Appears in *Proceedings of the 28th Annual Hawaii International Conference on System Sciences (HICSS-28), Vol. II (Software Technology), Maui, Hawaii, January 3-6, 1995, pp. 469-478.*

Fault-Diameter of the Star-Connected Cycles Interconnection Network

Marcelo M. Azevedo*, Nader Bagherzadeh
Dept. of Electrical and Computer Engr.
University of California, Irvine
Irvine, CA 92717
mazevedo, nader@ece.uci.edu

Shahram Latifi
Dept. of Electrical and Computer Engr.
University of Nevada, Las Vegas
Las Vegas, NV 89154-4026
latifi@jb.ee.unlv.edu

Abstract

Let G be a graph with vertex connectivity $k(G)$. An important measure of the fault tolerance of G is its fault-diameter $d_f(G)$, which is defined as the maximum diameter resulting from the deletion of any set of nodes containing less than $k(G)$ nodes. The robustness of G is often measured by comparing $d_f(G)$ with the diameter of the fault-free G, namely $d(G)$. In particular, a family of graphs G_n is dubbed strongly resilient if $d_f(G_n) \leq d(G_n) + c$, where c is a fixed constant independent of n.

This paper derives the fault-diameter of the star-connected cycles (SCC) interconnection network. We show that the SCC_n graph is strongly resilient, exhibiting a fault-diameter of $d_f(SCC_3) = d(SCC_3) + 4$, $d_f(SCC_4) = d(SCC_4)+5$, and $d_f(SCC_n) = d(SCC_n)+1$, for $n \geq 5$.

1 Introduction

The fault tolerance of interconnection networks has received quite some attention from different researchers lately, since maintaining a massively parallel processor (MPP) operating in the presence of faults is fundamental to any practical application.

A basic metric for the fault tolerance of a graph G is the *vertex connectivity* $k(G)$, which is defined as the minimum number of nodes whose removal results in a disconnected or trivial graph (i.e., a graph with disconnected components or an isolated node, respectively) [1]. Similarly, the *edge connectivity* $\lambda(G)$ is the minimum number of links whose removal results in a disconnected or trivial graph.

A shortest path between two nodes $u, v \in G$ is often called a *geodesic*. The *diameter* $d(G)$ of a graph G is the length of any longest geodesic in G [1].

An interconnection network modelled by a graph G with vertex connectivity $k(G)$ and edge connectivity $\lambda(G)$ can tolerate $k(G) - 1$ node failures or $\lambda(G) - 1$ link failures. This measure of fault tolerance, however, gives a poor indication about the impact of faults on the interconnection network. A more appropriate metric, which is often used for measuring the fault tolerance of a graph, is the *fault-diameter*, which is defined as the maximum diameter of any graph obtained from G by removing at most $k(G) - 1$ nodes from G [2].

Let G_n be a family of graphs defined recursively in terms of an integer number n, such that if G_n is obtained from a set of generators $\Omega_n = \{g_1, g_2, \ldots, g_\delta\}$ acting on some group Ψ_n, then G_{n+1} is obtained from a set of generators $\Omega_{n+1} = \Omega_n \cup \{g_{\delta+1}\} = \{g_1, g_2, \ldots, g_\delta, g_{\delta+1}\}$ acting on a group Ψ_{n+1}. A few examples of recursively defined graphs are the hypercube, the star graph and the SCC graph. While these two last graphs are reviewed in Section 2 of this paper, we refer the reader to [3] and [4] for a more formal description and additional examples of group graphs.

An interconnection network G_n is *strongly resilient* if its fault-diameter $d_f(G_n)$ is at most $d(G_n)+c$, where $d(G_n)$ is the diameter of the fault-free G_n and c is a fixed constant independent of n [2]. Many interconnection networks show the desired property of strong resilience, such as the hypercube [2], [5], the star graph [6], [7] and the cube-connected cycles [2].

This paper addresses the fault-diameter of a fixed-degree interconnection network referred to as star-connected cycles (SCC) [8], which was recently proposed as an attractive extension of the star graph [9], [3]. We show that the family of SCC_n graphs is strongly resilient, exhibiting a fault-diameter of $d_f(\text{SCC}_3) = d(\text{SCC}_3) + 4$, $d_f(\text{SCC}_4) = d(\text{SCC}_4) + 5$,

*This research was supported in part by Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq – Brazil), under Grant No. 200392/92-1.

Appears in *Proceedings of the 28th Annual Hawaii International Conference on System Sciences (HICSS-28), Vol. II (Software Technology), Maui, Hawaii, January 3-6, 1995, pp. 469-478.*

and $d_f(\text{SCC}_n) = d(\text{SCC}_n) + 1$, for $n \geq 5$. As noted in [2] for the case of the cube-connected cycles, this result is counter-intuitive due to the abundance of weakly resilient subgraphs (i.e., cycles) in the SCC_n graph.

2 Background

2.1 The star graph

An n-star graph (SG_n) contains $n!$ nodes that are labeled with the $n!$ possible permutations of n distinct symbols. In this paper, we use the digits $\{1, 2, \ldots, n\}$ to label the nodes of SG_n. A node $\pi = p_1 p_2 \ldots p_i \ldots p_n$ is connected to $(n-1)$ distinct nodes, respectively labeled with permutations $\pi_i = p_i p_2 \ldots p_{i-1} p_1 p_{i+1} \ldots p_n$, $2 \leq i \leq n$ (i.e., π_i is the permutation resulting from exchanging the digits occupying the first and the i^{th} position in π) [9], [3]. Each of these $(n-1)$ possible exchange operations is referred to as a *generator* of SG_n. Using finite group theory terminology, SG_n is defined by the generator set $\Omega_n = \{g_i | g_i = i23\ldots(i-1)1(i+1)\ldots n, 2 \leq i \leq n\}$, acting on the symmetric group[1] S_n [4], [10]. Two nodes π and π_i are connected by a link in SG_n if there is a generator $g_i \in \Omega_n$ such that $\pi \cdot g_i = \pi_i$. The link connecting π and π_i is referred to as an i^{th}-dimension link and is labeled i. Figure 1 shows SG_4.

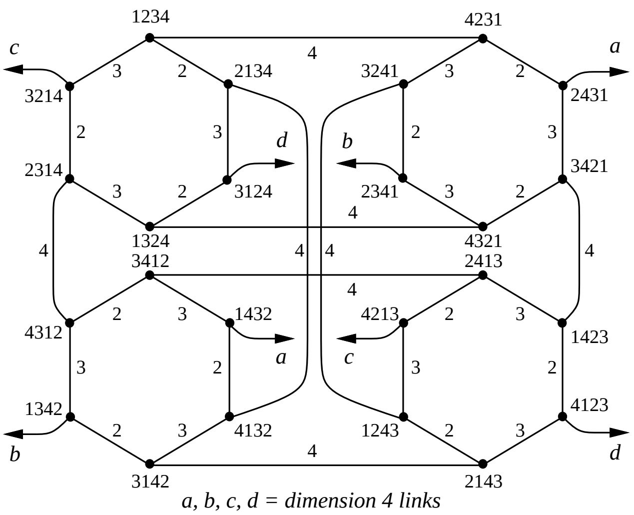

Figure 1: SG_4 graph

SG_n is a regular graph with degree $\delta(\text{SG}_n) = n - 1$. The star graph is vertex- and edge-symmetric, has hierarchical structure, simple routing, and presents a low diameter, given by $d(\text{SG}_n) = \lfloor 3(n-1)/2 \rfloor$ [9].

