DISTRIBUTED DESIGN OF STRONG STRUCTURALLY CONTROLLABLE AND MAXIMALLY ROBUST NETWORKS

by

Johir Suresh

APPROVED BY SUPERVISORY COMMITTEE:

____

Waseem Abbas, Chair

____

Justin Koeln

____

Justin Ruths

____

Tyler Summers

This thesis is dedicated to

my parents for their unconditional love and support,

to my advisor Dr. Waseem Abbbas for leading me in the right direction,

to all my teachers who made learning a wonderful experience so far,

to my research partner Priyansh Patel, with whom this research was possible,

to my loved ones for making life enjoyable and

to the almighty for this life and experience.

DISTRIBUTED DESIGN OF STRONG STRUCTURALLY CONTROLLABLE AND MAXIMALLY ROBUST NETWORKS

by

JOHIR SURESH, BTech

THESIS

Presented to the Faculty of

The University of Texas at Dallas

in Partial Fulfillment

of the Requirements

for the Degree of

MASTER OF SCIENCE IN

MECHANICAL ENGINEERING

THE UNIVERSITY OF TEXAS AT DALLAS

May 2023

ACKNOWLEDGMENTS

I would like to thank my advisor Dr. Waseem Abbas, for his guidance throughout my research and for the opportunity to learn about network control and graph theory. I also want to extend my hearty thanks to Dr. Justin Ruths, Dr. Justin Koeln and Dr. Tyler Summers for sparing the time and being part of my thesis committee and for their valuable input into its content. I could not have completed this thesis without the support of my friends, who provided stimulating discussions as well as happy distractions to rest my mind outside of my research.

May 2023

DISTRIBUTED DESIGN OF STRONG STRUCTURALLY CONTROLLABLE AND MAXIMALLY ROBUST NETWORKS

Johir Suresh, MS
The University of Texas at Dallas, 2023

Supervising Professor: Waseem Abbas, Chair

The design of multiagent networks with certain properties is in general a difficult problem. From a network control perspective, controllability and robustness are two important but opposing properties. In this dissertation, we address the problem of designing networks that are both structurally controllable and maximally robust. To achieve this objective, we propose several network constructions that are strong structurally controllable, for given network parameters the number of nodes N, leaders N_L, and diameter D. To measure controllability, we employ the zero-forcing process, and subsequently, maximize the number of edges in the networks. We also evaluate network robustness using Kirchhoff index. To validate our approach, we compare our network constructions with optimal clique chains and perform numerical evaluations.

Furthermore, we present a set of graph grammars that enable the distributed construction of these networks. Our work not only exploits the trade-off between controllability and robustness but also provides an optimal graph structure under specific conditions, and a near-optimal one for most cases.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

Combining the system- and graph-theoretic tools provides a powerful approach to understanding various aspects of complex dynamical systems across multiple domains, including engineering, computer science, and biology. In addition, this integrated approach enables us to capture the interdependency of the dynamical behavior of such systems with the underlying network structure, thus, allowing us to apply a range of techniques to gain insights and optimize the performance of such networked systems. Various network properties, such as controllability and robustness, affect the overall system's performance.

This work delves into the challenge of creating networks that meet multiple design standards. Achieving particular design objectives in network creation is often a demanding task. Two vital features from a network control perspective are controllability and robustness. Network controllability is the ability to steer and regulate a network towards desired configurations or states using external control signals or inputs. These inputs are delivered into the network via a subset of nodes designated as ***leaders*** (e.g., (Pasqualetti et al., 2014)). The concept of network robustness has been vastly studied, and it can be classified as functional and structural robustness (Abbas and Egerstedt, 2012). Functional robustness pertains to the network's ability to operate effectively in the presence of noise and perturbations, while structural robustness describes the network's ability to maintain its structural characteristics even in the event of node or edge failures (Young et al., 2016; Zelazo and Bürger, 2015). Remarkably, these two interpretations are interconnected in the context of network control systems and can be quantified using the Kirchhoff index (Ellens et al., 2011; Siami and Motee, 2013; Wang et al., 2010).

The conflict between network controllability and robustness is a well-researched topic (Abbas et al., 2020; Pasqualetti et al., 2020). This means that networks that require a small number of leaders to achieve complete controllability may have inadequate robustness properties for a given set of network parameters. For example, path graphs with a single leader node can achieve complete controllability, but they exhibit minimal robustness. Similarly, a fully connected graph of N nodes is has maximum robustness but requires $N-1$ leaders to be completely controllable (Abbas et al., 2020).

So, a crucial challenge is ***how can we design networks that achieve both complete controllability with few inputs and show high robustness?*** This challenge becomes even more compelling when considering network controllability in a strong structural sense, given the computational complexity involved (e.g., (Chapman and Mesbahi, 2013; Jia et al., 2019; Menara et al., 2017; Shabbir et al., 2022)). Typically, network controllability is contingent upon the edge weights, which measure the coupling strength between nodes; nonetheless, edge weights are irrelevant in the context of strong structural controllability (SSC). If a network is strong structurally controllable with a given set of leaders, it will be controllable regardless of the edge weights. Essentially, for SSC, the network structure is the key determinant and not the edge weights.

This work introduces a distributed approach to designing networks that exhibit both structural controllability for a specified number of nodes N and leaders N_L, as well as maximal robustness through maximizing the edge sets. One such class of networks that are maximally robust for given number of nodes N and diameter D are optimal clique chains (OCC), but they require $N-D$ leaders for SSC by zero forcing method. We construct our proposed networks in a distributed manner, by utilizing graph grammars (Jia et al., 2019; Klavins, 2007; Yim, 2007). We ensure SSC by leveraging the computationally efficient zero forcing method (Monshizadeh et al., 2014; Mousavi et al., 2017). Our proposed designs produce graphs with high robustness that closely approach optimal clique chains for given network parameters N, N_L, and D, evaluated using Kirchhoff index. In many cases, our designs achieve optimal robustness, while maintaining SSC with a much smaller leader set than OCC. Therefore, our network constructions aim to maximize robustness while preserving controllability for given network parameters. Our main contributions can be summarized as follows:

- For given N (total number of nodes), N_L (number of leaders) and D diameter, we design strong structurally controllable graphs with N_L leaders and maximal edge sets. For SSC, we utilize the idea of zero forcing sets.

- By controlling N, N_L, and D, we can achieve networks that are strong structurally controllable with varying degrees of robustness. Furthermore, we evaluate the robustness of these graphs using Kirchhoff index. Lastly, we conduct numerical evaluations to assess the effectiveness of the proposed constructions.

- In addition, we present distributed methods for constructing the aforementioned graphs using graph grammars, which are a set of rules implemented by nodes locally to achieve the desired network structure.

