the CC-Banyans, are also shown to be one of the subclasses of SK-Banyans. These two subclasses, which were thought to be unrelated, are unified under a single model, and shown to have common features. Also, significant understanding of the issue of network connectivity and distance properties have been attained by the study of SK-banyans.

The motivation to conduct this work came from the work of Menezes and Jenevein [10] regarding the distance properties of KYKLOS double-tree networks. These results showed that, by changing the connections of the second tree in an appropriate and regular way, the distance properties of this so modified double-tree were shown to be better than for the conventional double-tree. This represented a break from traditional symmetric interconnection schemes of the past which have lead to symmetric redundancy. The connections, while a break from the past trends, are regular and predictable from stage to stage. As the banyan networks have an embedded tree structure, they might also show some improvement by using some modified connection scheme, as will be shown in a latter section. An example of an SK-banyan is shown in Figure 3, in comparison to an SW-banyan.

The emphasis on the study of the distance properties of these new banyan networks is justified by the fact that the delay observed in the transmission of messages across an interconnection network is closely related to their distance properties. This is particularly useful when studying single-sided networks, because base-to-base distance becomes quite important, and improvements in the distance properties are essential for minimizing communication overhead. It is desirable in this case that the distance between these nodes be as low as possible, without aggravating the distance properties of

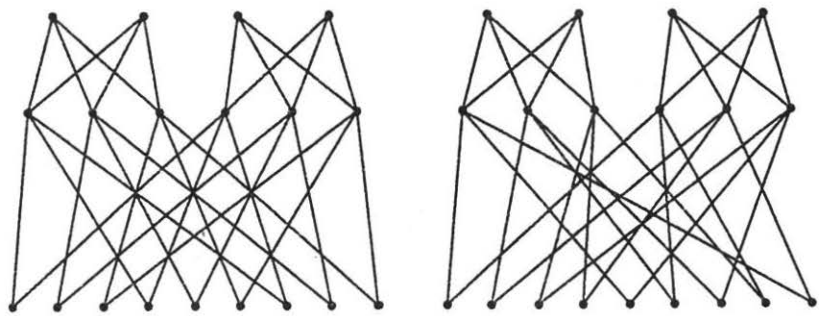


Figure 3: SW-banyan and SK-banyan.

the apex nodes.

2 Definition of SK-Banyans

We start our analysis by reviewing the basic properties of an (s, f, l) regular banyan, which were studied in detail in [14]. This graph is a Hasse diagram of a partial ordering in which the following properties hold:

- **banyan property**: there is one and only one path from any base to any apex;
- **l-level property**: all base-to-apex paths are of the same length l;
- **regularity property**: the indegree of every node, except the bases, is f and the outdegree of every node, except the apexes, is s.

Two basic results can be proved for this graph. First, the number of nodes at each level can be computed from the parameters of the graph. For an (s, f, l) regular banyan, the number of nodes at level i is given by:

$$n_i = s^i f^{l-i} \tag{1}$$

and the number of edges between levels i and $i-1$ is given by:

$$e_i = s^i f^{l-i+1} \tag{2}$$

Second, the nodes can be distinctively numbered within each level, from 0 up to $n_i - 1$. Nothing can be said though, about other properties of the graph as no connection scheme between levels is defined for an (s, f, l) regular banyan. Previous works have defined two connection schemes, leading to two banyan networks: the SW-banyan and the CC-banyan. In this work, we further generalize the connection scheme, and then investigate which networks are generated from this generalization and which relationship they have with the SW- and CC-banyan networks.

2.1 Labelling scheme

The labelling scheme consists of a tuple <*level*, *order*> in which the first number identifies the level in which this node is located, and the second number identifies the order of the node within that level. Normally, as is the case for multistage interconnection networks, the levels are numbered from

0 to l, and the nodes are numbered from 0 up to the maximum number of nodes in the particular level minus one. Typically, the numbering of nodes is done with a number system different from the decimal system to make it easier to specify routing algorithms.

Based on these considerations, we adopt the following labelling scheme for an (s, f, l) regular banyan, illustrated in Figure 4:

Level numbering:

- vertex levels are numbered using a decimal base, from 0 to l, and this number is called the (vertex) *level number*;
- the top vertex level, called the *apex* level, is numbered l;
- the bottom vertex level, called the *base* level, is numbered 0;
- edge levels are numbered like vertex levels, with level i being the edge level between vertex levels $i+1$ and i.