[1] S_n contains all $n!$ possible permutations of n distinct symbols.

Due to the node symmetry of the star graph, routing between two arbitrary nodes π_s, $\pi_d \in \text{SG}_n$ is equivalent to routing from π_{ds} to the identity node π_1, where $\pi_{ds} = \pi_d^{-1} \cdot \pi_s$, $\pi_1 = 123\ldots n$, and π_d^{-1} is the *inverse* or *reciprocal* of permutation π_d, such that $\pi_d \cdot \pi_d^{-1} = \pi_d^{-1} \cdot \pi_d = \pi_1$ [3], [10]. A shortest path between π_{ds} and π_1 can be obtained by successive application of the following rules:

1. If the first digit in π_{ds} is 1, move it to any position not occupied by the correct digit;
2. If i (i.e. any digit other than 1) is first, move it to its position.

We may organize the digits of permutation π_{ds} as a set of cycles – i.e. cyclically ordered sets of digits with the property that each digit's desired position is that occupied by the next digit in the set. For example, permutation $\pi_{ds} = 23154 \in \text{SG}_5$ can be represented in cyclic format by (1 2 3)(4 5). This cyclic representation has the particularity that the cycles can be listed in any order. In addition, any cyclic shift of the digits within a cycle can be used without affecting the unique permutation label of π_{ds}. For example, (1 2 3)(4 5), (1 2 3)(5 4), (2 3 1)(4 5), (2 3 1)(5 4), (3 1 2)(4 5) and (3 1 2)(5 4) are all valid cyclic representations of permutation $\pi_{ds} = 23154 \in \text{SG}_5$. For simplicity, we assume in this paper that all cycles are written in canonical form, i.e. the smallest digit appears first in each cycle. However, the cycles can be listed in any order.

A cycle containing r digits is dubbed an *r-cycle*. Note that any digit already in its correct position in π_{ds} appears as a *1*-cycle. Let $C_i \in \pi_{ds}$ be an r-cycle of the form $C_i = (p_1\ p_2\ \ldots\ p_r)$, $2 \leq r \leq n$. The *execution of* C_i corresponds to a path R in SG_n and can be expressed as a sequence of links as follows [11]:

- If $p_1 = 1$: $R = (p_2, p_3, \ldots, p_r)$.
- If $p_1 \neq 1$: $R = (p_{1+k \bmod r}, p_{1+(k+1) \bmod r}, \ldots, p_{1+(k+r-1) \bmod r}, p_{1+k \bmod r})$, for $0 \leq k \leq r-1$.

The execution of each cycle $C_i \in \pi_{ds}$ is part of a shortest path routing between the source and the destination nodes. For example, one possible shortest path R_{ds} from $\pi_{ds} = 23154 = (1\ 2\ 3)(4\ 5)$ to $\pi_1 = 12345 = (1)(2)(3)(4)(5)$ in SG_5 is the sequence of links (2, 3, 4, 5, 4):

$$23154 \xrightarrow{2} 32154 \xrightarrow{3} 12354 \xrightarrow{4} 52314 \xrightarrow{5} 42315 \xrightarrow{4} 12345$$

The length of a shortest path from π_{ds} to π_1 is [9]:

Appears in *Proceedings of the 28th Annual Hawaii International Conference on System Sciences (HICSS-28), Vol. II (Software Technology), Maui, Hawaii, January 3-6, 1995, pp. 469-478.*

$$|R_{ds}| = \begin{cases} c+m, & \text{if the 1st digit in } \pi_{ds} \text{ is 1} \\ c+m-2, & \text{if the 1st digit in } \pi_{ds} \text{ is not 1,} \end{cases}$$

where c is the number of cycles of length at least 2 in π_{ds} and m is the total number of digits in these cycles.

Note that according to the order chosen to execute the cycles in π_{ds} different paths result. Furthermore, the cycles do not have to be executed independently (i.e., it is possible to interleave the execution of the cycles in π_{ds}). Some possible shortest paths from $\pi_{ds} = 23154 = (1\ 2\ 3)(4\ 5)$ to $\pi_1 = 12345 = (1)(2)(3)(4)(5)$ are (2, 3, 4, 5, 4), (2, 3, 5, 4, 5), (4, 5, 4, 2, 3), (5, 4, 5, 2, 3), (2, 4, 5, 4, 3) and (2, 5, 4, 5, 3). While some of these paths share some common intermediate nodes (e.g., (2, 3, 4, 5, 4) and (2, 3, 5, 4, 5)), Day and Tripathi proved in [12] that there are $(n-1)$ node-disjoint paths between any two nodes in SG$_n$. These paths are of optimal length as listed in Table 1.

First digit in π_{ds} (p_1)	Number of *disjoint paths*	Length
$p_1 = 1$	m	$c+m$
	$n-m-1$	$c+m+2$
$p_1 \neq 1$	c	$c+m-2$
	$m-c-1$	$c+m$
	$n-m$	$c+m+2$

Table 1: Optimal node-disjoint paths in SG$_n$

SG$_n$ has a vertex connectivity of $k(\mathrm{SG}_n) = (n-1)$, and therefore can tolerate up to $(n-2)$ node failures. Exact values for the fault-diameter of the star graph were derived in [7] and are given below. This result is obtained via a worst-case analysis that considers the impact of up to $(n-1)$ arbitrary faults on the optimal-length node-disjoint paths listed in Table 1.

$$d_f(\mathrm{SG}_n) = \begin{cases} d(\mathrm{SG}_n)+1, & \text{if } n \text{ is odd or } n \geq 7 \\ d(\mathrm{SG}_n)+2, & \text{if } n=4 \text{ or } n=6 \end{cases}$$

2.2 The star-connected cycles graph

An *n-SCC* graph (SCC$_n$) is obtained by replacing each node of an n-star with a ring of $(n-1)$ nodes, namely a *supernode*. The i^{th} dimension link originally connected to a node of the star is referred to as a *lateral link* and is connected to the i^{th} node of the corresponding ring in the SCC graph. The connections between nodes inside the same supernode are referred to as *local links*. Figure 2 shows SCC$_4$.

