Path Failure in Cylinderical Mesh Networks *†

J. Ramachandran
Computer and Information Sciences
Ohio State University
Columbus, Ohio, 43210

Abstract

The problem of node failures in interconnection networks has been extensively studied. In this paper we define the notion of a more catastrophic form of failure — *path failure* — in which a sequence of nodes from a *source* to a *sink* in the network fail. The corresponding collapse in the structural properties of the underlying network is analyzed in quantitative terms, as a function of network size, by using the notion of *catastrophic failure.* We examine this problem in a fixed setting, that of the *cylinderical mesh*, a network designed to maximize the number of communication routes between a set of *producers* and a set of *consumers.* Our proofs use interesting graph theoretic and combinatorial techniques, and formally establish that connectivity is drastically effected by path failure.

1 Introduction

Fault tolerance in computing, an active and productive area of research in computer science these days, has been approached from a wide variety of perspectives. One viewpoint focuses on the robustness of an interconnection network to failures of switches in the network and the related effect on network connectivity between the components of such a system. The literature on such problems is extensive.

In this paper, however, we study a more catastrophic form of performance loss, the notion of *path failure*, in which a sequence of nodes may drop out of the network, drastically altering it's structural and communications properties. Path failures can be

*This work was supported in part by NSF Grant CCR-8909071
†This work was done when the author was a visitor at New Mexico State University

viewed as caused by a malicious agent whose progress through the network starts from some *source* node and leads to a *sink* node, and has in the process damaged or made unreliable all the nodes visited along the path taken. Path failures therefore can be viewed as a faulty sub-region within a larger network, that causes a partition of the underlying network.

To present concrete evidence of the effects of path failures, we study a particular class of networks that are interesting from two perspectives, firstly as solutions to the problem of connecting m producers to m consumers of a resource, and secondly, as a special class of the much-researched *mesh* architecture. These networks, called *cylinderical meshes*, are amenable to analysis combinatorially and have an elegant relationship with *Fibonacci networks*, first introduced by Martin Gardner in his *Mathematical Circus*, [Gar79], to study the paths of bees in a bee hive formed of hexagonal cells.

Geometrically, these networks are based on Schwarz's polyhedron, which, as described in [Wel91], shows that that, when computing the areas of surfaces through approximation intuition can be misleading and sometimes actually wrong.

Path failures can be viewed as time bounded phenomena in the case of circuit switching networks, where a message sender establishes a dedicated link to a receiver. The switches along the link are unavailable during transmission, thus altering the structure of the network temporarily.

Another useful application of these networks is presented in [Ram92], where these networks were encountered as the underlying structure for an algorithm to approximate *optimization* functions from the structural complexity class OptP, which contains optimal solution search analogs of decision problems in NP.

A cylinderical mesh network has a high degree of connectivity, measured in terms of the number of paths through the network. Our results are essentially negative: If the size of the network exceeds a certain limit, the failure of a sequence of nodes from a consumer to a producer, called a *path failure*, can result in a drastic drop in connectivity and a corresponding loss in performance. More formally: In a cylinderical mesh linking m producers to m consumers through k stages, all path failures reduce the number of reliable routes of communication between producers and consumers from $m \times 2^k$ to less than 2^k.

Our presentation is as follows:

1. We define 2-line networks, a simple form of the mesh architecture, and count the number of paths through a network of two consumers linked through k stages, to 2 producers. We bound this number by the Fibonacci number f_{k+2}; it is for this reason we call such networks *fibonacci networks.*

2. We introduce the cylinderical mesh architecture, a simple variant of the *mesh architecture.* We show that connectivity increases exponentially with the number of stages in the network (our measure of connectivity is the number of paths between producers and consumers). A cylinderical mesh of m producers and consumers with k stages has $m \times 2^k$ paths, thus it is desirable to have a large number of stages in the cylinderical mesh, given that maximizing connectivity between producers and consumers is an important..

3. We generalize the 2-producer-2-consumer fibonacci network to the case of m producers linked to m consumers.

4. We define formally the notion of path failure and present the following result: A path failure in a cylinderical mesh linking $m + 1$ producers to $m + 1$ consumers leaves the remaining network isomorphic to an m-producer-m-consumer fibonacci network. Thus we relate these two graph theoretic structures in an unusual fashion.

5. We now proceed to quantitatively analyze path failure by defining *catastrophic failure* : A cylinderical mesh has *catastrophic failure* if the path count drops from $m \times 2^k$ to less than 2^k as a result of path failure. Our main result is the following negative statement: Although it is desirable to have a high number of stages in a cylinderical mesh to increase the path count (and therefore connectivity) if the number of stages exceeds m^4, any path failure is *catastrophic.*

6. We finally present a long and complicated analysis of a specific path failure that improves the above bound to m^2.

We conjecture that *path failures* in arbitrary networks are invariably catastrophic; namely, they cause serious damage to the structure of the network. Network design principles related to providing fault tolerance may need to consider this issue in greater detail.

2 Some background and definitions

We first introduce two line fibonacci networks and describe their properties.