Node numbering:

- at each level, including the apex and the base levels, nodes are numbered from 0 to $n_i - 1$, where n_i represents the number of nodes within that level, as given by Equation 1; this number is called the *order number*;
- each order number is a numeric string composed of two substrings, one of them possibly empty, the rightmost substring in base f, and the leftmost substring in base s; an order number for a node at level $l - i$ is represented, according to this format, as a sequence of digits in the form:

$$(\cdots d_{i+2} d_{i+1} d_i)_s (d_{i-1} d_{i-2} \cdots)_f$$

  where the indexes are defined within the range $[0, l-1]$;
- each node in the graph is identified by a tuple <*level number, order number*>

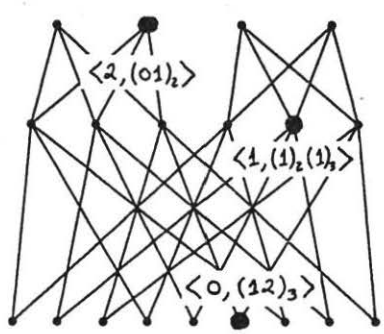


Figure 4: Level and node numbering schemes on an (s, f, l) regular banyan.

2.2 Connection scheme

The connection scheme to be used for SK-banyans will be defined in terms of a *connection formula*, which specifies which nodes at an adjacent, lower level are connected to a particular node at a higher level. This can also be written as a relation R between the two nodes, the relation being that if two nodes a and b are related (aRb), then there is an arc between these two nodes, the node at the upper level (a) being the final vertex, and the node at the lower level (b) being the initial vertex. We will write the relation R as "$\rightarrow$", to mean "*is connected to*". Assuming the node numbering described before, we can describe the connections as pairs of tuples in the form:

$$node < level, order\, number > \;\rightarrow node < level - 1, order\, number > \tag{3}$$

Using the notation given previously for writing the node number as a composition of two numbers in bases s and f, we can write the connection formula above as:

$$\begin{aligned} &node < l-i, (\cdots d_{i+2}d_{i+1}d_i)_s(d_{i-1}d_{i-2}\cdots)_f > \;\rightarrow \\ &\quad node < l-i-1, (\cdots d'_{i+2}d'_{i+1})_s(d'_i d'_{i-1}d'_{i-2}\cdots)_f > \end{aligned} \tag{4}$$

where:

- $l-i$ - level number of the node at the upper level ($0 \leq i \leq l-1$)
- d_i - digits in base s or f of the order number for a node at the upper level
- d'_i - digits in base s or f of the order number for a node at the lower level
- $d'_i = F(d_i)$ for some function F.

2.3 Recursive definition

One way to define a graph is by giving a construction algorithm to obtain it. This method will be adopted here because it leads naturally to a recursive definition, as is the case for an SW-banyan. For SK-banyans, we start with a basic graph, which corresponds to a one-level network, and proceed to define a recursion algorithm to obtain graphs for networks with higher number of levels. This definition is presented in a format that is not directly implementable, but it serves the purpose of defining graphs which are SK-banyans.

Algorithm 2.1

- **Basic step**: *an $(s,f,1)$ SK-banyan is $K_{n,m}$, the complete bipartite directed graph with n and m vertices in each set, and for which $n = s$ and $m = f$.*
- **Recursion step**: *an (s,f,l) SK-banyan is constructed from an $(s,f,l-1)$ SK-banyan by applying the following rules:*

  1. **Multiplicity rule**: *generate s copies of an $(s,f,l-1)$ SK-banyan, numbering them from 0 to $s-1$. Name these graphs the* top graphs.
  2. **Numbering rule**: *renumber every node in copy i of the top graphs by attaching digit i (in base s) to its order number in the most significant position, and by increasing its level number by one.*

3. **Null graph rule:** *generate a null graph of order f^l, and label every node with a number from 0 up to $f^l - 1$ (in base f) and assign them the level number 0. Name this graph the* bottom graph.

4. **Connection rule:** *connect every base node of the top graphs to f nodes of the bottom graph, according to the connection formula defined for this level.*

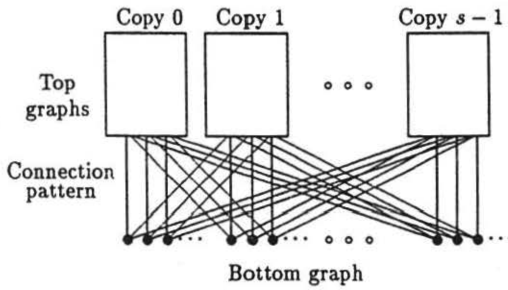


Figure 5: Illustration of the construction algorithm.