The nodes in each ring are identified by a label $\langle i, \pi \rangle$, where $2 \leq i \leq n$ and π is a permutation of n digits. Then two nodes $\langle i, \pi \rangle$ and $\langle i', \pi' \rangle$ are connected by a link $(\langle i, \pi \rangle, \langle i', \pi' \rangle)$ in an SCC$_n$ graph iff either

1. $(\langle i, \pi \rangle, \langle i', \pi' \rangle)$ is a local link, i.e. $\pi = \pi'$ and $\min(|i-i'|, n-1-|i-i'|) = 1$, or
2. $(\langle i, \pi \rangle, \langle i', \pi' \rangle)$ is a lateral link, i.e. $i = i'$ and π differs from π' only in the first and i^{th} digits such that $\pi(1) = \pi'(i)$ and $\pi(i) = \pi'(1)$.

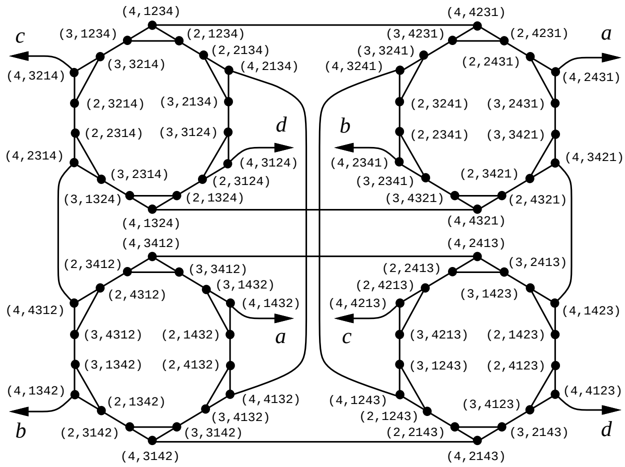

Figure 2: SCC$_4$ graph

SCC$_n$ contains $(n-1)n!$ nodes, $(n-1)n!$ local links and $(n-1)n!/2$ lateral links. SCC$_n$ is a vertex symmetric graph with degree $\delta(\mathrm{SCC}_3) = 2$, and $\delta(\mathrm{SCC}_n) = 3$ for $n \geq 4$. For $n \geq 4$, the vertex connectivity of SCC$_n$ is $k(\mathrm{SCC}_n) = 3$, and therefore up to 2 node failures can be tolerated. For $n = 3$, only 1 node failure can be tolerated, since in this case $k(\mathrm{SCC}_n) = 2$. In addition, the diameter of SCC$_n$ graph is given by:

$$d_{SCC} = \begin{cases} 6, & \text{for } n = 3 \\ \frac{1}{2}(n^2+n-4), & \text{for even } n \\ \frac{1}{2}(n^2+3n-8), & \text{for odd } n \geq 5 \end{cases} \tag{1}$$

Routing between two nodes $\langle i_s, \pi_s \rangle$ and $\langle i_d, \pi_d \rangle$ in an SCC graph reduces to routing from $\langle i_s, \pi_{ds} \rangle$ to $\langle i_d, \pi_1 \rangle$, where $\pi_{ds} = \pi_d^{-1} \cdot \pi_s$ and $\pi_1 = 123 \ldots n$ [8]. Routing from supernode π_s to supernode π_d can be accomplished with the same routing techniques that have been proposed for the star graph. However, the order chosen to execute the cycles in permutation π_{ds} affects the number of local links that have to be traversed in the corresponding path in the SCC graph. A routing algorithm that provides an optimal ordering of cycles (and consequently, a shortest path between $\langle i_s, \pi_s \rangle$ and $\langle i_d, \pi_d \rangle$) in SCC$_n$ was proposed in [8].

To illustrate the cost of local links in the routing, Figure 3 shows the path between nodes $\langle 3, 34125 \rangle$ and

$\langle 2, 12345\rangle$ in SCC_5. The cycle representation of permutation 34125 is (1 3)(2 4)(5), and in Figure 3 the order chosen to execute these cycles is (2, 4, 2, 3). The path shown in Figure 3 contains 4 lateral links and 7 local links. However, if the sequence of lateral links (3, 2, 4, 2) is used, a path with 4 lateral links and 5 local links results.

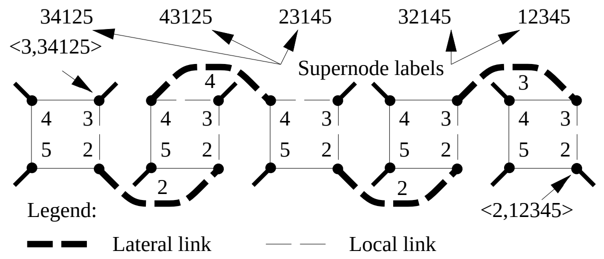

Figure 3: Example of routing in an SCC_5 graph

In general, the total cost of a path from $\langle i_s, \pi_s\rangle$ to $\langle i_d, \pi_d\rangle$ in an SCC_n graph along a sequence of s lateral links $R(\ell_1 \mapsto \ell_s) = (\ell_1, \ell_2, \ldots, \ell_s)$ is

$$|\mathcal{P}(\ell_1 \mapsto \ell_s)| = s + d(i_s, \ell_1) + \sum_{j=1}^{s-1} d(\ell_j, \ell_{j+1}) + d(\ell_s, i_d,) \quad (2)$$

where the function $d(i, j) = \min(|i-j|, n-1-|i-j|)$ represents the minimum cost required to traverse a supernode between its i^{th} and j^{th} nodes.

3 Derivation of $d_f(\mathbf{SCC}_n)$

3.1 Lower bound for $d_f(\mathbf{SCC}_n)$

The lower bound is easily shown to be $d(SCC_n)+1$ as follows. Let u and v be two nodes located $d(SCC_n)$ links away from each other. In addition, let w be a neighbor of v such that v is the only fault-free neighbor of w (i.e., the $k(SCC_n)-1$ faulty nodes in the graph are the other neighbors of w). Any path from w to u must pass through v and therefore is of length at least $d(SCC_n)+1$.

3.2 Exact value for $d_f(\mathbf{SCC}_3)$

The exact value of the fault-diameter of an SCC_3 graph can be obtained by inspection as follows. Considering that SG_3 is a hexagon [9], replacing each node of SG_3 with 2 connected nodes results in a dodecagon (i.e., SCC_3). Since $k(SCC_3) = 2$, only one node failure can be tolerated by SCC_3. If an arbitrary node failure in SCC_3 is chosen, the two neighbors of the faulty node remain connected via a path containing 10 links. Therefore, $d_f(SCC_3) = 10 = d(SCC_3) + 4$.