Our problem scenario is comparable to that presented in (Abbas et al., 2020); however, there are significant differences between the two. We analyze strong structural controllability using a simpler zero forcing method, while (Abbas et al., 2020) employs graph distances for this purpose. Additionally, in (Abbas et al., 2020), the proposed graphs for a given N and N_L have a fixed diameter, whereas we allow flexibility in choosing D as an input parameter. Furthermore, we propose distributed constructions of networks using graph grammars, which were not considered in (Abbas et al., 2020).

This thesis consists of the work we presented in (Patel et al., 2023) but with significant design improvements. In (Patel et al., 2023), we were able to create graphs of diameter $2 \leq D \leq N/N_L$, in this extension we provide flexibility in picking diameter as user input.

The rest of the thesis is structured into six chapters. Chapter 2 introduces preliminary concepts and defines the problem. Chapters 3 and 4 presents the proposed graph constructions and the results of numerical evaluations. Chapter 5 covers the graph grammar and showcases their application. Chapter 6, explains the experimental setup and results of numerical evaluation. Lastly, Chapter 7 concludes the thesis and suggests avenues for future research.

CHAPTER 2

NETWORK AND ITS PROPERTIES

This chapter provides an overview of the preliminaries of network systems, with a particular focus on systems modeled as leaders and followers. We discuss various properties such as controllability and robustness, which have a significant impact on the performance of network systems. Additionally, we define the measures that we use to evaluate networks based on their controllability and robustness. Furthermore, we review an essential family of graphs known as clique chains, which have distinct robustness properties and compare our designs with such graphs.

2.1 Control of Network Systems

Graphs, also called networks, are composed of vertices, or nodes, which are linked together through edges. These edges may be directed or undirected and may have weights or unitary weights. The various structural modifications mentioned can have an impact on the characteristics of networks. We consider a multiagent network system as an ***undirected graph*** of N number of nodes, denoted at $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{v_1, v_2, \cdots, v_N\}$ is the set of nodes and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the set of edges.

In an undirected graph, an edge between two nodes u and node v is represented by an unordered pair of those nodes denoted as (u, v). Similarly, if (u, v) is an edge in the graph, we can say that node u is a ***neighbor*** of node v and node v is a neighbor of node u.

The number of nodes in the neighborhood of u is the ***degree*** of u, defined as $\deg(u) = |\{v \in \mathcal{V} : (u, v) \in \mathcal{E}\}|$.

A ***path*** in a graph represents a sequence of connected nodes, where each node in the sequence is adjacent to the next node in the sequence. The length of a path is the number of edges that it contains. A path of length k in a graph $\mathcal{G}$ is denoted by $\mathcal{P}_k$, and can be represented as $\mathcal{P}_k = \langle u_0, u_1, u_2, \ldots, u_k \rangle$, where $u_0, u_1, u_2, \ldots, u_k$ are distinct nodes of $\mathcal{G}$ and $(u_i, u_{i+1}) \in \mathcal{E}$ for all $i = 0, 1, \ldots, k-1$.

The ***distance*** between two nodes u and v is defined as the length of the shortest path between them, where the length of a path is the number of edges that it contains. The distance between u and v is denoted by $d(u, v)$.

The ***diameter*** of a graph $\mathcal{G}$ is defined as the maximum distance between any two nodes in $\mathcal{G}$ and is denoted by $D = \max_{u,v 2 V}\{d(u, v)\}$.

A graph is ***weighted*** if edges are assigned numerical weights using a weighting function. This weight can represent a variety of quantities, such as the distance between two nodes, the cost of traveling from one node to another, or the strength of the connection between two nodes. The weighting function is typically defined as a mapping from the set of edges $\mathcal{E}$ to the set of real numbers.

$$w : \mathcal{E} \longrightarrow \mathsf{R}^+.$$

The weight of an edge represented as $w(i, j)$, is the weight of the edge between node i and node j. The ***adjacency matrix*** of graph $\mathcal{G}$ is defined as follows,

$$\mathcal{A}_{ij} = \begin{cases} w(i, j) & \text{if } i, j \in \mathcal{E} \\ 0 & \text{otherwise.} \end{cases} \tag{2.1}$$

The ***degree matrix*** of graph $\mathcal{G}$ is defined as,

$$\Delta = \begin{cases} \sum_{k 2 N_i} \mathcal{A}_{ik} & \text{if } i = j \\ 0 & \text{otherwise.} \end{cases} \tag{2.2}$$

The ***Laplacian*** of $\mathcal{G}$ is defined as,

$$\mathcal{L} = \Delta - \mathcal{A}. \tag{2.3}$$

2.2 Leader-follower System

The *leader-follower* system can be represented as a dynamical system where each node of the graph corresponds to a dynamical unit and the edges represent the interactions between these units. The goal of the system is for the followers to track the behavior of the leader. This system associated with the graph $\mathcal{G}$ is defined by the following state-space representation:

$$\dot{x}(t) = Mx(t) + Bu(t). \tag{2.4}$$

Here, the state vector $x(t) \in \mathsf{R}^N$ represents the state of all the nodes in the graph at time t. The system matrix $M \in \mathcal{M}(\mathcal{G})$ is the class of system matrices, where $\mathcal{M}(\mathcal{G})$ is a family of system matrices associated with an undirected graph $\mathcal{G}$ defined as below.

$$\mathcal{M}(\mathcal{G}) = \{M \in \mathsf{R}^{N \quad N} | M = M^T, \text{and for } i \neq j, M_{ij} \neq 0 \Leftrightarrow (i,j) \in \mathcal{E}(\mathcal{G})\}. \tag{2.5}$$

Note that $\mathcal{M}$ includes all matrices that can be used to describe the behavior of a dynamical system associated with the graph. These matrices can include, for example, the adjacency matrix, which encodes the connectivity of the graph, as well as other types of matrices that capture specific features of the system or the interactions between the nodes. The Laplacian matrix $\mathcal{L}$ presented in the above section is one such matrix that belongs to $\mathcal{M}$.

Furthermore, $u(t) \in \mathsf{R}^{N_L}$ is the input signal and $B \in \mathsf{R}^{N \quad N_L}$ is the input matrix containing information about the leader nodes through which inputs are injected into the network. For a set of leaders labelled $\mathcal{V}_L = \{\ell_1, \ell_2, \ldots, \ell_m\} \subseteq \mathcal{V}$, we define the input matrix as follows.

$$B_{ij} = \begin{cases} 1 & \text{if } v_i = \ell_j, \\ 0 & \text{otherwise.} \end{cases} \tag{2.6}$$

We are interested in designing networks with the above dynamics that are strong structurally controllable and maximally robust. Furthermore, we employ specific measures of controllability and robustness to assess the effectiveness of our network designs. In the following sections, we elaborate on these measures and their relevance in evaluating the performance of our graphs.

