*Computers Math. Applic.* Vol. 25, No. 12, pp. 87–98, 1993
Printed in Great Britain. 
0097-4943/93 \$6.00 + 0.00


# TOPOLOGICAL PROPERTIES OF STAR GRAPHS

SUMIT SUR AND PRADIP K. SRIMANI
Department of Computer Science, Colorado State University
Ft. Collins, CO 80523, U.S.A.

*(Received February 1992; revised and accepted July 1992)*

**Abstract**—Our purpose in the present paper is to investigate different topological properties of the recently introduced star graphs which are being viewed as attractive alternatives to $n$-cubes or hypercubes. These properties are interesting by themselves from a graph theory point of view as well as they have direct impact in generating vertex disjoint paths and minimal paths in a star graph. They can also be readily utilized to design routing algorithms and to compute contention and traffic congestion in networks that use star graphs as the underlying topology.

## 1. INTRODUCTION

The underlying topology of any multiple processor system is, in general, modeled as an undirected graph where the nodes denote the processing elements and the arcs (edges) denote the bi-directional communication channels. Design features for an efficient interconnection topology include properties like low degree, regularity, small diameter, high connectivity, efficient routing algorithms, high fault tolerance, low fault diameter, etc. The small diameter helps to keep the interprocessor communication delay low while the low degree of nodes is necessary to limit the number of input-output ports to some acceptable value. Various authors [1–5] have investigated the problem of network design with a view to achieving these goals. It has been observed that regular graphs in general and those with strong algebraic structures play the most important role in network design because of the ease of designing uniform routing algorithms as well as of mapping parallel algorithms on the networks. Until very recently one of the most efficient symmetric interconnection networks has been the well known binary $n$-cubes or the hypercubes; they have been used to design various commercial multiprocessor machines and they have been extensively studied.

Recently, a new interconnection topology, called the star graphs has been reported in the literature [1,6–9]. It is to be noted that these star graphs are a class of Cayley graphs as are $n$-cubes or the pancake graphs [1]. It is also to be noted that these star graphs are different from star graphs of [10]. These star graphs seem to be very attractive alternatives to the $n$-cubes in terms of almost all the desirable properties of an interconnection structure. It has been shown that these star graphs can accommodate more processors with less interconnection hardware and less communication delay (compared to $n$-cubes) and they are optimally fault-tolerant [8]. That is, the star graphs enjoy most of the desirable features of the $n$-cubes at a considerably less cost. Thus, it is of interest to further investigate the properties of these star graphs in order to make them truly viable alternatives for the $n$-cubes.

Our purpose in the present paper is to study various topological properties of these star graphs. These properties are interesting from graph theory point of view as well as they facilitate development of routing algorithms and computation of contention [11] and traffic congestion in the network. To design fault tolerant routing algorithms (with minimum traffic congestion) it is essential to know the vertex disjoint paths as well as all paths between two arbitrary vertices. The properties, developed in this paper, are easily translated into algorithms to compute such paths.

Typeset by $\mathcal{A}\mathcal{M}\mathcal{S}$-TEX

A star graph $S_n$, of dimension $n$, is defined to be a symmetric (undirected) graph $G = (V, E)$ where $V$ is the set of $n!$ vertices, each representing a distinct permutation of $n$ elements and $E$ is the set of symmetric edges such that two permutations (nodes) are connected by an edge iff one can be reached from the other by interchanging its first symbol with any other symbol. For example, in $S_3$, the node representing permutation $ABC$ have edges to two other permutations (nodes) $BAC$ and $CBA$. Throughout our discussion we denote the nodes by permutations of English alphabets. For example, the identity permutation is denoted by $I = (ABCD\ldots Z)$ ($Z$ is the last symbol, not necessarily the $26^{\text{th}}$).

It is easy to see that any permutation of $n$ elements can also be specified in terms of its cycle structure. For example, $CDEBAF = (ACE)(BD)(F)$. We use following notations throughout the paper:

n: Order of the star graph $S_n$.
N: Number of nodes in $S_n$, $N = n!$.
L: Number of edges in $S_n$, $L = (n-1)n!/2$.
$D_n$: Diameter of $S_n$, $D_n = \lfloor 3(n-1) \rfloor$.
c: Number of cycles of length at least 2 in any permutation.
m: Total number of symbols in these $c$ cycles.

It has been shown in [8] that the minimum distance $d(\pi)$ from a given node (permutation) $\pi$ to the identity permutation is given by:

$$d(\pi) = \begin{cases} c + m, & \text{when } A \text{ is the first symbol,} \\ c + m - 2, & \text{when } A \text{ is not the first symbol.} \end{cases}$$

Since star graphs are vertex symmetric [6], $d(\pi)$ will always indicate the distance of $\pi$ from the identity permutation $I$ without any loss of generality.

## 2. IDENTIFICATION OF VERTEX DISJOINT PATHS

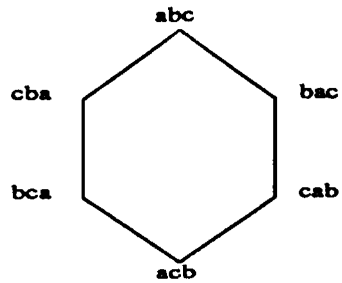
A star graph $S_n$ of order $n$ has vertex connectivity $n-1$ [1] i.e., given any two arbitrary nodes $u$ and $v$, there are $n-1$ vertex disjoint paths from $u$ to $v$. There are many applications like routing of messages with minimum traffic congestion or routing in presence of faulty nodes or links, it is essential to identify these vertex disjoint paths between the source and the destination nodes. It is to be noted that these vertex disjoint paths are not unique. Our purpose in this section is to develop an algorithmic approach to identify one such set of vertex disjoint paths in $S_n$ for any two arbitrary source and destination vertices $s$ and $d$.

A star graph $S_n$ can be partitioned into $n$ mutually disjoint component star subgraphs $S_{n-1}$, where each component is obtained by fixing a particular symbol at a particular position [1]. We use $V_X$ to denote the set of nodes (permutations) that end with the symbol $X$. $V_X$ is a star graph of order $n-1$. For example, in $S_3$ there are 3 orthogonal component subgraphs of order 2, e.g., $V_A = \{BCA, CBA\}$, $V_B = \{ACB, CAB\}$, and $V_C = \{ABC, BAC\}$ (see Figure 1a). Similarly, $V_\alpha$ denotes the set of nodes that end with $\alpha$ where $\alpha$ represents a sequence of symbols. $V_\alpha$ is a star graph of dimension $n - |\alpha|$ if $V_\alpha$ is a subgraph of $V_n$. For example, $V_{YZ}$ denotes the set of vertices that end with $YZ$. We need to define the *gateway points* between the orthogonal components of a star graph.