3.3 Outline of the proof method for $n \geq 4$

One possible method that can be used to derive the fault-diameter of a graph G_n consists of accomplishing a worst-case analysis onto node-disjoint paths in G_n [5], [7]. Given two nodes $u, v \in G_n$, and assuming that $k(G_n)$ node-disjoint paths from u to v exist, $k(G_n)-1$ of these paths are supposed to be faulty, such that only the longest path remains fault-free. Naturally, the lengths of these node-disjoint paths depend on the selected nodes u and v, which must be chosen such as to result in the longest possible fault-free path after the faults have been taken into account. The length of this longest possible path is the fault-diameter of $G(n)$, namely $d_f(G_n)$.

Since the degree of SCC_n, $n \geq 4$, is $\delta(SCC_n) = 3$, there are at most 3 node-disjoint paths between any two nodes in an SCC_n graph. The derivation of the fault-diameter of the SCC graph, however, seems to be more appropriately tackled by a variation of the method described above. Without loss of generality, we choose $u = \langle i, \pi\rangle$ and $v = \langle j, 123\ldots n\rangle$. The following aspects influence the derivation of $d_f(SCC_n)$:

- The permutation portion of u's label (π).
- The digit portion of u's and v's labels (i and j).
- The order of execution of the cycles in π such that paths of minimum length from u to v result.
- The placement of the 2 tolerable faults so as to maximize the length of an optimal fault-free path left between u and v.

Taking all the above aspects into account simultaneously can be a quite difficult task. To make the analysis more tractable, we use a tabular method in which the required aspects can be entered individually and then combined into a final worst-case analysis.

The first step in the method consists of selecting a permutation π requiring a large number of lateral and local links in the path from u to v. The number of lateral links depends on the number of cycles of length at least 2 in π (c) and on the number of digits in these cycles (m), as shown in Table 1. The number of local links also depends on c and m, but can be further increased by a proper selection of the internal composition of the cycles in π. Such an approach follows the same reasoning adopted in [8] to derive the diameter of the fault-free SCC_n. For example, the execution of a *2*-cycle $(2\ \ 3)$ in an SCC_7 graph requires 3 lateral and 2 local links. A *2*-cycle $(2\ \ 5)$, on the other hand, requires 3 lateral and 6 local links in SCC_7. Since it is not possible to determine a priori which permutation π will result in the longest path after faults have been taken into account, our method actually considers a few possible candidate permutations initially.

Once a particular permutation π is selected for analysis, an $(n-1)\times(n-1)$ table containing $(n-1)^2$ *partially node-disjoint* paths from $u = \langle i, \pi\rangle$ to $v = \langle j, 123\ldots n\rangle$ is built. Each entry in this table is the length of a path $\mathcal{Q}(\ell_1 \mapsto \ell_s)$ from node $\langle \ell_1, \pi\rangle$ to node $\langle \ell_s, 123\ldots n\rangle$ along the sequence of lateral links $R(\ell_1 \mapsto \ell_s) = (\ell_1, \ell_2, \ldots, \ell_s)$. In other words, $\mathcal{Q}(\ell_1 \mapsto \ell_s)$ is a path in which the initial and final lateral links (respectively, ℓ_1 and ℓ_s, for $2 \le \ell_1, \ell_s \le n$) are predetermined. The remaining lateral links that are required for such a path (i.e., $\ell_2, \ell_3, \ldots, \ell_{s-1}$) are selected by applying the optimal SCC routing algorithm proposed in [8] to the rules proposed in [12] for forming node-disjoint paths of optimal length in the star graph. This results in optimal paths that obey all $(n-1)^2$ possible choices for the initial and final local links ℓ_1 and ℓ_s. Note that the number of lateral links in these paths (s) may vary as shown in Table 1. However, we do not require that these paths are node-disjoint (in fact, they are only partially node-disjoint). Note also that for each possible selection of ℓ_1 and ℓ_s, we assume routing from $\langle \ell_1, \pi\rangle$ to $\langle \ell_s, 123\ldots n\rangle$ (i.e., we suppose that the values of the digits i and j in the labels of u and v are not initially known, and therefore the cost of the path $\mathcal{Q}(\ell_1 \mapsto \ell_s)$ is computed assuming that no local links are traversed in the source and destination supernodes). Hence, the cost of each path $\mathcal{Q}(\ell_1 \mapsto \ell_s)$ that is entered in the table is given by

$$|\mathcal{Q}(\ell_1 \mapsto \ell_s)| = s + \sum_{j=1}^{s-1} d(\ell_j, \ell_{j+1})$$

As mentioned above, the values of the digit portions of u and v (i.e., i and j respectively) are left initially unspecified. After the faults are taken into account, all $(n-1)^2$ possible combinations of digits i and j are considered in our method.

Another particularity of our method is the placement of the 2 tolerable faults in the SCC graph. In our analysis, we assume that the faults are located close to the source and destination nodes (more specifically, the 2 faults are supposed to occur anywhere within supernodes π or $123\ldots n$ or in a neighboring supernode connected to these supernodes). Such a fault hypothesis allows an extensive, yet simple, worst-case analysis based on our tabular method. Specifically, a table containing all $(n-1)^2$ possible shortest paths of cost $|\mathcal{Q}(\ell_1 \mapsto \ell_s)|$, $2 \le \ell_1, \ell_s \le n$ is inspected such that columns and/or rows containing the overall lowest cost paths are disabled due to the occurrence of properly placed faults. A fault located close to the source node disables an entire row of the table, and a fault located close to the destination node disables an entire column. The goal of the fault placement procedure is to maximize the length of the fault-free paths still left in the table.

The fact that our fault model considers only faults that are located close to the source and destination nodes does not invalidate the results presented in this paper. As it will become apparent when our technique is applied to the SCC graph, such faults can impair a larger number of possible paths than faults which are located farther from the source and destination nodes. This observation can be justified by the richness of partially node-disjoint paths in the star graph, and consequently in the SCC graph. If the number of misplaced digits in permutation π is m, there are about $m!$ different possible paths to the identity node. Each of these paths corrects the position of the misplaced digits in π in a different order. Although some of these paths share a few common intermediate nodes, they all share in common only the source and the destination nodes u and v. Hence, faults in the proximity of u and v will impair more paths than faults that are placed at random in the SCC graph.

3.4 Exact value for $d_f(\mathbf{SCC}_4)$

The exact value of the fault-diameter of SCC_4 can be obtained by application of the tabular method described above. Initially, we consider some candidates for permutation π that are likely to result in the largest possible number of links in the presence of faults. The values of c and m in the selected permutations result in the highest or close to the highest possible number of lateral links along the different routes from π to the identity node (see Table 1). Naturally, this selection criterion also increases the number of local links in these routes. For $n = 4$, our worst-case analysis selected the following candidate permutations: $(1)(2)(3\ \ 4)$, $(1)(2\ \ 3\ \ 4)$, $(1\ \ 2\ \ 3\ \ 4)$ and $(1\ \ 2)(3\ \ 4)$. For each of these permutations, we built a table listing the cost of all shortest paths $\mathcal{Q}(\ell_1 \mapsto \ell_s)$, for $2 \le \ell_1, \ell_s \le 4$ (see Tables 2a through 2d). Note that in any of the cases analyzed in this paper, there are other similar permutations that lead to exactly the same results (i.e., permutations having the same values of c and m but with

Appears in *Proceedings of the 28th Annual Hawaii International Conference on System Sciences (HICSS-28), Vol. II (Software Technology), Maui, Hawaii, January 3-6, 1995, pp. 469-478.*

different selections of digits within the cycles). Some different permutations that could have been used for the case $n = 4$, for example, are (1)(3)(2 4), (1)(2 4 3), (1 2 4 3) and (1 3)(2 4).