This algorithm is illustrated graphically in Figure 5. As can be inferred from the concept of the connection formula, not all of them preserve the banyan property, and it is clear from Algorithm 2.1 that the key to preserving this property is the definition of the connections between the base nodes of the top graphs and the nodes of the bottom graph. This is so because the only step in Algorithm 2.1 where a connection is specified is the connection rule, which is related only to the base edge level. For future reference, we also define what is meant by "upper" and "lower clusters". The set of f^{l-1} base nodes of each one of the s top graphs is called an *upper cluster*, and the set of f^{l-1} nodes of the bottom graph whose node numbers have the same most significant digit is called a *lower cluster*.

2.4 Examples of connection formulas

We now examine some examples of connection formulas, and the graphs that results from them. First, we make some observations. As was defined before, and according to Algorithm 2.1, there are s upper clusters and f lower clusters at the levels that correspond to the top and bottom graphs, and as a consequence, up to $s \times f$ bijections can be defined between them. This can be conveniently represented as an $s \times f$ matrix in which entry $[i, j]$ corresponds to the bijection between an upper cluster from copy i to a lower cluster whose MSD is equal to j. Also, because the cardinality of lower and upper clusters is given by f^{l-1}, the bijections between them will have to be defined between sets of varying cardinality, which will increase with the number of levels. This implies that although we still are dealing with $s \times f$ matrices, the cardinality of its elements will depend on the number of levels.

A second observation regards the relationship between connection formulas for different levels. If they are not related at all, we have a graph whose properties are not readily obtainable, as the irregular connections between levels would be hard to model analytically. On the other hand, if we impose some relationship between the connection formulas for different levels, a regularity may arise from it that makes it possible to study the graph with a tractable model. To avoid extending the subject, we restrict the analysis here to graphs whose connection formulas are the same for all levels. The extension to the more general case is left for future research.

Besides following a common formula we will say that, if the bijections defined between two levels are arbitrarily chosen and bear no relation to the ones defined between other two levels, we have what will be called a *non-uniform* SK-banyan. Accordingly, if the bijections at any level are chosen according to the same procedure, the graph will be called a *uniform* SK-banyan. Again, to avoid extending the subject, we will restrict the analysis here to only the uniform cases. As it will be seen, this already covers most of the interconnection networks being studied currently, as well as a large number of cases that have not been reported before. Examples of non-uniform SK-banyans will be provided later, after we present some examples of uniform SK-banyans. We formally define a uniform SK-banyan to emphasize the cases that will be analyzed in following chapters.

Definition 2.1 *An (s,f,l) uniform SK-banyan is an (s,f,l) SK-banyan for which the same connection formula and the same bijections are used throughout all levels.*

Besides specifying the connection formula, we will also make use of a *connection formula diagram*, which will serve as a visual aid to illustrate the action performed by the connection formula on the digits of a node's order number. In this diagram, the upper part represents the digits of a node at an upper level, and the lower part represents the digits of the nodes at the lower level to which it is connected. Thus, the upper part represents just one node, whereas the lower part represents a range of f nodes. The relation between the digits of the upper and lower level nodes will be represented by a down arrow ("$\downarrow$") to mean exactly the identity bijection, or by a thicker down arrow ("$\Downarrow$") to mean any bijection, including the identity.

A good example to illustrate these points is the SW-banyan. This is a very regular graph which has been studied extensively, and whose properties are well understood. For these reasons, we choose it as the first example in the definition of a connection formula.

SW-banyans

A construction definition for SW-banyans is usually given in terms of a connection rule: there is an arc from a vertex at level i $(0 \leq i < l)$ to a vertex at level $i+1$ if and only if their digit representation differs only at the i^{th} digit position. In our connection formula notation, this would be equivalent to the d's and d''s being different only at digit i. Formally, we can write it in the format shown in Figure 6.

$$node < l-i, (\cdots d_{i+2}d_{i+1}d_i)_s(d_{i-1}d_{i-2}\cdots)_f > \rightarrow$$
$$node < l-i-1, (\cdots d_{i+2}d_{i+1})_s(jd_{i-1}d_{i-2}\cdots)_f >$$

Connection formula diagram

$$\begin{array}{|c|c|c|c|c|c|c|}
\hline
d_{l-1} & \cdots & d_{i+1} & d_i & d_{i-1} & \cdots & d_0 \\
\hline
\multicolumn{1}{c}{\downarrow} & \multicolumn{1}{c}{\downarrow} & \multicolumn{1}{c}{\downarrow} & \multicolumn{1}{c}{} & \multicolumn{1}{c}{\downarrow} & \multicolumn{1}{c}{\downarrow} & \multicolumn{1}{c}{\downarrow} \\
\hline
d_{l-1} & \cdots & d_{i+1} & j & d_{i-1} & \cdots & d_0 \\
\hline
\end{array}$$

$$0 \leq j \leq f-1, 0 \leq i \leq l-1$$

Figure 6: Connection formula for SW-banyans.