Definition 2.1 *A two line network $N(y)$ is a directed acyclic graph (V, E) with $|y|+1$ stages, defined by a string $y \in \{0,1\}^*$ as follows. Let $|y| = k$, and let the i^{th} bit of y be y_i.*

$$
\begin{aligned}
V &= \{\langle 0,i\rangle, \langle 1,i\rangle \mid 0 \le i \le k\}, \\
E &= \{(\langle 0,i\rangle, \langle 0,i+1\rangle) \mid 0 \le i \le k\} \cup \\
&\quad \{(\langle 1,i\rangle, \langle 1,i+1\rangle) \mid 0 \le i \le k\} \cup \\
&\quad \{(\langle 0,i\rangle, \langle 1,i+1\rangle) \mid \text{ such that } \ y_{i+1} = 1\} \cup \\
&\quad \{(\langle 1,i\rangle, \langle 0,i+1\rangle) \mid \text{ such that } \ y_{i+1} = 0\}
\end{aligned}
$$

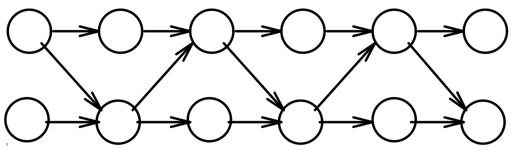


Figure 1: The fibonacci network $N(101010)$

A two line network links two producers to two consumers. The networks *connectivity* is the number of paths connecting producers to consumers. We make the following simplifying assumptions:

1. All networks considered in this paper are planar.

2. Of the several ways in which the two nodes of one stage can hook up with the nodes of the next stage, we restrict our attention to a subset of all possible networks that have a regular structure. This family of regular networks is of size 2^k for $k+1$ stage query networks, and each k bit binary string y uniquely determines a member of

the family, denoted by $N(y)$. We only use the transitions of Figure 2 between adjacent stages because we wish to relate our networks to the mesh network.

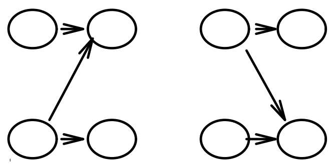


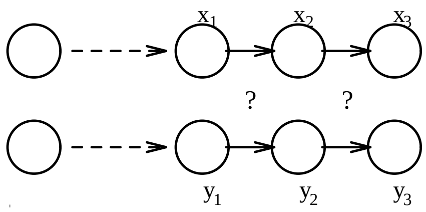
Figure 2: Possible transitions

2.1 Paths through fibonacci networks

We prove an upper bound on the number of paths through a 2-line network. The proof provides both a lower and upper bound and hence is called the *optimality theorem.* We denote the optimal $k+1$ stage network by $N(y_{opt})$ where $y_{opt} = (01)^{k/2}$ if k is even, or $y_{opt} = (01)^{\lfloor k/2 \rfloor}0$ otherwise. Label the producers and consumers p_1, p_2, c_1 and c_2 respectively. Recall that f_k is the k^{th} fibonacci number in the sequence $(1, 1, 2, 3, \ldots)$.

Theorem 2.2 *Let $y \in \{0,1\}^{k-1}$, let $N(y)$ be an arbitrary k-stage network, and let $S : \mathcal{N} \to \mathcal{N}$, where $S(k)$ is the number of paths through N. Then $S(k) \leq f_{k+2}$.*

Proof. We will show that that f_k, which satisfies the recurrence $f_k = f_{k-1} + f_{k-2}$, is an upper bound on $S(k)$. The proof is by induction on k. Note that $S(1) = 2$ and $S(2) \leq 3$ prove the basis. Consider the k stage network in Figure 2.1 below (note that the "?" denotes unknown crossing edges from stage $k-2$ to stage $k-1$, and from stage $k-1$ to stage k).



Let x_i (y_i) denote the number of paths ending at level 0 (1) at the end of stage $k-3+i$. By our induction hypothesis, we have $x_1 + y_1 \leq f_k$ and $x_2 + y_2 \leq f_{k+1}$. We need to show that $x_3 + y_3 \leq f_{k+2}$. Note that $x_2 \leq f_k$ and $y_2 \leq f_k$ as each (but not both simultaneously) could at most be the sum of all the paths at stage $k-2$.

If the transition between stages $k-1$ and k is similar to the first transition in Figure 2, then $x_3 = x_2 + y_2$ and $y_3 = y_2$. So $x_3 \leq f_{k+1}$ and $y_3 \leq f_k$, implying $x_3 + y_3 \leq f_{k+2}$. The proof is analogous if the transition between stages $k-1$ and k is similar to the second transition in Figure 2. Thus the number of paths $S(k)$ through any network is less equal f_{k+2}. $\square$

The above proof provides an upper bound on the number of paths though a k stage 2-line network, and as the fibonacci network $N(y_{opt})$ has exactly f_{k+2} paths, it also is a lower bound. In section 5 we will relate the above discussion to the notion of path failure.

We now describe the cylinderical mesh network.

3 Cylinderical meshes

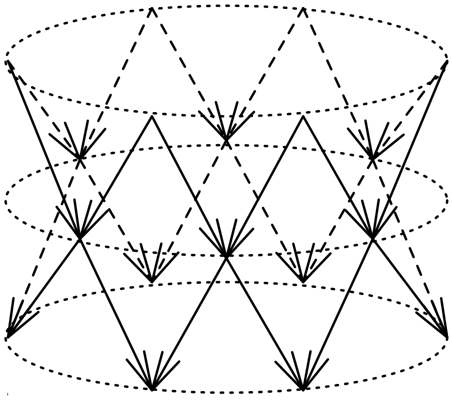
In this section we formally define and analyze cylinderical meshes of circumference m and height k (abbreviated from now on as (m,k)-cylinders). We view such networks as a special case of the mesh network, as they can be built from simple two-inputs-two-outputs switches that can route messages arriving along any input through either output path. The network connects m producers to m consumers. We are assuming that, in the interests of fault tolerance, the important property of such a network is connectivity, which we measure as a function of the number of stages in the network.

Definition 3.1 *An (m,k)-cylinder is a graph $C = (V, E)$ of $m \times (k+1)$ nodes. We call k the* height *of C, and m the* circumference *of C.*

$$\begin{aligned} V &= \{\langle i,j\rangle \mid 0 \leq i \leq m-1,\ 0 \leq j \leq k\}, \\ E &= \{(\langle i,j\rangle, \langle i,j+1\rangle) \mid 0 \leq i \leq m-1,\ 0 \leq j \leq k\} \cup \\ &\quad \{(\langle i,j\rangle, \langle (i+1) \bmod m, j+1\rangle) \mid \\ &\qquad 0 \leq i \leq m-1,\ 0 \leq j \leq k\} \end{aligned}$$

We describe C visually as follows:

Example 3.2 A (6,3)-cylinder.