Init. link	*Final link* 2	3	4
2	9	13	13
3	13	5	13
4	13	13	5

(a) $\pi = (1)(2)(3\ 4)$

Init. link	*Final link* 2	3	4
2	7	11	11
3	11	7	11
4	11	11	7

(b) $\pi = (1)(2\ 3\ 4)$

Init. link	*Final link* 2	3	4
2	13	9	5
3	9	9	13
4	9	9	13

(c) $\pi = (1\ 2\ 3\ 4)$

Init. link	*Final link* 2	3	4
2	15	7	7
3	7	11	11
4	7	11	11

(d) $\pi = (1\ 2)(3\ 4)$

Table 2: Cost of paths $\mathcal{Q}(\ell_1 \mapsto \ell_s)$ in SCC$_4$

For conciseness, we list in Table 3 only the sequences of lateral links $R(\ell_1 \mapsto \ell_s)$ used to build Table 2d. The sequences that are required for the remaining tables can be obtained similarly, i.e. by applying the optimal SCC routing algorithm proposed in [8] along with the techniques proposed in [12] for forming optimal node-disjoint paths in the star graph.

Init. link	*Final link* 2	3	4
2	$(2,3,4,2,3,2,3,2)$	$(2,3,4,3)$	$(2,4,3,4)$
3	$(3,4,3,2)$	$(3,2,3,2,4,3)$	$(3,2,4,3,2,4)$
4	$(4,3,4,2)$	$(4,2,3,4,2,3)$	$(4,3,2,4,2,4)$

Table 3: Sequences of lateral links $R(\ell_1 \mapsto \ell_s)$ used to build Table 2d

We now explain in further detail how to select a placement of the faults that results in a worst-case scenario from the viewpoint of routing in SCC$_4$. The longest path in this worst-case scenario yields the exact value of $d_f(\mathrm{SCC}_4)$.

Although the following procedure was accomplished for each of the Tables 2a through 2d, we will limit our discussion to the case of Table 2d. Application of the worst-case analysis technique to the remaining cases is in fact straightforward.

Each different fault placement causes a particular impact on Table 2d in terms of optimal paths that can still be taken in SCC$_4$. For each 2 faults that are selected, at most 2 rows and/or columns of Table 2d are disabled.

For example, in Table 2d we see that the shortest paths (of cost 7) are all located either in the first row or in the first column of the table. If we assume that node failures disable paths starting with $\ell_1 = 2$ or ending with $\ell_s = 2$, the possible paths left are shown in Table 4a.

Init. link	*Final link* 2	3	4
2	–	–	–
3	–	11	11
4	–	11	11

(a) Worst-case fault placement for $\pi = (1\ 2)(3\ 4)$

i	j 2	3	4
2	13	12	12
3	12	11	11
4	12	11	11

(b) Cost of optimal paths under the fault placement of Table 4a

Table 4: A worst-case fault placement in SCC$_4$

Recalling that Table 2d corresponds to permutation $\pi = (1\ 2)(3\ 4) = 2143$, two possible nodes that can cause the impairment shown in Table 4a are $\langle 2, 1243\rangle$ and $\langle 2, 2134\rangle$. Since under the fault placement shown in Table 4a the faults are external to the source and destination supernodes, any i, j can still be selected in supernodes π and 1234. Therefore, we must now verify which values of i and j maximize the cost of the path from $u = \langle i, \pi\rangle$ to $v = \langle j, 1234\rangle$ in the presence of the faults shown in Table 4a. The worst-case analysis in terms of i and j is done by observing which alternative paths can still be selected from Table 4a. To route from $\langle 2, \pi\rangle$ to $\langle 2, 1234\rangle$, for example, we can take the path $\langle 2, \pi\rangle \to \langle 3, \pi\rangle \to \cdots \to \langle 3, 1234\rangle \to \langle 2, 1234\rangle$.

The cost of this path is $1+11+1 = 13$. By applying the same reasoning to other combinations of i and j, we obtain Table 4b. As it can be verified in Table 4b, the length of the longest path in the presence of a worst-case fault placement is 13. This value actually corresponds to the fault-diameter of SCC$_4$. Hence, $d_f(\mathrm{SCC}_4) = d(\mathrm{SCC}_4) + 5 = 13$. The correctness of this result can be verified via an analysis of other fault placements onto Tables 2a through 2d.

It is also interesting to confirm how the worst-case faults $\langle 2, 1243\rangle$ and $\langle 2, 2134\rangle$ actually impair the routing from $\langle 2, 2143\rangle$ to $\langle 2, 1234\rangle$ by physically placing these faults onto Figure 2. To make the inspection easier, Figure 4 lists all possible paths containing either 4 or 6 lateral links between supernodes 2143 and 1234.

Appears in *Proceedings of the 28th Annual Hawaii International Conference on System Sciences (HICSS-28), Vol. II (Software Technology), Maui, Hawaii, January 3-6, 1995, pp. 469-478.*

Note that all paths containing only 4 lateral links either begin or end with lateral link 2. Since the faults resulting from our tabular method (i.e., $\langle 2, 1243\rangle$ and $\langle 2, 2134\rangle$) completely disable these paths, only paths containing 6 lateral links are left (in fact, 4 of these paths remain fault-free). Furthermore, in the presence of the indicated faults all the optimal paths begin and end with either lateral link 3 or 4. Therefore, by choosing $i = 2$ and $j = 2$, the longest overall path is from node $\langle 2, 2143\rangle$ to node $\langle 2, 1234\rangle$. The total length of this path is 13 as indicated by Equation 2.

