

THESIS





This is to certify that the

dissertation entitled

### DESIGNING DISTANCE-PRESERVING FAULT-TOLERANT TOPOLOGIES

presented by

SITARAMA SWAMY KOCHERLAKOTA

has been accepted towards fulfillment of the requirements for

PhD degree in Philosophy

<u>Major professor</u>

Date\_\_\_\_\_12/6/96

MSU is an Affirmative Action/Equal Opportunity Institution

0-12771

# LIBRARY Michigan State University

PLACE IN RETURN BOX to remove this checkout from your record. TO AVOID FINES return on or before date due.

DATE DUE	DATE DUE	DATE DUE

MSU is An Affirmative Action/Equal Opportunity Institution ctoircidinadus.pm3-p.1

## **DESIGNING DISTANCE-PRESERVING FAULT-TOLERANT TOPOLOGIES**

By

Sitarama Swamy Kocherlakota

### A DISSEERTATION

Submitted to Michigan State University in partial fulfillment of the requirements for the degree of

### DOCTOR OF PHILOSOPHY

Department of Computer Science

December 6, 1996

### Abstract

#### DESIGNING DISTANCE-PRESERVING FAULT-TOLERANT TOPOLOGIES

### By

### Sitarama Swamy Kocherlakota

The objective of designing a *fault-tolerant communication network* (CN) is to maintain its *functionality* in the presence of failures. The approaches taken in designing fault-tolerant systems can be classified, in general, by the functionality criteria. There are two *functionality criteria* that have received much attention. First, *preserving a topology*, which considers a CN functional as long as a desired topology is contained in the system. Second, *preserving a property*, which considers a CN functional as long as a given topological property is satisfied by the underlying topology. In this dissertation, we study the second functionality criterion. Specifically, designing fault-tolerant topologies that preserve *distance* between every pair of non-adjacent nodes in presence of edge failures.

We introduce and study a new family of *fault-tolerant* graphs called *distance preserving graphs*. A graph G is said to be *k-edge distance preserving* with respect to a spanning subgraph D, if there exist k edge-disjoint u-v paths in G of length at most  $d_D(u, v)$  for every pair of non-adjacent vertices u, v of D. We present few properties of distance preserving graphs along with a theorem that characterizes the *distance preserving* graphs. We study two design models each with a different optimality criterion. In the first model, minimizing the overall redundancy is considered. The second model considers regular fault-tolerant topologies with minimum regularity. We present tight lower and upper bounds for both design models. The focus of this dissertation is on designs based on the second model. In particular, we construct regular *distance preserving graphs* with respect to cycles, crowns, chordal rings, and n-ary 2-cubes. We also prove that the given constructions are optimal when  $D = C_p$  and  $k \leq \frac{p}{2}$ .

We also address the issue of the distance between adjacent vertices in D in presence of edge failures, and show that the distance between adjacent vertices is bounded by a constant. Finally, given a graph G and a spanning subgraph D, we have showen that there exist a polynomial time algorithm to determine if G is k-edge distance preserving with respect to D. This study will result in more cost-effective fault-tolerant systems where distance between communicating nodes is critical. © Copyright 1996 by Sitarama Swamy Kocherlakota

All Rights Reserved

## TABLE OF CONTENTS

LIST OF TABLES	vii
LIST OF FIGURES	viii
1 Introduction	1
1.1 Functionality Criteria	3
1.2 Terminology and Notation	4
1.3 Organization of the Dissertation	5
2 Measures of Fault Tolerance	6
2.1 Preserving a Desired Topology	. 7
2.2. Preserving a Property	
2.2 Preserving Connectedness	13
24 Distance	15
241 Preserving Diameter	15
24.2 Preserving Distance	13
2.5     Proposed Work	18
3 Distance Preserving Graphs	20
3.1 Introduction	20
3.2 Preliminaries	21
3.3 Characterization Result	21
3.4 Model 1	25
3.4.1 Bounds	
3.5 Model 2	
3.5.1 Bounds	
3.6 A Bound on the Diameter of $G$	35
4 Special Topologies	40
4.1 Cycles	42
4.2 Crowns	48
4.3 Chordal Graphs	
4.4 <i>n</i> -ary 2-cube	57
5 An Algorithm To Find k Given $D$ and $G$	67
5.1 Outline of the Algorithm	68
6 Conclusions	77
6.1 Future Work	78

## LIST OF TABLES

4.1	k edge-disjoint paths to vertices that are distance 2-away from $v_0$ .	55
4.2	k edge-disjoint paths to vertices that are distance 3-away from $v_0$ .	56
4.3	k-edge disjoint paths to vertices that distance two away from $(i, j)$	62
4.4	k-edge disjoint paths to vertices that are distance 3 away from $(i, j)$ .	66

### LIST OF FIGURES

2.1	Taxonomy of topological fault-tolerance.	8
3.1	$G \in \mathcal{F}_3(D)$	20
3.2	Example: $k = 3$ , $d_D(w, v) = 2$ and $d_G(u, w) > 3$ .	23
3.3	Graphs G and H are not in $\mathcal{F}_3(D)$	24
3.4	$G_{ont} \in \mathcal{F}_3(D)$ .	26
3.5	A complete binary tree of height 2	27
3.6	$K_2 + \overline{K}_5 \in \mathcal{F}_k$ (complete binary tree of height 2).	27
3.7	A 9-order star graph.	27
3.8	Case $\delta(H) = k$	28
3.9	Case $\delta(H) = k + i$	31
3.10	Case 1: $(u, x) \in E(G) \Rightarrow (u, v) \in E(G)$	32
3.11	Case 2: $N_G(u) = N_G(x) \Rightarrow N_G(u) = N_G(v)$ .	32
3.12	$K_{5,5} \in \mathcal{F}_5(K_{5,5})$ and diameter of $K_{5,5}$ is 2	35
3.13	Diameter of $K_{5,5} - \{0,1\}$ is 3	36
<i>A</i> 1	C	<b>A</b> 1
4.1	$C_{16}$	41
4.2		41
4.5	A C(10, 7)	42
4.4	$A \cup (10, < 0, 7 >)$ .	43
4.5	A 6-ary 2-cube	43
4.0	A set of h adapt division paths between $u_{i}$ and $u_{i}$ of length 2 in a $(h + 1)$ grown	44
4.1	A set of k edge-disjoint paths of length 3 in a $(k + 1)$ crown	4J 15
4.0	A set of A edge disjoint paths between $a_1$ and $a_2$	45
4.9	A set of 4 edge-disjoint pairs between $v_0$ and $v_3$	47
4.10	bedge disjoint paths between $w_{i}$ and $w_{i}$ when $2r = 1 \le i \le 4r = 5$	51
4.11	k edge-disjoint paths between $w_0$ and $w_1$ when $n = 2r \pm 1 \le i \le n = 3$	51
4.12	A set of 3 edge-disjoint paths between $a_1$ and $a_2$ when $p = 2i + 1 \le j \le p = 5$	53
A 1A	A set of 5 edge-disjoint pauls between $0_0$ and $0_7$ .	53
4 15	$C(28 < 3.7.9 >) \in F_0(C(28.7))$	54
4.15	Edge disjoint naths between 0 and 15 in a $C(28 < 3.7.9)$	56
4.10	Edge disjoint paths between 0 and 10 in a $C(28 < 3, 7, 9 >)$ .	57
4.17	$\Omega^{th}$ cycle and $5^{th}$ cycle in a 8x8 torus	59
4.10		57
	A 4-crown $(k = 3)$ for the 4 <sup>th</sup> cycle in a 8x8 torus	59
4.19	A 4-crown $(k = 3)$ for the 4 <sup>th</sup> cycle in a 8x8 torus.	59 60
4.19 4.20 4.21	A 4-crown $(k = 3)$ for the 4 <sup>th</sup> cycle in a 8x8 torus	59 60 61
4.19 4.20 4.21 4.22	A 4-crown $(k = 3)$ for the 4 <sup>th</sup> cycle in a 8x8 torus	59 60 61 62
4.19 4.20 4.21 4.22 4.23	A 4-crown $(k = 3)$ for the 4 <sup>th</sup> cycle in a 8x8 torus	59 60 61 62 63
4.19 4.20 4.21 4.22 4.23 4.23	A 4-crown $(k = 3)$ for the 4 <sup>th</sup> cycle in a 8x8 torus	59 60 61 62 63 64

4.25	k-edge disjoint paths between $(i, j)$ and $(i + 1, j - 2)$ in a n-ary 2-cube.	64
4.26	k-edge disjoint paths between $(i, j)$ and $(i, j - 3)$ in a n-ary 2-cube.	64
4.27	8-regular graph that belongs to $\mathcal{F}_3(n$ -ary 2-cube)	65
5.1	Graphs I and J and their corresponding I' and $J'$	69
5.2	Algorithm for finding max $k$ .	72
5.3	Graph $H$ and its $G'$	72
5.4	Case 1 of the Algorithm. $G$ and the resulting $G'$ .	73
5.5	Case 2 of the Algorithm. G and the resulting $G'$ .	73
<b>5.6</b>	Case 3 of the Algorithm. $G$ and the resulting $G'$ .	73

# Chapter 1

# Introduction

The necessity for high performance computing stems from several computationally intensive problems such as fuel combustion, three dimensional fluid flow, and the design of protein structures, etc [1]. Many of these computationally intensive problems were discovered in the past two decades. However, designing systems capable of meeting such high computational requirements has become increasingly feasible in recent years. This is due to the technological advances in VLSI and parallel processing. In parallel processing, several processors concurrently work on an application with each processor coordinating with others. This approach is promising for high performance computing because of the enormous amounts of computational power offered by parallel processing systems [1].

Several architectural approaches have been taken in designing parallel processing systems. Some of the important considerations in selecting an architecture for a parallel processing system include simple routing, scalability, cost-effectiveness and reliability [2]. Among the many architectures proposed for parallel processing systems, a particular class of systems called *point-to-point* networks has received much attention. The interest in these systems is mainly due to the scalability and cost-effectiveness [1]. Point-to-point networks can be defined as a collection of linked autonomous processors with the memory partitioned amongst the processors. The memory associated with each processor is local to that processor and is called *local memory*. Each processor has direct access to its local memory; indirect access to the memory located in other processors is facilitated through interprocessor communication. A processor can communicate with another processor by sending/receiving messages. These systems are widely known as *message passing systems*, marked by the nature of their interprocessor communication.

The processor and link layout of a message passing system is referred to as *communication* networks (CN). The underlying topology of a CN is usually modeled by a graph G(V, E) in which the vertex set V represents the set of processors and the edge set E represents the links.

One of the main reasons for the interest in CNs is due to their scalability. However, as the number of components increases, the probability of a failure in the system also increases<sup>1</sup>. With failures, the performance of the system may degrade. With degraded performance, the parallel algorithm running on the system may no longer be cost-effective to run on a parallel system. In other words, the application has a low ratio of performance to the resources consumed. Hence, in designing or selecting a topology for a CN, one fundamental consideration is *system-level fault-tolerance*. At this level, the type of faults to be tolerated are *processor* failures or *link* failures. A system is said to be *fault-tolerant* if it can remain *functional* in the presence of failures. It is, however, the topological requirements set by the application environment that essentially determines when a CN is considered to be functional. In the remaining part of the introduction, we briefly review the issues and classify the approaches taken in designing fault-tolerant CNs, followed by notation and the organization of the dissertation.

2

<sup>&</sup>lt;sup>1</sup>The rate of the failure probability is dependent on the particular CN under consideration.

3

## **1.1 Functionality Criteria**

Over the years researchers have considered different approaches in designing fault-tolerant CNs. In general, these approaches can be classified by the two *functionality* criteria which have received much attention. According to one of these criteria, a CN is considered functional as long as a certain topology is logically contained in the system. In the past few years, much work has been done in developing parallel algorithms and the best CNs for their executions [3]. For these algorithms, the existence of certain topologies is a significant factor in delivering the desired performance. Thus, for such applications, the system should be able to provide (logically) a specific topology throughout the execution of the algorithm. Designing fault-tolerant systems to guarantee a topology is referred to as *preserving a topology*. This criterion was first formulated by Hayes [4], and subsequently a number of topology preserving designs have been proposed [2, 5, 6, 7, 8, 9, 10, 11].

The second functionality criterion, the focus of this dissertation, considers a CN functional as long as the underlying topology of the non-faulty system satisfies one or more topological properties. Designing fault-tolerant systems to guarantee a topological property<sup>2</sup> is referred to as *preserving a property*. The existing work in preserving a property have mainly considered topological properties involving *connectivity* or *distance*. In this criterion, for instance, a CN is functional as long as there is a non-faulty communication path between each pair of non-faulty processors [12, 13, 14]. In other words, the underlying topology of a fault-tolerant CN should remain connected in presence of certain failures. Preserving a property is particularly applicable to large-scale CNs that are to execute concurrent algorithms whose performance is relatively insensitive to infrequent changes in system topology [13]. Such algorithms generally permit graceful degradation by providing the functionality required with degraded performance. In this dissertation, we study

<sup>&</sup>lt;sup>2</sup>Preserving a topology can also be viewed as a topological property. However, we distinguish these two terms for clarity.

designing "optimal" fault-tolerant CNs that preserve *distance* between every pair of non-adjacent nodes and tolerate edge failures. This study will result in more cost-effective fault-tolerant designs in networks where distance between communicating processors is critical.

### **1.2 Terminology and Notation**

A CN can be modeled as a graph by representing each processor as a vertex and each transmission link connecting a pair of processors as an edge. In the rest of the dissertation, we do not make a distinction between the terms "CN" and "graph". Hence, we use the terms processor (link) and vertex (edge) interchangeably. In this section, we briefly define the graph theoretic notation used in this dissertation. Terms not defined here can be found in [15].

All graphs considered here are connected, undirected, without any loops or multiple edges. Let G(V, E) be a graph with the vertex set V(G) and the edge set E(G). The order and size of the graph G are |V(G)| and |E(G)|, respectively. If an edge  $e = (u, v) \in E(G)$ , then vertices u and v are said to be adjacent. If  $F \subseteq V(G)$  (or E(G)) then the graph G - F represents the subgraph obtained by removing the vertices (edges) in F from G. If  $u \in V(G)$  then  $N_G(u)$  represents the set of all vertices adjacent to vertex u in G. The set  $N_G(u)$  is called the neighborhood of u; the closed neighborhood of u is defined as  $N_G[u] = N_G(u) \cup \{u\}$ . The degree of a vertex  $u \in V(G)$  is  $deg_G(u) = |N_G(u)|$ . The minimum degree of a graph G is  $\delta(G) = \min\{deg_G(u)|u \in V(G)\}$ . The maximum degree of the graph G is  $\Delta(G) = \max\{deg_G(u)|u \in V(G)\}$ . If  $\delta(G) = \Delta(G) = r$ , then all the vertices of G have the same degree and the graph G is called r-regular; r is also referred to as the degree of G. The distance  $d_G(u, v)$  between two distinct vertices u and v in G is defined as the length (in number of edges) of a shortest path joining these vertices. The complete graph with p vertices is denoted by  $K_p$ . A complete bipartite graph with m and n vertices in each partition is denoted by  $K_{m,n}$ . The completent of a graph G, denoted as  $\overline{G}$ , is a

graph with  $V(\overline{G}) = V(G)$  and for every  $(u, v) \notin E(G)$ , we have  $(u, v) \in E(\overline{G})$ . The join of two graphs G and H, denoted as G + H, is a graph with vertex set  $V(G) \cup V(H)$  and edge set  $E(G) \cup E(H) \cup \{(u, v) | \text{ for every } u \in V(G) \text{ and for every } v \in V(H) \}$ . We denote by  $C_p$  a cycle on p vertices, and by  $P_p$  a path on p vertices.

### **1.3 Organization of the Dissertation**

In Chapter 2, we present a brief survey of the work to date in designing topology preserving and property preserving graphs. In Chapter 3, we introduce the distance preserving graphs and present a theorem that characterizes the distance preserving graphs. We present two design models in designing "optimal" distance preserving graphs in Chapter 3. Also, in Chapter 3, we show that preserving the distance between non-adjacent vertices also sets a bound, in presence of edge removal, on the distance between vertices which were adjacent before any edge removal. Distance preserving graphs with respect to some special topologies are presented in Chapter 4. An algorithm to find the maximum value of k given a graph G and a spanning subgraph D is presented in Chapter 5. In Chapter 6, we conclude by outlining the contributions of this dissertation.

In the rest of the dissertation, the terms "fault-tolerance" or "functional" are meaningful in the context of the criterion under consideration.

# Chapter 2

# **Measures of Fault Tolerance**

The degree of fault-tolerance can be quantified either *probabilistically* or *deterministically*. In *probabilistic measures*, the objective is to find the probability of a system being functional, given the failure probabilities of its components. To decrease the complexity of the analysis, it is usually assumed that failures are statistically independent and equiprobable. However, components may fail, in general, with different probabilities. Probabilistic measures of fault-tolerance seem to model the real world better than their deterministic counterparts [13]. However, it has been shown that all probabilistic measures of reliability which are of practical significance are computationally NP-hard [16]. Consequently, a number of heuristic algorithms for the computation of these measures have been proposed.