□

1. Stack $k+1$ identical circles vertically. Place m nodes at equidistant intervals on each circle. Label the circles 0 to k from the top downwards.

2. The m nodes of level j form an independent set. Level 0 nodes are producers; level k nodes are consumers.

3. The nodes of level j, $0 \leq j < k$, are connected to those of level $j+1$ by $2m$ edges. Each node is connected to two nodes on the next level.

Definition 3.3 *A* path *through an (m,k)-cylinder C is a pair (i,y), for some $i \leq m$ and $y \in \{0,1\}^{=k}$.*

A path is a sequence of $k+1$ nodes, beginning from the top level of the cylinder and ending at the bottom. The path (i,y) starts from the producer $\langle 0,i\rangle$, and proceeds from this node, using the bits of y to choose a path, to arrive at some consumer $\langle k,j\rangle$ as follows: If the l^{th} bit of y is a 1, we pick the right edge of the two directed edges out of the l^{th} stage of the path, and pick the left edge otherwise.

Proposition 3.4 *An (m,k)-cylinder has exactly $m \times 2^k$ paths through it.*

We now capture the notion of path failure:

Definition 3.5 *A* cut *through an (m,k)-cylinder is a specific path (i,y).*

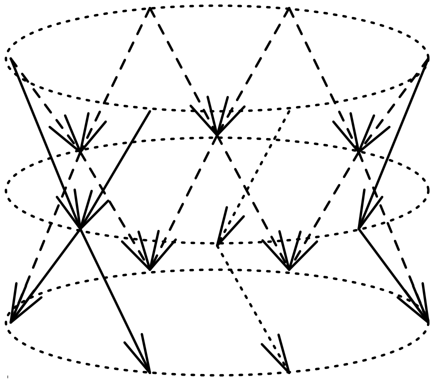
A cylinder is said to be *cut open* if all the nodes from producer i to some consumer j, reached along the path (i, y), fail. The producer may be faulty, or may corrupt the switches along the cut, so that messages passing through these switches cannot be relied upon to be propagated correctly. We say that the path from producer to consumer is *damaged*. As a result, all other producers and consumers delete all edges into the switches on this path. We also remove the node representing the producer, to simulate the unreliability of a message from this producer, and the node representing the consumer because it is unproductive to produce messages for a faulty consumer.

As a result, we have drastically altered the structural properties of the underlying mesh, and have changed its connectivity considerably.

For any cut (i, y), we say other path (j, z) *crosses* (i, y) if there exists a substring of z, $w = w_1 \ldots w_n$ such that the nodes corresponding to these bits have the following properties: (1) The first and last node of the sequence of n nodes on (j, z) corresponding to w do not lie on the path (cut) (i, y), (2) all the intermediate nodes of w lie on (i, y), and (3) the first and last nodes of w lie on "opposite" sides of (i, y). Intuitively, if (i, y) is a cut then a path (j, z) that crosses it runs through it from left to right, or right to left, at least once. When the path (i, y) fails, all such crossing paths (j, z) are lost and producers and consumers have fewer options for paths of communication.

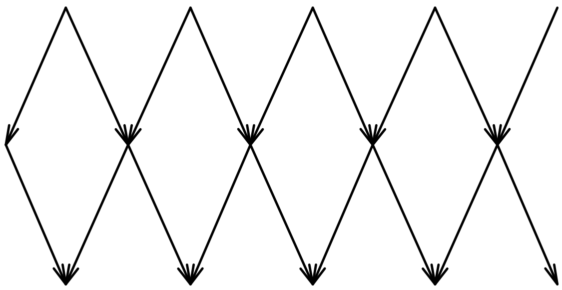
Definition 3.6 *The operation* Cut *takes as input three arguments, an (m, k)-cylinder C, a producer's ID say m and a bitstring y of length k and generates a graph Cut(C, y) as follows: Pick an arbitrary node, say m. Let the nodes on path (m, y) be $v_0, v_1, \ldots, v_k$. Delete all these nodes from the cylinder and all the edges incident upon them.*

Example 3.7 The (6,3)-cylinder of the last figure cut along the bitstring 01. Note: the dotted line denotes the path along which we cut the cylinder.



□

Example 3.8 The corresponding network, formed by "unwrapping" the cylinder.



□

A cylinderical mesh C can experience many different degrees of path failure, one for each choice of a cut. In the following section we describe a graph theoretic structure that condenses the family of resultant networks $Cut(C, y)$ into a single framework, easing quantitative analysis of the effects of cuts on cylinderical meshes.

4 On m-line networks

In this section we define and analyze m-line networks.

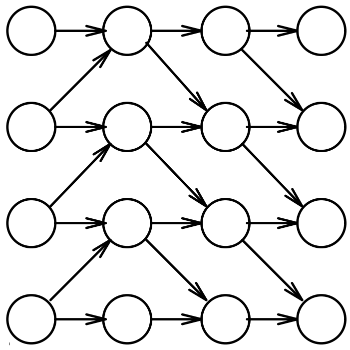
Definition 4.1 *An m-line network $N(m, y)$ is a directed acyclic graph (V, E) with $|y|+1$ stages, defined by a string y as follows. Let $|y| = k$, and let the i^{th} bit of y be y_i.*

$$\begin{aligned} V &= \{\langle i, j\rangle \mid 0 \le i \le m-1,\ 0 \le j \le k\}, \\ E &= \{(\langle i, j\rangle, \langle i, j+1\rangle) \mid\ 0 \le i \le m-1,\ 0 \le j \le k\} \cup \end{aligned}$$

$$\{(\langle i,j\rangle,\langle i+1,j+1\rangle)\mid$$
$$0\le i\le m-2,\ \forall j \text{ s.t } y_{j+1}=1\}\cup$$
$$\{(\langle i,j\rangle,\langle i-1,j+1\rangle)\mid$$
$$1\le i\le m-1,\ \forall j \text{ s.t } y_{j+1}=0\}$$

Intuitively we draw $N(m,y)$ as a sequence of $|y|+1$ stages from left to right, each stage of at most m nodes. Each stage is said to have m levels.