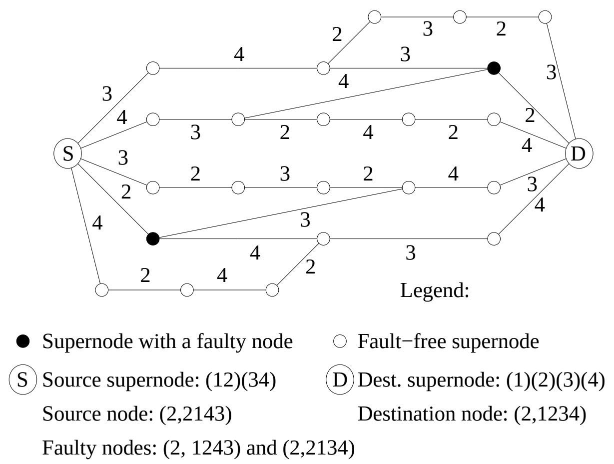

Figure 4: Alternate routes in an SCC_4 graph affected by 2 faults

3.5 Exact value for $d_f(\mathbf{SCC}_5)$

The exact value of the fault-diameter of an SCC_5 graph can also be obtained by application of the tabular method described previously. Our worst-case analysis was carried out with the following candidate permutations: $(1)(2\ 4\ 3\ 5), (1\ 2)(3\ 5\ 4), (1)(2\ 4)(3\ 5)$ and $(1\ 2\ 4)(3\ 5)$. Tables 5a through 5d list the cost of all shortest paths $\mathcal{Q}(\ell_1 \mapsto \ell_s)$, $2 \le \ell_1, \ell_s \le 5$, for these permutations. For conciseness, we list in Table 6 only the sequences of lateral links $R(\ell_1 \mapsto \ell_s)$ used to build Table 5d.

The shortest paths $\mathcal{Q}(\ell_1 \mapsto \ell_s)$ in Tables 5a and Table 5c are distributed along the diagonal of those tables. Since the 2 tolerable faults act on a combination of at most 2 columns and/or rows, it is not possible to disable all these paths. We can therefore limit our worst-case analysis to Tables 5b and 5d. Due to space constraints, only the case $\pi = (1\ 2\ 4)(3\ 5) = 24513$ (Table 5d) will be considered here.

The shortest paths in Table 5d are of cost 11 or 12 and are all located either in the 1st row or in the

In. link	*Final link* 2	3	4	5
2	11	16	15	14
3	16	11	14	15
4	17	16	11	14
5	16	17	14	11

(a) $\pi = (1)(2\ 4\ 3\ 5)$

In. link	*Final link* 2	3	4	5
2	19	10	11	10
3	10	15	16	13
4	11	16	17	14
5	10	15	16	13

(b) $\pi = (1\ 2)(3\ 5\ 4)$

In. link	*Final link* 2	3	4	5
2	14	15	20	15
3	15	14	15	20
4	20	15	14	15
5	15	20	15	14

(c) $\pi = (1)(2\ 4)(3\ 5)$

In. link	*Final link* 2	3	4	5
2	17	12	11	12
3	16	15	12	15
4	17	16	17	16
5	16	15	12	15

(d) $\pi = (1\ 2\ 4)(3\ 5)$

Table 5: Cost of paths $\mathcal{Q}(\ell_1 \mapsto \ell_s)$ in SCC_5

Init. link	*Final link* 2	3	4	5
2	$(2,4,2,5,3,5,2)$	$(2,4,3,5,3)$	$(2,3,5,3,4)$	$(2,4,5,3,5)$
3	$(3,5,4,2,4,3,2)$	$(3,5,4,3,2,4,3)$	$(3,5,3,2,4)$	$(3,2,4,5,3,2,5)$
4	$(4,2,4,5,3,5,2)$	$(4,3,5,2,4,2,3)$	$(4,5,3,5,4,2,4)$	$(4,5,3,4,2,4,5)$
5	$(5,3,4,2,4,5,2)$	$(5,4,3,5,2,4,3)$	$(5,3,5,2,4)$	$(5,3,4,5,2,4,5)$

Table 6: Sequences of lateral links $R(\ell_1 \mapsto \ell_s)$ used to build Table 5d

3rd column of that table. If we assume that node failures disable paths $\mathcal{Q}(\ell_1 \mapsto \ell_s)$ beginning with $\ell_1 = 2$ or ending with $\ell_s = 4$, the possible remaining paths are given by Table 7a. Two possible nodes that can cause such an impact in Table 5d are $\langle 2, 42513\rangle$ and $\langle 4, 42315\rangle$. Since under the fault placement given in Table 7a the faults are external to the source and destination supernodes, any values of i, j can still be selected. Table 7b gives the costs of the optimal paths for all possible values of i and j, in the presence of the fault placement depicted in Table 7a.

Another possible worst-case fault placement that can be applied to Table 5d consists of the failure of the 2 local link neighbors of node $\langle 4, 24513\rangle$ (i.e., $\langle 3, 24513\rangle$ and $\langle 5, 24513\rangle$). Such a fault placement disables the 2nd and 4th rows of Table 5d, yielding Table 8a. Note that with this fault placement we can

Appears in *Proceedings of the 28th Annual Hawaii International Conference on System Sciences (HICSS-28), Vol. II (Software Technology), Maui, Hawaii, January 3-6, 1995, pp. 469-478.*

In. link	*Final link* 2	3	4	5
2	–	–	–	–
3	16	15	–	15
4	17	16	–	16
5	16	15	–	15

(a) Worst-case fault placement for $\pi = (1\,2\,4)(3\,5)$

i	*j* 2	3	4	5
2	17	16	17	16
3	16	15	16	15
4	17	16	17	16
5	16	15	16	15

(b) Cost of optimal paths under the fault placement of Table 7a

Table 7: A worst-case fault placement in SCC$_5$

In. link	*Final link* 2	3	4	5
2	17	12	11	12
3	–	–	–	–
4	17	16	17	16
5	–	–	–	–

(a) Another worst-case fault placement for $\pi = (1\,2\,4)(3\,5)$

i	*j* 2	3	4	5
2	13	12	11	12
3	–	–	–	–
4	17	16	17	16
5	–	–	–	–

(b) Cost of optimal paths under the fault placement of Table 8a

Table 8: Another worst-case fault placement in SCC$_5$

only choose $i = \{2, 4\}$ and $j = \{2, 3, 4, 5\}$. The costs of the optimal paths for all possible combinations of digits i and j under the fault placement shown in Table 8a are given in Table 8b. Note that in some cases the costs of these optimal paths can be lower than those listed in the original $Q(\ell_1 \mapsto \ell_s)$ table (i.e., Table 5d in this case), since better alternate routes can still be found despite of the presence of faults. Such is the case, for example, of the routing from $\langle 2, 24513\rangle$ to $\langle 2, 12345\rangle$. A better alternate route which is still available after the fault placement given in Table 8a is $\langle 2, 24513\rangle \rightarrow \cdots \rightarrow \langle 3, 12345\rangle \rightarrow \langle 2, 12345\rangle$.

Note that the cost of the longest path in both Table 7b and Table 8b is 17. Hence, $d_f(\mathrm{SCC}_5) = d(\mathrm{SCC}_5) + 1 = 17$. The correctness of this result can be verified by analysis of other fault placements onto Tables 5b and 5d.