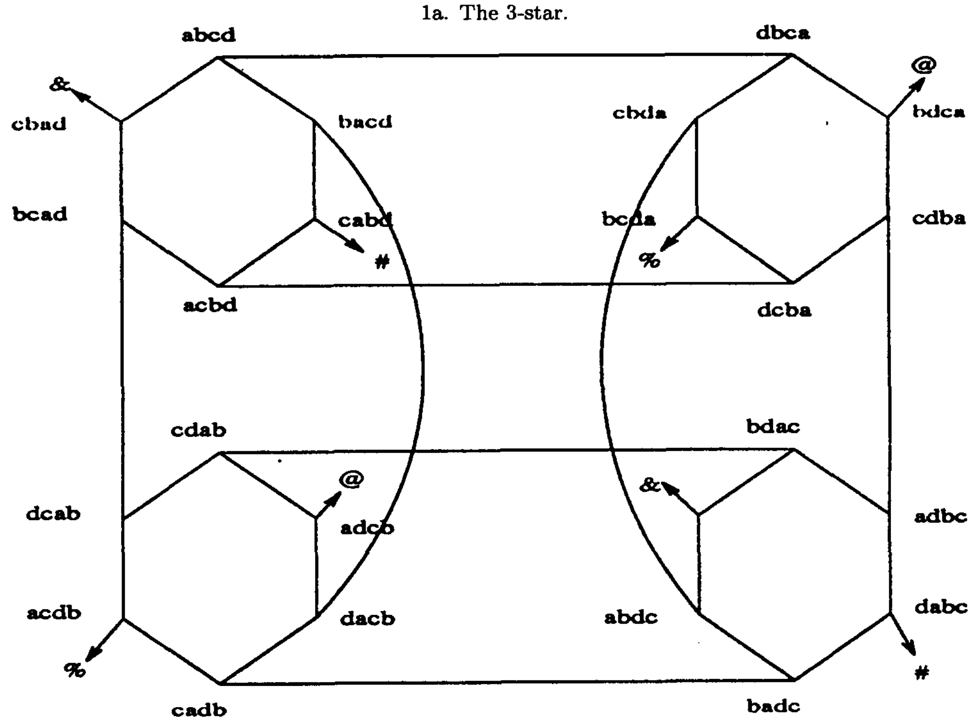
DEFINITION 1. *Consider any two mutually disjoint subgraphs $V_X$ and $V_Y$ of a star graph $S_n$. The set of nodes of $V_X$ that are directly connected to some node of $V_Y$ are called the gateway nodes of $V_X$ with respect to $V_Y$. We denote this set by $G_{X,Y}$. In general, either or both of $X$ and $Y$ may be sequence of symbols instead of single symbol.*

For example, in $S_4$, $G_{B,C} = \{\text{CDAB, CADB}\}$ and $G_{C,B} = \{BDAC, BADC\}$ (see Figure 1b). We can easily observe following properties of the gateway points.

- In $S_n$, any node $x \in G_{X,Y}$ has $X$ as its last symbol and $Y$ as its first symbol and any node $y \in G_{Y,X}$ has $Y$ as its last symbol and $X$ as its first symbol.
- In any star graph $S_n$, $|G_{X,Y}| = |G_{Y,X}| = (n-2)!$, when $X$ and $Y$ are single symbols, and there is a one-to-one and onto mapping from the set $G_{X,Y}$ to the set $G_{Y,X}$.




1a. The 3-star.




1b. The 4-star.

Figure 1. Star graphs of dimensions 3 and 4.

Now consider two arbitrary source and destination nodes $s$ and $d$ in $S_n$. There are two cases: $s$ and $d$ belong to different components $V_X$ and $V_Y$ or both of them belong to the same $V_X$ for some $X$ and $Y$. We consider the case that $s \in V_X$ and $d \in V_Y$, where $X \neq Y$.

LEMMA 1. *In $S_n$, for any node $s \in V_X$, where $X$ and $Y$ are single symbols, ($V_X$ is a star graph of dimension $n-1$), there exists vertex disjoint paths from $s$ to $n-2$ gateway nodes in $G_{X,Y}$ with the following properties.*

- *These $n-2$ gateway points have different symbols in their second positions.*
- *The symbols in all these gateway nodes other than the first, second and last are alphabetically ordered.*

*These vertex disjoint paths are identified by the following algorithm A.*

ALGORITHM A.

Step 1. If $Y$ is not the first symbol in $s$, then interchange the first symbol of $s$ with all other symbols (except the second) in $s$ (except the last symbol $X$) to arrive at $n-2$ neighboring nodes $s_1^1, \cdots, s_{n-2}^1$ else interchange $Y$ (first symbol of $s$) with other symbols in $s$ (except the last symbol $X$) to arrive at $n-3$ neighboring nodes $s_1^1, \cdots, s_{n-3}^1$ and consider $s$ to be the node $s_{n-2}^1$.

Step 2. For all nodes $s_\ell^1$, $1 \leq \ell \leq n-2$ do the following: interchange the first symbol of $s_\ell^1$ with its second symbol to arrive at node $s_\ell^2$ except when (a) the interchange takes the node back to $s$ or (b) second symbol of $s_\ell^1$ is $Y$ (in such cases consider $s_\ell^1$ to be $s_\ell^2$).

Step 3. At the beginning of this step each node $s_\ell^2$, $1 \le \ell \le n-2$, has a different second symbol. Now for each $s_\ell^2$ move along the same constant second symbol to reach the corresponding desired gateway point. To go to the desired gateway point, continue putting the first symbol in its desired position. And when $Y$ becomes the first symbol, interchange it with the smallest symbol (in alphabetic order) not in position.

PROOF. To prove the lemma, we need to prove the correctness of Algorithm $A$. Step 1 is feasible by the definition of the edges in a star graph of dimension $n$. For Step 2, we consider two cases.

(a) $Y$ is not the second symbol of $s$. The nodes $s_1^1, \cdots, s_{n-2}^1$ have the same second symbol as $s$ except one which has the first symbol of $s$ as its second symbol. At the end of Step 2, the nodes $s_1^2, \cdots, s_{n-2}^2$ have symbols in their second position other than the first or second symbols of $s$. In other words, we have vertex disjoint paths from $s$ to the nodes $s_1^2, \cdots, s_{n-2}^2$. Also at the end of Step 2 all these nodes $s_1^2, \cdots, s_{n-2}^2$ have different symbols in their second positions.