Example 4.2 *The 4-line network $N(4,011)$. The transition between adjacent stages corresponding to $y_i=0$ is called an* up *stage; likewise, the transition corresponding to $y_i=1$ is called a* down *stage.*



□

In the next section we prove our first result: Path failures in cylinderical meshes of circumference m reduce the cylinderical mesh network to an $m-1$-line fibonacci network. We therefore reduce the problem of analyzing path failures in cylinderical meshes, to that of counting the number of paths in an $m-1$ line network.

5 Relating cylinderical meshes to m-line networks

There are 2^k different $(m-1)$-line $k+1$ stage networks $N(m-1,y)$, one for each bitstring $y\in\{0,1\}^{=k}$. We now show that the operation $Cut(C,y)$ creates a network $N(m-1,y)$.

Theorem 5.1 *Let $|y|=k$, and let C be an (m,k)-cylinder. Then the network $Cut(C,y)$ is isomorphic to $N(m-1,y)$.*

Proof. We need only verify that cutting C along (m, y) maintains the property that, for all $i < k$, the subgraph between levels i and $i+1$ in C is isomorphic to the subgraph between stages i to $i+1$ in $N(m-1, y)$.

The number of nodes in each level of $Cut(C, y)$ matches that of each stage in $N(m-1, y)$, namely $m-1$. It is straightforward to note that the operation of cutting C along any cut removes one of the $2m$ edges and leaves the remaining edges untouched. The resulting $2m-1$ edges are isomorphic to an "up" stage if the i^{th} bit of y is a 0, and is isomorphic to a "down" stage if the i^{th} bit of y is a 1. □

Having established the above theorem, the following corollary follows easily.

Corollary 5.2 *Every $(m-1)$-line k stage network can be obtained by cutting an (m, k)-cylinder along an appropriately chosen cut.*

We have, thus, established a bijection between the set of all networks, and the range of the *Cut* operation on cylinders.

Recall the following familiar combinatorial structure:

Definition 5.3 *Pascal's triangle is the ordering of all binomial coefficients into a triangle whose m^{th} level consist of the $m+1$ coefficients: $\binom{m}{0}, \ldots, \binom{m}{m}$.*

Definition 5.4 *An (m, k) Pascal Triangle (an (m, k)-PT) is a graph (V, E) with $k+1$ levels, $m+i$ nodes at level i connected, mesh-like to the $m+i+1$ nodes of the next level.*

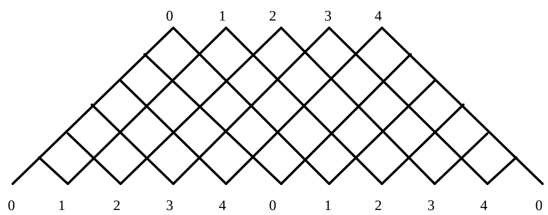


Figure 3: Five Pascal's triangles "overlaid" on each other, a (5,6)-PT

Consider the following alternate view of a cylinderical mesh: The paths through the above graph are isomorphic to the paths through a $(5, 6)$-cylinder; both have 5 producer nodes at the top, from which there are 2^6 possible paths. The numbers at the bottom of the above figure are matched with the 5 consumers.

When we cut a $(6,6)$-cylinder (recall that that leaves us 5 producers and 5 consumers), along a path, say $(0, 010111)$ (i.e. start at node 0, and follow the path formed by the bits of 010111), by Lemma 5.1, the resulting graph (shown between the heavy lines in Figure 4) is isomorphic to $N(5, 010111)$:

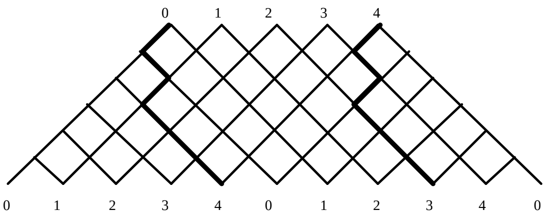


Figure 4: Embedding $N(5, 010111)$ in a 5-PT

We show a one to one correspondence between some m coefficients in the k^{th} row of Pascal's triangle, and the number of paths originating from any single producer, and terminating at a fixed consumer.

Lemma 5.5 *We can embed the structure of an arbitrary $k+1$ stage m-line network $N(m, y)$, into an $(m, |y|)$-PT.*

Proof. Assume that the consumers correspond to the top m nodes of the (m, k)-PT. Inductively, assume that some m consecutive nodes in the m-PT correspond to the nodes of N's $(i-1)^{\text{th}}$ stage. Then, if y_i, the i^{th} bit of y, is a 0 choose all the left-hand children of the m nodes in the m-PT, otherwise choose the m right-hand children.

Continuing in a similar fashion, we can always find a one to one correspondence between the nodes of the network and the nodes of the m-PT, and induce a correspondence between the edges of the network and the edges of the m-PT. $\square$

Thus every path through the network $Cut(C, y)$ corresponds to a unique path through an (m, k)-PT. This indicates that we can bound from above the number of paths ending at each of the nodes, at the last stage of the network, by the the number of paths from the top m nodes to the bottom m nodes reached, in the last level (level $k+1$) respectively.

(By the embedding process of Lemma 5.5 that embeds $Cut(C, y)$ into an (m, k)-Pascal's triangle). These values are the sums of at most m arbitrary binomial coefficients $\binom{k}{k'}$. From this we deduce that the total number of paths in the network is dominated by m times the sum of these m coefficients. We now show that this upper bound grows slower than 2^k, for all choices of m, for $k = \mathcal{O}(m^4)$. This establishes an upper bound on the path count through an arbitrary m-line network, with at least this many stages.