For *deterministic measures* researchers have mainly used graph invariants such as connectivity, diameter, *etc.* In this dissertation, we focus on designing fault-tolerant CNs and use deterministic measures to quantify their fault-tolerance.

### **The Problem**

The general question in designing fault-tolerant topologies is:

"Given a non-negative integer k, design a graph G such that after k failures in G, the remaining graph is still functional".

It is often helpful to minimize the cost of the fault-tolerant graph G. This can be achieved by imposing one or more *optimality criteria* in designing G. There are three important optimality criteria [17]. The first and second criteria are minimizing the number of vertices and/or minimizing the number of edges, respectively. Over the years, the above two criteria have received much attention. In the third optimality criterion, only regular graphs are considered and the objective is to minimize the regularity of the fault-tolerant graph. The degree of the fault-tolerant graph is minimized to reduce the complexity of the system. Also, there are some advantages in requiring the topologies to be regular. Regular topologies may allow simple routing [14] . Further, VLSI building blocks such as transputer and iWarp [1] with communication links can be used in designing scalable faulttolerant point-to-point networks. Hence, the specifications for the above design problem should include some optimality criteria. This formulation makes designing fault-tolerant topologies more challenging. A sketch of the issues in fault-tolerance is outlined in Figure 2.1.

In the remaining part of this Chapter, we present a brief survey of the research in the two functionality criteria discussed earlier. Studies in both processor and link failures are presented. For each of the two functionality criteria, the problems presented in this chapter mainly differ by the optimality criteria and/or design approach taken. Chapter 3 presents the proposed work in detail.

### 2.1 Preserving a Desired Topology

Designing systems that guarantee a desired topology is referred to as preserving a topology. Several researchers have proposed various design methods to preserve a given topology. The basic design approach to preserve a topology is to employ systemwide redundancy and reconfiguration [4]. In other words, a fault-tolerant graph is designed by augmenting vertices and/or edges to a





given desired graph. Let us consider the designs that tolerate processor failures. This criterion can be formally defined as follows [4].

**Definition 1** Let G(V, E) represent the topology of a CN and D(V, E) be a desired structure. Then, G is k-node fault-tolerant (k-NFT) with respect to D, if for any set  $F \subset V(G)$ , with  $|F| \leq k$ , the graph D is isomorphic to a subgraph of G - F.

Using the above definition, several classes of problems can be formulated by imposing certain optimality criteria on the graph G and/or some restrictions on the *faulty set* F. Hayes [4] considered minimizing the number of vertices first and then the number of edges. It is easy to see that G must contain at least k spares. The problem originally proposed in [4] can be stated as follows.

**Problem 1** Given a graph D and a positive integer k, construct an "optimal" graph G, which is k-NFT with respect to D. The optimality criterion is:

- (a) |V(G)| = |V(D)| + k
- (b) for any G' that is k-NFT with respect to D satisfying (a), we have  $|E(G)| \leq |E(G')|$ .  $\Box$

Let us compute the extremal part of the above problem, *i.e.*, to find the minimum number of edges in a graph G which is k-NFT with respect to D. From G, remove any set, say F, of k vertices adjacent to a vertex v, with  $deg_G(v) = \delta(G)$ . Since G is k-NFT with respect to D, the resulting subgraph should contain the graph D. Hence  $deg_{G-F}(v) \ge \delta(D)$ , which implies  $\delta(G) \ge \delta(D) + k$ . This also implies that the degree of each vertex in G is proportional to the number of faults to be tolerated. In [4], three candidates for the graph D are considered, namely, cycles, trees, and simple paths. In particular, a construction is given for G when D is an even cycle. In this case, G is a (k + 2)-regular graph. A similar construction is also given for the case of D being an odd cycle. The work of Hayes was followed by [2, 5, 6, 7, 8, 9, 10, 11]. In these papers, the authors have

adhered to the same optimality criterion, but imposed additional restrictions on the set F. For example, in [6] the case of D being a binary tree is considered with the additional restriction that failures only occur at *different* levels of the tree.

Since the current technology limits the number of possible connections at each processing node <sup>1</sup>, the optimality criterion of Problem 1 may not be viable in certain situations [18, 19]. Fault-tolerant designs which limit the fault-tolerant graph to be regular are studied in [20]. The size and complexity of the CN is reduced by introducing additional redundancy. Formally,

**Problem 2** Given a desired graph D and positive integers k and r, construct a graph G, such that

- (a) G is r-regular,
- (b) for any set  $F \subset V(G)$  with  $|F| \leq k$ , D is isomorphic to a subgraph of G F,
- (c) no graph H satisfying (a) and (b) above has fewer vertices than G.  $\Box$

This approach is more viable since current technology limits the number of possible connections to each node [11, 20]. It is clear that this problem may not have any solution for certain choices of D, k, and r. A case in point, when r = 2 and D is a cycle on n vertices, there is no solution G for any k. Thus 3 is the smallest value of r for which a solution may exist for the case of  $D = C_n$ . When  $D = C_n$  and r = 3, it is shown in [20] that any solution G must satisfy  $|V(G)| \ge |V(D)| + 2k$ . Also, a construction is given for a class of graphs for which this lower bound is achieved for k = 1, 2, and 3. Very little is known about fault-tolerant topologies when k > 3, r = 3, and  $D = C_n$ . Also, no solution is proposed so far for other desired graphs such as grids, hypercubes, etc.

Now let us consider the designs that tolerate edge failures and preserve a topology. The edge failure version was recently introduced by Harary and Hayes [21]. The following definition of a k-edge fault-tolerant graph is due to [21].

<sup>&</sup>lt;sup>1</sup>it appears that this constraint will persist for some time to come [18, 19].

**Definition 2** Let G(V, E) represent the topology of an CN and D(V, E) be a desired structure. Then, G is k-edge fault-tolerant (k-EFT) with respect to D, if for any set  $F \subset E(G)$ , with  $|F| \le k$ , the graph D is isomorphic to a subgraph of G - F.

In [21], k-EFT graphs with minimum number of edges were studied. This problem can be formally stated as follows.

**Problem 3** Given a p-order graph D and a positive integer k, construct an "optimal" p-order graph, G, such that

- (a) G is k-EFT with respect to D,
- (b) for any p-order graph G' satisfying (a), we have  $E(G) \leq E(G')$ .

Note that in Problem 3, both D and G are p-order graphs. In other words, in this approach a fault-tolerant graph is constructed using only spare links. It is easy to verify that  $\delta(G) \ge \delta(D) + k$  in any approach that uses only spare links. Further, for certain large values of k the resulting fault-tolerant graph G is a multigraph. A construction is given in [21] for a class of optimal k-EFT graphs when the desired graphs are simple paths and cycles. In particular, when either k or n is even a (k + 2)-regular graph is constructed for  $D = C_n$ . Also, when the desired topology is an n-dimensional hypercube, an optimal solution to tolerate one edge failure is given in [21]. We can also restrict fault-tolerant graph to be regular by using spare processors. Formally,

**Problem 4** Given a desired graph D and positive integers k and r, construct a graph G, such that

(a) G is r-regular,

- (b) for any set  $F' \subset E(G)$  with  $|F'| \leq k$ , D is isomorphic to a subgraph of G F',
- (b) no graph H satisfying (a) and (b) above has fewer vertices than G.

The above design methods which primarily aim to preserve a desired topology have some drawbacks. We will refer to two of them here. First, these topologies are expensive in terms of the number of redundant processors/links. For example, when minimum number of redundant processors are used, the best known approach to tolerate two processor failures and preserve a  $4 \times 4$ mesh requires 235% more redundant links [2]. Second, there are no tight bounds on the size of the graph that could preserve topologies such as hypercubes and 2D meshes [2]. On the other hand, the existing approaches for preserving topological properties are relatively less expensive. However, preserving a property is not a complete solution to the topological fault-tolerance problem.

### 2.2 Preserving a Property

This functionality criterion is useful in designing systems that allow graceful degradation. Researchers have taken two approaches in designing property preserving graphs. In one approach, an optimal property preserving graph is designed, given the order of the fault-tolerant graph. We will refer to this approach as *intrinsic design*. This approach is taken because there are some classes of graphs which are, by definition, property preserving. For example,  $K_{m,n}$  with  $m \leq n$ , can tolerate up to m - 1 processor failures and still preserve the diameter of  $K_{m,n}$ . Similarly, a cycle can tolerate one processor or link failure and guarantee a communication path between each pair of non-faulty processors. There are several research issues in designing property preserving graphs with this approach, some of which have been addressed in [22, 23, 24, 25].

Notice that some popular topologies such as path, cycle, k-ary n-cube, may not preserve a given property after a few failures. In such cases, some redundancies may have to be introduced to make a given topology preserve a property under failures [26]. This approach is referred to as *redundant* design.

As mentioned earlier, topological properties involving connectedness and distance have been

studied. Further, in preserving distance, researchers have considered preserving maximum distance (diameter) of the graph and distance between every pair of non-adjacent vertices.

### 2.3 Preserving Connectedness

Preserving connectedness considers a CN functional as long as every pair of non-faulty processors is joined by a non-faulty path. A measure for this property can be given by the connectivity of the underlying graph. The vertex-connectivity,  $\kappa(G)$ , of a graph G is defined as the least cardinality |S| of a set  $S \subset V$  such that G - S is either disconnected or trivial (i.e., a single-vertex graph). The edge-connectivity,  $\lambda(G)$ , is defined similarly with S being a set of edges. Using connectivity as a measure of fault-tolerance implies that if a graph G is the underlying topology of a CN then that CN can tolerate  $\kappa(G) - 1$  processor failures or  $\lambda(G) - 1$  link failures. Hence designing an optimal system that inherently tolerates (k - 1) processor (link) failures and guarantees connectedness can be designed by an optimal k-vertex (edge) connected graph. Now, we will state the design problems in preserving connectivity to tolerate vertices/link failures.

**Problem 5** Given two positive integers p and k, design a p-order graph G, such that,

- (a)  $\kappa(G) = k$ ,
- (b) for any p-order graph G' satisfying (a), we have  $E(G) \leq E(G')$ .

**Problem 6** Given two positive integers p and k, design a graph p-order graph G such that,

(a) 
$$\lambda(G) = k$$
,

(b) for any p-order graph G' satisfying (a), we have  $E(G) \leq E(G')$ .

Let us consider the extremal part of the above problems. A well known result due to Whitney [27] is  $\kappa(G) \leq \lambda(G) \leq \delta(G)$ . That is, in any graph the vertex connectivity can at most be equal to edge connectivity, and the edge connectivity can at most be equal to the degree of the graph. It is clear from Whitney's result that we must have  $\delta(G) \ge k$ . Hence, the minimum number of edges in a k-connected (k-edge connected) graph is at least  $\lceil \frac{kp}{2} \rceil$ . The solutions to the above problems were first given by Harary [22] by constructing a special class of k-connected (k-edge connected) graphs with  $\lceil \frac{kp}{2} \rceil$  edges. This class of graphs are known as *Harary graphs*. For a survey on this class of graphs, called circulants, and their connectivities, the reader is referred to [28]. Now, we will briefly review the problems in designing graphs that preserve connectedness by augmenting edges to a given graph D.

**Problem 7** Given a p-order graph D and a positive integer k, construct a p-order graph G, such that:

- (a)  $D \subseteq G$ ,
- (b) G is k-connected,

(c) for any G' satisfying (a) and (b) above, we have  $E(G) \leq E(G')$ .

**Problem 8** Given a p-order graph D and a positive integer k, construct a p-order graph, G, such that:

- (a)  $D \subseteq G$ ,
- (b) G is k-edge connected,
- (c) for any G' satisfying (a) and (b) above, we have  $E(G) \leq E(G')$ .

Problem 7 is known as *k*-connectivity augmentation problem. A solution for the 2-connectivity augmentation problem is presented in [29]. Recently, Watanabe and Nakamura [30] solved the 3-connectivity augmentation problem. When k > 3, this augmentation problem remains open.

Problem 8 is known as the k-edge-connectivity augmentation problem. When the given graph is a tree, the k-edge-connectivity augmentation problem is solved in [31]. Cai and Sun [32] recently solved the problem when the given graph is a general graph. In [26], it is shown that the weighted version of this problem is NP-complete.

While preserving the connectivity guarantees the communication between any non-faulty processors, the number of intermediate processors involved in message traversal may increase randomly. For instance, let us consider a CN with  $C_p$  as its underlying topology. Such a CN can tolerate one link failure and guarantee a non-faulty path between every pair of processors. Consider two adjacent vertices u and v. It is easy to see that if the link (u, v) fails the distance between vertices u and v is p - 1. To compensate this shortcoming, one can make use of several other measures in preserving distance.

### 2.4 Distance

There are two important aspects in preserving distance which have attained the attention of several researchers. They are the *maximum distance* in the graph and the *distance* between every pair of non-adjacent vertices in the graph. In this section, we briefly review some of the important problems in these two aspects.

### 2.4.1 Preserving Diameter

As defined earlier, the diameter of a graph is the maximum distance among all pairs of vertices in the graph. The delay in transmitting a message in a CN may be measured by the number of intermediate processors between the communicating processors [33]. In this measure, the upper bound on the maximum transmission delay is a function of the diameter of the underlying topology of the CN. Hence, to minimize the maximum transmission delay is to minimize the number of intermediate processors involved between any pair of communicating processors. Research in preserving a diameter is motivated from designing fault-tolerant CNs in which maximum transmission delay is critical to keep the system performance. Our interest is in optimal topologies which can tolerate certain number of failures and still preserve the upper bound on the transmission delays. Intrinsic design problems to preserve diameter under faults can be stated as follows.

**Problem 9** Given positive integers p, k, and d, design a p-order graph G such that

- (a) diam(G) = d,
- (b) for any  $F \subset V(G)$  with  $|F| \leq k$ , we have diam(G F) = d,
- (c) for any G' satisfying (a) and (b) above we have  $E(G) \leq E(G')$ .

**Problem 10** Given positive integers p, k, and d, design a p-order graph G such that

- (a) diam(G) = d,
- (b) for any  $F \subset E(G)$  with  $|F| \leq k$ , we have diam(G F) = d,
- (c) for any G' satisfying (a) and (b) above we have  $E(G) \leq E(G')$ .

These problems were first posed by Murty and Vijayan [23], which later received considerable attention. However, the exact solutions to the above problems have been given only for some special cases. For example, for Problem 10, the known results are for  $(d = 2, k \ge 1, p \ge 1), (d \le 4, k = 1, p \ge 1)$ , and  $(d = p - 1, k \ge 1, p \ge 1)$  [25]. Very recently, F.R.K. Chung [25] solved another special case when  $(d \ge 1, k = 1, p \ge 1)$ . There are several variations of the above two problems; However, many problems remain unsolved [25, 34]. A survey of these problems with recent results can be found in [34].

Similar to Problems 7 and 8, we can construct a diameter preserving graph G by augmenting a minimum number of edges to a given graph D.

**Problem 11** Given a p-order graph D with diameter d and a positive integer k, construct a p-order graph G such that,

- (a)  $D \subseteq G$ ,
- (b) for any  $F \subset V(G)$ , we have diam(G F) = d,
- (c) any other G' satisfying (a) and (b) above we have  $E(G) \leq E(G')$ .

**Problem 12** Given a p-order graph D with diameter d and a positive integer k, construct a p-order graph G such that,

(a)  $D \subseteq G$ ,

- (b) for any  $F \subset E(G)$ , we have diam(G F) = d,
- (c) any other G' satisfying (a) and (b) above we have  $E(G) \leq E(G')$ .

Preserving the diameter is a good criterion to limit the maximum transmission delay. However, a better design is to be able to keep the transmission delay intact between any communicating processors. In other words, the interest is in the design of fault-tolerant topologies in which the distance between the non-faulty processors is not increased after failures.

### 2.4.2 Preserving Distance

As defined earlier, the distance  $d_G(u, v)$  between two distinct vertices in a graph G is the length of the shortest path joining the vertices in the graph. Intrinsic designs that preserve distance between every pair of non-adjacent vertices were first considered by Entringer *et al.* [35]. A special class of graphs in which there are k vertex-disjoint paths of minimum length are studied in [35]. These graphs are called k-geodetically connected graphs. The following definition of a k-geodetically connected graph is due to Entringer *et al.* [35]. **Definition 3** A graph G is said to be k-geodetically connected if for every pair of non-adjacent vertices, say u and v, we have at least k vertex-disjoint paths of length  $d_G(u, v)$ .

In other words, in a k-geodetically connected graph, removal of at least k vertices is necessary to increase the distance between any pair of non-adjacent vertices. A k-geodetically line connected graph can be defined similarly. Some properties of k-geodetically vertex (line) connected graphs are given in [35]. There are two properties that are of interest to us [35].