There are two observations to make. First, the d''s have been substituted by their images under some bijection, which are in this case the corresponding values for the d's. The function F as defined in Equation 4 is the identity function, except for digit i, whose values span the range $[0, f-1]$. Secondly, as suggested previously, the connection formula is the same regardless of the level, the only difference being on which digit it acts. This digit itself is defined by the edge level, and because only one digit is affected this leads to the regularity of the properties of the SW-banyan, as mentioned before. A $(3,3)$ SW-banyan is shown in Figure 7.

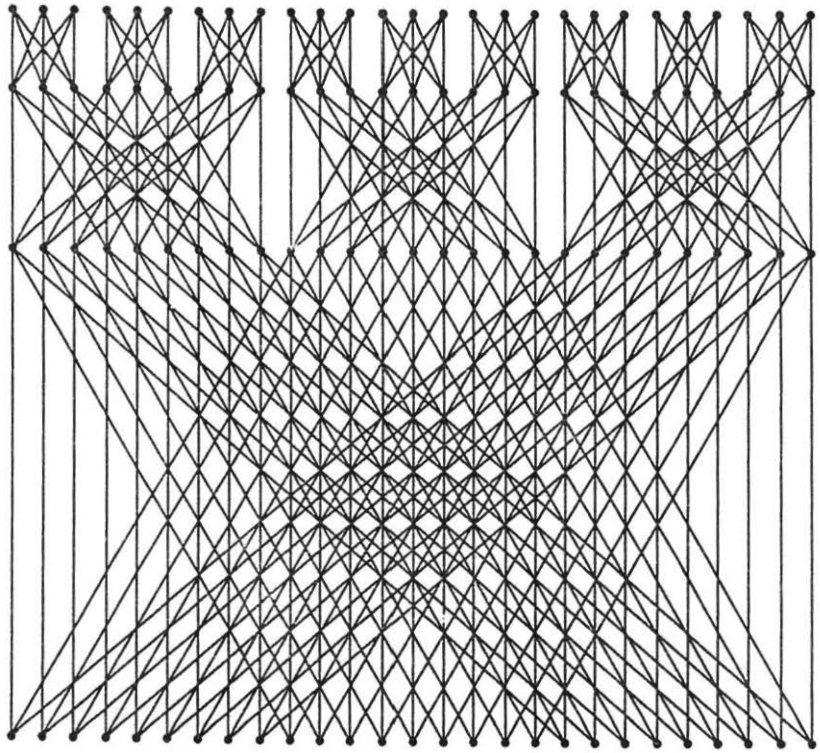


Figure 7: A (3,3) SW-banyan.

SK-banyans

Although SW-banyans are actually one of the subclasses of SK-banyans, they were presented independently in order to demonstrate the unifying concept of SK-banyans. In fact, one connection formula that is fundamental to the concept of SK-banyans is based on the connection formula for SW-banyans. The difference is that, instead of using the identity as the function that relates the d_i''s to the d_i's, we allow any arbitrary bijection, as stated in the recursive algorithm. Following the work provided in [3], we present the case in which just one digit is submitted to a bijection, and we will refer to the graph as a *uniform, single-digit* SK-banyan.

Uniform, single-digit SK-banyans

The basic principle of SK-banyans, as presented in section 1, is to rearrange connections between nodes at two adjacent levels, with the objective of improving the distance properties of the graph by avoiding redundant connections. In the connection formula for SW-banyans this is not the case because nodes that differ in just one digit at the upper level will be connected to the same nodes at the lower level. One way to avoid this is to submit the same digit in the two nodes' order numbers to different bijections. This way, they can be connected to different nodes at the lower level.

An example of a uniform, single-digit SK-banyan is shown in Figure 9.

$$node < l - i, (\cdots d_{i+2}d_{i+1}d_i)_s, (d_{i-1}d_{i-2}\cdots)_f > \rightarrow$$
$$node < l - i - 1, (\cdots d_{i+2}d_{i+1})_s, (jd'_{i-1}d_{i-2}\cdots)_f >$$

Connection formula diagram

d_{l-1}	$\cdots$	d_{i+1}	d_i	d_{i-1}	$\cdots$	d_0
$\downarrow$	$\downarrow$	$\downarrow$		$\Downarrow$	$\downarrow$	$\downarrow$
d_{l-1}	$\cdots$	d_{i+1}	j	d'_{i-1}	$\cdots$	d_0

Figure 8: Connection formula for uniform, single-digit SK-banyans.