Lemma 5.6 *For any fixed choice of m, if B is the largest binomial coefficient in row k of Pascal's triangle, for sufficiently large k, $m^4 \times B < 2^k$.*

Proof. Without loss of generality assume k is even, say $k = 2k'$. The largest coefficient in the k^{th} row of Pascal's triangle is the value $\binom{k}{k'}$. The number of paths in any m-line k-stage network is bounded by

$$m^2 \times \binom{k}{k'} = m^2 \times \frac{(k)!}{(k')!(k')!} = m^2 \times 2^k/\sqrt{2 \times \pi \times k'}$$

(using Stirlings's approximation for $n!$). If this number is less than 2^k then $m < \times\sqrt[4]{2\pi k}$. Thus a choice of k of the order of $\mathcal{O}(m^4)$ suffices. [Note: The fraction $k!/((k'!)^2 \times 2^k)$, when viewed as defining an infinite product, diverges to zero as k tends to infinity. Thus we can make the fraction as small as we wish; we again use Stirling's approximation for $n!$ to show this.] $\square$

We can now state the main result of this section: catastrophic failure is inevitable in large cylinderical meshes, if a path fails.

Theorem 5.7 *Let C be a cylinderical mesh connecting* m *producers to* m *consumers. If the number of stages in the network is greater than m^4, then any arbitrary path failure in C causes the path count to drop from $m \times 2^k$ to less than 2^k.*

As we measure the fault tolerance of a cylinderical mesh as a function of the number of stages from producer to consumer, we can state:

Corollary 5.8 *The path fault tolerance of a cylinderical mesh is inversely proportional to the number of stages in the mesh.*

In the next section we proceed to show an improvement to this loose bound of m^4 by analyzing a specific network, which intuitively has more paths than any other m-line network, with an equal number of states. This network is $N(m, y_{opt})$ has the *best* case property of preserving more paths than from any other path failure. This follows by the observation that embedding this network into an $(m, |y_{opt}|)$-PT reaches the largest values at the bottom of the m-PT.

6 Upper bounds on best case failure

In this section we study a specific m-line k'-stage network, and bound, from above, the number of paths through the network. Without loss of generality we assume that m and $k' = 2k + 1$ is odd. Let k_{limit} denote the least number of stages of the network at the point when any path failure is catastrophic; the point where the path count drops from $m2^k$ to less than 2^k. We present a technical analysis of this special network, leading to the following stronger statement, showing the negative effects of path failure happen earlier than predicted by Theorem 5.7:

Theorem 6.1 *If an arbitrary path fails in a cylinderical mesh connecting* m *producers to* m *consumers, where the number of stages in the network is greater than m^2, the path count drops from $m \times 2^k$ to less than 2^k.*

The specific network chosen is the m stage generalization of the 2-stage network studied in Lemma 2.2, and is of independent interest as a generalization of Martin Gardner's [Gar79] description of a bee's walk through a hexagonal grid.

Definition 6.2 *For all $k \geq 1$ the string $y_{opt}(k)$ is $(01)^{k \text{ div } 2}$ if k is even and is the string $(01)^{k \text{ div } 2}0$ if k is odd.*

We state, without proof, that the optimality result for the 2-line network also holds for the case of m-line network: Of all m line networks $N(m, y)$, none has more paths from producers to consumers than $N(m, y_{opt})$.

Let $B(m)$ be the m line query network $N(m, 01)$. The network $B(m)$ defined above is called a *building block stage*. We build the special network we are interested in, by connecting sequences of building blocks together.

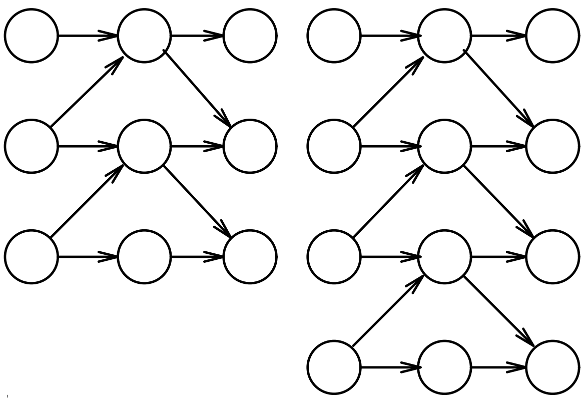
Definition 6.3 *Let $y \in \{0,1\}^{=k}$. A* path *through $N(m,y)$ is a list of $k+1$ nodes, one from each stage, such that each is connected to the next via an edge. The* path matrix *for $N(m,y)$ is a $m \times m$ matrix $M(m,y) = (a_{ij})$ where, for $1 \leq i,j \leq m$, a_{ij} is the number of paths from level i in stage 0 of the network to level j in stage $k+1$ (connectivity between consumer i and producer j). Let $S(m,y) = \sum_{i=0}^{m-1} \sum_{j=0}^{m-1} a_{ij}$ to be the total number of paths through $N(m,y)$.*

Given two networks $N(m,y_1)$ and $N(m,y_2)$, we can *connect* them by overlapping the last stage of $N(m,y_1)$ with the first stage of $N(m,y_2)$. (This situation would occur if the consumers themselves were producers of a resource for m new secondary consumers). The resultant network is formed by merging these two stages into a single new stage and all incoming edges from $N(m,y_1)$'s penultimate stage come into this stage and all outgoing edges of $N(m,y_2)$'s first stage start from this stage. Thus the resultant network $N(m,y_1y_2)$ has $(|y_1|+|y_2|)+1$ stages.

Proposition 6.4 $\forall m, y_1, y_2,\ M(m,y_1y_2) = M(m,y_1) \times M(m,y_2)$.