**Property 1** A graph G is k-geodetically line-connected if and only if G is k-geodetically connected.

**Property 2** For any pair of vertices u and v in a k-geodetically connected graph G we have  $N_G(u) \cap N_G(v) \ge k.$ 

By Property 1, it is clear that designing an optimal k-geodetically line connected graph is equal to designing an optimal k-geodetically connected graph. The problem of designing an optimal k-geodetically line connected graph can be formally stated as follows.

**Problem 13** Given two positive integers p and k, design a p-order graph G, such that

- (a) G is k-geodetically line connected,
- (b) for any p-order graph G' that satisfies (a) above, we have  $E(G) \leq E(G')$ .

Exact solution to the above problem is known only for a special case, namely, when diam(G) = 2 and  $2k \le p$ ; this case was solved by Bollabas and Eldridge [36]. Entringer and Jackson [37] solved the case when diam(G) = 2 and 2k > p. The problem remains unsolved for diam(G) > 2.

### 2.5 Proposed Work

The main subject of this dissertation is the corresponding augmentation problem to tolerate edge failures and preserve distance when a graph D is given. That is, we construct a distance preserving

graph G by augmenting edges to D. Note that by adding an edge to the given graph D, the previous distance between the end vertices of each new edge is changed [38, 39]. Hence, we propose to construct a graph in which, between every pair of non-adjacent vertices u and v in D, there are at least k edge-disjoint paths, each having length at most  $d_D(u, v)$ . We call the resulting graph as k-edge distance preserving with respect to D.

The objective of this dissertation is to study the aspects in designing distance preserving graphs. In Chapter 3, we introduce the problem definition and our design models. In Chapter 3, we show that preserving the distance between non-adjacent vertices also sets a bound, in presence of edge removal, on the distance between vertices which were adjacent before any edge removal. Distance preserving graphs with respect to some special topologies are presented in Chapter 4.

# **Chapter 3**

# **Distance Preserving Graphs**

# 3.1 Introduction

**Definition 4** Given a connected graph D and a non-negative integer k, a graph G is said to be k-edge distance preserving with respect to the given graph D, if:

- 1. D is a spanning subgraph of G and,
- 2. every pair of non-adjacent vertices u and v in D are joined in G by at least k edge-disjoint paths, each having length at most  $d_D(u, v)$ .

Let  $\mathcal{F}_k(D)$  denote the set of all graphs that are k-edge distance preserving with respect to a given graph D. Figure 3.1 shows two graphs D and G; where  $G \in \mathcal{F}_3(D)$ .



Figure 3.1:  $G \in \mathcal{F}_3(D)$ .

The results provided in the remaining part of this section can be used to check the validity of the constructions presented in this dissertation.

### 3.2 Preliminaries

**Property 3** For any graph D on p vertices we have  $\mathcal{F}_{p-1}(D) = \{K_p\}$ .

**Property 4** For any graph D on p vertices we have  $\mathcal{F}_{k+1}(D) \subseteq \mathcal{F}_k(D)$  for any k,  $1 \leq k < p-1$ .

**Property 5** For any graph D on p vertices the set  $\mathcal{F}_k(D)$  is not empty for any k,  $1 \le k < p$ .

**Lemma 1** Let G be a k-edge distance preserving graph with respect to a given graph D. Then for every pair of vertices u and v with  $d_D(u, v) = 2$ , we have  $|N_G(u) \cap N_G(v)| \ge k - 1$ .

**Proof:** By the definition of a k-edge distance preserving graph, we know that G contains k edge-disjoint u-v paths of length at most 2. Since G is a simple graph, there can be at most one edge connecting the vertices u and v. There remain at least k - 1 paths and each path must have only one intermediate vertex. Hence  $|N_G(u) \cap N_G(v)| \ge k - 1$ . Note that if it turns out that the edge  $(u, v) \notin E(G)$ , then we have  $|N_G(u) \cap N_G(v)| \ge k$ , since in this case all the k edge-disjoint u-v paths in G are of length two.

**Corollary 1** Every graph G in  $\mathcal{F}_k(D)$  is at least k-connected.

### **3.3 Characterization Result**

**Theorem 1** Let D be a p-order graph and k < p be a non-negative integer. Further, let G be a graph having D as a spanning subgraph, and if u and v are non-adjacent vertices of D such that  $d_D(u, v) \leq 3$ , then there are k edge-disjoint paths between u and v in G, each having length  $d_D(u, v)$  or less. Then  $G \in \mathcal{F}_k(D)$ .

**Proof:** We will consider an arbitrary pair of vertices u and v in D and inductively construct a set of k edge-disjoint u-v paths in G of length at most  $d_D(u, v)$ . The following notation is useful. For any two vertices x and y in a path P, we denote by  $P_{x\sim y}$ , the portion of P connecting the vertices x and y.

Basis : If  $d_D(u, v) \leq 3$ , then by the way G is defined there are k edge-disjoint u-v paths in G of length at most  $d_D(u, v)$ .

*Hypothesis*: For every pair of non-adjacent vertices u and v in D, with  $d_D(u, v) < n$  and  $n \ge 4$ , there are k edge-disjoint u-v paths in G of length at most  $d_D(u, v)$ .

Induction Step : Consider a pair of non-adjacent vertices u and v in D such that  $d_D(u, v) = n$ ; if no such pairs exist, we are done. Consider the vertex w on a shortest u-v path in D such that  $d_D(u, w) = n - 2$ . By the induction hypothesis, there are k edge-disjoint u-w paths in G of length at most  $d_D(u, w)$ . Let  $\mathcal{P} = \{P_1, P_2, \ldots, P_k\}$  be such a set of k edge-disjoint paths. From the way we chose w, the vertices w and v are distance two apart in D and hence there are k edge-disjoint w-vpaths in G of length at most 2. Let  $\mathcal{Q} = \{Q_1, Q_2, \ldots, Q_k\}$  be such a set of k edge disjoint paths. Also by Lemma 1 we know that at least k - 1 of the paths in  $\mathcal{Q}$  are of length two. Without loss of generality, let  $Q_i = w, q_i, v, 1 \le i < k$ . Further let  $q_k = w$  if  $Q_k = w, v$ ; otherwise,  $Q_k = w, q_k, v$ . Using the paths in  $\mathcal{P}$  and  $\mathcal{Q}$  we construct,  $\mathcal{R}$ , a set of k edge-disjoint u-v paths in G of length at most  $d_D(u, v)$ .

We call a path P in  $\mathcal{P}$  as used if P is used to construct a u-v path in  $\mathcal{R}$ . Initially all the paths in  $\mathcal{P}$  are unused. Also, we mark a vertex  $q_i$ ,  $1 \le i \le k$ , as visited if the edge  $(q_i, v) \in E(G)$  is used in constructing a u-v path in  $\mathcal{R}$ . Initially, all vertices  $q_i$ ,  $1 \le i \le k$ , are marked unvisited.

**Step 1:** For each path  $P \in \mathcal{P}$ ,



Figure 3.2: Example: k = 3,  $d_D(w, v) = 2$  and  $d_G(u, w) \ge 3$ .

Case:  $v \in V(P)$ 

If v is an intermediate vertex in P then  $P_{u \sim v} \in \mathcal{R}$ ; mark P as used. The path  $P_{u \sim v}$  is of length at most  $d_D(u, v)$ , because P is of length at most  $d_D(u, w) < d_D(u, v)$ . Note that any such path  $P_{u \sim v}$  can have at most one edge  $(q_j, v)$ ,  $1 \leq j \leq k$ . If such a vertex  $q_j$  exists, mark  $q_j$  as visited.

Case:  $v \notin V(P)$ 

In this case, the path P may contain zero or more unvisited vertices  $q_i$ ,  $1 \le i \le k$ .

Subcase: P contains one or more unvisited vertices

Let  $q_j$  be the first unvisited vertex encountered when traversing the path P from u to w. Then we construct a new u-v path in  $\mathcal{R}$  by augmenting the path  $P_{u \sim q_j}$  with edge  $(q_j, v)$ , and then mark the vertex  $q_j$  as visited. It is easy to see that such a u-v path is of length at most  $d_D(u, v)$ . Further, the constructed u-v path is mutually edge-disjoint with all the existing paths in  $\mathcal{R}$  as all paths in  $\mathcal{P}$  are edge-disjoint and the vertex  $q_j$  is not visited before.

Subcase: P contains no unvisited vertex

In this case the path will remain unused and will be processed in Step 2.

Note that, each time we construct a path in  $\mathcal{R}$  we mark at most one more vertex  $q_j$ ,  $1 \le j \le k$ , as visited. Hence, after the above construction, the number of unvisited vertices in G is at least as



Figure 3.3: Graphs G and H are not in  $\mathcal{F}_3(D)$ .

many as the number of unused paths in  $\mathcal{P}$ . If a vertex  $q_j$ ,  $1 \le j \le k$ , is still unvisited, it implies that both the edges  $(w, q_j)$  and  $(q_j, v)$  do not exist in any unused path in  $\mathcal{P}$ .

**Step 2:** For each unused path in  $P \in \mathcal{P}$ 

Construct a new u-v path by appending the path  $w, q_j, v$  to P, where  $q_j$  is some unvisited vertex. It is easy to see that each such u-v path in G is mutually edge-disjoint with the paths in  $\mathcal{R}$  and of length at most  $d_D(u, w) + 2 \leq d_D(u, v)$ . Hence, by using each of the k paths in  $\mathcal{P}$  we construct a set of k edge-disjoint u-v paths in G of length at most  $d_D(u, v)$ .

An example explaining the above construction is given in Figure 3.2. Figure 3.2 represents a portion of a graph  $G \in \mathcal{F}_3(D)$ , for some D. Let  $\mathcal{P} = \{P_1, P_2, P_3\}$ , where  $P_1 = \{u, \ldots, q_1, v, q_2, w\}$ ,  $P_2 = \{u, \ldots, q_1, w\}$  and  $P_3 = \{u, \ldots, q_3, w\}$ . The above construction first considers  $P_1$ . Since  $v \in P_1$ ,  $R_1 = P_{1_{u\sim v}}$  and vertex  $q_1$  is marked visited. The path  $P_2$  does not contain any *unvisted* vertex. Hence, path  $P_2$  will still be marked unused and path  $P_3$  is considered next. Since  $P_3$  contains one unvisited vertex,  $q_3$ , the new path  $R_3$  is  $\{P_{3_{u\sim q_3}}, v\}$ . The vertex  $q_3$  is marked visited. Now in step 2 the remaining unused path,  $P_2$  is considered. A new u-v path is constructed by augmenting  $P_2$  with path  $w, q_2, v$ .

The above theorem implies that to prove a graph G belongs to  $\mathcal{F}_k(D)$ , it is enough to show that every pair of vertices u and v with  $d_D(u, v) \leq 3$  are joined in G by k edge-disjoint paths of length at most  $d_D(u, v)$ . Figure 3.3 shows an example in which every pair of vertices having distance two in D are joined by 3 edge-disjoint paths of length at most two in G. However, for the vertices u and w, with  $d_D(u, w) = 3$ , there are only 2 edge-disjoint paths of length at most three in G. Similarly, all the vertices which are distance three apart in D are joined by 3 edge-disjoint paths of length at most three in H; but the vertices u and v with  $d_D(u, v) = 2$ , are not joined by 3 edge-disjoint paths of length at most two in H. Hence, the above two examples show that the sufficiency condition given in Theorem 1 is sharp.

In this chapter, we present two models in designing distance preserving topologies that tolerate edge failures. In the first model, the optimality criterion is to minimize the number of edges. The second model considers regular fault-tolerant topologies with minimum regularity. In Sections 3.4 and 3.5, we present our models and lower and upper bounds when the given graph is a general graph. We also prove that the given bounds are tight. In Chapter 4, we present constructions for cycles, crowns, chordal rings and n-ary 2-cubes.

### **3.4 Model 1**

**Model 1** Given a graph D and a positive integer k, construct a graph, G, such that:

- (a)  $G \in \mathcal{F}_k(D)$ ,
- (b) for any  $G' \in \mathcal{F}_k(D)$ , we have  $|E(G)| \leq |E(G')|$ .

#### 3.4.1 Bounds

Since we consider only simple and connected graphs, it is assumed in the rest of the dissertation that k < |V(D)|. For any graph D on p vertices, if G belongs to  $\mathcal{F}_k(D)$  then it is easy to see that  $\delta(G) \ge k$ . Hence, for any  $G \in \mathcal{F}_k(D)$ , we have  $|E(G)| \ge \lceil \frac{kp}{2} \rceil$ . Note that the example graph G in Figure 3.1 belongs to  $\mathcal{F}_3(D)$ ; however, it is not of minimum size. The graph  $G_{opt}$  in Figure 3.4 is a


Figure 3.4:  $G_{opt} \in \mathcal{F}_3(D)$ .

minimum size graph. The graph  $G_{opt}$  in Figure 3.4 uses  $\left\lceil \frac{kp}{2} \right\rceil$  edges.

The following lemma gives an upper bound on the size of the augmentation. Before we present the upper bound on the size of the augmentation let us define a graph  $H \cong K_k + \overline{K}_{p-k}$ .

**Lemma 2** Given any p-order graph D and a non-negative integer k < p, there exists a graph  $G \in \mathcal{F}_k(D)$  such that  $|E(G)| - |E(D)| \le {k \choose 2} + k(p-k) - |E(H) \cap E(D)|$ .

**Proof:** We will construct a new graph G from D by adding at most  $\binom{k}{2} + k(p - k)$  edges. Since D is a spanning subgraph of G, we start with V(G) = V(D) and E(G) = E(D). Label the vertices of G as  $v_1, v_2, \ldots, v_p$  such that  $deg_D(v_i) \ge deg_D(v_{i+1}), 1 \le i < p$ . Let  $V_k = \{v_1, v_2, \ldots, v_k\} \subseteq V(G)$ . If  $(v_i, v_j) \notin E(G)$ , for  $1 \le i \le k$  and  $1 \le j \ne i \le p$ , then add the edge  $(v_i, v_j)$  to E(G). Hence, the total number edges added is  $\le (\sum_{i=1}^k (p-i)) - |E(H) \cap E(D)| = \frac{k(2p-k-1)}{2} - |E(H) \cap E(D)| = \binom{k}{2} + k(p-k) - |E(H) \cap E(D)|$ . It is easy to see that such a graph G belongs to  $\mathcal{F}_k(D)$ . Note that if k = p - 1, the above construction results in a complete graph.

Figure 3.5 shows a complete binary tree of height two. Figure 3.6 shows a construction for k = 2 and when the graph D is a complete binary tree of height two; the set  $V_k$  is shown in Figure 3.6. Now we will show that the given upper bound is tight for certain values of k when the given graph, D, is a star graph.

26



Figure 3.5: A complete binary tree of height 2.



Figure 3.6:  $K_2 + \overline{K}_5 \in \mathcal{F}_k$  (complete binary tree of height 2).

**Definition 5** A p-order star graph, say  $S_p$ , can be defined a graph with vertex set  $\{v_0, v_1, \ldots, v_{p-1}\}$ and for each vertex  $v_i$ ,  $1 \le i < p$ , we have  $(v_0, v_i) \in E(S_p)$ . Vertex  $v_0$  is referred as the source vertex.

A 9-order star graph is shown in Figure 3.7.

**Theorem 2** Given two non-negative integers p, k where  $k \leq \frac{p}{3}$ , and a graph  $D = S_p$ , the minimum



Figure 3.7: A 9-order star graph.

27



Figure 3.8: Case  $\delta(H) = k$ .

number of edges in a graph  $G \in \mathcal{F}_k(D)$  is at least  $\frac{k(2p-k-1)}{2}$ .

**Proof:** Let  $V(D) = \{v_0, v_1, \dots, v_{p-1}\}$ , where  $v_0$  is the source of  $S_p$ . Let us assume that there exist a graph  $H \in \mathcal{F}_k(D)$  with  $|E(H)| < \frac{k(2p-k-1)}{2}$ .

Consider a vertex in H, say  $v_1$ , such that  $deg_H(v_1) = \delta(H)$ . Let  $X = N(v_1)$  and  $Y = V(H) - X - \{v_1\}$ . Note that  $v_1$  is connected to every vertex in X. And vertex  $v_0$ , the source of the star graph, is connected every vertex in  $\{X\} + \{v_1\} - \{v_0\}$ . First let us consider the case when  $\delta(H) = k$  (Figure 3.8). Since there are k edge-disjoint paths from  $v_1$  to each vertex in  $V(H) - \{v_0, v_1\}$ , each vertex in  $X - \{v_0\}$  is connected to p - 3 vertices in  $V(H) - \{v_0, v_1\}$ . In other words, the degree of each of the k vertices in X is p - 1. Hence the total number edges in H is  $\frac{k(p-1)+(p-k)k}{2} < \frac{k(2p-k-1)}{2}$  a contradiction.

Now consider the case when  $\delta(H) = k + i$ , for some non-negative integer i > 0; Figure 3.9 shows such a graph H. The total number of edges in H is the sum of,

$$(k+i)$$
: Number of edges from  $v_1$ .