3.6 Exact value for $d_f(\mathbf{SCC}_6)$

The worst-case analysis for the SCC$_6$ graph was carried out with the following candidate permutations: (1)(2 4)(3 5 6), (1 2)(3 5)(4 6), (1)(2)(3 5)(4 6) and (1 3 5 2)(4 6). Tables 9a through 9d list the cost of all shortest paths $Q(\ell_1 \mapsto \ell_s)$ $(2 \le \ell_1, \ell_s \le 6)$ for these permutations. Due to space constraints, we show in Table 10 only the sequences of lateral links $R(\ell_1 \mapsto \ell_s)$ used to build Table 9d.

One possible worst-case fault placement that can be applied to Table 9d is shown in Table 11a. The costs of the optimal paths for all possible combinations of digits i and j are listed in Table 11b.

In. link	*Final link* 2	3	4	5	6
2	–	18	19	19	18
3	–	–	–	–	–
4	–	20	19	20	19
5	–	19	20	20	19
6	–	20	20	20	19

(a) Worst-case fault placement for $\pi = (1\,3\,5\,2)(4\,6)$

i	*j* 2	3	4	5	6
2	19	18	19	19	18
3	20	19	20	20	19
4	20	20	19	20	19
5	20	19	20	20	19
6	20	19	20	20	19

(b) Cost of optimal paths under the fault placement of Table 11a

Table 11: A worst-case fault placement in SCC$_6$

Note that the cost of the longest path in Table 11b is 20. Hence, $d_f(\mathrm{SCC}_6) = d(\mathrm{SCC}_6) + 1 = 20$. The correctness of this result can be verified by analysis of other fault placements onto Tables 9a through 9d.

3.7 Proof for general values of n

Intuitively, the tabular method of proof used so far in this paper indicates that, for $n > 4$, there is little possibility that faults will cause a significant impairment in the SCC$_n$ graph. Such a reasoning can be justified by the fact that the 2 tolerable faults can affect at most 2 rows and/or columns of an $(n-1)\times(n-1)$ table containing the costs of optimal paths $Q(\ell_1 \mapsto \ell_s)$, for $2 \le \ell_1, \ell_s \le n$. We now extend our previous results for the case of general values of n as follows:

Theorem 1 *The fault-diameter of the SCC_n graph is*

$$d_f(SCC_n) = \begin{cases} d(SCC_n) + 4, & for\ n = 3 \\ d(SCC_n) + 5, & for\ n = 4 \\ d(SCC_n) + 1, & for\ n > 4, \end{cases}$$

where $d(SCC_n)$ is the diameter of the fault-free SCC_n.

Proof: The proof for the cases $n = 4$, $n = 5$ and $n = 6$ uses the tabular method described earlier and was presented in Subsections 3.4 through 3.6. The proof for $n > 6$ is by induction on n and uses the fact that the diameter of a fault-free SCC$_n$ graph relates

In.	*Final link*				
link	2	3	4	5	6
2	16	17	22	18	17
3	17	16	17	22	22
4	22	17	16	17	18
5	18	22	17	16	21
6	17	22	18	21	16

(a) $\pi = (1)(2\ 4)(3\ 5\ 6)$

In.	*Final link*				
link	2	3	4	5	6
2	21	16	17	17	16
3	16	22	18	21	17
4	17	18	22	19	21
5	17	21	19	22	18
6	16	17	21	18	22

(b) $\pi = (1\ 2)(3\ 5)(4\ 6)$

In.	*Final link*				
link	2	3	4	5	6
2	18	19	20	20	19
3	19	14	15	20	16
4	20	15	14	15	20
5	20	20	15	14	15
6	19	16	20	15	19

(c) $\pi = (1)(2)(3\ 5)(4\ 6)$

In.	*Final link*				
link	2	3	4	5	6
2	19	18	19	19	18
3	14	22	16	19	15
4	15	20	19	20	19
5	18	19	20	20	19
6	16	20	20	20	19

(d) $\pi = (1\ 3\ 5\ 2)(4\ 6)$

Table 9: Cost of paths $\mathcal{Q}(\ell_1 \mapsto \ell_s)$ in SCC$_6$

Init.	*Final link*				
link	2	3	4	5	6
2	$(2,6,4,6,2,3,5,2)$	$(2,3,4,6,4,5,2,3)$	$(2,3,5,2,3,4,6,4)$	$(2,6,4,6,5,2,3,5)$	$(2,6,4,5,2,3,5,6)$
3	$(3,4,6,4,5,2)$	$(3,4,6,4,5,3,2,3,2,3)$	$(3,5,2,4,6,4)$	$(3,4,6,4,2,5,2,5)$	$(3,5,2,6,4,6)$
4	$(4,6,4,3,5,2)$	$(4,6,4,5,2,3,5,3)$	$(4,6,5,2,4,3,5,4)$	$(4,6,4,3,2,5,2,5)$	$(4,3,5,2,6,4,3,6)$
5	$(5,6,4,5,3,5,6,2)$	$(5,6,4,6,2,3,5,3)$	$(5,2,3,5,3,4,6,4)$	$(5,6,4,6,2,5,3,5)$	$(5,2,6,4,5,3,5,6)$
6	$(6,4,6,3,5,2)$	$(6,4,6,5,2,3,5,3)$	$(6,5,2,4,6,3,5,4)$	$(6,4,6,2,5,2,3,5)$	$(6,4,5,2,6,3,5,6)$

Table 10: Sequences of lateral links $R(\ell_1 \mapsto \ell_s)$ used to build Table 9d

to the diameter of a fault-free SCC$_{n-2}$ graph by the recurrence below. This recurrence holds for $n \geq 6$ and can be verified from Equation 1.

$$d(\mathrm{SCC}_n) = \begin{cases} d(\mathrm{SCC}_{n-2}) + 2n - 1, & \text{for even } n \\ d(\mathrm{SCC}_{n-2}) + 2n + 1, & \text{for odd } n \end{cases}$$

Let a candidate permutation used in the derivation of the fault-diameter of SCC$_{n-2}$ be referred to as π_{n-2}. If the tabular method were to be used for an SCC$_n$ graph, a new candidate permutation π_n could be obtained by extending π_{n-2} via the following rules:

R1 Form π_n by adding a *2*-cycle $(n-1\ n)$ to π_{n-2}.

R2 Rearrange the digits in the cycles of π_n, such that the number of local links that must be traversed during the execution of these cycles is maximized.

Note that if the number of cycles of length at least 2 in π_{n-2} is c_{n-2} and the the number of digits in these cycles is m_{n-2}, then the corresponding values for π_n are $c_n = c_{n-2} + 1$ and $m_n = m_{n-2} + 2$. Rules **R1** and **R2** guarantee a maximum number of additional lateral and local links while forming π_n from π_{n-2}. As an example, the candidate permutations used in the derivation of $d_f(\mathrm{SCC}_5)$ can be extended such as to result in the following candidate permutations for SCC$_7$: $(1)(2\ 5\ 3\ 6)(4\ 7)$, $(1\ 2)(3\ 5\ 6)(4\ 7)$, $(1)(2\ 5)(3\ 6)(4\ 7)$ and $(1\ 2\ 5)(3\ 6)(4\ 7)$.