(b) $Y$ is the second symbol of $s$. Here, $Y$ is the 2nd symbol of the nodes $s_1^1, \cdots, s_{n-2}^1$, except the first one. In Step 2, $Y$ is interchanged with the first symbols of $s^1$ nodes, which are all different. Hence, nodes $s_1^2, \cdots, s_{n-2}^2$, that are created in Step 2 are different from the previous ones and are themselves different because of their second position symbols. Again, we have vertex disjoint paths from $s$ to the nodes $s_1^2, \cdots, s_{n-2}^2$ and all these nodes $s_1^2, \cdots, s_{n-2}^2$ have different symbols in their second positions. Since in Step 3 further traversal of paths from each node $s_\ell^2$ is obviously vertex disjoint (since the second symbols of the nodes in each path are different), we have identified $n-2$ vertex disjoint paths from $s$ to the gateway points $G_{X,Y}$ with the desired properties.

∎

EXAMPLE. Suppose BCYFEDX is the source node. Step 1 generates the nodes CBYFEDX, YCBFEDX, FCYBEDX, ECYFBDX, DCYFEBX. In Step 2 we leave the two nodes CBYFEDX and YCBFEDX alone, and arrive at CFYBEDX, CEYFBDX and CDYFEBX from the other three by swapping the 1st and 2nd symbol. Now notice that all these nodes, CBYFEDX, YCBFEDX, CFYBEDX, CEYFBDX and CDYFEBX have different symbols in their second positions. Now in Step 3 keeping the 2nd symbol constant one can arrive at the respective gateway points, YBCDEFX, YCBDEFX, YFBCDEX, YEBCDFX and YDBCEFX. Figure 2 illustrates this example.

LEMMA 2. *For any source node, $s \in V_X$ and destination node $d \in V_Y$, $n-2$ vertex disjoint paths can be identified by using the previous lemma twice. From $s$, reach $n-2$ gateway points in $G_{X,Y}$, from $d$, reach the corresponding $n-2$ gateway points in $G_{Y,X}$ and then include the connections between corresponding nodes in $G_{X,Y}$ and $G_{Y,X}$ to obtain $n-2$ vertex disjoint paths.*

PROOF. Obvious, since use of the previous lemma the first time will generate vertex disjoint paths restricted to $V_X$, and the second time it will define vertex disjoint paths restricted to $V_Y$. The connections between the gateway points are direct. Hence, all the $n-2$ paths generated are vertex disjoint. ∎

The star graph $S_n$ has a vertex connectivity $n-1$, i.e., there exist $n-1$ vertex disjoint paths between any two nodes. Lemma 2 identifies $n-2$ of them; we need to identify the $(n-1)^{st}$ vertex disjoint path between $s \in V_X$ and $d \in V_Y$.

REMARK. From the orthogonality of the component subgraphs of a star graph, we can easily observe the following: if $s \notin G_{X,Y}$ and $d \notin G_{Y,X}$, go from $s$ to $s'$ and from $d$ to $d'$ by interchanging the first symbol with the last symbol in the nodes. $s'$ and $d'$ belong to two different orthogonal components (if $s$ and $d$ have the same first symbol, $s'$ and $d'$ are in the same component subgraph different from $V_X$ and $V_Y$). Any connection restricted to the two components (one component in the special case) between $s'$ and $d'$ identifies the $(n-1)^{st}$ vertex disjoint path between $s$ and $d$.

EXAMPLE. Consider a $S_7$ where $s = BCYFEDX$ and $d = DCBEXFY$. To identify the last vertex disjoint path between the source and the destination by the previous lemma, we go from $s$ to $XCYFEDB$ which is in $V_B$ ($s \in V_X$). We go from $d$ to $YCBEXFD$ which is in $V_D$ ($d \in V_Y$). Any path between $XCYFEDB$ and $YCBEXFD$ through $G_{B,D}$ and $G_{D,B}$ identifies the last vertex disjoint path between $s$ and $d$.

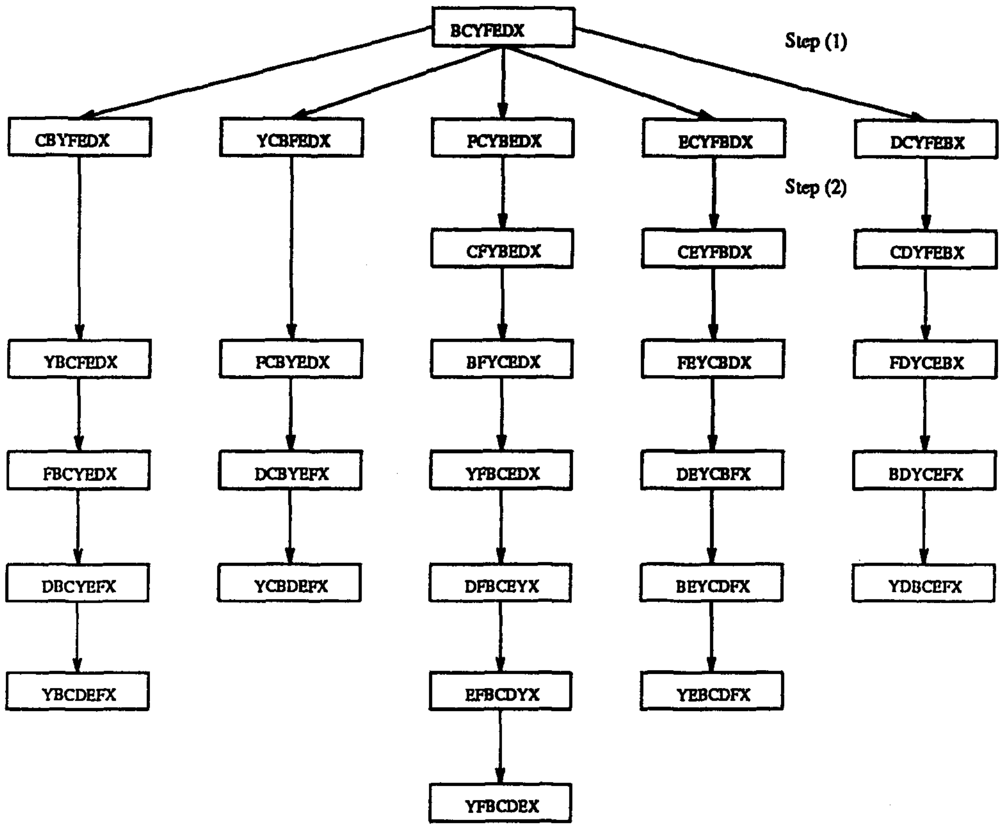



Figure 2. Vertex disjoint paths from source to gateway points.