2.3 Network Controllability

The controllability of networked systems describes the ability to steer the system's state from any initial condition to any desired final state using appropriate input signals. A networked system is said to be controllable if it can be fully controlled by the available input signals, regardless of the initial state of the system. Considering the system mentioned in (2.4), a state $x(t) \in \mathsf{R}^n$ is said to be reachable if there exists some input that could drive the system from origin to x in a finite amount of time. One widely used measure of controllability is structural controllability, which is based on the network's topology and the input matrix B. A set of all reachable states constitutes the ***controllable subspace***, which is the range space of the following matrix.

$$\Gamma(\mathcal{G}, \mathcal{V}_L) = [B \quad (-M)B \quad (-M)^2B \quad \cdots \quad (-M)^{N \;\; 1}B].$$

The dimension of the controllable subspace is the rank of Γ, which needs to be N for complete controllability. The rank of Γ depends not only on the edge set of the graph but also on the edge weights. A graph that is controllable for one set of edge weights might not remain controllable if edge weights are changed.

A given graph $\mathcal{G}$ with a set of leader nodes $\mathcal{V}_L \subseteq \mathcal{V}$, and the corresponding B matrix is said to be ***strong structurally controllable*** if and only if (M, B) is a controllable pair $\forall\, M \in \mathcal{M}$ irrespective of the choice of non-zero edge weights. When this condition is met, we refer to the pair $(\mathcal{G}, \mathcal{V}_L)$ as strong structurally controllable. Therefore, in a network that exhibits strong structural controllability, changes in edge weights do not affect the dimension of the controllable subspace. In (Monshizadeh et al., 2014), the authors provide a graph theoretical characterization of SSC in networks. They introduce the concept of ***zero forcing*** to establish a lower bound on controllability. However, in (Abbas et al., 2020), another tight lower bound on controllability is provided using a ***distance-based bound***. The distance-based bound is known to be more computationally expensive compared to the zero-forcing bound. Interestingly, as demonstrated in (Patel et al., 2023), a graph that is both strong structurally controllable and maximally robust, as measured using the zero-forcing and distance-based bounds, for a given number of nodes N and number of leaders N_L, are co-spectral. In the next section, we will discuss the zero forcing process.

2.3.1 Zero forcing

In this section, we discuss the concept of zero-forcing based lower bound on the rank of the controllability matrix. In (AIM Minimum Rank Special Graphs Work Group, 2008), the authors showed that zero-forcing provides a tight lower bound on the rank of the controllability matrix, making it a robust metric for evaluating the controllability of networks. If the lower bound achieved by the zero-forcing process is the complete graph, then it guarantees controllability, specifically strong structural controllability (Monshizadeh et al., 2014). We will further explore this concept in the following sections.

Definition *(Zero forcing process)* Consider a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where each node is initially either colored white or black, with the black nodes representing leaders, i.e., $\mathcal{V}_L = \ell_1, \ell_2, \ldots, \ell_m \subseteq \mathcal{V}$. The zero forcing process can be defined as follows: if a black node $v_i \in \mathcal{V}$ has exactly one white neighbor $v_j \in \mathcal{V}$, then v_i forces v_j to be colored black.

For a given zero forcing process there can be multiple ways of executing the zero forcing process; however, the set of black nodes at the end of the zero forcing process remains the same. If there is a unique way of proceeding with the zero forcing process in a graph $\mathcal{G}$, then we call it a unique zero forcing process.

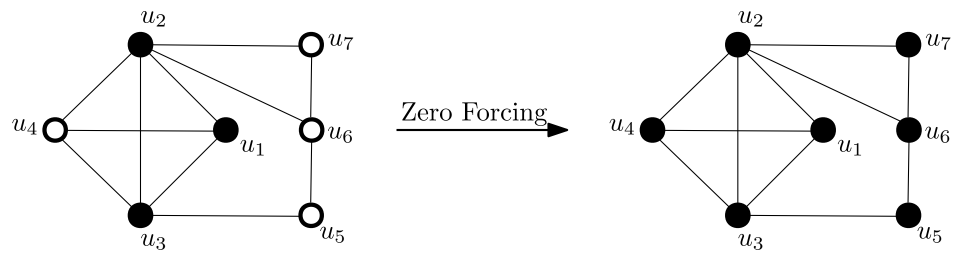


Figure 2.1: Zero forcing process with zero forcing set $\{u_1, u_2, u_3\}$.

Definition *(Derived set)* Consider a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with a leader set $\mathcal{V}_L \subseteq \mathcal{V}$, then the set of black nodes obtained at the end of the zero forcing process is the derived set, denoted by $\mathcal{V}^0(\mathcal{G}, \mathcal{V}_L)$.

Definition *(Zero Forcing Set)* Consider a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with a leader set $\mathcal{V}_L \subseteq \mathcal{V}$, if the derived set $\mathcal{V}^0 = \mathcal{V}$ then the leader set $\mathcal{V}_L$ is called the zero forcing set.

For example, consider a set of nodes $\{u_1, \cdots, u_7\}$, where $\{u_1, u_2, u_3\}$ are colored black and the rest $\{u_4, \cdots, u_7\}$ are uncolored. Through the zero-forcing process, at every step, if there exists a black node with only one white node in its neighborhood, the black node can color the white node. This process continues until there are no more black nodes with exactly one white neighbor. Consider the network in Figure 2.1, u_1 has exactly one white neighbor u_4, so u_1 colors u_4. Then u_3 would have exactly one white neighbor u_5, so u_3 colors u_5. This process continues until all seven nodes are colored.

The problem of characterizing the minimum leader set for SSC in terms of ZFS is shown in (Monshizadeh et al., 2014). The authors have shown that if a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with a leader set $\mathcal{V}_L \subseteq \mathcal{V}$ is strong structurally controllable if and only if $\mathcal{V}_L$ is a ZFS of $\mathcal{G}$. A strong structurally controllable network remains unaffected by changes in edge weights, implying that the size of the controllable subspace remains constant. This attribute of strong structural controllability proves to be valuable in situations where the exact values of edge weights are unknown due to factors such as numerical inaccuracies, uncertainties, and imprecise system parameters. Consequently, the objective of our work is to determine the smallest number of leaders that can render a network strongly structurally controllable.

In the following section, we discuss the concept of robustness in network systems and measures employed to evaluate robustness in networked systems numerically.