- $\frac{(k+i)(k-1)}{2}:$  Number of edges between vertices in X. Since there are k edge-disjoint paths from  $v_1$  to each vertex in  $X - \{v_0\}$ , each vertex in X is connected to at least (k - 1) other vertices in X.
- (p-k-i-1)k: Number of edges between vertices in X to the vertices in Y. Since there are k edge-disjoint paths from  $v_1$  to each vertex in Y, each of the (p-k-i-1) vertices in Y is connected to at least k vertices in X.

$$\frac{(p-k-i-1)i}{2}$$
: The degree of each of the  $(p-k-i-1)$ 

vertices in Y is at least k + i.

Which implies,

$$(k+i) + \frac{(k+i)(k-1)}{2} + (p-k-i-1)k + \frac{(p-k-i-1)i}{2} < \frac{k(2p-k-1)}{2}$$
  

$$\Rightarrow \frac{2k+2i+k^2-k+ki-i+2kp-2k^2-2ki-2k+pi-ki-i^2-i}{2} < \frac{2kp-k-k^2}{2}$$
  

$$\Rightarrow pi - 2ki - i^2 < 0$$
  

$$\Rightarrow p - k < k + i$$

Hence,

$$p - k + 1 \le \delta(H) \tag{3.1}$$

Hence the number of edges in H is at least

$$\frac{(p-1)+(p-1)(p-k+1)}{2} < \frac{k(2p-k-1)}{2}$$
$$\Rightarrow \frac{(p-1)(p-k+2)}{2} < \frac{k(2p-k-1)}{2}$$

$$\Rightarrow p^{2} + p(1 - 3k) + k^{2} + 2k - 2 < 0 \Rightarrow p < \frac{(3k - 1) \pm \sqrt{(1 - 3k)^{2} - 4(k^{2} + 2k - 2)}}{2} \Rightarrow p < \frac{(3k - 1) \pm \sqrt{5k^{2} - 14k + 9}}{2}$$

Case:  $p < \frac{(3k-1)+\sqrt{5k^2-14k+9}}{2}$ Since k > 0,  $p < \frac{(3k-1)+\sqrt{9k^2}}{2}$  $\Rightarrow p < \frac{6k-1}{2}$ A contradiction.

Case : 
$$p < \frac{(3k-1)-\sqrt{5k^2-14k+9}}{2}$$
  
Since  $k > 0$ ,  $p < \frac{(3k-1)}{2}$   
A contradiction.

	-		
L			
L		1	

**Corollary 2** Given a graph  $D = S_p$ , and a non-negative integer  $k \le \frac{p}{3}$  the minimum size of an augmentation to construct a graph  $G \in \mathcal{F}_k(D)$  is  $\frac{k(2p-k-1)}{2} - (p-1)$ .

From Corollary 2, it is easy to see that the bound given in Lemma 2 is tight.

The optimality criterion used in Model 1 is to add a minimum number of edges. But, regular topologies are desired in designing modular systems [20]. Hence, we shift our optimality criterion from minimum number of edges to minimum regularity. In the rest of the dissertation, we study designing optimal graphs in this model.

# 3.5 Model 2



Figure 3.9: Case  $\delta(H) = k + i$ .

Model 2 Given a graph D and a positive integer k, construct a graph G such that:

- (a)  $G \in \mathcal{F}_k(D)$ ,
- (b) G is r-regular, and
- (c) for any r'-regular graph  $G' \in \mathcal{F}_k(D)$ , we have  $r \leq r'$ .

### 

### 3.5.1 Bounds

Let p = |V(D)|. Since  $K_p \in \mathcal{F}_k(D)$ , it is obvious that the upper bound for r is p-1. For the lower bound, since a graph  $G \in \mathcal{F}_k(D)$  is at least k-edge connected, we have  $r \ge k$ . In the remaining part of this section, we prove that for certain values of k, there are no k-regular graphs in  $\mathcal{F}_k(D)$ , in which case the lower bound for r is (k+1). But before proving this statement, we give some useful properties of k-regular graphs in  $\mathcal{F}_k(D)$ .



Figure 3.10: Case 1:  $(u, x) \in E(G) \Rightarrow (u, v) \in E(G)$ .



Figure 3.11: Case 2:  $N_G(u) = N_G(x) \Rightarrow N_G(u) = N_G(v)$ .

**Property 6** For a given graph D, if there exists a k-regular graph  $G \in \mathcal{F}_k(D)$  then for every pair of non-adjacent vertices u and v in D, we have either  $(u, v) \in E(G)$  or  $N_G(u) = N_G(v)$ .

**Proof:** Let us consider an arbitrary pair of non-adjacent vertices u and v in D. We will prove the above property by induction on  $d_D(u, v)$ .

Basis : If  $d_D(u, v) = 2$  then the validity of the property can be deduced from the proof of Lemma 1. If  $d_D(u, v) = 3$ , we will show that the edge (u, v) must belong to E(G). Consider the vertex won a shortest u-v path in D such that  $d_D(u, w) = 2$ . Since the graph G is k-regular, the edge (w, v)must be used in one of the k edge-disjoint u-w paths. Further, since any such u-w path is of length at most two, the edge (u, v) must belong to E(G). This implies  $(u, v) \in E(G)$ . This completes the verification of the basis.

*Hypothesis*: For every pair of non-adjacent vertices u and v in D with  $2 \le d_D(u, v) < n$ , either  $(u, v) \in E(G)$  or  $N_G(u) = N_G(v)$ .

Induction Step : Consider a pair of vertices u and v in D such that  $d_D(u, v) = n$ , and not satisfying the hypothesis; if no such pairs exist, we are done. Let x be the vertex on a shortest u-vpath in D such that  $d_D(u, x) = n - 2$ . By induction hypothesis, we know that either  $(u, x) \in E(G)$ or  $N_G(u) = N_G(x)$ . Note that from the way we choose x, the vertices x and v are distance two apart in D. Therefore, we have k edge-disjoint x-v paths in G of length at most two. Now, we have two cases to consider.

**Case 1**  $(u, x) \in E(G)$ : Since G is k-regular, one of the k edge-disjoint x-v paths must use the edge (u, x). Further, any such path can be of length at most two (Figure 3.10). Hence the edge (u, v) must belong to E(G). This implies  $(u, v) \in E(G)$ .

**Case 2**  $N_G(u) = N_G(x)$ : Let such a set be  $\{y_1, y_2, \ldots, y_k\}$  (Figure 3.11). Since G is k-regular, each of the k edge-disjoint x-v paths in G must start with an edge  $(x, y_i)$ , where  $1 \le i \le k$ . Since any such path can be of length at most two, the edge  $(y_i, v)$  must be in E(G) for all  $i, 1 \le i \le k$ . This implies  $\{y_1, y_2, \dots, y_k\} \subseteq N_G(v)$ . Since  $deg_G(v) = k$ , we have  $N_G(v) = \{y_1, y_2, \dots, y_k\}$ . Hence  $N_G(u) = N_G(v) = \{y_1, y_2, \dots, y_k\}$ . This concludes the proof of the property.  $\Box$ 

**Corollary 3** <sup>1</sup> Given a graph G and vertex set  $\{u, x, v\} \subseteq V(G)$ , if there exists a set of k edgedisjoint u-x paths in G of length at most d and a set of k edge-disjoint x-v paths in G of length at most 2, then there are k edge-disjoint u-v paths in G, each having length d + 2 or less.  $\Box$ 

**Property 7** Given a graph D, if there exists a k-regular graph  $G \in \mathcal{F}_k(D)$  then the diameter of G is at most 2.

**Proof:** The proof of this property immediately follows from Property 6.

**Theorem 3** Given a graph D, if there exists a k-regular graph  $G \in \mathcal{F}_k(D)$  then the order of the graph D (and G) is at most 2k.

**Proof:** Let p = |V(D)| = |V(G)| and  $V_i$  be the set of all vertices that are distance i > 0 away from u in G. By Property 7, we have  $i \le 2$ . Since the graph G is k-regular  $|V_1| = k$  and  $|V_2| = p - k - 1$ . By Property 6, for every vertex  $v \in V_2$  we have  $N_G(u) = N_G(v)$ ; *i.e.*, every vertex in  $V_2$  must be connected to all the k vertices in  $V_1$ . But each vertex in  $V_1$  can be connected to at most k - 1 vertices in  $V_2$ , hence  $p - k - 1 \le k - 1$ . This implies  $p \le 2k$ .

**Corollary 4** For any p-order graph D and any non-negative integer  $k < \frac{p}{2}$ , there does not exist a k-regular graph  $G \in \mathcal{F}_k(D)$ .

<sup>&</sup>lt;sup>1</sup>This corollary follows immediately from the "Induction Step" of proof for Theorem 1.



Figure 3.12:  $K_{5,5} \in \mathcal{F}_5(K_{5,5})$  and diameter of  $K_{5,5}$  is 2.

It is clear from Corollary 4 that for any  $k < \frac{p}{2}$ , the degree of regularity of a graph in  $\mathcal{F}_k(D)$  is at least (k + 1). Before we present our designs for some special topologies, we address the issue of the distance between adjacent vertices upon edge removal.

# **3.6 A Bound on the Diameter of** G

It is interesting to note that there exist graphs D and G such that  $G \in \mathcal{F}_k(D)$  and after removing a set of F edges, |F| < k, in G we have diam(G-F) > diam(D). For instance, consider a complete bipartite graph  $K_{n,n}$ , for any n > 1;  $diam(K_{n,n}) = 2$ . In  $K_{n,n}$ , every pair of non-adjacent vertices are joined by n edge-disjoint paths of length 2. Therefore, we can remove up to n - 1 edges and still preserve distance between non-adjacent vertices in  $K_{n,n}$ . In other words,  $K_{n,n} \in \mathcal{F}_n(K_{n,n})$ . Since n > 1, remove any edge, say (u, v), then we have  $diam(K_{n,n} - \{(u, v)\}) > 2$ . Figure 3.12 shows a complete bipartite graph of order 10. Figure 3.13 shows a  $K_{5,5} - \{(0, 1)\}$  in which vertices 0 and 1 are distance 3 apart. Hence preserving distance may not necessarily preserve the diameter. In the remaining part of this section, we present an upper bound on diam(G - F). First we present a property which is crucial in providing an upper bound for diam(G - F).



Figure 3.13: Diameter of  $K_{5,5} - \{0,1\}$  is 3.

**Property 8** Given a graph D and a graph  $G \in \mathcal{F}_k(D)$  with k < |V(D)|. For any  $F \subseteq E(G)$ with |F| < k, if diam(G - F) > diam(D) then for every pair of vertices u and v in G with  $d_{G-F}(u,v) = diam(G - F)$ , we have  $(u,v) \in E(D)$ .

**Proof:** Suppose otherwise and let  $(u, v) \notin E(D)$ . This implies u and v are non-adjacent vertices in D. Since  $G \in \mathcal{F}_k(D)$ , by definition, there are k edge-disjoint u-v paths in G of length at most  $d_D(u, v)$ . Implying  $diam(G - F) = d_{G-F}(u, v) \leq d_D(u, v) \leq diam(D)$ —a contradiction.  $\Box$ 

**Corollary 5** Given a graph D and a graph  $G \in \mathcal{F}_k(D)$  and a set of edges F, with |F| < k, if diam(G-F) > diam(D) then diam(G-F) is bounded by  $max\{d_{G-F}(u,v)|(u,v) \in E(D)\}$ .  $\Box$ 

For any pair of adjacent vertices u and v in D, the following theorem presents an upper bound on the distance between u and v in G - F. To follow the proof of the theorem a brief definition of eccentricity is necessary. Given a connected graph D and a vertex  $u \in V(D)$ , the eccentricity<sub>D</sub>(u) = max{ $d_D(u, v) | v \in V(D)$ }.

**Theorem 4** Given a graph D and a graph  $G \in \mathcal{F}_k(D)$  with k < |V(D)|, for any pair of adjacent vertices u and v in D and for any  $F \subseteq E(G)$  with |F| < k, we have  $d_{G-F}(u, v) \le 5$ .

**Proof:** The proof is given by constructing a set of k edge-disjoint u-v paths in G of length at most 5. Without loss of generality, assume that  $eccentricity_D(u) \ge eccentricity_D(v)$ . First consider the case when  $eccentricity_D(u) \ge 3$ . Let x be a vertex such that  $d_D(u, x) = 3$ . Let P = u, l, m, x be a u-x path of length 3 in D. Since  $(u, v) \in E(D)$ , we have  $d_D(v, x) \ge 2$ ; otherwise,  $d_D(u, x) < 3$ . Similarly,  $d_D(v, x) \le 4$  because v and x are joined in G by a path of length 4 namely, v, u, l, m, x. In other words, we have  $2 \le d_D(v, x) \le 4$ .

**Case**  $d_D(v, x) = 2$ : We have  $d_D(u, x) = 3$  and  $d_D(v, x) = 2$  and since  $G \in \mathcal{F}_k(D)$ , by Corollary 3, there are k edge-disjoint u-v paths in G, each having length 5 or less.

- **Case**  $d_D(v, x) = 3$ : Let us consider the vertex m in the path P. We know that  $d_D(u, m) = 2$ . Notice that  $d_D(v, m) > 1$ ; Otherwise,  $d_D(v, x) < 3$ . Also,  $d_D(v, m) \le 3$  because v, u, l, mis a v-m path in D. In other words, we have  $2 \le d_D(v, m) \le 3$ . Since  $G \in \mathcal{F}_k(D)$  and  $d_D(u, m) = 2$ , by Corollary 3 we have k edge-disjoint u-v paths in G, each having length at most 5.
- Case  $d_D(v, x) = 4$ : Consider the vertex m in the path P; we have  $d_D(u, m) = 2$ . Since we have a v-m path v, u, l, m in D, we have  $d_D(v, m) \leq 3$ . But note that if  $d_D(v, m) < 3$ , then  $d_D(v, x) < 4$ , a contradiction. Hence  $d_D(v, m) = 3$ . By Corollary 3, we have at least kedge-disjoint u-v paths in G, each having length at most 5.

Now let us consider the case when  $eccentricity_D(u) \le 2$  and  $eccentricity_D(v) \le 2$ . If  $eccentricity_D(u) = 1$  or  $eccentricity_D(v) = 1$ , then all the vertices in D are adjacent to either uor v. However, if  $eccentricity_D(u) = 2$  and  $eccentricity_D(v) = 1$ , since  $\delta(G) \ge k$ , it is easy to see that  $|N_G[u] \cap N_G[v]| \ge k$ . This implies that there k edge-disjoint u-v paths of length at most 2. Hence, without loss of generality, we assume that  $eccentricity_D(u) = 2$  and  $eccentricity_D(v) =$ 2. If there exist a vertex w which is distance 2 away from both u and v in D then, by Corollary 3, we have k edge-disjoint u-v paths in G, each having length at most 4 and we are done. If such a w does not exist then each vertex in V(D) is either adjacent to u or adjacent to v. In such a case, we claim that there are k edge-disjoint u-v paths in G of length at most 3. In the remaining part of the proof, we enumerate a set of k edge-disjoint u-v paths in G, each having length at most 3.

Since  $(u, v) \in E(G)$ , there is one u-v path of length 1 in G. Let  $L = N_G(u) \cap N_G(v)$  and let  $l = |L| \ge 0$ . By using the vertices in L, we can construct a set of l edge-disjoint u-v paths of length 2. So far, we have l + 1 number of edge-disjoint paths of length at most 2. If  $k \le l + 1$ , we are done. Otherwise, since  $\delta(G) \ge k$  we have at least (k - l - 1) unused vertices adjacent to u. Similarly, there are at least (k - l - 1) unused vertices adjacent to v. Let us denote by M and N the set of unused vertices adjacent to u and v, respectively; It is easy to see that  $M \cap N_G(v) = \emptyset$ and  $N \cap N_G(u) = \emptyset$ . Since k > l + 1, we have  $M \neq \emptyset$ ,  $N \neq \emptyset$ . Notice that each vertex in M is distance 2 away from v in D, implying that there are at least k edge-disjoint paths in G of length at most 2 from each vertex in M to v. Consider any vertex  $u_i \in M$ . Since  $M \cap N_G(v) = \emptyset$ , by lemma 1, we must have  $N_G(u_i) \cap N_G(v) \ge k$ . However,  $u_i$  can at most be connected to l+1vertices of  $L \cup \{u\}$ . The remaining (k - l - 1) vertices in  $N_G(u_i) \cap N_G(v)$  must belong to N. In other words, for each  $u_i \in M$  we have  $|N_G(u_i) \cap N| \ge (k-l-1)$ . Since  $|M| \ge (k-l-1)$ there exist a matching of size at least (k - l - 1) from vertices in M to the vertices in N. Since all vertices in M are connected to u and all vertices in N are connected to v, there are (k - l - 1)internally disjoint u-v paths, each having length 3 in G. To summarize this case, there is one u-vpath of length 1,  $l u \cdot v$  paths of length 2 and  $(k - l - 1) u \cdot v$  paths of length 3 in G. 

**Corollary 6** Given a graph D and a graph  $G \in \mathcal{F}_k(D)$ , with k < |V(D)|. For any  $F \subseteq E(G)$  with |F| < k, we have  $diam(G - F) \le max\{5, diam(D)\}$ .