We will count the number of paths in this particular network by summing the elements of its *path matrix.* If the network has m lines and $2k+1$ stages, then the network is denoted by N_k^m, it's path matrix is denoted by M_k^m, and the sum of elements of M_k^m is denoted by S_k^m.

Example 6.5 For $m = 3$ and 4, $B(3)$ and $B(4)$ are as below along with their associated path matrix



$$M_1^3 = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 2 & 1 \\ 0 & 1 & 2 \end{pmatrix} \text{ and } M_1^4 = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & 2 & 1 & 0 \\ 0 & 1 & 2 & 1 \\ 0 & 0 & 1 & 2 \end{pmatrix}$$

□

In general $M_1^m = (a_{ij})$ has the following structure.

1. The diagonal from element a_{00} to $a_{m-1,m-1}$, which we will call the *left major diagonal*, is composed of one "1" at a_{00} and "2"s everywhere else.

2. The elements above and below the left major diagonal (those running from a_{01} to $a_{m-2,m-1}$, and from a_{10} to $a_{m-1,m-2}$ respectively) are composed of "1"s.

3. All the remaining elements are zeros.

M_k^m is symmetric. The recursive definition of N_k^m yields the following:

Proposition 6.6 $M_k^m = M_{k-1}^m \times M_1^m$

We thus have a method of computing M_k^m iteratively. However constructing a similar recurrence for S_k^m, from which we may derive a bound on it's value, is not so easy. Our strategy for doing so is the following:

1. By symmetry, we need only compute the upper triangle (above the left major diagonal) of M_k^m.

2. Establish a *self referential* description for M_k^m. By this we mean that, independent of k, every element not in the first row of M_k^m can be expressed as some linear combination of the elements $\{a_{00}, a_{01}, \ldots, a_{0,m-1}\}$ of the the first row of M_k^m.

3. More concretely, define two functions *Start* and *Finish* such that, for each i and j:
$$a_{ij} = \sum_{n=Start(i,j)}^{Finish(i,j)} a_{0n}.$$
We do so by induction on the number of blocks k joined together; i.e.: We show the above relationship holds for M_1^m, then assume that it holds for M_{k-1}^m and then show that $M_k^m = M_{k-1}^m M_1^m$ has the same structure.

4. Estimate the total number of paths S_k^m by summing up the elements of M_k^m, using the self referential property to "ignore" k.

5. Build a recurrence for the total number of paths S_k^m in N_k^m from the relationship between $M_k^m = M_{k-1}^m M_1^m$ and other properties of the self referential structure of the two matrices.

6. Show that, in the limit, as k tends to infinity, S_k^m, for all m, grows slower than 2^{2k+1}, and from this show that for any m, the point at which it happens $(2k+1)$ is of the order of $\mathcal{O}(m^2)$.

To help relate the element $a_{ij}, i \neq 0$ and $j \neq 0$ to the elements of the first row of the matrix $a_{00}, a_{01} \ldots a_{0,m-1}$, consider the following examples

Example 6.7 The matrices M_2^3, M_3^3, M_4^3, and M_5^3 are respectively equal to:

$$\begin{pmatrix} 2 & 3 & 1 \\ 3 & 6 & 4 \\ 1 & 4 & 5 \end{pmatrix}, \begin{pmatrix} 5 & 9 & 5 \\ 9 & 19 & 14 \\ 5 & 14 & 14 \end{pmatrix},$$

$$\begin{pmatrix} 14 & 28 & 19 \\ 28 & 61 & 47 \\ 19 & 47 & 42 \end{pmatrix}, \begin{pmatrix} 42 & 89 & 66 \\ 89 & 197 & 155 \\ 66 & 155 & 131 \end{pmatrix}$$

Note that for any arbitrary k, M_k^3 has the following structure:

$$M_k^3 = \begin{pmatrix} a & b & c \\ b & a+b+c & b+c \\ c & b+c & a+b \end{pmatrix}$$

□

Example 6.8 It can be similarly observed that M_k^4 can be expressed self referentially as:

$$M_k^4 = \begin{pmatrix} a & b & c & d \\ b & a+b+c & b+c+d & c+d \\ c & b+c+d & a+b+c+d & b+c \\ d & c+d & b+c & a+b \end{pmatrix}$$

□

In the following we assume we are looking at m-line networks and therefore will drop the superscript m.

Consider the following functions on a pair of indices (i, j)

Definition 6.9 *Let Start*(i,j) *to be the following algorithm:*

begin

Input two integers i and j, $j \geq i$.

Return $j - i$

end

Definition 6.10 *Let Finish*(i,j) *to be the following algorithm:*

begin

Input two integers i and j, $j \geq i$.

if $i + j < m - 1$ **then**

return $i + j$

else

return $2m - 1 - (i + j)$

end$(*$ **if** $*)$

end

We will use the above functions to define the $StartFinish$ matrix:

Definition 6.11 *The* $m \times m$ *matrix* $SF_m = (p_{ij})$ *is a matrix of ordered pairs such that* $p_{ij} = (Start(i,j), Finish(i,j))$.

Example 6.12 For the 8-line network, the corresponding start-finish matrix is as below:

$$\begin{pmatrix} 0,0 & 1,1 & 2,2 & 3,3 & 4,4 & 5,5 & 6,6 & 7,7 \\ 1,1 & 0,2 & 1,3 & 2,4 & 3,5 & 4,6 & 5,7 & 6,7 \\ 2,2 & 1,3 & 0,4 & 1,5 & 2,6 & 3,7 & 4,7 & 5,6 \\ 3,3 & 2,4 & 1,5 & 0,6 & 1,7 & 2,7 & 3,6 & 4,5 \\ 4,4 & 3,5 & 2,6 & 1,7 & 0,7 & 1,6 & 2,5 & 3,4 \\ 5,5 & 4,6 & 3,7 & 2,7 & 1,6 & 0,5 & 1,4 & 2,3 \\ 6,6 & 5,7 & 4,7 & 3,6 & 2,5 & 1,4 & 0,3 & 1,2 \\ 7,7 & 6,7 & 5,6 & 4,5 & 3,4 & 2,3 & 1,2 & 0,1 \end{pmatrix}$$

□

We now will prove some properties of the *Start* and *Finish* functions.