2.4 Network Robustness

Robustness in networks can be classified into *functional robustness* and *structural robustness* (Abbas and Egerstedt, 2012). Firstly, functional robustness is the robustness to noise of linear dynamics over networks. Consider a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ that exhibits the dynamics described in equation 2.4. In the presence of noise, the system has a certain probability of attaining the desired states. The capacity of the system to withstand this disruption and function correctly is termed functional robustness. Secondly, structural robustness is the ability of the network to retain its structural attributes in the case of edge or node deletion. The study of structural robustness is critical for designing resilient networks that can withstand perturbations and continue to function efficiently under adverse conditions (Young et al., 2016; Zelazo and Bürger, 2015).

To analyze the robustness of our proposed networks, we use *Kirchho index* (Ellens et al., 2011; Siami and Motee, 2013; Wang et al., 2010). In the following section, we present Kirchhoff index and its relevance in analyzing both the functional and structural robustness of a network system.

2.4.1 Kirchhoff index

Kirchhoff index is an important and widely used robustness measure. It is denoted by $K_f(\mathcal{G})$ and is defined as the sum of all the non-zero eigenvalues of the network's Laplacian matrix. It can be computed as,

$$K_f(\mathcal{G}) = N \sum_{i=2}^{N} \frac{1}{\lambda_i}, \tag{2.7}$$

where, N is the total number of nodes in the graph $\mathcal{G}$ and $\lambda_2 \leq \lambda_3 \leq \cdots \lambda_N$ are the non-zero eigenvalues of the Laplacian of graph $\mathcal{G}$. Robustness and the value of Kirchhoff index are inversely related. A lower Kirchhoff index indicates a more robust and well-connected graph. Therefore, minimizing the Kirchhoff index of a graph is one way to increase its robustness. Therefore, increasing the number of edges of a graph is one way to minimize the Kirchhoff index and hence improve robustness to failure and noise (Ellens et al., 2011; Siami and Motee, 2013; Young et al., 2016).

Network robustness measured using Kirchhoff index is a monotonically increasing function of the number of edges in a network (Ellens et al., 2011; Wang et al., 2010). Hence, our objective is to increase the number of edges while preserving SSC. However, it is crucial to maintain a balance between controllability and robustness as controllability decreases with increasing edges while robustness increases, and vice versa. Thus, it is necessary to achieve a trade-off between these factors while expanding the network. This sets up the stage to motivate the problem, we are trying to design networks that are maximally robust by having the most number of edges while guaranteeing SSC, as measured by zero forcing, for a given number of nodes N, number of leaders N_L and diameter D.

2.5 Clique Chains

In graph theory, ***clique chains*** are networks that are maximally robust for any given N and D. As shown in (Ellens et al., 2011), clique chains have the minimum Kirchhoff index among all graphs of N nodes and D diameter. Consider a set of positive integers $n_1, n_2, \cdots, n_D, n_{D+1}$, such that the total number of nodes in the network is $N = \sum_{i=1}^{D+1} n_i$. In (Abbas et al., 2020), the authors define the construction of a clique chain involves creating a path graph of length $D+1$, and then replacing each node in the path with a clique of size n_i, such that the vertices in the cliques are adjacent if and only if the corresponding vertices in the original path graph are adjacent. This results in a network structure with cliques connected in

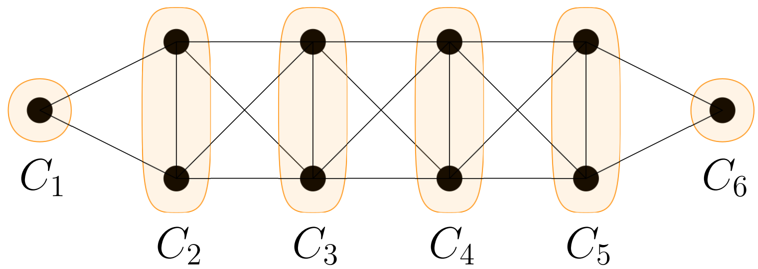


Figure 2.2: Optimal clique chain of $N = 10, D = 5$.

a chain-like fashion, with a diameter of D. (Ellens et al., 2011) optimises the network robustness for given N and D for optimal edge addition. This network is a specific form of clique chain with $n_1, n_{D+1} = 1$, this network with the minimum Kirchhoff index is called ***optimal clique chains***. The optimal clique chain (OCC) for an example with $N = 10$ and $D = 4$ is illustrated in Figure 2.2. The cliques in the OCC are denoted as $C_1, C_2, \cdots, C_6$, with corresponding sizes $n_1 = 1, n_2 = 2, \cdots, n_5 = 2, n_6 = 1$.

CHAPTER 3

DESIGN OF CONTROLLABLE AND ROBUST NETWORKS

Networks are pervasive in various domains and play a crucial role in understanding and analyzing complex systems. From supply chains to social networks, biological systems to the internet, many real-world phenomena can be effectively represented and analyzed using network models. One important property of networks is controllability, which refers to the ability to steer the dynamics of a networked system by manipulating a subset of its nodes or edges. Controllability is critical in many applications, including multiagent robotic systems, where coordinating the actions of multiple agents is necessary to achieve desired outcomes. Controllability allows a central controller to guide the collective behavior of the swarm by influencing a few key robots. Moreover, in networks where the behavior of each system depends on the actions of the neighbors, reliable data is crucial for accurate evolution. Given the importance of controllability and robustness as key properties of networks, it would be ideal to achieve a maximum level of both of these properties simultaneously. This leads us to the question, *\How do controllability and robustness interrelated with each other, and how does one property a ect the other one?"*.

In previous sections, we discussed that the trade-off between controllability and robustness is an important consideration in the study of networked systems. A path graph can be highly controllable with only one leader node. However, such a network may exhibit poor robustness, as the failure or removal of the single leader node can result in the loss of controllability. On the other hand, a fully connected graph can be the most robust in terms of its ability to withstand failures or disruptions. However, achieving complete controllability in a fully connected graph of N nodes may require a large number of leader sets of $N-1$ nodes. This raises the question, *"What is the most robust network that is also strong structurally controllable?"*. (Abbas et al., 2020) presented a network design that is strong structurally controllable in terms of distance-based bounds and maximally robust for given N and N_L or

D. We also know that optimal clique chains produce graphs with the highest robustness for given N and D, but require $N - D$ number of leaders to achieve strong structural controllability. Our work (Patel et al., 2023), try to investigate the question,

> Can we design a network that is both strong structurally controllable and maximally robust for given inputs N and $N_L \leq N - D$?

In the following sections, we will discuss three network designs from (Patel et al., 2023), that are both strong structurally controllable and maximally robust for given N and N_L, each with different characteristics and performances.