In this chapter, we formulated two design models and their tight lower and upper bounds. In the next chapter we consider certain special topologies as D and construct distance preserving graphs for Model 2.

# Chapter 4

# **Special Topologies**

In this chapter, we design fault-tolerant topologies for cycles, crowns, chordal rings and n-ary 2cubes. These graphs are highly symmetric and attractive for communication networks due to their regularity, symmetry, and small diameter. The connectivity and reliability of this class of graphs have been studied extensively by many researchers [2, 28, 42, 43, 44, 45].

**Definition 6** A cycle of order p, referred as  $C_p$ , can be defined as a graph with vertex set  $\{v_0, v_1, \ldots, v_{p-1}\}$  and for each vertex  $v_i$ ,  $0 \le i < p$ , we have  $(v_i, v_{(i+1) \mod p}) \in E(C_p)$ .

A  $C_{16}$  is shown in Figure 4.1.

**Definition 7** <sup>1</sup> A r-regular crown, called r-crown, of order p is a bipartite graph with partitions  $V_0$ = { $v_0, v_2, \ldots, v_{p-2}$ } and  $V_1$  = { $v_1, v_3, \ldots, v_{p-1}$ }; each  $v_i$  in  $V_0$  is connected to  $v_{(i+2j-1) \mod p}$ ,  $0 \le j < r$ .

A 5-crown of order 16 is shown in Figure 4.2.

**Definition 8** A chordal ring C(p,h), where p is even and  $h \leq \frac{p}{2}$  is odd, can be defined as a 3-regular graph with vertex set  $\{v_0, v_1, \ldots, v_{p-1}\}$ , and for each even vertex  $v_i$  in V(C(p,h)),

<sup>&</sup>lt;sup>1</sup>Since  $|V_1| = \frac{p}{2}, p \ge 2r$ .



Figure 4.1:  $C_{16}$ .



Figure 4.2: A 5-regular crown of order 16.



Figure 4.3: A C(16, 7).

 $(v_i, v_{(i+1) \mod p}) \in E(C(p, h)) \text{ and } (v_i, v_{(i+h) \mod p}) \in E(C(p, h)).$ 

A generalized chordal ring  $C(p, < h_1, h_2, ..., h_r >)$  can similarly be defined with multiple odd chords  $h_1 < h_2 < ... < h_r \le \frac{p}{2}$ . Figures 4.3 and 4.4 show a C(16, 7) and C(16, < 5, 7 >), respectively.

**Definition 9** A n-ary 2-cube can be defined as a graph, say G, of order  $n \times n$  with vertex set  $\{v_{(i,j)}|0 \leq i, j < n\}$ . And for each vertex  $v_{(i,j)}$  we have  $(v_{(i,j)}, v_{(((i+1) \mod n),j)}) \in E(G)$  and  $(v_{(i,j)}, v_{(i,((j+1) \mod n))}) \in E(G)$ .

A 8-ary 2-cube is shown in Figure 4.5.

### 4.1 Cycles

In this section, we show that a class of (k + 1)-regular graphs is k-edge distance-preserving with respect to  $C_p$ , a cycle of order p. For even values of k, the regularity of the distance-preserving graphs is odd. For a regular graph, both the degree and order cannot be odd. Hence to give a general construction, we assume that the order of the graph is even.

42



Figure 4.4: A C(16, < 5, 7 >).



Figure 4.5: A 8-ary 2-cube.



Figure 4.6: A 3-crown on 12 vertices.

**Theorem 5** A p-order (k + 1)-crown belongs to  $\mathcal{F}_k(C_p)$ .

**Proof:** Since crowns are Hamiltonian [28],  $C_p$  is a spanning subgraph of a *p*-order (k + 1)-crown. Due to Theorem 1, it suffices to show that each pairs of vertices that are distance two apart in  $C_p$  are joined by k edge-disjoint paths of length at most two, and vertices that are distance three apart in  $C_p$  are joined by k edge-disjoint paths of length at most three. Note that in a  $C_p$ , only vertices of form  $v_i$  and  $v_{i+2}$  are distance two apart, and vertices of form  $v_i$  and  $v_{i+3}$  are distance three apart.

The following notation will be useful in the remaining part of the proof. If  $\{v_0, v_1, \ldots, v_{p-1}\}$ is the vertex set of a graph then the vertex  $v_i$  is referred to as an *even* vertex if i is even and odd otherwise. In a k-crown, if  $v_i$  is an even vertex, we call the edge  $(v_i, v_{i+2j-1}), 0 \le j < k$ , as the  $j^{th}$  forward chord (FC) from vertex  $v_i$  and the  $j^{th}$  backward chord (BC) from vertex  $v_{i+2j-1}$ . Specifically, if  $v_l$  is an odd vertex, the edge  $(v_l, v_{l-2j+1})$  is the  $j^{th}$  BC from  $v_l$ . Each even vertex in a k-crown has k FCs, and each odd vertex has k BCs.

An example of a 3-regular crown on 12 vertices is shown in Figure 4.6. The edge (0,1) in Figure 4.6 is called the first FC from vertex 0 and the same edge is called as the first BC from vertex 1.

First let us consider a pair of vertices, say  $v_i$  and  $v_{i+2}$ , that are distance two apart in  $C_p$ . Since crowns are known to be *point-symmetric graphs* [42], without loss of generality, we assume that  $v_i$ is an even vertex. To prove that there are k edge-disjoint paths of length two between  $v_i$  and  $v_{i+2}$ ,



Figure 4.7: A set of k edge-disjoint paths between  $v_i$  and  $v_{i+2}$  of length 2 in a (k + 1)-crown.



Figure 4.8: A set of k - 3 edge-disjoint paths of length 3 in a (k + 1)-crown.

it suffices to show that  $|N(v_i) \cap N(v_{i+2})| = k$ . By definition, in a (k+1)-crown we have  $N(v_i) = \{v_{i+2j-1} \in V(G) \mid 0 \le j \le k\}$ . Similarly,  $N(v_{i+2}) = \{v_{i+2j+1} \in V(G) \mid 0 \le j \le k\}$ . It is easy to see that  $|N(v_i) \cap N(v_{i+2})| = k$ .

A set of k edge-disjoint paths of length 2 between  $v_i$  and  $v_{i+2}$  is shown in Figure 4.7.

Let us consider a pair of vertices,  $v_i$  and  $v_{i+3}$ , that are distance three apart in  $C_p$ . First we will construct a set of three paths between vertices  $v_i$  and  $v_{i+3}$  of length at most three. These paths are,  $P_1 = \{v_i, v_{i+1}, v_{i+2}, v_{i+3}\}, P_2 = \{v_i, v_{i+3}\}, \text{ and } P_3 = \{v_i, v_{i+2k-1}, v_{i+4}, v_{i+3}\}$ . It is obvious that the above three paths between  $v_i$  and  $v_{i+3}$  are pair wise edge (vertex) disjoint. The next (k-3) paths are as follows. For each  $j, 3 \le j \le k - 1$ , we have,  $P_j = \{v_i, v_{i+2j-1}, v_{i+2j-2k}, v_{i+3}\}$ . Now we will show that the above (k - 3) paths are edge (vertex) disjoint.

Let  $A_i = \{v_{i+5}, v_{i+7}, \dots, v_{i+2j-1}, \dots, v_{i+2k-3}\} = \{v_{i+2j-1} \in V(G) \mid 3 \le j \le k-1\} \subseteq N(v_i)$ . Note that all the vertices in  $A_i$  are odd, distinct, and unused. Similarly, let  $A_{i+3} = \{v_{i-2}, v_{i-4}, \dots, v_{i+2j-2k}, \dots, v_{i+6-2k}\} = \{v_{i+2j-2k} \in V(G) \mid 3 \le j \le k-1\} \subseteq N(v_{i+3})$ . Note that all the vertices in  $A_{i+3}$  are even, distinct, and unused.

Now take the  $k^{th}$  BC from each vertex in  $A_i$ . The set of vertices reached by using the  $k^{th}$  BC from each vertex in  $A_i$  can be given as  $\{v_{(i+2j-1)-(2k-1)} \mid 3 \leq j \leq k-1\} = \{v_{(i+2j-2k)} \mid 3 \leq j \leq k-1\} = A_{i+3}$ . Hence from the (k-3)-vertices in  $A_i$  we can reach the (k-3) vertices in  $A_{i+3}$ , using k-3 distinct edges. In a (k+1)-crown we have  $N(v_i) \cap N(v_{i+3}) = \emptyset$ . This implies that the constructed set of (k-3) paths between  $v_i$  and  $v_{i+3}$  are pair wise edge (in fact, vertex) disjoint. The set of (k-3) edge-disjoint paths between  $v_i$  and  $v_{i+3}$  is shown in Figure 4.8. This concludes the proof of the theorem.

Using Corollary 4, it is easy to see that the above construction is optimal. Figure 4.9 shows a set of 4 edge-disjoint paths of length at most 3 between  $v_0$  and  $v_3$  in a 5-regular 16-order crown.

**Theorem 6** A complete bipartite graph on p vertices,  $K_{\frac{p}{2},\frac{p}{2}} \in \mathcal{F}_{\frac{p}{2}}(C_p)$ .

**Proof:** It is easy to see that a  $C_p$  is spanning subgraph of a complete bipartite graph. Due to Theorem 1, it suffices to show (1) that the vertices that are distance two apart in  $C_p$  are joined by k edge-disjoint paths of length at most two, and (2) vertices that are distance three apart in  $C_p$  are joined by k edge-disjoint paths of length at most three. Note that in a  $C_p$ , only vertices of form  $v_i$ and  $v_{i+2}$  are distance two apart, and vertices of form  $v_i$  and  $v_{i+3}$  are distance three apart. Also, the vertices that are distance 2 apart in  $C_p$  belong to the same partition of the bipartite graph. Hence, it is easy to see that any such vertex pair has exactly  $\frac{p}{2}$  vertices common to both vertices. Now let



Figure 4.9: A set of 4 edge-disjoint paths between  $v_0$  and  $v_3$ .

us consider the vertices that are distance 3 away. Note that in a  $C_p$  the vertex pairs that are distance 3 apart are of the form  $v_i$  and  $v_{i+3}$ . Since the bipartite graphs are point symmetric, without loss of generality, assume that such a pair is  $v_0$  and  $v_3$ . The k edge-disjoint paths can be given as,

 $1st path : v_0, v_1, v_2, v_3.$ 

 $2nd path : v_0, v_3.$ 

 $3rd - k^{th}paths: v_0, v_{2i+1}, v_{2i}, v_3$ , where  $2 \le i < k - 1$ .

The proof of Theorem 5 can be used to show that the given k edge-disjoint paths are edge (vertex) disjoint.

The above two theorems provide an optimal solution for Model 2 when  $D = C_p$  and  $k \le \frac{p}{2}$ . In the remaining part of this section, we construct distance preserving graphs for other topologies such as crowns, chordal rings, and *n*-ary 2-cubes.

47

### 4.2 Crowns

**Theorem 7** A p-order (r + k - 1)-crown belongs to  $\mathcal{F}_k(r$ -crown), for any non-negative integer  $k \leq \frac{(p-2r+2)}{2}$ .

**Proof:**<sup>2</sup> By definition, a *p*-order *r*-crown is a spanning subgraph of a *p*-order (r + k - 1)-crown. Due to Theorem 1, it suffices to show that (1) each pairs of vertices that are distance two apart in an *r*-crown are joined by *k* edge-disjoint paths of length at most two, and (2) vertices that are distance three apart in an *r*-crown are joined by *k* edge-disjoint paths of length at most three. Since crowns are known to be *point-symmetric graphs* [42], without loss of generality, we assume that our "source" vertex is  $v_0$ . First, we will consider vertices that are distance two apart in an *r*-crown. Note that in an *r*-crown all the pairs of vertices that are distance two apart are of the form  $v_0$  and  $v_{2l}$ ,  $1 \le l < r$ . Figure 4.10 shows a 5-crown and vertices that distance 2 away from  $v_0$ . It is easy to see that in an (r + k - 1)-crown, the *k* edge-disjoint paths joining  $v_0$  to  $v_{2l}$ ,  $1 \le l < r$ , are  $v_0, v_{2j-1}, v_{2l}$ , where  $r - 1 \le j \le r + k - 2$ .

Now, we will consider all the vertices in an r-crown that are distance 3 away from  $v_0$ . In an r-crown, it is easy to verify that vertices  $v_{2r-2}$  and  $v_{p-2r+2}$  span all the vertices that are distance 3 away from  $v_0$ . In other words, a shortest path to all the distance 3 away vertices from  $v_0$  can be given by,

(a)  $v_0, v_{2r-3}, v_{2r-2}, v_{2r+2l-3}$ , for some  $l, 1 \le l < r$ , and

(b)  $v_0, v_{p-1}, v_{p-2r+2}, v_{p-2r+2l-1}$ , for some  $l, 1 \le l < r$ .

Therefore, if a vertex  $v_j$  is distance 3 away from  $v_0$ , we have either  $2r - 1 \le j \le 4r - 5$  or,  $p - 2r + 1 \le j \le p - 3$ . Figure 4.10 shows a 5-crown and vertices that distance 3 away from  $v_0$ .

<sup>&</sup>lt;sup>2</sup>Note that a *p*-order (r + k - 1)-crown is only defined when  $2(r + k - 1) \le p$ .



Figure 4.10: Vertices that are distance 2 and 3 away from vertex  $v_0$ .

Given a vertex  $v_j$  which is distance 3 away from  $v_0$  in an r-crown, the k edge-disjoint paths joining  $v_0$  and  $v_j$  in an (r + k - 1)-crown can be given as:

Case  $2r - 1 \le j \le 4r - 5$ : We construct a set of k edge-disjoint paths by considering two sets of k vertices that are adjacent to  $v_0$  and  $v_j$ . Since all the vertices adjacent to  $v_0$  are odd<sup>3</sup> and all the vertices adjacent to  $v_j$  are even, by definition of an (r + k - 1)-crown, we have  $N(v_0) \cap N(v_j) = \emptyset$ . However, we have to show that all the k vertices used in  $N(v_0)$  are distinct and all the k vertices used in  $N(v_j)$  are distinct. The construction is divided into three subcases. A brief overview of each subcase is as follows. In subcase 1, we construct a set of  $min(r, k)^4$  paths. In these min(r, k) paths,  $v_0$  is adjacent to odd vertices between<sup>5</sup>  $v_0$  and  $v_{2r-2}$  and  $v_j$  is adjacent to even vertices between  $v_0$  and  $v_{2r-2}$ . In subcase 2, we construct an r + 1<sup>th</sup> path.

<sup>&</sup>lt;sup>3</sup>We call a vertex  $v_i$  odd(even), if i is odd(even).

 $<sup>^{4}</sup>min(r, k)$  refers to the minimum of r and k

<sup>&</sup>lt;sup>5</sup>By "between  $v_i$  and  $v_j$ " we refer to the vertices on the clockwise  $v_i - v_j$  path in the spanning cycle of the given graph, as can be seen in Figure 4.11.

In subcase 3, we construct the rest of the k - r - 1 paths. In these paths  $v_0$  is adjacent to  $v_{2(r+k-2)-3}, v_{2(r+k-2)-5}, \ldots, v_{2(r+k-2)-2(k-r-1)-1}$ . Consider the vertex adjacent to  $v_0$  in the  $(k-r-1)^{st}$  path. Since 2(r+k-2) - 2(k-r-1) - 1 > 2r - 3, the paths constructed in subcase 3 would not reuse the odd vertices used in subcase 1. Hence all the selected k vertices adjacent to  $v_0$  are distinct. Similarly in the paths constructed in this subcase,  $v_j$  is adjacent to  $v_{p-2}, v_{p-4}, \ldots, v_{p-2(k-r-1)}$ . Consider the vertex adjacent to  $v_j$  in the  $(k-r-1)^{st}$  path. By definition of an (r+k-1)-crown we have,  $p \ge 2(r+k-1)$  and since r is a positive integer, we have p - 2(k-r-1) > 2r - 2, which guarantees that the even vertices used in subcase 1 are not reused. Now we are ready to construct a set of k edge-disjoint  $v_0 - v_j$  paths of length at most 3 in an (r + k - 1)-crown.