CC-banyans

This class of banyan networks has different connection rules than those for SW-banyans. Their name comes from the fact that their underlying graph can be laid out on the surface of a Cylinder as a Crosshatch pattern. They present a richer set of shift-rotate permutations than SW-banyans, as well as better distance properties for the base nodes. Although they are generally defined using an intricate graphical construction algorithm, it was shown in [2] that they can also be constructed recursively, via an algorithm similar to the one used for SK-banyans. In this way, CC-banyans can be thought of as another of the subclasses of SK-banyans, with a different set of connection formulas than those used for uniform SK-banyans. We can write the connection formula for CC-banyans as shown in Figure 10, and the corresponding graph is shown in Figure 11.

2.5 Optimal SK-banyans

Different CGMs will generate graphs with different base-to-base average distances, and hence there should be some criterion upon which one can find an equivalence class of graphs that satisfies a number of properties. In [4], such criterion was defined, and it eliminated the need to actually compute the distance properties of the graphs as, once it was defined, analytical expressions for the distance properties could be used instead. In the case of MINs, two properties are most important: that the average distance be the lowest and that the graph be base-symmetric, that is, that any base nodes have the same distance properties. The networks generated within such conditions were called *optimal SK-banyans*, and they are considered in the next section when comparisons are drawn between them and previous networks (SW- and CC-banyans).

3 Distance Properties of SK-banyans

In the study of an interconnection network, an often used figure for performance evaluation is the computation of its distance properties, usually in terms of average distance. Other measures of importance include finding the *diameter* of the graph or finding a regular graph (of degree d) with maximal order and minimum diameter k. In our case, we will restrict the analysis to the computation of the average distance.

3.1 Average distance

The computation of the average distance properties of a graph involves, in general, computing the *distance matrix* D of the graph, each entry d_{ij} in D corresponding to the distance between nodes i and j. Given the *nxn* adjacency matrix A of a graph, its distance matrix can be computed from it by using one of several algorithms. In the case of SK-banyans, since computing resources will be assumed to be present only in apex and base nodes, we will be interested in computing the distance properties

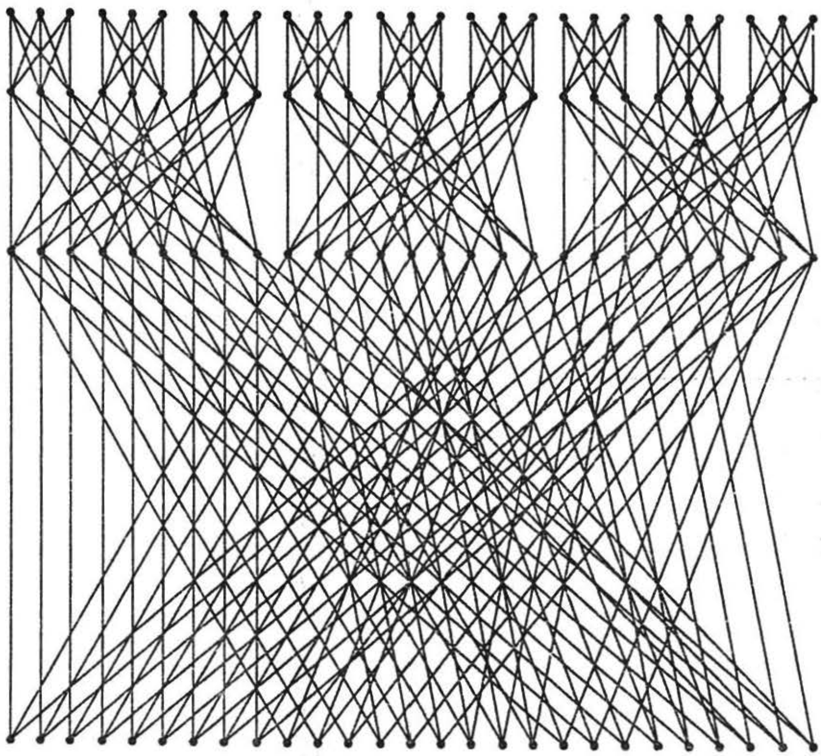


Figure 9: A (3,3) uniform, single-digit SK-banyan.