LEMMA 3. *If $s \in G_{X,Y}$, or $d \in G_{Y,X}$, or both (i.e., either the source or the destination node is a gateway node), go from $s$ to $s'$ and from $d$ to $d'$ by interchanging the first symbol with the second symbol in the nodes. $s'$ and $d'$ belong to two different orthogonal components (if $s$ and $d$ have the same second symbol, $s'$ and $d'$ are in the same component subgraph different from $V_X$ and $V_Y$). Any connection restricted to the two components (one component in the special case) between $s'$ and $d'$ identifies the $(n-1)^{\text{st}}$ vertex disjoint path between $s$ and $d$.*

PROOF. Note that, if $s \in G_{X,Y}$, $Y$ is the first symbol and $X$ is the last symbol in $s$. The node created by interchanging $Y$ with the $2^{\text{nd}}$ symbol (say $Z$) is not visited in Step 1 of Algorithm $A$ to compute the first $(n-2)$ paths. Interchanging $Z$ with $X$ will produce a node in $V_Z$. Rest is obvious from the disjointness of the components. ∎

EXAMPLE. Let $YFBCDEX$ be the source node $s$; then interchange $Y$ and $F$ to go to the node $FYBCDEX$ and then go to $XYBCDEF$ (by using the last remark) which is in $V_F$. Same can be done for the destination node if needed.

REMARK. So far we have assumed that the source and destination nodes belong to different orthogonal components. When they belong to the same component, the above procedures to identify vertex disjoint paths need be modified. We can easily tackle this problem by making different orthogonal decomposition of the graph so that the source and the destination nodes are kept in different orthogonal components. For example, instead of keeping the last symbol fixed we can keep any other symbol, say the third symbol fixed, to get a different orthogonal decomposition. If, for example, $BCYFEDX$ is the source node and $CDYFDEX$ is the destination node, then consider orthogonal decomposition based on the $6^{\text{th}}$ position which implies BCYFEDX $\in V_D$ and CDYFDEX $\in V_E$. Now the previous lemmas are applicable to identify $n-1$ vertex disjoint paths.

## 3. NUMBER OF OPTIMAL PATHS

In a star graph $S_n$, given any two vertices, there are $n-1$ vertex disjoint paths between them. This set of vertex disjoint paths is not unique and we have outlined an algorithm to compute one such set for two given arbitrary vertices. It is to be noted that these vertex disjoint paths are not optimal (minimal) and in general they are of different lengths. Knowledge of vertex disjoint paths plays an important role in routing messages between source and destination when some nodes in the network are faulty or malfunctioning and the number of vertex disjoint paths gives the measure of robustness of the graphs under multiple failures of arbitrary nodes.

On the other hand, there are situations where we are interested to compute the shortest path between two arbitrary nodes in a network graph for message routing in a fault free situation. We are also interested to know about all the alternate optimal paths between nodes in order to bypass a heavily crowded node (to avoid traffic congestion delays) or a malfunctioning node. Optimal routing algorithm for arbitrary node pairs is known for star graphs [1]. In general, there are many different optimal paths between two arbitrary nodes in a star graph (not all of them are necessarily vertex disjoint). In this section, we are interested to compute the total number of optimal paths between two arbitrary nodes in a star graph. The analysis reveals many interesting topological and algebraic properties of the star graph as well as it provides a new methodology to study algebraic properties of network graphs.

Because of the vertex symmetry of the star graphs [1], any node can be mapped to the identity node $I$ by renaming the symbols. This renaming defines a specific mapping that is then to be applied to all other nodes. For example, let the source node be $BDECA$ and the destination node be $ECDAB$. Then to map the destination node to the identity node ($I = ABCDE$), we need the following renaming of symbols:

$$E \rightarrow A, \quad C \rightarrow B, \quad D \rightarrow C, \quad A \rightarrow D, \quad B \rightarrow E$$

which maps the source node into $ECABD$. Now, all the paths between the two original nodes are isomorphic to those between $ECABD$ and the identity node in the renamed graph. So, without loss of generality, we assume that the destination node is always the identity node.

Our aim is to count the total number of optimal paths between the identity node and any node at a maximum distance (diameter of the graph) from it in a given star graph. The reason we consider the maximum distance node is that once those optimal paths are identified, optimal paths between any other nodes can be shown to be subpaths of the earlier optimal paths. This number of optimal paths very much depends on the cycle structure of the source node. We characterize the cycle structure of nodes at a maximum distance from the identity node by the next two lemmas.

LEMMA 4. *For odd $n$, a node (permutation) $\pi$ in $S_n$ is at the maximum distance $\lfloor 3(n-1)/2 \rfloor$ from $I$, iff its cycle structure satisfies the following:*

(i) *$A$ is the first symbol in $\pi$ (i.e. $A$ forms a 1-cycle)*
(ii) *All other cycles are 2-cycles*

PROOF. If $A$ is the first symbol in $\pi$, $d(\pi)$ is given by $c+m$. If conditions (i) and (ii) hold for $\pi$, then $c = (n-1)/2$ and $m = n-1$ and hence, $d(\pi) = 3(n-1)/2$, which is the diameter for $S_n$. To prove the only if part, note that if A is in the first position of $\pi$, then $c+m$ is maximized only when all symbols in $\pi$ other then $A$ form 2-cycles. If $A$ is not in the first position of $\pi$, then maximum possible value of $m$ is $n$ and that of $c$ is still $(n-1)/2$, but $d(\pi)$ is then given by $c+m-2$. Hence, $\pi$ cannot be at a maximum distance from $I$. ∎

For example, the node $AFEGCBD$ in $S_7$ has maximum distance of 9 ($D_7 = \lfloor 3(7-1)/2 \rfloor$. Note that $AFEGCBD = (A)(BF)(CE)(DG)$ where, $c = 3$ and $m = 6$.