- Subcase 1: For each  $0 \le l < min(r, k)$ , there exist a path  $v_0, v_{(2r-2)-2l-1}, v_{(2r-2)-2l}, v_j$ . By definition  $v_0$  is adjacent to  $v_{(2r-2)-2l-1}$ . In an r-crown vertex  $v_j$  is adjacent to  $v_{2r-2}$ , which implies that in an (r + k - 1)-crown vertex  $v_j$  is connected to at least k even vertices before  $v_{2r-2}$ . In other words,  $v_j$  is connected to  $v_{(2r-2)-2l}$ . Note that when  $k \ge r$ , and l = r - 1, the above given path is  $v_0, v_{-1}, v_0, v_j$ . In this case, we should consider the edge  $v_0$ - $v_j$  as a path.
- Subcase 2: If k = r + 1, there exists a path  $v_0, v_{2(r+k-2)-1}, v_{j+1}, v_j$ . It is easy to see that the edge  $(v_0, v_{2(r+k-2)-1})$ , and  $(v_{j+1}, v_j)$  exist in a (r + k - 1)-crown. It remains to show that  $v_{j+1}$  is joined to  $v_{2(r+k-2)-1}$  in an (r + k - 1)-crown. By definition, in an (r + k - 1)-crown the vertex  $v_{j+1}$  is joined to all odd vertices in between  $v_{(j+1)}$  and  $v_{(j+1)+2(r+k-2)-1}$ . Since j is positive,  $2r - 1 \le j \le 4r - 5$ , and k = r + 1, we can see that  $v_{2(r+k-2)-1}$  is an odd vertex in between  $v_{(j+1)}$  and  $v_{(j+1)+2(r+k-2)-1}$ . Which implies, vertex  $v_{j+1}$  is joined to  $v_{2(r+k-2)-1}$ .



Figure 4.11: k edge-disjoint paths between  $v_0$  and  $v_j$  when  $2r - 1 \le j \le 4r - 5$ .



Figure 4.12: k edge-disjoint paths between  $v_0$  and  $v_j$  when  $p - 2r + 1 \le j \le p - 3$ .

Subcase 3: If k > r + 1, there exists a path  $v_0, v_{(p-2l-2)+2(r+k-2)-1}, v_{p-2l-2}, v_j$ , for each  $0 \le l < k - r - 1$ . Since k > r + 1, vertex  $v_j$  is connected to  $v_{p-2l-2}i, 0 \le l < k - r - 1$ . Hence, it is enough to show that  $v_0$  is adjacent to  $v_{(p-2l-2)+2(r+k-2)-1}$ , for each  $0 \le l < k - r - 1$ . We know that vertex  $v_0$  is adjacent to all odd vertices between  $v_1$  and  $v_{2(r+k-2)-1}$ . Note that  $p - 2l - 2 + 2(r+k-2) - 1 \ge p$  and since vertex index operations are taken over modulo p, it remains to show that  $1 \le 2(r+k-2)-1-2l-2 \le 2(r+k-2) - 1$ . Since k > r + 1 > 2 and  $0 \le l < k - 3$ , the above inequality holds.

Figure 4.11 shows a set of k edge-disjoint paths between  $v_0$  and  $v_j$  when k > r.

Case  $p - 2r + 1 \le j \le p - 3$ : In this case, we consider vertices  $v_{(j-2l+1)}$ ,  $0 \le l < k$ , which are adjacent to  $v_j$ . From each of the k vertices, traverse through the  $(r + k - 2)^{th}$  chord to reach a vertex adjacent to  $v_0$ . Such a set of k edge-disjoint paths can be given as follows. For each  $0 \le l < k$ ,  $v_0, v_{(j-2l+1)+2(r+k-2)-1}, v_{j-2l+1}, v_j$ . Now, we will show that vertex  $v_0$  is adjacent to vertex  $v_{(j-2l+1)+2(r+k-2)-1}$ . Since,  $p - 2r + 1 \le j \le p - 3$  and vertex index operations are taken over modulo p, the vertex  $v_{(j-2l+1)+2(r+k-2)-1}$  is in between  $v_{2k-2l-3}$ and  $v_{2(r+k-2)-(2l+3)}$ . It is easy to see that for all values of l, vertex  $v_0$  is adjacent to odd vertices in between  $v_{2k-2l-3}$  and  $v_{2(r+k-2)-(2l+3)}$ . Since we use the  $(r + k - 2)^{th}$  chord from each of the k distinct vertices to reach a vertex adjacent to  $v_0$ , all the vertices used in  $N(v_0)$  are distinct.

Figure 4.12 shows a set of k edge-disjoint paths between  $v_0$  and  $v_j$ . Figure 4.13 shows a 5-regular crown and a set of 3 edge-disjoint paths between vertices  $v_0$  and  $v_7$  which are distance 3 apart in a 3-regular crown.



Figure 4.13: A set of 3 edge-disjoint paths between  $v_0$  and  $v_7$ .



Figure 4.14: A C(28, 7).



Figure 4.15:  $C(28, < 3, 7, 9 >) \in \mathcal{F}_2(C(28, 7))$ .

### 4.3 Chordal Graphs

**Theorem 8** Given non-negative integers p, k, and h, and a three regular chordal graph C(p, h), we have  $C(p, < 1, 3, ..., 2k - 1, h, h + 2, ..., h + 2k >) \in \mathcal{F}_k(C(p, h))$ .

**Proof:** By definition, a C(p, h) is a spanning subgraph of C(p, < 1, 3, ..., 2k-1, h, h+2, ..., h+2k >). Also, by definition, C(p, < 1, 3, ..., 2k-1, h, h+2, ..., h+2k >) has the following edges,

- 1.  $(v_i, v_{(i+2l-1)}), 0 \le l \le k$  for any even vertex  $v_i$ .
- 2.  $(v_i, v_{(i+h+2l)}), 0 \le l < k$  for any even vertex  $v_i$ .
- 3.  $(v_j, v_{(j-2l+1)}), 0 \le l \le k$  for any odd vertex  $v_j$ .
- 4.  $(v_j, v_{(j-h-2l)}), 0 \le l < k$  for any odd vertex  $v_j$ .

Destination	k edge-disjoint paths	
v <sub>2</sub>	$v_0, v_{(h+2i)}, v_{(h+2i)-h-2(i-1)}$ $v_0, v_1, v_2$	$1 \le i < k$ $k^{th}$ path
$v_{h-1}$	$v_0, v_{(h+2i)}, v_{(h+2i)-2(i+1)+1}$	$0 \leq i < k$
$v_{h+1}$	$v_0, v_{(h+2i)}, v_{(h+2i)-2i+1}$	$0 \leq i < k$

Table 4.1: k edge-disjoint paths to vertices that are distance 2-away from  $v_0$ .

Due to Theorem 1, it suffices to show that (1) vertices that are distance two apart in a C(p, h)are joined by k edge-disjoint paths of length at most two, and (2) vertices that are distance three apart in a C(p, h) are joined by k edge-disjoint paths of length at most three. Since chordal rings are point symmetric [42], without loss of generality, we assume that our source vertex is  $v_0$ . First let us consider vertices that are distance two apart from  $v_0$ . In a C(P, h) all vertex pairs that are distance two away from each other are of the form  $v_0$  and  $v_2$ ,  $v_0$  and  $v_{h-1}$  and,  $v_0$  and  $v_{h+1}$ . The k edge-disjoint paths of length at most two for the above three pairs are given in Table 4.1. It is easy to see that the given k paths are edge (in fact, vertex) disjoint.

Now let us consider the vertices that are distance three apart. Note that, in a C(p, h), all the pairs of vertices that are distance 3-apart from  $v_0$  are odd and of the form  $(v_0, v_3)$ ,  $(v_0, v_{h-2})$ ,  $(v_0, v_{h+2})$ ,  $(v_0, v_{2h-1})$ ,  $(v_0, v_{2h+1})$ ,  $(v_0, v_{-2-h})$ ,  $(v_0, v_{-h})$ ,  $(v_0, v_{2-h})$  and  $(v_0, v_{p-3})$ . The k edge-disjoint paths of length at most three for the above pairs are given in Table 4.2. By the definition of a generalized chordal ring, for any vertex,  $v_l$ , that is distance 3 away from  $v_0$ , we have  $N(v_0) \cap N(v_l) = \emptyset$ . Hence, it is easy to see that the given k paths are edge (in fact, vertex) disjoint.

A C(28,7) is shown in Figure 4.14. A graph in  $\mathcal{F}_k(C(28,7))$  is shown in Figure 4.15. Figures 4.16 and 4.17 show the two edge-disjoint paths of length 3 between vertex pairs 0, 15 and 0, 19 respectively.

Destination	k edge-disjoint paths	
υვ	$ \begin{array}{c} v_0, v_{(h+2l)}, v_{(h+2l)-(h+2(k-2))}, v_{(h+2l)-(h+2(k-2))+(2(k-l)-1)} \\ v_0, v_1, v_2, v_3 \\ v_0, v_3 \end{array} $	$0 \le l < k - 2$ (k - 1) <sup>th</sup> path k <sup>th</sup> path
$v_{h-2}$	$ \begin{array}{c} v_0, v_{(2l-1)}, v_{(2l-1)-(2k-1)}, v_{(2l-1)-(2k-1)+(h+2(k-l-1))} \\ v_0, v_h, v_{h-1}, v_{h-2} \end{array} $	$1 \le l < k$ $k^{th}$ path.
$v_{h+2}$	$v_0, v_{(2l+1)}, v_{(2l+1)-(2k-1)}, v_{(2l+1)-(2k-1)+(h+2(k-l))}$ $v_0, v_1, v_2, v_{h+2}$	$1 \le l < k$ $k^{th}$ path
$v_{2h-1}$	$v_0, v_{(h+2l)}, v_{(h+2l)-(2k-1)}, v_{(h+2l)-(2k-1)+(h+2(k-l-1))}$	$0 \leq l < k$
$v_{2h+1}$	$v_0, v_{(h+2l)}, v_{(h+2l)-(2(k-1)-1)}, v_{(h+2l)-(2(k-1)-1)+(h+2(k-l-1))}$	$0 \leq l < k$
$v_{-2-h}$	$v_0, v_{(2l-1)}, v_{(2l-1)-(h+2(k-1))}, v_{(2l-1)-(h+2(k-1))+2(k-l-1)-1}$	$0 \leq l < k$
$v_{-h}$	$v_0, v_{(2l-1)}, v_{(2l-1)-(h+2(k-1))}, v_{(2l-1)-(h+2(k-1))+2(k-l)-1}$	$0 \leq l < k$
$v_{2-h}$	$v_0, v_{(2l+1)}, v_{(2l+1)-(h+2(k-1))}, v_{(2l+1)-(h+2(k-1))+2(k-l)-1}$	$0 \leq l < k$
$v_{p-3}$	$ \begin{array}{l} v_0, v_{(2l-1)}, v_{(2l-1)-(h+2(k-1))}, v_{(2l-1)-(h+2(k-1))+(h+2(k-l-2))} \\ v_0, v_{(2k-3)}, v_{(2k-3)-(2k-1)}, v_{-3} = v_{p-3} \end{array} $	$0 \le l < k - 1$ $k^{th}$ path

Table 4.2: k edge-disjoint paths to vertices that are distance 3-away from  $v_0$ .



Figure 4.16: Edge disjoint paths between 0 and 15 in a C(28, <3, 7, 9>).



Figure 4.17: Edge disjoint paths between 0 and 19 in a C(28, < 3, 7, 9 >).

# 4.4 *n*-ary 2-cube

**Theorem 9** Given a n-ary 2-cube, say D, and a non-negative integer k < n, we show that there exist a (2k + 2)-regular graph in  $\mathcal{F}_k(D)$ .

**Proof:** First we will present the construction of a (2k + 2)-regular graph. We partition the edge set of the given *n*-ary 2-cube into *n* cycles of size 2n each. Using the solution for  $C_{2n}$ , we construct a *k*-edge distance preserving graph for each of the cycles. We will then show that the line disjoint sum of these graphs is (2k + 2)-regular and *k*-edge distance preserving with respect to the given *n*-ary 2-cube. Note that, the edges are partitioned in such a way that the distance between any two vertices in any partition is same as the distance between them in the original graph D. The n cycles of size 2n are given as follows:

Given a vertex (i, j) it belongs to  $(i - j)^{th}$  and  $(i - j - 1)^{th}$  cycles<sup>6</sup>. Similarly given any edge of D, say  $((i_1, j_1), (i_2, j_2))$  it belongs to  $(i_1 - j_1 - b)^{th}$  cycle, where b=1 if  $i_1 = i_2$ . It is easy to see that each of the partitions are mutually edge disjoint. Figure 4.18 shows cycle 0 and cycle 5 in a 8-ary 2-cube.

Now we will present some notation that will be useful in the remaining part of the proof. We label a vertex (i, j) even if (i + j) is even and odd otherwise<sup>7</sup>. Two cycles, say p and q, are said to be neighbors if |p - q| = 1. Note that in any two neighboring cycles, we have either the same set of even vertices or the same set of odd vertices.

For each cycle we construct a (k + 1)-crown with even vertices in one partition and odd vertices in another partition. We will name the edges of the resulting (k + 1)-crown as forward or backward

<sup>&</sup>lt;sup>6</sup>All the vertex indices and cycle numbers are taken over modulo n.

<sup>&</sup>lt;sup>7</sup>By the way we partitioned, each cycle contains an alternative sequence of even and odd vertices.



Figure 4.18: 0<sup>th</sup> cycle and 5<sup>th</sup> cycle in a 8x8 torus.



Figure 4.19: A 4-crown (k = 3) for the 4<sup>th</sup> cycle in a 8x8 torus.



Figure 4.20: A 4-crown (k = 3) for the 5<sup>th</sup> cycle in a 8x8 torus.

chord as follows:

$$\begin{array}{ll} \text{If } (i,j) \text{ is even } & ((i,j),(i+l,j+l-1)) & 0 \leq l \leq k & l^{th} \text{ FC from } (i,j) \text{ in cycle } (i-j). \\ & ((i,j),(i+l-1,j+l)) & 0 \leq l \leq k & l^{th} \text{ FC from } (i,j) \text{ in cycle } (i-j-1). \\ & \text{If } (i,j) \text{ is odd } & ((i,j),(i-l+1,j-l)) & 0 \leq l \leq k & l^{th} \text{ BC from } (i,j) \text{ in cycle } (i-j). \\ & ((i,j),(i-l,j-l+1)) & 0 \leq l \leq k & l^{th} \text{ BC from } (i,j) \text{ in cycle } (i-j-1). \end{array}$$

Note that in an  $(i - j)^{th}$  cycle, the  $l^{th}$  FC from (i, j) can also be called  $l^{th}$  BC from (i + l, j + l - 1). Figures 4.19 and 4.20 show the  $4^{th}$  and  $5^{th}$  cycles in a 8-ary 2-cube and the corresponding 4-crowns. These two neighboring cycles have the same set of odd vertices. Figure 4.20 also shows the FCs and BCs from vertices (4, 6) and (3, 6), respectively. By construction, an even and odd vertex pair can belong to no more than one cycle. And since a (k + 1)-crown is constructed by connecting even vertices and odd vertices, it is easy to see that the (k + 1)-crowns are mutually edge disjoint. Since each vertex belongs to two cycles, the degree of the resulting line disjoint union of the (k + 1)-crowns is a (2k + 2)-regular graph.

60



Figure 4.21: Vertices that are distance 2 away from (i, j) in *n*-ary 2-cube.

In the remaining part of the proof, we will show that the above constructed graph is k-edge distance preserving with respect to n-ary 2-cube. Using Theorem 1, it is enough to show that there are k-edge disjoint paths of length at most 2 and 3 between vertices that are distance 2 and 3 apart, respectively. Since a n-ary 2-cube is point symmetric, without loss of generality, we assume that (i, j) is even.

As shown in Figure 4.21, vertex (i, j) has the following vertices that are distance 2 away: (i, j + 2), (i + 1, j + 1), (i + 2, j), (i + 1, j - 1), (i, j - 2), (i - 1, j - 1), (i - 2, j) and (i - 1, j + 1). By the way we partitioned, vertices (i - 1, j - 1), (i, j) and (i + 1, j + 1) belong to  $(i - j - 1)^{th}$  cycle and the vertices (i - 1, j - 1) and (i, j) are distance two apart in the cycle. Similarly, vertices (i, j) and (i + 1, j + 1) are also distance two apart in the same cycle. Hence, by construction, there are k-edge disjoint paths of length at most 2. Since a n-ary 2-cube is a point symmetric graph it is enough to show that there are k-edge disjoint paths from (i, j) to each of (i + 2, j), (i + 1, j - 1) and (i, j - 2).

Between (i, j) and (i + 2, j), we show that there are k-edge disjoint paths of length two by considering two neighboring cycles containing (i, j) and (i + 2, j), respectively. Consider the
Destination	Cycles	Common vertices
(i+2,j)	(i-j) and $(i-j+1)$	$(i+l,j+l-1), 1 \le l \le k$
[(i+1,j+1)]	(i-j) and $(i-j+1)$	$(i+l, j+l-1), 0 \le l < k$
(i, j-2)	(i-j) and $(i-j+1)$	$(i+l, j+l-1), 0 \le l < k$

Table 4.3: k-edge disjoint paths to vertices that distance two away from (i, j).



Figure 4.22: Vertices that are distance 3 away from (i, j) in *n*-ary 2-cube.

 $(i-j)^{th}$  cycle and  $(i-j+1)^{th}$  cycle. It is easy to see that both of these cycles contain the same set of odd vertices. Similarly, the neighborhood of N(i,j) in the  $(i-j)^{th}$  cycle is (i+l,j+l-1),  $0 \le l \le k$ . The neighborhood of N(i+2,j) in the  $(i-j+1)^{th}$  cycle is (i+l+1,j+l),  $0 \le l \le k$ . Hence, the k common vertices are (i+l,j+l-1),  $1 \le l \le k$ . A similar proof can be given for the k-edge disjoint paths from (i,j) to (i+1,j-1) and (i,j-2). Table 4.3 summarizes the k-edge disjoint paths that are distance 2 away from (i,j).