of these nodes only, and for this reason the corresponding distance matrix will be a submatrix of the distance matrix for the whole graph. From this distance submatrix, the average distance from a node i to the other nodes can be computed from:

$$\bar{d}_i = \frac{\sum_{j=1}^{M} d_{ij}}{M} \tag{5}$$

where M is the cardinality of the subset of nodes involved, and indices i and j refers now to entries from the distance submatrix. If we want to compute the average distance between all nodes to all nodes,[1] we can compute it from:

$$\bar{d} = \frac{\sum_{i=1}^{M} \sum_{j=1}^{M} d_{ij}}{M^2} \tag{6}$$

3.2 Distance Properties of SW-, CC-, and optimal SK-Banyans

The values of the average distance were computed from the equations derived in [4] for these subclasses, and are plotted in Figures 12(a) to 12(d) for each fanout. For $f = 2$, CC- and optimal SK-banyans have the same distance properties, for any number of levels, and the average distance for

[1] Again, we are referring to a subset of nodes.

$$node < l-i, (d_{l-1}\cdots d_{i+1}d_i d_{i-1}d_{i-2}\cdots d_0)_f > \rightarrow$$
$$node < l-i-1, (d_{l-1}\cdots d_{i+1}[d_i+j][d_{i-1}+c_i][d_{i-2}+c_{i-1}]\cdots[d_0+c_1])_f >$$

CC-banyan connection formula diagram

d_{l-1}	$\cdots$	d_{i+1}	d_i	d_{i-1}	d_{i-2}	$\cdots$	d_0
$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$
d_{l-1}	$\cdots$	d_{i+1}	d_i+j	$d_{i-1}+c_i$	$d_{i-2}+c_{i-1}$	$\cdots$	d_0+c_1

$$c_i = \left\lfloor \frac{d_i+j}{f} \right\rfloor$$
$$c_k = \left\lfloor \frac{d_k+c_{k+1}}{f} \right\rfloor$$
$$0 \le j \le f-1, 0 \le i \le l-1, 0 \le k \le i-1$$

Figure 10: Connection formula for a CC-banyan.

SW-banyans is higher than the average distance of both. For higher values of fanout and number of levels, the average distance for CC-banyans approaches that of SW-banyans. This shows that optimal SK-banyans are, with regard to average distance, a more efficient topology for single-sided networks than the other two topologies.

3.3 Distribution of average distance for $(3,3)$ SK-Banyans

As an example of the wide variation of distance properties of SK-banyans with different CGMs, we present here the distribution in the average base-to-base distance of all 10,077,696 possible $(3,3)$ uniform, single-digit SK-banyans, shown in Figure 13. This serves to illustrate some points. First, the equivalence classes under isometrism do not have the same cardinality. This is in sharp contrast with the case for $f = 2$, where only two equivalence classes exist. It can be conjectured that this is the only case for which this is true. For larger values of f ($f > 2$) it is expected that the equivalence classes will not have the same cardinality, as exemplified here for $f = 3$.

Another point worth commenting on is that, in accordance with the expressions derived for the base-to-base average distance, the lowest value in Figure 13 corresponds to an optimal SK-banyan, whereas the highest value corresponds to SW-banyans. The value for CC-banyans is also shown in the figure. As can be seen, the cardinality of isometric topologies for these subclasses is high, and it is expected that many isomorphisms exist for each of these subclasses.

From the results presented in this section, one can appreciate the wide variety of topologies provided by SK-banyans. Besides SW- and CC-banyans, which have been studied extensively, and have a very large range of applications, optimal SK-banyans have superior properties that may be useful in many of these applications. This is true also for non-rectangular SK-banyans, and SK-banyans belonging to other subclasses, specially the multiple-digit case.

4 Conclusions

New subclasses with very distinct and improved properties over the previously known SW- and CC-banyan networks were proposed. By introducing the concept of a connection formula and the use of any bijection to determine the final arrangement of connections in a banyan, we extended the treatment of multistage interconnection networks enormously. This classification further allows to