LEMMA 5. *For even $n$, any permutation $\pi$ in $S_n$ has the maximum distance $D_n$ iff its cycle structure satisfies one of the following two:*

(i) *$A$ is the first symbol in $\pi$ (i.e., $A$ forms a 1-cycle), three other symbols form a 3-cycle and the rest form 2-cycles,*
(ii) *$A$ is not in the first position in $\pi$, and all cycles are of length 2 ($n/2$ of them).*

PROOF. For even $n$, $D_n$ is given by $\lfloor 3(n-1)/2 \rfloor = (3n-4)/2$. When $A$ is in the first position of $\pi$, maximum value of $m$ is $n-1$ and maximum value of $c$ is $(n-2)/2$ and hence, $d(\pi)$ is $(3n-4)/2$. Obviously this maximum value for $c$ can be achieved only when the symbols other than $A$ form one 3-cycle and the rest 2-cycles. Similarly, when $A$ is not in the first position of $\pi$, maximum possible value of $m$ is $n$ and that of $c$ is $n/2$ and this can be achieved only when the symbols all form 2-cycles. ∎

For example, if $\pi = ACDBFE = (A)(BCD)(EF)$, then $c = 2$, $m = 5$, and hence $d(\pi) = 7 = D_6$. Similarly for $\pi = CDABFE$, whose cycle structure is $(AC)(BD)(EF)$, $c = 2$, $m = 6$, and hence $d(\pi) = c + m - 2 = 7$.

In a star graph, the adjacent nodes of a given node $u$ are determined by interchanging the first symbol in $u$ with any other symbol in the permutation. It has been shown in [1,7] that an optimal path from an arbitrary node $u$ to the identity permutation can be traced by using two simple rules: if the first symbol of the node is not A, do an interchange to place it in its proper position; else interchange $A$ with some symbol which is yet to be in its proper position. It is to be noted that in computing the optimal path the first symbol $X$ (when not $A$) can also be interchanged with any other symbol $Y$ when $X$ and $Y$ don't belong to the same cycle and $Y$ is not in its proper position. It has also been shown in [1] that these rules exhaustively enumerate all possible optimal paths. The possibility tree enumerates all possible of optimal paths from a source node $u$ to the destination node $I$. We define this formally as follows.

DEFINITION 2. *The* **Possibility Tree** $PT_u$ *of any node* $u$ *in a star graph is defined to be a tree rooted at* $u$ *where each path from the root to a leaf node is an optimal path from* $u$. *The immediate successors of any node* $v$ *in this tree are those that can be reached from* $v$ *by the optimal routing algorithm (note that the nodes in this tree may not be distinct; we view this graph as a tree for convenience of later descriptions). The leaf nodes are the ones with only one cycle of length greater than 1.*

REMARKS.

- The possibility tree of the node $ACDBFE$ in $S_6$ is shown in the Figure 3. For each node we have shown the permutation as well as the cycle structure (omitting the 1-cycles as usual).
- We considered the nodes with only one cycle of length greater than 1 as the leaf nodes, since once such a node is encountered there is a unique optimal route from this node to the identity node.

DEFINITION 3. *A* **type I** *node (permutation) is one whose first symbol is* $A$ *(i.e.,* $A$ *forms a 1-cycle) and all other cycles are 2-cycles or 1-cycles.*

DEFINITION 4. *A* **type II** *node (permutation) is one whose first symbol is not* $A$ *and all cycles excepting the one containing* $A$ *are 2-cycles or 1-cycles (the cycle containing* $A$ *can have any length* $\geq 2$*).*

DEFINITION 5. *A* **type III** *node (permutation) is one whose cycle structure consists of one 1-cycle (containing* $A$*), one 3-cycle and zero or more 2-cycles and 1-cycles.*

DEFINITION 6. *A* **type IV** *node (permutation) is one whose cycle structure consists of a* $m$*-cycle* ($m \geq 2$) *containing* $A$*, one 3-cycle and zero or more 2-cycles and 1-cycles.*

EXAMPLE. $ACBED = (A)(BC)(DE)$ and $ACBDE = (A)(BC)(D)(E)$ are type I nodes; $BCAED = (ABC)(DE)$ and $DCBAE = (AD)(BC)(E)$ are type II nodes; $ACDBFE = (A)(BCD)(EF)$ is a type III node; $BADECF = (AB)(CDE)(F)$ and $BFDECA = (ABF)(CDE)$ are type IV nodes.

The number of optimal paths depends on the cycle structure of the source node and we found in Lemmas 5 and 6 that the cycle structure of the source node (at a maximum distance from the identity node) is different for odd and even $n$. So, we consider star graphs $S_n$ for odd and even $n$ separately.

### *3.1. Star Graph $S_n$ with odd $n$*

When $n$ is odd, first we make the following three interesting observations:

- The arbitrary source node $u$, which is at a maximum distance from the identity node $I$, is always a type I node (see Lemma 8).