Now we will consider the vertices that are distance 3 away. As shown in Figure 4.22, the vertex (i, j) is distance 3 away from (i, j + 3), (i + 1, j + 2), (i + 2, j + 1), (i + 3, j), (i + 2, j - 1), (i + 3, j)



Figure 4.23: k-edge disjoint paths between (i, j) and (i + 3, j) in a n-ary 2-cube.

1, j-2), (i, j-3), (i-1, j-2), (j-2, j-1), (i-3, j), (i-2, j+1), and (i-1, j+2). Vertices (i-2, j-1) and (i+1, j+2) belong to  $(i-j-1)^{th}$  cycle and hence by construction we have k-edge disjoint paths of length at most 3. Similarly, vertices (i-1, j-2) and (i+2, j+1) belong to (i-j) cycle and by construction we have k-edge disjoint paths of length at most 3. Since a n-ary 2-cube is a point symmetric graph, it is enough to show that there are k-edge disjoint paths from (i, j) to each of (i + 3, j), (i + 2, j - 1), (i + 1, j - 2), and (i, j - 3).

For the k-edge disjoint paths between (i, j) and (i + 3, j), consider the following three cycles. (i - j), (i - j + 1), and (i - j + 2). Note that (i, j) and (i + 3, j) are in  $(i - j)^{th}$  and  $(i - j + 2)^{th}$ cycles, respectively. And the cycle (i - j + 1) connects the odd vertices of  $(i - j)^{th}$  cycle to the even vertices of  $(i - j + 2)^{th}$  cycle. The k-edge disjoint paths can be constructed as follows. From (i, j) take the  $l^{th}, 1 \le l \le k$ , forward chord in cycle (i - j) to reach an odd vertex. From each odd vertex, take the  $(k - 1)^{st}$  backward chord in cycle (i - j + 1) to reach an even vertex. From each such even vertex take the  $(k - l + 1)^{st}$  chord in cycle (i - j + 2) to reach (i + 3, j). Figure 4.23 shows the k-edge disjoint paths between (i, j) and (i + 3, j).

To see that the above given k-paths are edge disjoint (in fact, vertex disjoint), consider the vertices  $\{(i + l, j + l - 1) | 1 \le l \le k\} \subseteq N[(i, j)]$ —these vertices are all distinct by construction.



Figure 4.24: k-edge disjoint paths between (i, j) and (i + 2, j - 1) in a n-ary 2-cube.



Figure 4.25: k-edge disjoint paths between (i, j) and (i + 1, j - 2) in a n-ary 2-cube.



Figure 4.26: k-edge disjoint paths between (i, j) and (i, j - 3) in a n-ary 2-cube.



Figure 4.27: 8-regular graph that belongs to  $\mathcal{F}_3(n$ -ary 2-cube).

Since we took the  $(k-1)^{st}$  BC from each vertex in the above set, it is easy to see that the vertices  $\{(i + l - k + 2, j + l - k) | 1 \le l \le k\}$  are distinct. A similar proof can be given to vertices (i + 2, j - 1), (i + 1, j - 2) and (i, j - 3), which are distance 3 away from (i, j). Table 4.4 and Figures 4.24-4.26 summarize the k-edge disjoint paths to vertices that are distance 3 away from (i, j).

Figure 4.27 shows the part of a graph that is 3-edge distance preserving with respect to an infinite torus. In this chapter we presented distance preserving graphs with respect to cycles, crowns, chordal graphs, and n-ary 2-cubes. In the next chapter we present a polynomial algorithm that can

Source	Next odd vertex is	Next even vertex is	Destination	
vertex	in Cycle $(i + j + 1)$	in Cycle $(i + j + 2)$	vertex	
(i,j)	(i+l,j+l-1)	(i+l-k+2,j+l-k)	(i+3,j),	$1 \leq l \leq k$
take <i>l<sup>th</sup></i> FC	take $(k-1)^{st}$ BC	take $(k - l + 1)^{th}$ FC		
(i,j)	(i+l,j+l-1)	(i+l-k+1, j+l-k-1)	(i+2, j-1),	$1 \leq l \leq k$
take <i>l<sup>th</sup></i> FC	take $k^{th}$ BC	take $(k - l + 1)^{th}$ FC		
(i,j)	(i+l,j+l-1)	(i+l-k+1, j+l-k-1)	(i+1, j-2),	$1 \leq l \leq k$
take <i>l<sup>th</sup></i> FC	take $k^{th}$ BC	take $(k-l)^{th}$ FC		
(i,j)	(i+l,j+l-1)	(i+l-k+2,j+l-k)	(i, j - 3),	$0 \leq l < k$
take <i>l<sup>th</sup></i> FC	take k <sup>th</sup> BC	take $(k-l-1)^{st}$ FC		

Table 4.4: k-edge disjoint paths to vertices that are distance 3 away from (i, j).

find the maximum k, when a graph G and a spanning subgraph D is given.

## **Chapter 5**

# **An Algorithm To Find** k **Given** D **and** G

It is well known that the problem of finding the maximum number of edge-disjoint paths of bounded length, say K, between a pair of vertices in a graph is NP-hard [46]. However, the problem is solvable in polynomial time when  $K \leq 5$  [46]. Hence, given a graph G and a spanning subgraph D, Theorem 1 and the polynomial algorithm in [46] can be used to determine if G belongs to  $\mathcal{F}_k(D)$ for some non-negative integer k. Details of such an algorithm are presented in this chapter.

We present an algorithm which takes graphs D and G as input and computes the maximum possible value k such that  $G \in \mathcal{F}_k(D)$ . Since the subgraph isomorphism problem is NP-complete [47], we assume that the given graph G contains D as a spanning subgraph. The algorithm  $Find_k$ is given in Figure 5.2. An outline of the algorithm is presented here.

In general, the max flow problem can be used to compute the maximum number of edge-disjoint paths joining two vertices in a graph [48]. However, a similar technique cannot be used to find the maximum number of edge-disjoint paths of bounded length. For example, consider graphs I and Jin Figure 5.1(a) and 5.1(b), respectively. We are interested in the maximum number of edge-disjoint u-v paths of length at most 3 in I and J. Let us use the max flow approach to compute the edgedisjoint paths of length at most 3 in the given graph. For graph I, if u, x, v is selected as the first u-v path then the only remaining edge-disjoint path connecting u and v is u, w, x, y, v which is of length 4. However, there are two edge-disjoint paths of length at most 3 namely, u, w, x, v and u, x, y, v. Similarly in J, if u, w, x, v is selected as the first path then there is no other edge-disjoint path connecting u and v. But there are 2 edge-disjoint paths of length at most 3 between u and v in J. Hence the well known maximum flow algorithm cannot give the maximum number of edge-disjoint paths of bounded lenght, even for small values of K.

### 5.1 Outline of the Algorithm

Due to Theorem 1, k can be computed by using only non-adjacent vertices that are at most distance 3 apart in D. Step 1 and Step 2 of the algorithm considers every pair of non-adjacent vertices u and v in D, with  $d_D(u, v) = 2$  and  $d_D(u, v) = 3$ , respectively. In each step, the algorithm computes the minimum of the maximum number of edge-disjoint paths. When  $d_D(u, v) = 2$ , it is easy to see that  $|N_G[u] \cap N_G[v]|$  represent the maximum number of edge-disjoint u-v paths in G of length at most 2. In Step 1, k is set to the minimum value of all such u,v pairs. Now, we explain how to compute the maximum number of edge-disjoint paths in G between vertices that are distance 3 apart in D.

First we present some useful properties of edge-disjoint u-v paths of length at most 3 in G. Let us denote by  $\mathcal{L}$ , a set of edge-disjoint u-v paths of length at most 3 in G. Let  $L_1$  and  $L_2$  be any two paths in  $\mathcal{L}$ .

### **Property 9** $|(V(L_1) \cap V(L_2)) \setminus \{u, v\}| \le 1.$

**Proof:** Let  $|(V(L_1) \cap V(L_2)) \setminus \{u, v\}| > 1$ . This implies that both  $L_1$  and  $L_2$  are of length 3. Let  $L_1 = u, w, x, v$ . Since the graph G is a simple graph we have only one choice for another edge-disjoint u-v path using vertices w and x. That is,  $L_2 = u, x, w, v$ . But then  $L_1$  and  $L_2$  share a common edge (w, x)—a contradiction.



Figure 5.1: Graphs I and J and their corresponding I' and J'.

**Property 10** If  $(V(L_1) \cap V(L_2)) \setminus \{u, v\} \neq \emptyset$  then  $|E(L_1)| = |E(L_2)| = 3$ . Also, if  $|E(L_1)| = 2$ , then for any other  $L_i \in \mathcal{L}$ , we have  $(V(L_1) \cap V(L_i)) = \emptyset$ .

**Proof:** Assume otherwise and without loss of generality, let  $L_1$  be a path of length 2; And let  $\{V(L_1) \cap V(L_2)\}\setminus\{u,v\} = \{w\}$ . Since  $|E(L_1)| = 2$  we have both (u,w) and  $(w,v) \in E(L_1)$ . Since  $w \in V(L_2)$  we must have either  $(u,w) \in E(L_2)$  or  $(v,w) \in E(L_2)$ . Implying that  $L_1$  and  $L_2$  are not edge-disjoint—a contradiction.

**Property 11** If  $(V(L_1) \cap V(L_2)) \setminus \{u, v\} = \{w\}$  Then  $w \in \{N_G(u) \cap N_G(v)\}$ .

**Proof:** Since  $L_1$  and  $L_2$  are edge-disjoint w cannot be adjacent to u (similarly to v) in both  $L_1$  and  $L_2$ . Hence, if  $(u, w) \in E(L_1)$  then  $(v, w) \in E(L_2)$ . This implies  $w \in \{N_G(u) \cap N_G(v)\}$ .  $\Box$ 

69

**Property 12** Any vertex in  $w \in V(G) \setminus \{u, v\}$  can at most be in two paths in  $\mathcal{L}$ . Further, any such vertex is adjacent to u in one path and adjacent to v in the other path.

**Proof:** Since all the paths in  $\mathcal{L}$  are of length at most 3, any intermediate vertex, w, in any path in  $\mathcal{L}$  has only two choices. Either w must be adjacent to u or to v. Hence vertex w can be adjacent to u in one path and to v in another path. That is  $L_1 = (u, w, x, v), L_2 = (u, y, w, v)$  for some  $y \in V(G) \setminus \{u, v\}.$ 

Our algorithm computes the maximum number of edge-disjoint paths of length at most 3 by counting the number of vertex-disjoint paths in a new bipartite graph G' which is constructed from  $H = G[N_G[u] \cup N_G[v]]$ . Before we explain the construction of G' notice that the maximum number of edge-disjoint u-v paths of length at most 3 in graphs G and H are equal this is because, all the vertices used in any u-v path in G of length at most 3 must belong to  $N_G[u] \cup N_G[v]$ . To avoid some special cases we assume that  $(u, v) \notin E(G)$ ; if  $(u, v) \in E(G)$  then k is incremented by 1 in the last step of the algorithm Find\_k.

We now construct a bipartite graph G' from H with  $V(G') = V(H) \cup \{w' | \text{for every } w \in N_H(u) \cap N_H(v)\}$ . In other words, for every vertex w in  $N_H(u) \cap N_H(v)$  we have two copies, w and w', in G'. Also, w and w' are referred first copy vertex and second copy vertex, respectively. The edge set of G' is given as follows. For each  $w \in N_H(u) \cap N_H(v)$  we have,  $\{(u,w),(w,w'),(v,w')\} \subseteq E(G')$  (Figure 5.3); And any edge  $(w,x) \in E(H)$  is replaced in G' by (x,w') and (w,x'), if  $x \in N_H(u) \cap N_H(v)$  (Figure 5.4) and (x,w'), if  $x \in N_H(u) \setminus N_H(v)$ (Figure 5.5) and (w,x), if  $x \in N_H(v) \setminus N_H(u)$ (Figure 5.6).

For every  $w \notin N_H(u) \cap N_H(v)$ , if  $(u, w) \in E(H)$  then  $(u, w) \in E(G')$  and if  $(v, w) \in E(H)$ then  $(v, w) \in E(G')$ . The following observations are immediate from the construction of G'.

1. Any first copy vertex is always adjacent to u in G' and any second copy vertex is always

adjacent to v in G'.

- 2. The first and second copies of a vertex  $w \in N_H(u) \cap N_H(v)$  are always connected in G'.
- 3. If  $(u, w) \in E(H)$ , for some  $w \in V(H)$ , then  $(u, w) \in E(G')$ .
- 4. If  $(w', v) \in E(G')$  then  $w \in N_H(u) \cap N_H(v)$ . (Otherwise,  $w' \notin V(G')$ ).
- 5.  $(w, v) \in E(G')$ , if and only if  $w \in N_H(v) \setminus N_H(u)$ .
- 6. If  $(w, x) \in E(H)$  and if both w and  $x \notin N_G(v)$  then  $(w, x) \notin E(G')$ . Similarly, If  $(w, x) \in E(H)$  and if both w and  $x \notin N_G(u)$  then  $(w, x) \notin E(G')$ .
- 7. G' is a bipartite graph.

Such a G' for the graphs I and J is given in 5.1(c) and 5.1(d), respectively. We claim that the maximum number of edge-disjoint paths of length at most 3 in H is equal to the maximum number of vertex-disjoint paths in G'. A proof for the above claim is given in Theorem 10.

**Theorem 10** Given a graph D and a graph G having D as a spanning subgraph the algorithm Find\_k computes the maximum k such that  $G \in \mathcal{F}_k(D)$ .

**Proof:** In this proof we show that every pair of non-adjacent vertices u and v are joined in G by at least k edge-disjoint paths; Also we show that such a k is maximum. In other words, for no other k' > k we have  $G \in \mathcal{F}_{k'}(D)$ . By Theorem 1, we know that such a k is the minimum of the maximum number of edge-disjoint u-v paths between every pair of non-adjacent vertices u and v with  $d_D(u, v) \leq 3$ .

After the completion of Step 1, it is clear that k will hold minimum of the maximum number of edge-disjoint paths in G between every pair of vertices u and v with  $d_D(u, v) = 2$ .

As mentioned earlier, to compute the maximum number of edge-disjoint paths between a pair of non-adjacent vertices u and v with  $d_D(u, v) = 3$ , the step 2 of the algorithm constructs a graph

#### Algorithm: Find\_k

**Input:** Graphs G and D (D is a spanning subgraph of G). **Output:** Maximum k such that  $G \in \mathcal{F}_k(D)$ .

Step 1:  $k = \min \{ |N_G[u] \cap N_G[v] | \text{ for all } u \text{ and } v \text{ with } d_D(u, v) = 2 \}.$ 

**Step 2:** For every pair of vertices u and v with  $d_D(u, v) = 3$ 

Construct a graph G' with  $V(G') = \{N_G[u] \cup N_G[v]\} \cup \{w' | \text{ for each } w \in N_G(u) \cap N_G(v)\}.$ The edge set E(G') is constructed as follows: For each  $w \in N_G(u) \cap N_G(v)$  $(u, w), (w, w'), (w', v) \in E(G')$ End For each  $w \in N_G(u) \cap N_G(v)$ If  $(w, x) \in E(G)$  Then **Case:** 1:  $x \in N_G(v) \cap N_G(u)$  $(w, x'), (x, w') \in E(G')$ **Case:** 2:  $x \in N_G(u) \setminus N_G(v)$  $(x,w') \in E(G')$ **Case:** 3:  $x \in N_G(v) \setminus N_G(u)$  $(w,x) \in E(G')$ End For each  $w \notin N_G(u) \cap N_G(v)$ If  $(u, w) \in E(H)$  Then  $(u,w) \in E(G')$ If  $(v, w) \in E(H)$  Then  $(v,w) \in E(G')$ End

End

If  $(u, v) \in E(G)$  Then  $k \leftarrow \min(k, \text{ size of the max matching of } (G' - \{u, v\}) + 1).$ Else  $k \leftarrow \min(k, \text{ size of the max matching of } (G' - \{u, v\})).$ 

End





Figure 5.3: Graph H and its G'.



Figure 5.4: Case 1 of the Algorithm. G and the resulting G'.



Figure 5.5: Case 2 of the Algorithm. G and the resulting G'.

G' from  $H = G[N_G[u] \cup N_G[v]]$ . By construction, G' is a bipartite graph; Also, vertex u is adjacent to all vertices in one partition of  $V(G') - \{v\}$ . Similarly, vertex v is adjacent to all vertices in one partition of  $V(G') - \{u\}$ . It is known that the size of the maximum matching in the bipartite graph  $G' - \{u, v\}$  is equal to the maximum number of vertex-disjoint paths between u and v in G' [48]. Further, any such path is of length 3. In the remaining part of the proof we will show that the maximum number of edge-disjoint u-v paths of length at most 3 in H is equal to the maximum number of vertex-disjoint u-v paths in G'.