3.1 Network Design 1

We know that for a given N and D, clique chains give a graph of optimum robustness and a path graph is strong structurally controllable with just one leader. Then for given, N_L number of leaders and $N = N_L \times D$ number of nodes, where D is the diameter, we generate a network G_1, to meet the design criteria. To construct $\mathcal{G}_1$, we make N_L path graphs of length $D - 1$, where the end nodes of each path is a leader. Then we make each leader pairwise adjacent, making a clique between leaders. More formally, let us consider the following vertex set for $\mathcal{G}_1$,

$$V = \{\ell_i\} \cup \{u_{i,j}\},$$

where $i \in \{1, 2, \ldots, k\}$ and $j \in \{1, 2, \ldots, D - 1\}$. Vertices with label $\{\ell_1, \ell_2, \ldots, \ell_k\}$ are leaders and the rest $\{u_{1,1}, \ldots, u_{i,j}, \ldots, u_{k,D\;\;1}\}$ are followers. Let us connect these nodes in the following manner:

- All the leaders ℓ_i have a link between them and generate a complete graph among them.

- For all $i \in \{1, 2, \dots, k\}$, there exists a link between ℓ_i and $u_{i,1}$.

- For all $i \in \{1, 2, \dots, k\}$ and $j \in \{1, 2, \dots, D-2\}$, there is a link between $u_{i,j}$ and $u_{i,j+1}$.

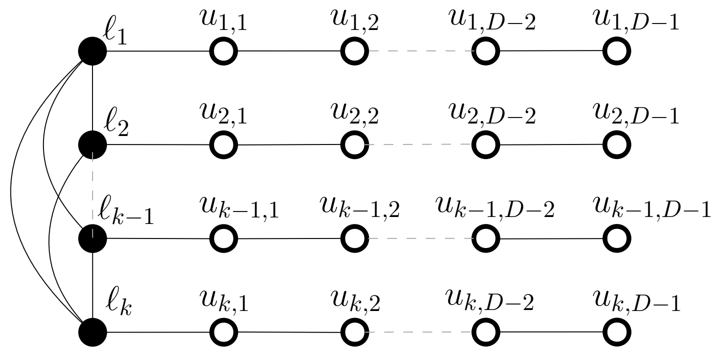


Figure 3.1: Graph $\mathcal{G}_1$ with N_L number of leaders, D diameter and $N = (N_L \times D)$ nodes.

This construction is portrayed in Figure 3.1. This graph $\mathcal{G}_1$, is strong structurally controllable, but notice that we can add edges to $\mathcal{G}_1$ without affecting SSC. Let us refer to this graph as $\bar{\mathcal{G}}_1$, the following are the sets of edges that can be added to maximize the edge set.

- All the leaders ℓ_i has an edge with $u_{q,1}$ $\forall q < i$, where $i, q \in \{1, 2, \dots, k\}$.

- Similarly, all the nodes in $u_{i,j}$ have an edge with nodes in $u_{q,j+1}$ $\forall q < i$, where $i, q \in \{1, 2, \dots, k\}$ and $j \in \{1, 2, \dots, D-2\}$.

- Also, for a fixed j, all nodes in $u_{i,j}$ generate a complete graph, where $i \in \{1, 2, \dots, k\}$ and $j \in \{1, 2, \dots, D-1\}$

Figure 3.2 illustrates the construction of $\bar{\mathcal{G}}_1$ from $\mathcal{G}$. The newly added edges are shown in blue and orange.

Lemma 1.1. *The leader set $\{\ell_1, \ell_2, \cdots, \ell_{N_L}\}$ is a ZFS of $\bar{\mathcal{G}}_1$ (described above) with N nodes and D diameter.*

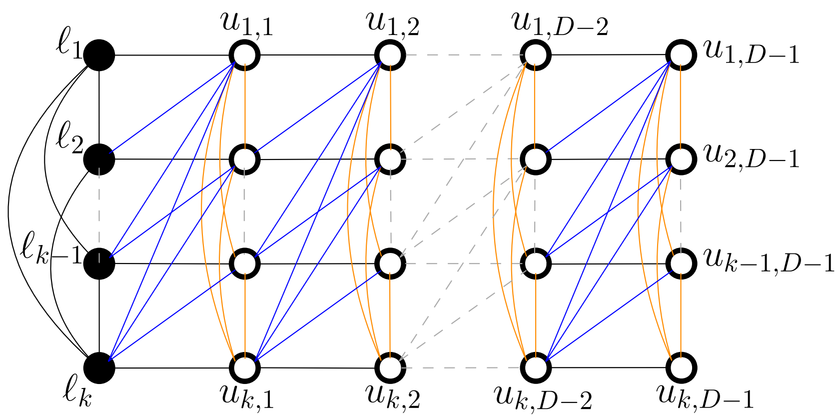


Figure 3.2: Graph $\bar{\mathcal{G}}_1$ with maximal edges.

Proof. We observe that in $\bar{\mathcal{G}}_1$, all leaders, except ℓ_1, have more than one white neighbor in their neighborhoods. So, only ℓ_1 can initiate the zero forcing process. Leader (ℓ_2) adjacent to ℓ_1 has exactly two white neighbors one of which, $u_{1,1}$, it shares with ℓ_1. Consequently, the third leader has exactly three white neighbors and shares one white neighbor each with ℓ_1 and ℓ_2. This continues for all N_L number of leaders.

As discussed, ℓ_1 starts the zero forcing process by coloring its only white neighbor, $u_{1,1}$. As a result, ℓ_2 is now left with only one white neighbor, $u_{2,1}$, and thus ℓ_2 colors it. Similarly, ℓ_3 is now left with only one white neighbor that is $u_{3,1}$ because the other two white neighbors $u_{2,1}$ and $u_{1,1}$, which it had in common with ℓ_2 and ℓ_1, respectively, are now colored. This continues until all the nodes in $u_{i,1}$ are colored. Note that there exists a complete graph between all the followers in $u_{i,1}$, therefore, $u_{1,1}$ will only be able to color $u_{1,2}$ once all the other nodes in $u_{i,1}$ are colored. We also know that the nodes in $u_{i,1}$ and in $u_{i,2}$ have similar connections between them as ℓ_i and $u_{i,1}$. So, it follows from above discussion that the process continues until all the nodes are colored, implying that the given leader set is a ZFS of $\bar{\mathcal{G}}_1$, which is the desired claim. □

Lemma 1.2. *For xed number of nodes N and diameter D, a graph of construction $\bar{\mathcal{G}}_1$ has maximal edges, i.e., by adding any additional edge the leader set $\{\ell_1, \ell_2, \cdots, \ell_{N_L}\}$ will no longer be a ZFS of $\bar{\mathcal{G}}_1$.*

Proof. Let us show that the above statement holds for a subgraph $\bar{\mathcal{G}}_1^0$ containing only the leader set and the first set of followers $u_{i,1}$ and all the edges between them. As mentioned in Lemma 1.1 the zero forcing process in $\bar{\mathcal{G}}_1$ is unique and it propagates from first follower $u_{1,1}$ to all the nodes in $u_{i,1}$ till $u_{k,1}$, in this particular order. Now, considering $\bar{\mathcal{G}}_1^0$, we observe that adding any edge would disturb this zero forcing process because an additional edge would result in some leader having more than one white neighbor at a particular time step. This results in the leader set not being a ZFS of the subgraph $\bar{\mathcal{G}}_1^0$.