Consider the ordered pairs p_{ij} from the upper triangle of the matrix SF_m (inclusive of the left major diagonal). To help describe the parts of the matrix SF_m, (1) analogous to the *left major* diagonal, let the *right major diagonal* consist of the elements $p_{i,m-1-i}$ for all i. (2) Let the r^{th} upper left diagonal (abbreviated r-uld) consist of the elements $p_{i,i+r}$ for all r. (3) Let *region 1* be the elements p_{ij} that are strictly above the right major diagonal, and (5) let *region 2* be the elements p_{ij} that are on or below the right major diagonal.

Lemma 6.13 *The following are properties of the two algorithms, Start and Finish, and therefore of SF_m:*

1. *If $i \neq 0$ then Start$(i,j) <$ Finish(i,j)*

2. *Every element of the r^{th} upper left diagonal of SF_m has r as it's first component.*

3. *There are $m-r$ elements in the r^{th} upper left diagonal of SF_m. Every number from r to $m-1$ occurs once and only once in the set of second components of each of these ordered pairs.*

4. *The mapping $(i,j) \to ($Start$(i,j), \mathrm{Finish}(i,j))$ is a bijection on such pairs. Thus this mapping is invertible.*

Proof.

1. This follows from the fact that if we are in region 1 then $\mathrm{Start}(i,j) < \mathrm{Finish}(i,j)$ as $j-i < j+i$, and if we are in region 2, then

$$\begin{aligned}&\mathrm{Finish}(i,j) - \mathrm{Start}(i,j)\\ &\quad = \quad 2m-1-(i+j)-(j-i)\\ &\quad = \quad 2m-1-2j > 0.\end{aligned}$$

2. This follows from the definition, if $p_{ij} = (x,y) \in SF_m$ then $x = j-i$ but for the r^{th} upper left diagonal, $j-i$ is always r.

3. Recall that m is assumed odd. Assume that r is odd (the case when r is even is similar). The elements on the r^{th} upper left diagonal are

$$\{p_{0,r}, p_{1,r+1}, p_{2,r+2}, \ldots, p_{i,i+r}, \ldots, p_{m-r-1,m-1}\}$$

We split this list into two at its center element $p_{\frac{m-1-r}{2},\frac{m-1+r}{2}}$ on the right major diagonal. The Finish values for the first half of these elements, from left to right (all belong in Region 1 by virtue of being above the right major diagonal) are $Finish(i,j) = i + j$ implying that the list is

$$\{r, r+2, r+4, \ldots, m-1\}$$

The Finish values for the second half are $Finish(i,j) = 2m - 1 - (i+j)$, from left to right (in Region 2), implying that the list is

$$\{m-2, m-4, \ldots, r+3, r+1\}$$

We thus see that the two sets together cover the set $\{r, r+1, \ldots, m-1\}$.

4. This follows from the fact that every r-uld has the same start value and within a diagonal the finish values are all different (the proof above), and from the fact that the range of the Start and Finish functions is between 0 and $m-1$.

□

Theorem 6.14 *Let $m \in \mathcal{N}$. Let M_k be the path matrix of the $(2k+1)$ stage network N_k obtained by appending k copies of the network B together. Then, for all $0 < i \leq j \leq m-1$ the following holds (note we are only describing the upper triangle of the matrix).*

$$a_{ij} = \sum_{n=Start(i,j)}^{Finish(i,j)} a_{0n}$$

Proof. We omit the proof for conciseness, but sketch a proof by induction on the number of stages instead. We divide the matrix M_k into seven regions on the basis of the relative behavior of the Start and Finish functions in each of these. Three of these regions are on the left major diagonal, the first half of the m elements, the middle element, and the second half of the m elements (recall m is odd). The remaining four regions are the elements strictly above both the left and right major diagonals, the elements on the right major diagonal, the elements on the diagonal directly beneath the right major diagonal, and finally the remaining elements of M_k. In each case we show that the relation holds for any element of that region. $\square$

We now proceed to analyze the size of S_k, the sum of the elements of M_k. We now use the self-referential nature of M_k (every element can be expressed as the sum of some consecutive sequence of elements out of the first row) to express S_k as a linear function of *only* the elements of the first row. Rename the elements of the first row by dropping the first subscript, i.e. a_{0i} is now a_i.

Lemma 6.15 *For all $m > 1$,*

$$S_k^m = \sum_{i=0}^{m-1} (m-i)(2i+1) \times a_i$$

Proof. From Lemma 6.13 .3 we have that every element p_{ij} of the r^{th} upper left diagonal of SF has the same first component (equal to $\text{Start}(i,j)$) and the set of second component values (all the values of $\text{Finish}(i,j)$ on r-uld) is the set $\{r, r+1, \ldots, m-1\}$. The coefficient of a_i in the above summation is the number of times a_i appears in SF_m. We can estimate this by counting the number of elements of SF_m, $p = (s, f)$ such that $s \leq i \leq f$. Note that this element a_i appears in $2i+1$ left diagonals about the left major diagonal, and in each appears $m - i$ times. The lemma follows. $\square$

Let C_i be the coefficient of a_i in the previous lemma; i.e. $C_i = (m-i)(2i+1)$. We now possess all we need to build our recurrence inequality.

Lemma 6.16 $4S_{k-1} > S_k$.