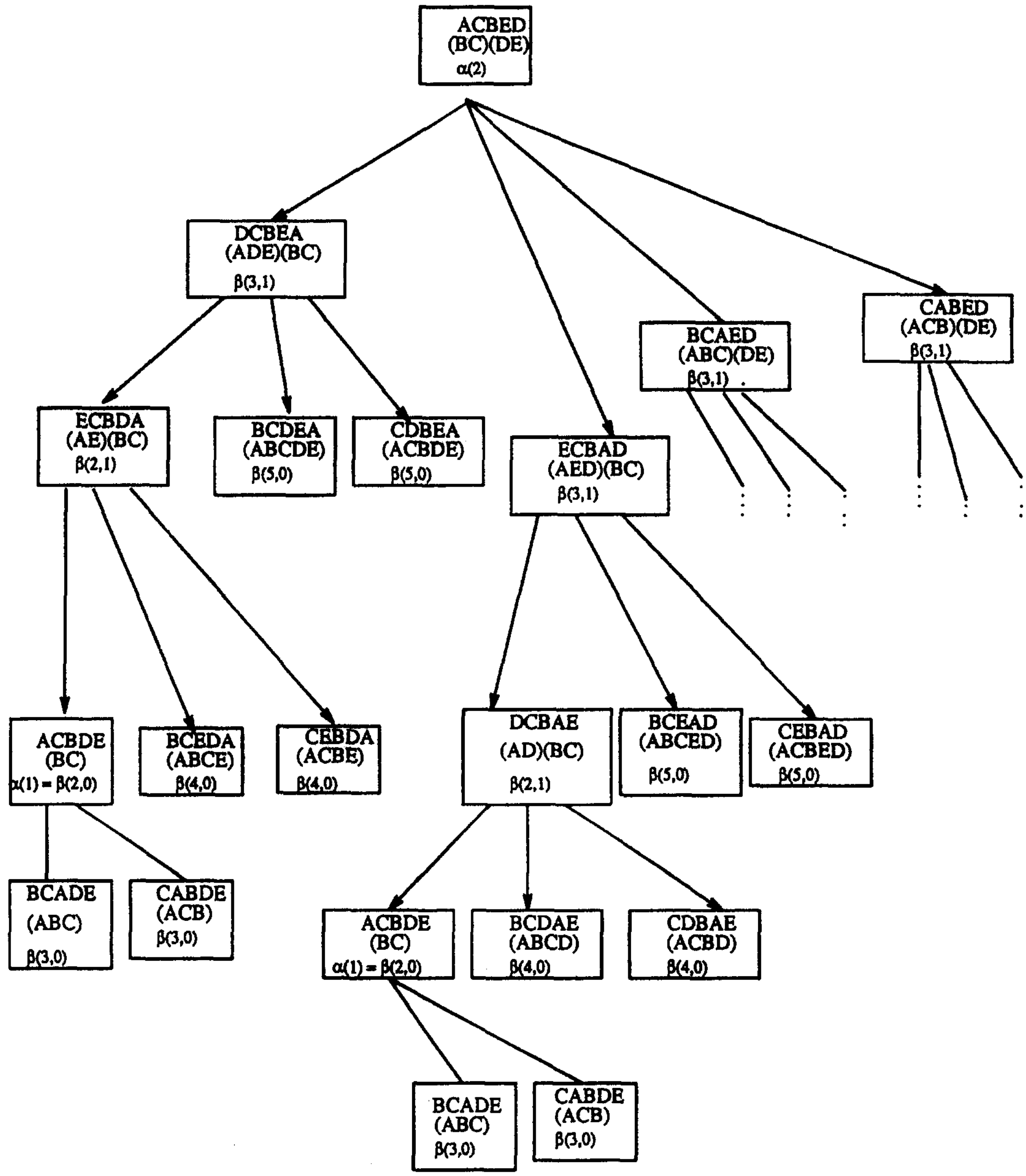



Figure 3. Possibility tree of the node $ACBED$.

- The possibility tree $PT_u$ consists of only type I and type II nodes. This follows directly from the nature of the optimal routing principles as described earlier.
- The structure of the possibility tree does not depend on the actual permutation of the root node, but is fully determined by its cycle structure, i.e., two different nodes with identical cycle structure have identical possibility trees.

We define two functions $\alpha$ and $\beta$ to essentially capture the information of number of optimal paths in a possibility tree rooted at a type I or a type II node. We use $\alpha(i)$ to denote the number of optimal paths in a tree, rooted at a type I node (A is the first symbol) with $i$ 2-cycles. We use $\beta(m, i)$ to denote the number of optimal paths in a tree, rooted at a type II node with $i$ 2-cycles and one more cycle of length $m$.

REMARKS.

- It is to be noted that $\alpha(i) = \beta(1, i)$ for all $i$ since when the length of the $m$-cycle in a type II node is one ($m = 1$), it is also a type I node.

- A leaf node is a type II node which has only one cycle of length more than 1. So, $\beta(m,0) = 1$, for all $m$, since there is only one possible optimal path from any node with a single cycle of length greater than 1.
- $\alpha(1) = 2$ since there are only two ways to optimally go to the destination from a type I node with only one 2-cycle.

EXAMPLE. From Figure 3, we see that $\beta(3,0) = 1$, $\beta(2,1) = 4$, $\beta(3,1) = 6$, and $\alpha(2) = \beta(1,2) = 24$ as few examples.
We need some mechanism to compute the values of $\alpha(i)$ and $\beta(m,i)$ for all $i$ and $m$. The following lemma establishes the relationship between the $\alpha$'s and the $\beta$'s.

LEMMA 6.

$$\alpha(i) = 2i\beta(3, i-1).$$

PROOF. By definition $\alpha(i)$ is the number of optimal paths in the possibility tree rooted at a node that has $A$ in the first position and $i$ 2-cycles. In order to generate optimal paths, the first symbol $A$ must interchange with any element from the 2-cycles, which forces all the immediate successor nodes to be type II nodes with 3 elements in the first cycle and $i-1$ 2 cycles. There can be exactly $2i$ such immediate successors since there are $i$ 2-cycles. Number of optimal paths generated from such a type II node is by definition $\beta(3, i-1)$. Hence, the total number of optimal paths from a type I node with $i$ 2-cycles is $2i\beta(3, i-1)$. ∎

EXAMPLE. We can verify from Figure 3 that $\alpha(2) = 2.2.\beta(3,1) = 4.6 = 24$.
We can compute $\alpha(i)$'s, iff we know the appropriate $\beta(m,i)$'s. We have the following recurrence for the function $\beta$.

LEMMA 7.

$$\beta(m,i) = \beta(m-1,i) + 2i\beta(m+2,i-1).$$

PROOF. This follows from the definition of a type II node which has $m$ elements in the first cycle and $i$ 2-cycles. To generate an optimal path from a type II node, there are two options.

(1) The first symbol is put in its correct position; the successor node is then a type II node with an $(m-1)$-cycle and $i$ 2-cycles and the number of optimal paths from such a type II node is by definition $\beta(m-1,i)$.
(2) The first symbol is interchanged with any symbol in the 2-cycles; this can be done in $2i$ possible ways since there are that many symbols in $i$ 2-cycles and in each case the successor is a type II node with a $(m+2)$-cycle and $i-1$ 2-cycles (such a type II node has $\beta(m+2, i-1)$ optimal paths. Adding up for all the possibilities proves the lemma.

∎

Now we have the recurrence relations to compute $\alpha$ and $\beta$; we still need appropriate initial values for $\beta$'s. The following lemma is useful.

LEMMA 8.

$$\beta(m,1) = 2m.$$

PROOF. By the previous lemma $\beta(m,1) = \beta(m-1,1) + 2\beta(m+2,0)$. By our previous observation $\beta(k,0) = 1$ for all $k$. So we get the one-dimensional recurrence relation $\beta(m,1) = \beta(m-1,1) + 2$. Solving this one gets $\beta(m,1) = \beta(1,1) + 2(m-1)$. But $\beta(1,1) = \alpha(1) = 2$. Hence, the proof. ∎

EXAMPLE. Using above lemmas repeatedly one can evaluate $\alpha(3) = 2.3.\beta(3,2) = 6[\beta(2,2) + 2.2.\beta(5,1)] = 576$.