Next, we assume that the above argument is true for all nodes in $\bar{\mathcal{G}}_1$ until the set of nodes in $u_{i,D\ 2}$. This means that all the nodes in $u_{i,D\ 2}$ are colored. Since the nodes in $u_{i,D\ 1}$ are further ahead in the zero forcing process than nodes in $u_{i,D\ 2}$, we can say that nodes in $u_{i,D\ 2}$ are not dependent on nodes in $u_{i,D\ 1}$ for getting colored. Furthermore, we notice that edge set between $u_{i,D\ 2}$ and $u_{i,D\ 1}$ is same as that between the leader set and $u_{i,1}$. We use the same reasoning as above to show that we cannot add any other edge between these two node sets without disrupting the zero forcing process, implying that not all the nodes will get colored. Therefore, by induction, adding any extra edge in $\bar{\mathcal{G}}_1$ would result in the leader set not being a ZFS, which is the desired claim. □

Remark 1.3. Graph $\bar{\mathcal{G}}_1$ is same (isomorphic) as the graph produced in (Abbas et al., 2020). We call the graph constructed in (Abbas et al., 2020) as $\mathcal{G}_{PMI}$. It is interesting to note that even though we constructed $\bar{\mathcal{G}}_1$ using the zero forcing method, we arrive at the same result, whereas (Abbas et al., 2020) uses the distance-based approach in their design.

In $\bar{\mathcal{G}}_1$, the diameter is $\lceil \frac{N}{N_L} \rceil$, i.e., by changing the total number of nodes N and the number of leaders N_L, the diameter varies. So, the interesting question is, ***can we design graphs with improved robustness while constraining/ xing the diameter without deteriorating controllability?*** In the following subsection, we answer this by designing strong structurally controllable graphs $\bar{\mathcal{G}}_2$ with diameter $D = 2$, N total nodes, $N_L \geq 2$ leaders, and improved robustness.

3.2 Network Design 2

In this section, we construct a maximally robust graph $\bar{\mathcal{G}}_2$ by fixing $N_L \geq 2$ and adding $N_F = (N - N_L)$ number of other nodes (followers) one-by-one to the graph. This design is different from $\bar{\mathcal{G}}_1$ in that the maximum distance between any two nodes is two. This partial construction is shown in Figure 3.3, and can be noticed that it has a complete derived set but with possibilities of addition of edges. Next, lets explore the construction of $\bar{\mathcal{G}}_2$ considering the following vertex set,

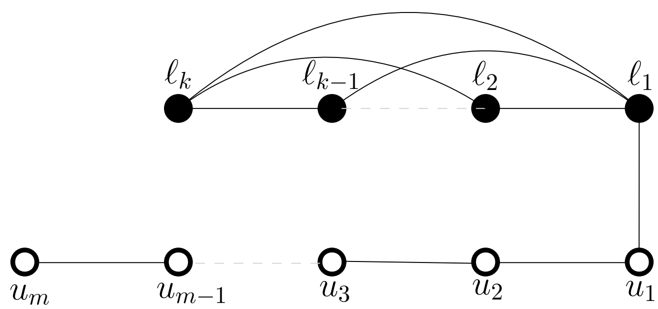


Figure 3.3: Graph $\mathcal{G}_2$ with N nodes and N_L leaders.

$$V = \{\ell_i\} \cup \{u_j\},$$

where $i \in \{1, 2, \ldots, k\}$ and $j \in \{1, 2, \ldots, m\}$, where $k = N_L$ and $m = N_F$. Vertices labeled $\{\ell_1, \ell_2, \ldots, \ell_k\}$ are leaders and $\{u_1, u_2, \ldots, u_m\}$ are followers.

We connect the vertices as follows,

- Leader ℓ_1 and followers $\{u_1, u_2, \ldots, u_m\}$ are connected through a path graph starting from ℓ_1.
- All the leaders ℓ_i are connected with all the nodes u_j, where $i \in \{2, \ldots, k\}$ and $j \in \{1, \ldots, m\}$

Figure 3.4 illustrates the construction of $\bar{\mathcal{G}}_2$ with maximal edges.

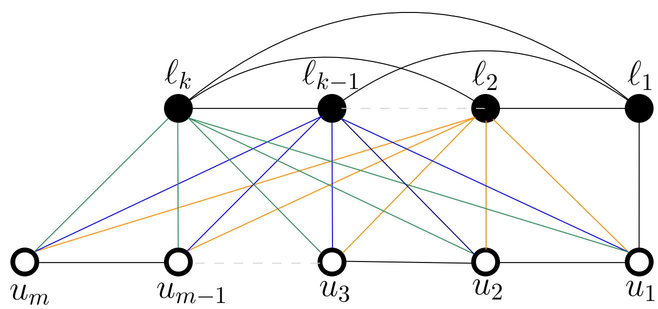


Figure 3.4: Graph $\bar{\mathcal{G}}_2$ with maximal edges.

Lemma 2.1. *For a graph $\bar{\mathcal{G}}_2$ (as described above) with N number of nodes, the proposed leader set $\{\ell_1, \ell_2, \cdots, \ell_{N_L}\}$ is a ZFS.*

Proof. From the construction of $\bar{\mathcal{G}}_2$, we observe that all leaders, except ℓ_1, are pair-wise adjacent to all the followers, $u_j\ \forall j$. This means that except ℓ_1, all leaders will generally have more than one white neighbor. Since ℓ_1 has only one white neighbor u_1, it will start the zero forcing process by coloring u_1.

Next, the rest of the leaders will still have multiple white neighbors in their neighborhoods. However, u_1 has only one white neighbor u_2. That will allow u_1 to color u_2. Subsequently, u_2 will also have only one white neighbor u_3. This stands true for rest of the follower nodes in u_j, where $1 \leq j \leq (N - N_L)$. We note that as a result of this unique zero forcing process, the entire graph gets colored, implying that the leader set is a ZFS of $\bar{\mathcal{G}}_2$. □

Lemma 2.2. *For xed number of nodes N and given leader set $\{\ell_1, \ell_2, \cdots, \ell_{N_L}\}$ the graph generated using the construction $\bar{\mathcal{G}}_2$ has maximal edge set, i.e., by adding any additional edge the leader set $\{\ell_1, \ell_2, \cdots, \ell_{N_L}\}$ will no longer be a ZFS of $\bar{\mathcal{G}}_2$.*

Proof. As shown in Lemma 2.1, the Graph $\bar{\mathcal{G}}_2$ has a unique zero forcing process. There are two types of edges that we can add to $\bar{\mathcal{G}}_2$. They include,

- leader to follower (non-leader) edges, i.e., (ℓ_1, u_j), where $j \neq 1$, and
- follower to follower edges, i.e., $(u_j, u_{j'})$, where $j \neq j'$.