Let  $\mathcal{L} = \{L_1, L_2, \dots, L_l\}$  and  $\mathcal{M} = \{M_1, M_2, \dots, M_m\}$  be the set of edge-disjoint paths of length at most 3 between u and v in H and G', respectively. It suffices to show that  $l \leq m$  and  $m \leq l$ .

Using the properties mentioned in the outline of the algorithm, we will show that  $l \leq m$ . For



Figure 5.6: Case 3 of the Algorithm. G and the resulting G'.

Case  $L_i = u, w, v$  we have  $M_i = u, w, w', v$ .

Since  $E(L_i) = 2$  the vertex w cannot be used in any other u-v path in  $\mathcal{L}$  (Property 9). Hence the vertices w and w' in the corresponding path  $M_i = u, w, w', v$ , are not used by any other path in  $\mathcal{M}$ . Hence  $M_i$  is internally vertex-disjoint with any other path in  $\mathcal{M}$ .

Case  $L_i = u, w, x, v$ : Then we have two possibilities.

Subcase  $x \in N_H(u)$  we have  $M_i = (u, w, x', v)$ .

In this case, both  $x \in N_H(u) \cap N_H(v)$ . By construction we have  $x' \in V(G')$  and  $(x',v) \in E(G')$ . Also, since  $(w,x) \in E(H)$ , we have  $(w,x') \in E(G')$ . Hence the path  $M_i$  given above exists in G'. Now, we will prove that  $M_i$  is internally vertex-disjoint with any other path in  $\mathcal{M}$ . Assume otherwise and let  $V(M_i) \cap V(M_j) = \emptyset$ , for some  $i \neq j$ . We have  $(u,w) \notin M_j$  because  $(u,w) \notin L_j$  for any  $L_j \in \mathcal{L}$ . Similarly,  $(x',v) \notin M_j$  because  $(x,v) \notin L_j$  for any  $L_j \in \mathcal{L}$ . Similarly,  $(x',v) \notin M_j$  because  $(x,v) \notin L_j$  for any  $L_j \in \mathcal{L}$ . In such a case, at least  $(u,x') \in E(M_j)$  or  $(w,v) \in E(M_j)$ . But  $(u,x') \notin E(M_j)$  because in G', by construction, no second copy vertex is adjacent to u. If  $(w,v) \in E(M_j)$  then the edge  $(w,v) \in E(G')$  and E(H), which implies  $w \notin N_H(u)$ —a contradiction, because  $(u,w) \in L_i$  implies  $w \in N_H(u)$ .

Subcase  $x \notin N_H(u)$  we have  $M_i = (u, w, x, v)$ .

Since  $x \notin N_H(u)$  and  $(x, v) \in E(H)$ , we have edge  $(x, v) \in G'$ . Hence it is easy to see that path  $M_i$  exists in G'. Now, we will prove that  $M_i$  is internally vertex-disjoint with any other path in  $\mathcal{M}$ . Assume that there exists another path  $M_j \in \mathcal{M}$ , for some  $i \neq j$ , such that  $V(M_i) \cap V(M_j) \neq \emptyset$ . With out loss of generality, let  $w \in V(M_i) \cap V(M_j)$ . In this case, either  $(u, x) \in M_j$  or  $(w, v) \in M_j$ . If  $(u, x) \in M_j$  then  $(u, x) \in L_j$ , implying that  $x \in N_H(u)$ —a contradiction. If  $(w, v) \in M_j$  then this implies that  $w \notin N_H(u)$ —a contradiction, because  $(u, w) \in L_i$  implies  $w \in N_H(u)$ .

Now we will show that  $m \leq l$  by constructing a set  $\mathcal{L}$  of edge-disjoint paths in H. We denote by copy(X) the original vertex in H; that is copy(X) = x whether X = x or x'. For each  $M_i \in \mathcal{M}$ we construct a path in H, say  $L_i$ , using the vertices in  $M_i$ . For each  $M_i = u, w, x', v \in \mathcal{M}$ , if there exists another path  $M_j = u, x, w', v$  then  $L_i = u, w, v$  and  $L_j = u, x, v$ ; Otherwise,  $L_i = u, w, copy(X), v$ . If w = copy(X) then  $L_i = u, w, v$ . It is easy to see that any edge used in the resulting u-v path exist in E(H).

Now we will prove that the set of paths  $\mathcal{L}$  constructed from  $\mathcal{M}$  are edge-disjoint. Assume otherwise and let  $E(L_i) \cap E(L_j) \neq \emptyset$ , for some  $L_i$  and  $L_j$  in  $\mathcal{L}$ . Note that both  $L_i$ and  $L_j$  are of length three. Otherwise, if  $L_i = u, w, v$  then either  $(u, w, w', v) \in \mathcal{M}$  or  $\{(u, w, x', v), (u, x, w', v)\} \subseteq \mathcal{M}$  and w or w' cannot be used by any other path in  $\mathcal{M}$ . Hence, vertex w is used by at mot one path in  $\mathcal{L}$ . Without loss generality, let  $L_i = u, w, x, v$ . Notice that these two paths cannot share the edge (u, w) or (w, x) because  $M_i$  is internally vertex-disjoint with any other path in  $\mathcal{M}$ . Hence,  $E(L_i) \cap E(L_j) = \{(x, v)\}$ . If (x, v) belongs to two paths in  $\mathcal{L}$ then the first copy of vertex x is adjacent to v in one path in  $\mathcal{M}$ , say  $M_i$ . And the second copy of vertex x is adjacent to v to  $M_j$ . But from the construction of G' only second copy is adjacent to v. Therefore,  $(x, v) \notin E(G')$ ; hence  $(x, v) \notin E(M_i)$ —a contradiction.

## **Chapter 6**

# Conclusions

In this dissertation, we introduced a new family of fault-tolerant graphs called *distance preserving graphs*. We studied some properties of distance preserving graphs and a result that can be used to characterize distance preserving graphs is presented. For designing distance preserving graphs we considered two design models with different optimality criteria. Designs in the first model minimize the overall redundancy. The second model restricts the designs to regular fault-tolerant topologies with minimum regularity. For both of the design models, we present lower and upper bounds and show that the given bounds are tight.

Our focus was mainly on designs based on Model 2. Our solutions include fault-tolerant designs with respect to cycles, crowns, chordal rings, and *n*-ary 2-cubes. In particular, we prove that our designs are optimal when  $D = C_p$  and  $k \leq \frac{p}{2}$ . We have also shown that the distance between adjacent vertices, upon edge removal, is bounded by a constant. Finally, given a graph G and a spanning subgraph D, we present a polynomial algorithm that can be used to find the maximum k such that  $G \in \mathcal{F}_k(D)$ .

The results presented in this dissertation can be used in designing fault-tolerant communication networks in which distance between communicating nodes is critical.

### 6.1 Future Work

The given fault-tolerant designs for crowns, chordal rings, and n-ary 2-cubes are optimal for lower values of k. However, we believe that the given constructions are optimal for any k. Thus, we state the following conjectures.

**Conjecture 1** For any p-order r-crown, there does not exist an i-regular p-order graph  $G \in \mathcal{F}_k(r\text{-}crown)$ , for any i < r + k - 1.

**Conjecture 2** For any C(p,h), there does not exist an *i*-regular graph  $G \in \mathcal{F}_k(r - crown)$ , for any

- 1. i < 2k + 1, if  $h \ge 2k$ .
- 2.  $i < k + \lfloor \frac{h}{2} \rfloor + 1$ , if h < 2k.

**Conjecture 3** For any n-ary 2-cube, there does not exist an i-regular graph  $G \in \mathcal{F}_k$  (n-ary 2-cube), for any i < 2k + 2.

The fault-tolerant design given for  $C_p$  when  $k \leq \frac{p}{2}$ , is optimal for Model 2 designs. We believe that the same fault-tolerant graph is optimal even for Model 1 designs. Formally,

**Conjecture 4** For any p-order cycle,  $C_p$ , there does not exist a  $\frac{p(k+1)}{2} - 1$  size graph  $G \in \mathcal{F}_k(C_p)$ .

Regular topologies seem to be popular for their simplicity and scalability in designing faulttolerant CNs [18, 20]. However, fault-tolerant designs involving minimum redundancy prevail in CNs where the cost of the CN is an important factor. Optimal designs for Model 1 need to be studied in more detail.

Throughout the dissertation, we studied deterministic measures of fault-tolerance. The foundations laid out in this dissertation can be used to investigate probabilistic measures of fault-tolerance. The fault-tolerant designs given for n-ary 2-cubes are based on partitioning the edge set of the n-ary 2-cube into n partitions, where each partition is a cycle of order 2n. A similar technique can be used to develop a framework in designing topologies for any given arbitrary graph.

# **Bibliography**

- [1] A. L. DeCegama, The Technology of Parallel Processing, vol. 1. Prentice Hall, 1989.
- [2] S. Dutt and J. P. Hayes, "Some practical issues in the design of fault-tolerant multiprocessors," *IEEE Transactions on Computers*, vol. 41, pp. 588–598, May 1992.
- [3] M. J. Quinn, Designing Efficient Algorithms for Parallel Computers. McGraw-Hill, 1987.
- [4] J. P. Hayes, "A graph model for fault-tolerant computing systems," IEEE Transactions on Computers, vol. 25, pp. 875–884, Sept. 1976.
- [5] C.-L. Kwan and S. Toida, "Optimal fault-tolerant realizations of some classes of hierarchical tree systems," *The* 11<sup>th</sup> *International Conference on Fault-Tolerant Computing*, pp. 176–178, 1981.
- [6] C. S. Raghavendra, A. Avizienis, and M. D. Ercegovac, "Fault tolerant in binary tree architecture," *IEEE Transactions on Computers*, vol. 33, pp. 568–571, June 1984.
- [7] M. B. Lowrie and W. K. Fuchs, "Reconfigurable tree architectures using subtree oriented fault tolerance," *IEEE Transactions on Computers*, vol. 36, pp. 1172–1183, Oct. 1987.
- [8] S. Dutt and J. P. Hayes, "On designing and reconfiguring strategies for k-fault tolerant tree architectures," *IEEE Transactions on Computers*, vol. 39, pp. 490–503, Apr. 1990.

- [9] J. Bruck, R. Cypher, and C.-T. Ho, "On the construction of fault-tolerant cube-connected cycles networks," in *Proceedings of the 1991 International Conference on Parallel Processing*, vol. 1, pp. 692–693, 1991.
- [10] J. Bruck, R. Cypher, and C.-T. Ho, "Fault-tolerant meshes with minimal numbers of spares," in Proceedings of the 1991 Symposium on Parallel and Distributed Processing, pp. 288-295, 1991.
- [11] J. Bruck, R. Cypher, and C.-T. Ho, "Fault-tolerant de Bruijn and shuffle-exchange networks." Unpublished Document.
- [12] F. Boesch, "Synthesis of reliable networks a survey," *IEEE Transactions on Reliability*, vol. 35, pp. 240–246, Aug. 1986.
- [13] A.-H. Esfahanian, "Generalized measures of fault-tolerance with application to n-cube networks," *IEEE Transactions on Computers*, vol. 38, pp. 1586–1591, Nov. 1989.
- [14] D. K. Pradhan and S. M. Reddy, "A fault-tolerant communication architecture for distributed systems," *IEEE Transactions on Computers*, vol. 31, pp. 863–869, Sept. 1982.
- [15] F. Harary, Graph Theory. Addison-Wesley Publishing Company, 1968.
- [16] M. O. Ball, "Complexity of network reliability computation," NETWORKS, vol. 10, pp. 153– 165, 1980.
- [17] R. J. Dawson and A. A. Farrag, "Fault-tolerant extensions of star networks," NETWORKS, vol. 21, pp. 373–385, 1991.
- [18] D. V. Chudnovsky, G. V. Chudnovsky, and M. M. Denneau, "Regular graphs with small diameter as models for interconnection networks," in *Proceedings of the 3<sup>rd</sup> International Conference on Supercomputing*, vol. 3, pp. 232–239, 1988.

- [19] W. J. Dally, "Performance analysis of k-ary n-cube interconnection networks," IEEE Transactions on Computers, vol. 39, pp. 775–785, June 1990.
- [20] G. Zimmerman and A.-H. Esfahanian, "Chordal rings as fault-tolerant loops," Journal of Discrete Applied Mathematics, vol. 99, pp. 563–573, Dec. 1992.
- [21] F. Harary and J. P. Hayes, "Edge fault tolerance in graphs," NETWORKS, vol. 23, pp. 135–142, 1993.
- [22] F. Harary, "The maximum connectivity of a graph," Proceedings of National Academy of Sciences, vol. 48, pp. 1142–1146, 1962.
- [23] K. Vijayan and U. S. R. Murty, "On accessibility in graphs," Sankhya, Series A, vol. 26, pp. 299–302, 1965.
- [24] B. Bollobas, Extremal Graph Theory. Academic Press, 1978.
- [25] F. R. K. Chung, "Graphs with small diameter after edge deletion," Discrete Applied Mathematics, vol. 37-38, pp. 73-84, 1992.
- [26] K. P. Eswaran and E. Tarjan, "Augmentation problems," SIAM Journal of Computing, vol. 5, pp. 653–665, Dec. 1976.
- [27] H. Whitney, "Congruent graphs and the connectivity of graphs," American Journal of Mathematics, vol. 54, pp. 150–168, 1932.
- [28] F. Boesch and R. Tindell, "Circulants and their connectivities," Journal of Graph Theory, vol. 8, pp. 487–499, 1984.
- [29] A. Rosenthal and A. Goldner, "Smallest augmentations to biconnect a graph," SIAM Journal of Computing, vol. 6, pp. 55–66, 1977.

- [30] T. Watanabe and A. Nakamura, "A smallest augmentation to 3-connect a graph," Discrete Applied Mathematics, vol. 28, pp. 183–186, 1990.
- [31] S. Ueno, Y. Kajitani, and H. Wada, "Minimum augmentation of a tree to a k-edge-connected graph," NETWORKS, vol. 18, pp. 19–25, 1988.
- [32] G. Cai and Y. Sun, "The minimum augmentation of any graph to a k-edge-connected graph," NETWORKS, vol. 19, pp. 151–172, 1989.
- [33] A. Sengupta, P. D. Joshi, and S. Bandyopadhyay, "A synthesis approach to design optimally fault tolerant network architecture," *IEEE Transactions on Computers*, vol. 40, pp. 94–100, Jan. 1991.
- [34] F. R. K. Chung, "Diameters of graphs: Old problems and new results," Congressus Numerantium, vol. 60, pp. 295–317, 1987.
- [35] R. C. Entringer, D. Jackson, and P. J. Slater, "Geodetic connectivity of graphs," IEEE Transactions on Circuits and Systems, vol. 24, pp. 460–463, Aug. 1977.
- [36] B. Bollabas and S. Eldridge, "On graphs with diameter 2," Journal of Combinatorial Theory, Series B, vol. 21, pp. 201–205, 1976.
- [37] R. Entringer and D. E. Jackson, "Minimum k-geodetically connected graphs," Congressus Numerantium, vol. 36, pp. 303–309, 1982.
- [38] S. Sitarama and A.-H. Esfahanian, "Designing optimal distance-preserving topologies," Department of Computer Science, Michigan State University, vol. 95, Aug. Technical Report CPS-94-12.

- [39] S. Sitarama and A.-H. Esfahanian, "Designing distance-preserving fault-tolerant topologies,"
  22nd International Workshop on Graph-Theoretic Concepts in Computer Science, pp. 150– 164, 1996.
- [40] S. Sitarama and A.-H. Esfahanian, "Fault-tolerant designs that preserve distance," *Department* of Computer Science, Michigan State University, vol. 96, Oct. Technical Report CPS-96-34.
- [41] S. Sitarama and A.-H. Esfahanian, "Distance-preserving graphs," Journal of Discrete Mathematics, vol. 96, Oct. Submitted for publication.
- [42] F. Boesch and R. Tindell, "Connectivity and symmetry in graphs," Graphs and Applications, Proceedings of 1st Colorado Symposium on Graph Theory, pp. 53-67, 1982.
- [43] F. Boesch and J. Wang, "Reliable circulant networks with minimum transmission delay," IEEE Transactions on Circuits and Systems, vol. 32, pp. 1286–1291, Dec. 1985.
- [44] F. Buckley and F. Harary, Distance in Graphs. Addison-Wesley Publishing Company, 1990.
- [45] X. Hu and F. Hwang, "Reliabilities of chordal rings," NETWORKS, vol. 22, pp. 487-501, 1992.
- [46] A. Itai, Y. Perl, and Y. Shiloach, "The complexity of finding maximum disjoint paths with length constraints," NETWORKS, vol. 12, pp. 277–286, 1982.
- [47] M. R. Garey and D. S. Johnson, *Computers and Intractability*. W. H. Freeman and Company, 1979.
- [48] G. Chartrand and L. Lesniak, *Graphs and Digraphs*. Wadsworth and Brooks/cole mathematics series, 1979.