Let the extra *2*-cycle obtained by application of rules **R1** and **R2** to π_{n-2} be C_e. **R2** requires that:

$$C_e = \begin{cases} (\frac{n}{2}\ n) \text{ or } (\frac{n+2}{2}\ n), & \text{for even } n \\ (\frac{n+1}{2}\ n), & \text{for odd } n \end{cases}$$

Our goal is to compute the cost difference between the families of paths $\mathcal{Q}(\ell_1 \mapsto \ell_s)$ in SCC$_n$ and SCC$_{n-2}$, via an analysis of the transformations carried onto π_{n-2} that lead to π_n. This can be done by taking into account each of the following aspects:

1. Equation 2 indicates that the execution of the extra cycle C_e requires $|C_e|$ links, where $|C_e| = n+1$, for even n, and $|C_e| = n+2$, for odd n.
2. The cycle structure of π_{n-2} is such that:
   - For even n, π_{n-2} contains at most $(n-4)/2$ r-cycles C_i, $r \geq 2$, such that $C_i \neq (1\ p_i)$, $2 \leq p_i \leq n-2$. Rearranging the digits within these cycles to form π_n requires at most $(n-4)$ additional local links.
   - For odd n, π_{n-2} contains at most $(n-3)/2$ r-cycles, $r \geq 2$. Rearranging the digits within these cycles to form π_n requires at most $(n-3)$ additional local links.

3. As explained earlier, the internal composition of the cycles in π_n maximizes the number of local links that must be traversed during the execution of these cycles in SCC_n. On the other hand, the distribution of the digits over the cycles of π_n favors cycle orderings in which only one local link is required to move from one cycle to the other. Therefore, the extra cycle C_e introduced while forming π_n also contributes with one additional local link to the cost of a path $\mathcal{Q}(\ell_1 \mapsto \ell_s)$ from $u = \langle \ell_1, \pi_n \rangle$ to $v = \langle \ell_s, 123 \ldots n \rangle$.
4. Finally, one extra local link is required in the fault analysis of SCC_n when compared to SCC_{n-2}, due to the fact that the node labels within a supernode of SCC_n range from 2 to n as opposed to 2 to $(n-2)$ in SCC_{n-2}.

Adding the cost increments of items 1 to 4 above, the resulting total cost increment is at most $(2n-1)$ for even n, and at most $(2n+1)$ for odd n. Hence, a fault analysis carried onto SCC_n with a permutation π_n formed according to rules **R1** and **R2** results in a fault-diameter of at most:

- For even $n \geq 8$: $d_f(\mathrm{SCC}_n) \leq d_f(\mathrm{SCC}_{n-2}) + 2n - 1$
- For odd $n \geq 7$: $d_f(\mathrm{SCC}_n) \leq d_f(\mathrm{SCC}_{n-2}) + 2n + 1$

Based on the fact that $d_f(\mathrm{SCC}_5) = d(\mathrm{SCC}_5) + 1$ and $d_f(\mathrm{SCC}_6) = d(\mathrm{SCC}_6) + 1$, assume that the hypothesis $d_f(\mathrm{SCC}_{n-2}) = d(\mathrm{SCC}_{n-2}) + 1$ holds. Hence:

- For even $n \geq 8$: $d_f(\mathrm{SCC}_n) \leq d(\mathrm{SCC}_{n-2}) + 1 + 2n - 1 = d(\mathrm{SCC}_n) + 1$
- For odd $n \geq 7$: $d_f(\mathrm{SCC}_n) \leq d(\mathrm{SCC}_{n-2}) + 1 + 2n + 1 = d(\mathrm{SCC}_n) + 1$

By combining the lower bound established in Subsection 3.1 with the above inequality, we prove that $d_f(\mathrm{SCC}_n) = d(\mathrm{SCC}_n) + 1$, for $n \geq 7$. □

4 Conclusion

This paper derived exact values for the fault-diameter of the star-connected cycles (SCC) interconnection network. We showed that the SCC graph is strongly resilient, which is not intuitive due to the utilization of a topology where rings abound. This result can be attributed to the robustness of the star graph, which dominates over the weak resilience of the ring. Such a dominance is made clear through a tabular method of proof, which demonstrates how the richness of paths in the star graph contributes to minimize the impact of faults onto the corresponding SCC graph.

References

[1] F. Harary, *Graph Theory*, Addison-Wesley, 1969, pp. 14, 43.

[2] M.S. Krishnamoorthy and B. Krishnamurthy, "Fault Diameter of Interconnection Networks," *Computers & Mathematics with Applications*, Vol. 13, No. 5/6, 1987, pp. 577-582.

[3] S. B. Akers and B. Krishnamurthy, "A Group-Theoretic Model for Symmetric Interconnection Networks," *Proc. Int'l Conf. on Parallel Processing*, 1986, pp. 216-223.

[4] S. Lakshmivarahan, J-S. Jwo and S. K. Dhall, "Symmetry in Interconnection Networks Based on Cayley Graphs of Permutation Groups: a Survey," *Parallel Computing*, 1993, Vol. 19, pp. 361-407.

[5] S. Latifi, "Combinatorial Analysis of the Fault-Diameter of the n-Cube," *IEEE Transactions on Computers*, 1993, Vol. 42, No. 1, pp. 27-33.

[6] S. B. Akers and B. Krishnamurthy, "The Fault Tolerance of Star Graphs," *Proc. 2nd Int'l. Conf. on Supercomputing*, 1987, pp. 270-276.

[7] S. Latifi, "On the Fault-Diameter of the Star Graph," *Information Processing Letters*, 46 (1993), pp. 143-150.

[8] S. Latifi, M.M. Azevedo and N. Bagherzadeh, "The Star-Connected Cycles: a Fixed-Degree Interconnection Network for Parallel Processing," *Proc. Int'l. Conf. Parallel Processing*, 1993, Vol. 1, pp. 91-95.

[9] S. B. Akers, D. Harel and B. Krishnamurthy, "The Star Graph: An Attractive Alternative to the n-Cube," *Proc. Int'l Conf. on Parallel Processing*, 1987, pp. 393-400.

[10] W. Ledermann, *Introduction to the Theory of Finite Groups*, Oliver and Boyd, London, 1964, pp. 1-3, 62-95.

[11] S. Latifi, "Parallel Dimension Permutations on Star Graph," *IFIP Transactions A: Computer Science and Technology*, 1993, A23, pp. 191-201.

[12] K. Day and A. Tripathi, "A Comparative Study of Topological Properties of Hypercubes and Star Graphs," *IEEE Transactions on Parallel and Distributed Systems*, Vol. 5, No. 1, January 1994, pp. 31-38.