Both categories of edges belong to a unique zero forcing process. Adding an edge between any two non-adjacent nodes in this unique zero forcing process will not allow the preceding node to continue the zero forcing process since it will now have more than one white neighbor. This means that the addition of any edge other than the ones already existing in $\bar{\mathcal{G}}_2$ will result in a graph for which the given leader set is not a ZFS. □

It is interesting to note that, the above two constructions, $\bar{\mathcal{G}}_1$ and $\bar{\mathcal{G}}_2$ generate equal number of edges for the same parameters (N and N_L). The number of edges in $\bar{\mathcal{G}}_1$ is,

$$E_{\bar{\mathsf{G}}_1} = \underbrace{D \times \frac{N_L \times (N_L - 1)}{2}}_{E_1} + \underbrace{(D-1) \times \frac{N_L \times (N_L + 1)}{2}}_{E_2}. \tag{3.1}$$

There are D cliques $\bar{\mathcal{G}}_1$, each of size N_L. E_1 is the total number of edges in these cliques, and E_2 is the number of remaining edges in $\bar{\mathcal{G}}_1$. Similarly, the number of edges in $\bar{\mathcal{G}}_2$ is given by,

$$\begin{aligned} E_{\bar{\mathsf{G}}_2} = &\underbrace{(N - N_L) \times \ (N_L - 1)}_{E_3} + \underbrace{N - N_L}_{E_4} \\ &+ \underbrace{\frac{N_L \times (N_L - 1)}{2}}_{E_5}. \end{aligned} \tag{3.2}$$

Here, E_3 is the number of edges between $N_L - 1$ leaders and $(N - N_L)$ followers, E_4 is the number of edges in the path induced by leader ℓ_1 and followers, and E_5 is the number of edges in the complete graph induced by the leader nodes. Now, simplifying (3.1) and (3.2) gives

$$E_{\bar{\mathsf{G}}_1} = E_{\bar{\mathsf{G}}_2} = N_L \times \left(N - \frac{(N_L + 1)}{2} \right). \tag{3.3}$$

As discussed previously, the diameter of $\bar{\mathcal{G}}_1$ depends on N_L and N, whereas the diameter of $\bar{\mathcal{G}}_2$ is constant regardless of N_L and N. So, next, we explore a graph construction that combines $\bar{\mathcal{G}}_1$ and $\bar{\mathcal{G}}_2$ and provides the option of choosing the diameter D of the graph.

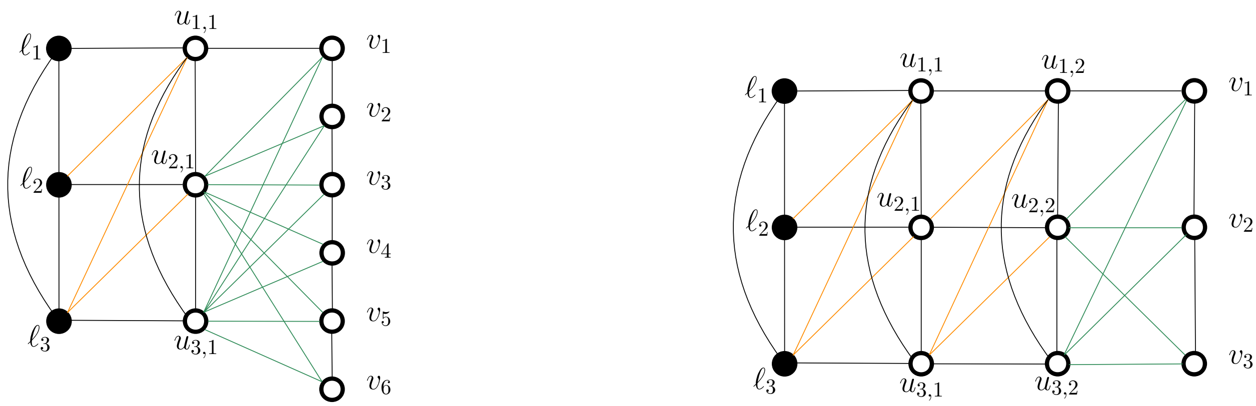
We will see in Subsection 3.4 that in many cases $\bar{\mathcal{G}}_2$ provide higher robustness than $\bar{\mathcal{G}}_1$, but require the same amount of leaders to achieve SSC. So, it is advantageous to have $\bar{\mathcal{G}}_1$ combined with $\bar{\mathcal{G}}_2$ for achieving higher robustness while meeting a design requirement in terms of the diameter.

3.3 Network Design 3 (Combining Designs 1 and 2)

In this section, we construct a graph that is a combination of $\bar{\mathcal{G}}_1$ and $\bar{\mathcal{G}}_2$, meaning partial nodes follow the rules of construction of $\bar{\mathcal{G}}_1$ and rest of them follow the construction rules of $\bar{\mathcal{G}}_2$ (as discussed in the previous subsections). We define the construction with three different parameters, N (total number of nodes), N_L(number of leaders), and D (diameter of the graph). For these given parameters, we construct a graph $\bar{\mathcal{G}}_3$ that is strong structurally controllable, maximally robust and has equal number of edges as $\bar{\mathcal{G}}_1$ or $\bar{\mathcal{G}}_2$ for the same N and N_L.

Let $\bar{\mathcal{V}} = \bar{\mathcal{V}_1} \cup \bar{\mathcal{V}_2}$, be the set of all the nodes in $\bar{\mathcal{G}_3}$, where $\bar{\mathcal{V}_1}$ and $\bar{\mathcal{V}_2}$ are the subset of nodes that follow construction of $\bar{\mathcal{G}}_1$ and $\bar{\mathcal{G}}_2$, respectively. Then, the graph $\bar{\mathcal{G}}_3$ is constructed as follows:

- The leader set $\{\ell_1, \ell_2, \cdots, \ell_{N_L}\} \subset \bar{\mathcal{V}}_1$.
- Let the end nodes of the construction of $\bar{\mathcal{G}}_1$, i.e., $u_{i,D\ \ 2}$, where $1 \leq i \leq N_L$, be the pseudo-leaders of $\bar{\mathcal{G}}_2$.
- Let $u_{i,D\ \ 2} = \bar{\mathcal{V}}_1 \cap \bar{\mathcal{V}}_2$, where $1 \leq i \leq N_L$, then the total number of nodes become $|\bar{\mathcal{V}}| = |\bar{\mathcal{V}_1}| + |\bar{\mathcal{V}_2}| - (\bar{\mathcal{V}}_1 \cap \bar{\mathcal{V}}_2)$.
- The first pseudo-leader of $\bar{\mathcal{G}}_2$, i.e.,$u_{1,D\ \ 2}$, belongs to the same zero forcing path as of the first leader (ℓ_1) in $\bar{\mathcal{G}}_1$.