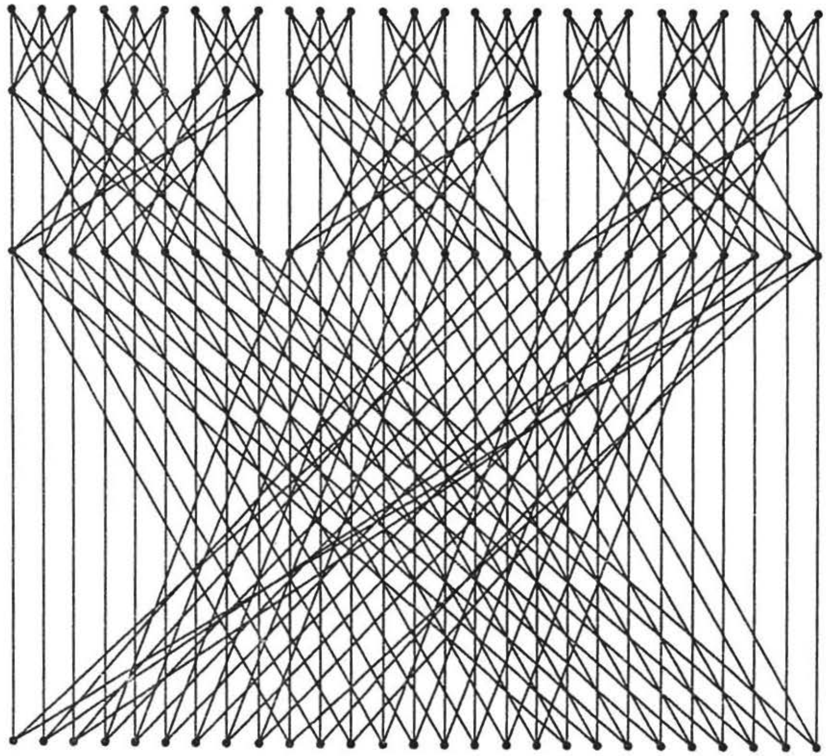


Figure 11: A (3,3) CC-banyan.

define which subclasses are best suited for a given application. The distance properties of optimal SK-banyan networks show that this subclass would have a definitive advantage over other subclasses in single-sided networks.

5 References

References

[1] K. E. Batcher, "The Flip Network in STARAN," *Proceedings of the 1976 International Conference on Parallel Processing*, pp 65-71.

[2] V. Cherkassky and M. Malek, "Analysis of CC-Banyan Networks," *Proceedings of the 1986 International Conference on Parallel Processing*, pp. 115-118.

[3] J. de Melo and R.M. Jenevein, "SK-Banyans: A Unified Class of Banyan Networks," *Proceedings of the 1986 International Conference on Parallel Processing*, pp. 100-107.

[4] Julio C. D. de Melo, *A Unified Theory of Banyan Networks*, Ph.D. Dissertation, University of Texas at Austin, 1989.

[5] M. A. Franklin, "VLSI Performance Comparison of Banyan and Crossbar Communication Networks," *IEEE Transactions on Computers*, Vol. C-30, No. 4, April 1981, pp. 283-290.

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

[7] R. M. Jenevein, D. Degroot and G. J. Lipovski, "A Hardware Support Mechanism for Scheduling Resources in a Parallel Machine Environment," *Proceedings of the 8th. Annual International Symposium on Computer Architecture*, May 1981, pp. 57-66.

[8] D. H. Lawrie, "Access and Alignment of Data in an Array Processor," *IEEE Transactions on Computers*, Vol. C-24, No. 12, December 1975, pp. 1145-1155.

[9] G. J. Lipovski and M. Malek, *Parallel Computing, Theory and Comparisons*, John Wiley & Sons, 1987.

[10] B. L. Menezes and R.M. Jenevein, "KYKLOS : A Linear Growth Fault-tolerant Interconnection Network," *Proceedings of the 14th. International Conference on Parallel Processing*, 1985, pp. 498-502.

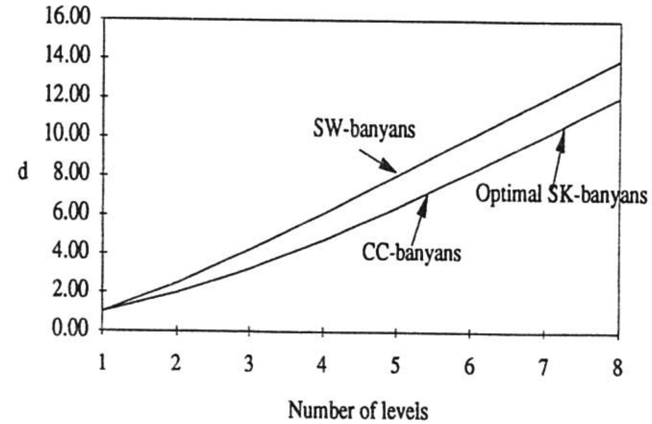
[11] E. Opper, *Fault-Diagnosis of Banyan Networks*, Ph.D. Dissertation, University of Texas at Austin, 1984.

[12] J. H. Patel, "Performance of Processor-Memory Interconnections for Multiprocessors," *IEEE Transactions on Computers*, Vol. C-30, No. 10, October 1980, pp. 771-780.