REMARKS. To compute the values of $\alpha(i)$ for any $i$, a bottom-up approach seems effective. To compute $\alpha(k)$, compute $\beta(m,1)$, $\beta(m,2)$ etc. up to $\beta(m,k-1)$, in that order for all $m$ ranging from 1 to $2k$. Knowing these values $\alpha(k)$ can be easily computed. Some typical values of $\alpha(i)$, number of optimal paths in star graph $S_n$ for odd $n$ are given in Table 1.

Table 1.

| Number of optimal paths | | | | |
|---|---|---|---|---|
| $i =$ | 1 | 2 | 3 | 4 |
| $\alpha(i)$ | 2 | 24 | 576 | 21120 |
| $\gamma(i)$ | 4 | 56 | 1440 | 54912 |
| $\eta(i)$ | 3 | 42 | 1080 | 41184 |

### *3.2. Star Graph $S_n$ with even n*

Again, we assume that the destination is the identity node and the source node is at a maximum distance from the destination. Since $n$ is even, we can see from Lemma 9 that the source node must have one of the following two cycle structures:

(A) $A$ is not the first symbol and all cycles are 2-cycles.
(B) $A$ is the first symbol, one 3-cycle and all other cycles are 2-cycles.

Computation of number of optimal paths is different in these two cases and we treat them separately as follows.

CASE (A). In this case, the source node $u$ is a type II node with a first cycle of length 2, i.e., $m = 2$ and with say, $i$ other 2-cycles, where $i$ is 0 or greater.

We use $\gamma(i)$ to denote the number of paths in the possibility tree rooted at a type II node (A not in the first position) with a first cycle of length 2 ($m = 2$) and with $i$ other 2-cycles.

REMARK. It directly follows from the definition that $\gamma(i) = \beta(2,i)$. For example, if the source node has three 2-cycles ($A$ is not in the first position and $n = 6$), $\gamma(2) = \beta(2,2) = \beta(1,2) + 2.2.\beta(4,1) = \alpha(2) + 4.(2.4) = 24 + 32 = 56$, i.e., there are 56 different optimal paths from the source to the destination. Typical values of $\gamma(i)$ for different values of $i$ are given in Table 1.

CASE B. In this case, the source node $u$ is a type III node that has $A$ in the first position, one 3-cycle and zero or more 2-cycles. Again it can be easily checked that the possibility tree $PT_u$ consists of only type I, II, III and IV nodes. An example possibility tree for a type III node $ACDBFE$ in $S_6$ is shown in the Figure 4. We need to define two more functions $\eta$ and $\delta$ in the same flavor of $\alpha$ and $\beta$ as we defined before.

We use $\eta(i)$ to denote the number of optimal paths in a possibility tree, rooted at a type III node with $i-1$ 2-cycles, one 3-cycle and $A$ in the first position. We use $\delta(m,i)$ to denote the number of optimal paths in a possibility tree, rooted at a type IV node with $i-1$ 2-cycles, one 3-cycle and one cycle containing $A$ of length $m$.

REMARKS.

- $\eta(1) = 3$, by definition, since $A$ is in the first position of the permutation and there is only one 3-cycle, $A$ can interchange with any one of the three symbols in the 3-cycle and hence there are exactly 3 optimal paths to the destination.
- $\eta(i) = \delta(1,i)$, for all $i \geq 1$, since when the length of the m-cycle in a type IV node is one ($m = 1$), it is also a type III node.
- We can see from Figure 4 that $\eta(1) = 3$, $\delta(2,1) = 6$, $\delta(3,1) = 9$, and $\eta(2) = \delta(1,2) = 42$ as few examples.

Again we need some mechanism to compute the values of the functions $\eta$ and $\delta$.

LEMMA 9.
$$\eta(i) = 2(i-1)\delta(3,i-1) + 3\beta(4,i-1).$$

PROOF. $\eta(i)$ is the number of optimal paths in the possibility tree rooted at a type III node that has $A$ in the first position, one 3-cycle and $i-1$ 2-cycles. In order to generate optimal paths, the first symbol $A$ must interchange with either (a) any of the three elements in the 3-cycle or (b) any of the 2(i-1) elements in the 2-cycles. In case (a), each successor is a type II node with a 4-cycle ($m = 4$) and $(i-1)$ 2-cycles; so, this case generates $3\beta(4,i-1)$ optimal paths. In case (b), each successor is a type IV node with a m-cycle ($m = 3$), one 3-cycle and $i-2$ 2 cycles; this