Proof. Let $\langle \ldots, a_i, \ldots \rangle$ be the elements of the first row of M_{k-1}, by Lemma 6.15

$$S_{k-1} = C_0 a_0 + C_1 a_1 + \cdots + C_{m-1} a_{m-1}$$

By symmetry, let $\langle \ldots, a'_i, \ldots \rangle$ be the elements of the first row of M_k. By Lemma 6.15

$$S_k = C_0 a'_0 + C_1 a'_1 + \cdots + C_{m-1} a'_{m-1}$$

From the fact that $M_{k-1} \times M_1 = M_k$ observe that the elements of the first row of M_k depend in a simple fashion on those of M_{k-1}, i.e.

$$S_k = C'_0 a_0 + C'_1 a_1 + \cdots + C'_{m-1} a_{m-1}$$

We establish the lemma by establishing $4 \times C_i > C'_i$, from which the recurrence follows.

1. $a'_0 = a_0 + a_1$. Therefore
$$\begin{aligned} C'_0 &= C_0 + C_1 \\ &= m + 3(m-1) \\ &= 4m - 3 \end{aligned}$$
and so $C'_0 < 4m$ which equals $4 \times C_0$.

2. $\forall\, i, 1 \leq i \leq m-2, a'_i = a_{i-1} + 2a_i + a_{i+1}$. Therefore
$$\begin{aligned} C'_i &= C_{i-1} + 2C_i + C_{i+1} \\ &= (m-i+1)(2i-1) + 2(m-i)(2i+1) \\ &\quad +(m-i-1)(2i+3) \\ &= 8mi + 4m - 8i^2 - 4i - 4 \end{aligned}$$
and so $C'_i < 8mi + 4m - 8i^2 - 4i$ which equals $4 \times C_i$.

3. $a'_{m-1} = a_{m-2} + 2a_{m-1}$. Therefore
$$\begin{aligned} C'_{m-1} &= C_{m-2} + 2C_{m-1} \\ &= (2m-3)2 + 2(2m-1) \\ &= 8m - 8 \end{aligned}$$
and so $C'_{m-1} < 8m - 4$ which equals $4 \times C_{m-1}$.

□

Corollary 6.17 $\lim_{k\to\infty}(\frac{S_k}{4^k}) = 0$

Proof. Note in the proof of Lemma 6.16 that $\forall i, 0 \leq i \leq m-1$, we actually show that $4 \times C_i - C'_i \geq 3$. We may therefore conclude that

$$4 \times S_{k-1} - S_k \geq 3(a_0 + a_1 + \cdots + a_{m-1})$$

where, as before a_i is the element in the i^{th} column of the first row of M_{k-1}. Therefore:

$$S_{k-1} \times (4 - \frac{3 \times \sum_{i=0}^{m-1} a_i}{S_{k-1}}) \geq S_k$$

For a fixed m the coefficients of Lemma 6.15 are also fixed, therefore let big be an index such that for all i, $C_i \leq C_{big}$ then

$$S_{k-1} = \sum_{i=0}^{m-1} C_i a_i < C_{big} \times \sum_{i=0}^{m-1} a_i$$

Therefore

$$\begin{aligned} & S_{k-1} \times \left(4 - \frac{3 \times (\sum_{i=0}^{m-1} a_i)}{C_{big} \times (\sum_{i=0}^{m-1} a_i)}\right) \\ > \; & S_{k-1} \times \left(4 - \frac{3 \times (\sum_{i=0}^{m-1}}{S_{k-1}}\right) \end{aligned}$$

implying that

$$S_{k-1} \times \left(4 - \frac{3}{C_{big}}\right) > S_k \Rightarrow S_0 \times \left(4 - \frac{3}{C_{big}}\right)^k > S_k$$

from which, after substituting $S_0 = m$ and dividing both sides by 4^k, we get

$$m \times \frac{(4 - \frac{3}{C_{big}})^k}{4^k} = m \times \left(1 - \frac{3}{4 \times C_{big}}\right)^k > \frac{S_k}{4^k}$$

But the LHS term is of the form $m \times r^k$ where $r < 1$. Therefore as $k \to \infty$, the given ratio tends to 0. $\square$

We now proceed to prove that k_{limit} is quadratic in m.

Lemma 6.18 $k_{limit} = \mathcal{O}(m^2)$.

Proof. From the proof of Lemma 6.16 and the succeeding corollary we have that (using $k = k_{limit}$)

$$m \times \frac{(4 - \frac{3}{C_{big}})^k}{4^k} = m \times \left(1 - \frac{3}{4 \times C_{big}}\right)^k < \frac{1}{2}$$

Using Bernoulli's inequality $(1 + x)^n > (1 + nx)$ we have that,

$$2m \times (1 - 3k/C_{big}) < 1$$

It is easy to see $C_{big} = (2m + 1)^2/8$ from the definition of C_i. We can use this in the above to see that $k_{limit} = \mathcal{O}(m^2)$. $\square$

7 Acknowledgements

We thank Tim Long, Ming Sheu, J. Ramanujam, and Mahendra Ramachandran for comments on many of the results in this paper.

References

[BDG88] J. Balcázar, J. Diaz, and J. Gabarró. *Structural Complexity I.* Springer-Verlag, Berlin, 1988.

[Ben65] V. E. Beneŝ. *Mathematical Theory of Connecting Networks.* Academic Press, New York, 1965.

[Gar79] M. Gardner. *Mathematical Circus.* Alfred A. Knopf, New York, 1979.

[Ogi72] C. Stanley Ogilvy. *Tomorrow's Math.* Oxford Univ. Press, New York, 1972. 2nd edition.

[Ram92] J. Ramachandran. Set bit enumeration is hard. In *Proceedings, Seventh Annual Structure in Complexity Theory*, pages 58–70, 1992. IEEE Computer Society Press.

[Sch87] M. Schwartz. *Telecommunication Networks: Protocals, Modeling and Analysis.* Addison-Wesley Pub. Co., Reading, MA, 1987.

[Wel91] David Wells. *The Penguin Book of Curious and Interesting Geometry.* Penguin Books, Toronto, 1991.