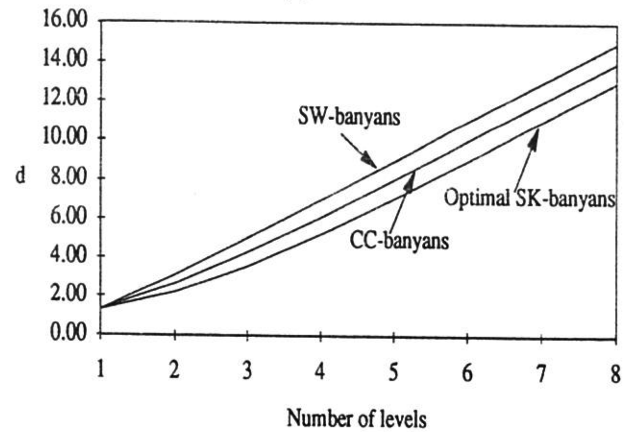
[13] M. C. Pease, "The Indirect Binary n-Cube Microprocessor Array," *IEEE Transactions on Computers*, Vol. C-26, No. 5, May 1977, pp. 458-473.

[14] U. V. Premkumar, *A Theoretical Basis for the Analysis and Partitioning of Regular SW-Banyans*, Ph.D. Dissertation, University of Texas at Austin, August 1981.

[15] C. Wu and T. Feng, "On a Class of Multistage Interconnection Networks," *IEEE Transactions on Computers*, Vol C-29, No. 8, August 1980, pp. 694-702.

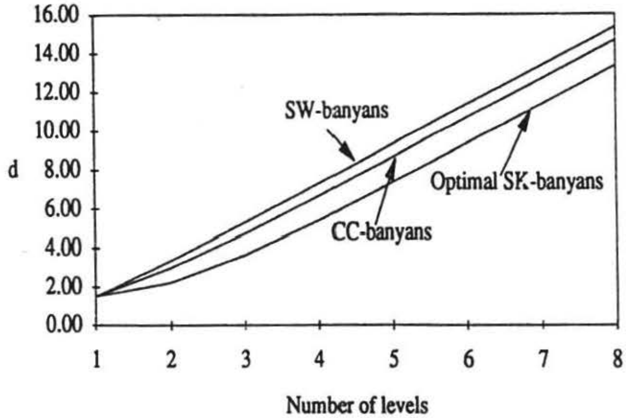


(a) Fanout = 2

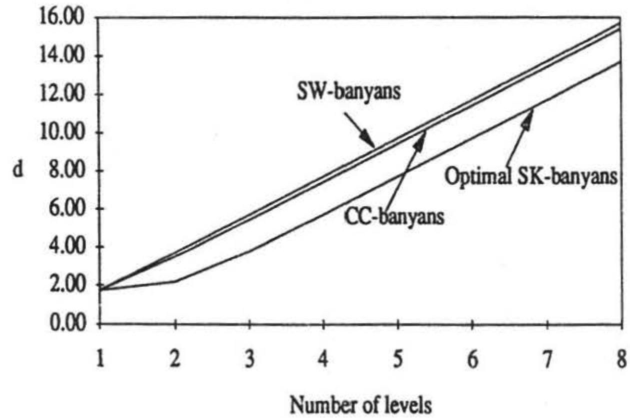


(b) Fanout = 3

Figure 12: Base-to-base average distance for SW-, CC-, and optimal SK-banyans (cont'd.).



(c) Fanout = 4



(d) Fanout = 8

Figure 12: Base-to-base average distance for SW-, CC-, and optimal SK-banyans.

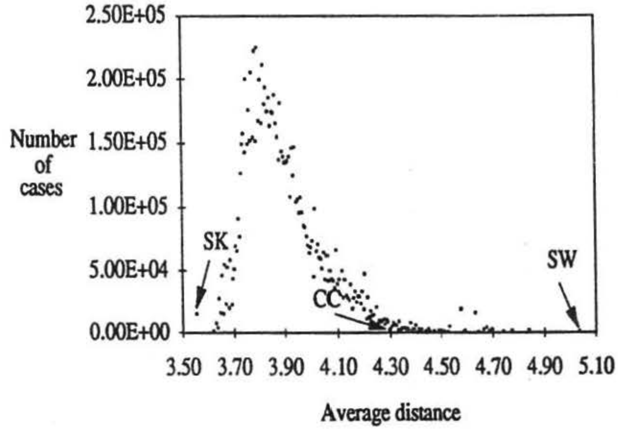


Figure 13: Distribution of average distances for all (3, 3) uniform, single-digit SK-banyans.