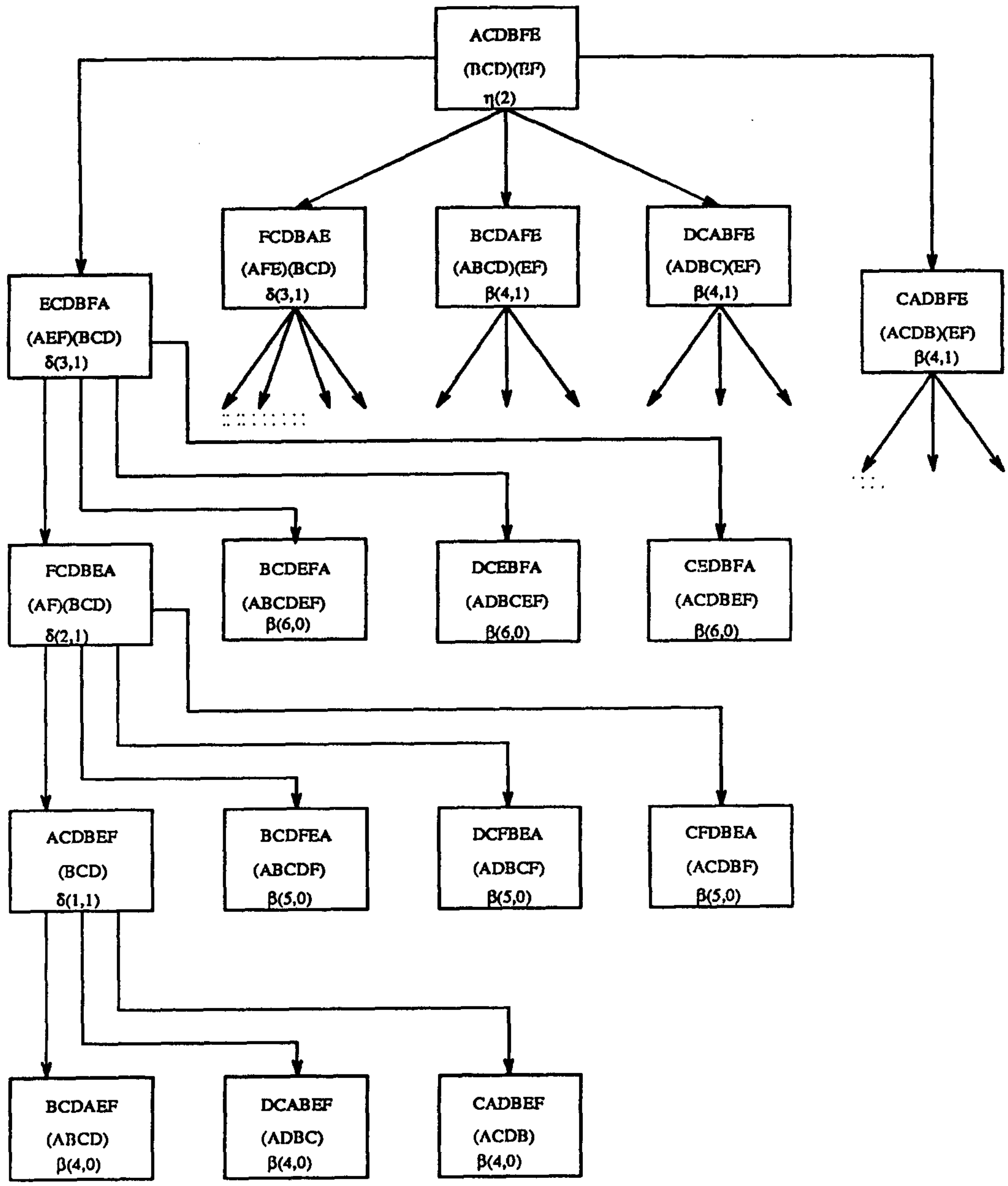



Figure 4. Possibility tree of the node $ACDBFE$.

case generates $2(i-1)\delta(3, i-1)$ optimal paths. Summing up the possibilities in two cases proves the lemma. ∎

EXAMPLE. $\eta(2)$ is given by $2.1.\delta(3,1) + 3.\beta(4,1)$, which is 42.

LEMMA 10.

$$\delta(m,i) = \delta(m-1,i) + 2(i-1)\delta(m+2,i-1) + 3\beta(m+3,i-1).$$

PROOF. $\delta(m,i)$ is the number of optimal paths in the possibility tree rooted at a type IV nodes which has $m$ elements in the first cycle, one 3-cycle and $i-1$ 2-cycles. To generate an optimal path, there are three possibilities.

(1) The first symbol is put in its right position; this can be done in only one way. The successor node is a type IV node with a $(m-1)$-cycle. The number of optimal paths from such a type IV node is given by $\delta(m-1,i)$.

(2) The first symbol is interchanged with any of the $2(i-1)$ symbols in the 2-cycles. Each successor node is a type IV node with a $(m+2)$-cycle, one 3-cycle and $i-2$ 2-cycles. The number of optimal paths from such a type IV node is given by $\delta(m+2, i-1)$.
(3) The first symbol is interchanged with any one of the three elements in the 3-cycle. Each successor node is a type II node with one $(m+3)$-cycle and $i-1$ 2-cycles. The number of optimal paths from such a type II node is given by $\beta(m+3, i-1)$. Summing up the number of optimal paths in all the three cases proves the lemma.

∎

Again, we have obtained the recurrence relations; we still need to know some initial values. The following lemma does just that.

LEMMA 11.

$$\delta(m, 1) = 3m.$$

PROOF. Similar to proof of Lemma 9. ∎

REMARK. To evaluate $\eta(i)$'s, similar bottom up approach described in the remark for odd $n$ can be followed. A table of values of number of optimal paths for this case is given in Table 1.

## 4. CONCLUSION

We have proposed an algorithm to identify all the $n-1$ vertex disjoint paths between two arbitrary nodes in a Star-graph $S_n$. In the second part, we have computed the total number of optimal paths between two arbitrary nodes; in the process we have developed interesting recurrence relations and studied various algebraic properties of star graphs. These properties are interesting from graph theoretical point of view as well as they can be useful in designing computer networks using the star graph topology.

## REFERENCES


1. S.B. Akers and B. Krishnamurthy, A group-theoretic model for symmetric interconnection networks, *IEEE Trans. Comput.* **38**, 555–565 (April 1989).
2. R. Finkel and M.H. Soloman, The lens interconnection strategy, *IEEE Trans. Comput.* **C-30**, 960–965 (December 1981).
3. K. Hwang and J. Ghosh, Hypernet: A communication-efficient architecture for constructing massively parallel computers, *IEEE Trans. Comput.* **C-36**, 1450–1466 (December 1987).
4. M. Imase and M. Itoh, A design method for directed graphs with minimum diameter, *IEEE Trans. Comput.* **C-32** (September 1983).
5. D.K. Pradhan, Fault-tolerant multiprocessor link and bus network architectures, *IEEE Trans. Comput.* **C-34**, 33–45 (1985).
6. S.B. Akers and B. Krishnamurthy, A group theoretic model for symmetric interconnection networks, In *Proceedings of Intl. Conf. on Parallel Processing*, pp. 213–216, (1986).
7. S.B. Akers and B. Krishnamurthy, The fault tolerance of star graphs, In *Proceedings of 2nd Intl. Conf. on Supercomputing*, (1987).
8. S.B. Akers, D. Harel and B. Krishnamurthy, The star graph: An attractive alternative to $n$-cube, In *Proceedings of Intl. Conf. on Parallel Processing*, p. 393–400, (1987).
9. M.S. Krishnamurthy and B. Krishnamurthy, The fault diameter of interconnection networks, *Journal of Comput. Math. Appl.* **13**, 577–582 (1987).
10. F. Harary, *Graph Theory*, Addison-Wesley, Reading, MA, (1972).
11. D.M. Pase, Contention and the star graph as a network topology, Technical Report No. CS/E 88-023, Oregon Graduate Center, (1988).
12. Y. Saad and M.H. Schultz, Topological properties of hypercubes, *IEEE Trans. Comput.* **37**, 867–872 (July 1988).