(a) $D = 3$ (b) $D = 4$

Figure 3.5: Examples of network design 3 ($\bar{\mathcal{G}}_3$).

- The edges between nodes x and y, where $x \in \{u_{i,D\ \ 2}, \forall i\}$ and $y \in \{u_{i,D\ \ 2}, \forall i\}$, are according to the construction of $\bar{\mathcal{G}}_1$. Similarly, the edges between nodes in $\{u_{i,D\ \ 2}, \forall i\}$ and $\{v_j\}$, where $1 \leq j \leq (|\bar{\mathcal{V}}_2| - N_L)$, are according to the construction of $\bar{\mathcal{G}}_2$.

Figure 3.5 illustrates two examples of the construction of $\bar{\mathcal{G}}_3$ for $N = 12$ and $N_L = 3$. The diameters of graphs in (a) and (b) are 3 and 4, respectively.

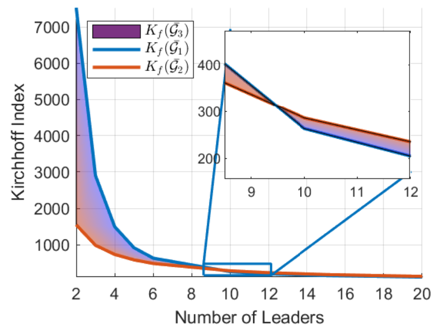
We make the following observations from the examples:

- Both graphs have the same number of edges, which is also equal to the number of edges in graphs generated according to $\bar{\mathcal{G}}_1$ and $\bar{\mathcal{G}}_2$ for the same N and N_L.
- Both graphs are strong structurally controllable with the given leader sets.
- In general, changing $|\bar{\mathcal{V}}_1|$ and $|\bar{\mathcal{V}}_2|$ will result in graphs ranging from diameter $D = 2$ to diameter of $\bar{\mathcal{G}}_1$ for the same N and N_L, i.e.,

$$2 = D(\bar{\mathcal{G}}_2) \leq D(\bar{\mathcal{G}}_3) \leq D(\bar{\mathcal{G}}_1) = N/N_L. \tag{3.4}$$

  Thus, if $|\bar{\mathcal{V}}_1| = N_L$, $\bar{\mathcal{G}}_3 = \bar{\mathcal{G}}_2$, and similarly, if $|\bar{\mathcal{V}}_2| = N_L$, $\bar{\mathcal{G}}_3 = \bar{\mathcal{G}}_1$.

In this following section, we will discuss the robustness performance of the network designs $\bar{\mathcal{G}}_1$, $\bar{\mathcal{G}}_2$ and $\bar{\mathcal{G}}_3$ in terms of Kirchhoff index for various leaders of a network with N nodes.



(a)

Figure 3.6: Kirchhoff index as a function of number of leaders in $\bar{\mathcal{G}}_1$, $\bar{\mathcal{G}}_2$ and $\bar{\mathcal{G}}_3$.

3.4 Numerical Evaluation and Robustness Analysis

Here, we numerically evaluate the performance of graphs $\bar{\mathcal{G}}_1$, $\bar{\mathcal{G}}_2$, $\bar{\mathcal{G}}_3$, in terms of robustness and controllability for $N = 60$. Figure 3.6 plots the robustness performance of the proposed graphs in terms of the Kirchhoff index. Interestingly, for a lower number of leaders, the Kirchhoff index of $\bar{\mathcal{G}}_2$ is significantly lower (indicating improved robustness) than that of $\bar{\mathcal{G}}_1$. However, for a higher number of leaders, this trend changes, and $\bar{\mathcal{G}}_1$ has a lower Kirchhoff index than $\bar{\mathcal{G}}_2$, though the difference between the values remains relatively small.

Finally, as discussed in the previous subsection, with $\bar{\mathcal{G}}_3$ we can generate graphs whose diameters are lower bounded by $\bar{\mathcal{G}}_2$ and upper bounded by $\bar{\mathcal{G}}_1$. Similarly, Kirchhoff index for $\bar{\mathcal{G}}_3$ are also bounded by those of $\bar{\mathcal{G}}_1$ and $\bar{\mathcal{G}}_2$. This is shown in Figure 3.6 as a gradient between robustness values of $\bar{\mathcal{G}}_1$ and $\bar{\mathcal{G}}_2$. This reveals an important design trade-off, i.e., for a specific diameter requirement, we can utilize the design $\bar{\mathcal{G}}_3$ while affording more robustness than $\bar{\mathcal{G}}_1$ for the same N and N_L. However, their diameters were either dependent on N and N_L or fixed. In the next chapter, we propose such a design with D as a user input and answer the following problem statement,

Given input parameters N, N_L and D, can we design a graph that is strong structurally controllable and maximally robust?

CHAPTER 4

ENHANCED NETWORK DESIGN FOR INCREASED FLEXIBILITY

This chapter delves into the creation of our strong structurally controllable and maximally robust graph $\mathcal{G}$, designed specifically for a given number of nodes N, number of leaders N_L, and diameter $D \leq min(N_L, N_F)$, where $N_F = N - N_L$ represents the number of followers. To construct the graph, our approach focuses on solving a number partitioning problem, which allows us to partition the nodes in a manner that satisfies the specified diameter requirement. In the following subsection, we will delve deeper into the partitioning problem and provide an algorithm that effectively solves it.

4.1 The partitioning problem

The number partitioning problem is an extensively studied combinatorial optimization problem in mathematics, known for its NP-hard classification (Mertens, 2006). Despite its computational complexity, it is a critical aspect of this research as it involves dividing the nodes into partitions, which is akin to solving a number partitioning problem for a given number of nodes N and diameter D. While obtaining an analytical solution for this problem remains elusive, we have developed a method that produces a sub-optimal solution. As we will discuss in Section 4.3, this solution performs close to optimum robustness in most cases and even achieves optimum robustness in certain cases.

Let us consider N number of nodes that are to be divided into q partitions denoted as $P \in \{P_1, P_2, \cdots, P_q\}$, where P_i represents the number of nodes in i^{th} partition. By analyzing the optimal partitioning structure for maximal robustness, as discussed in (Ellens et al., 2011), we observe that it follows a specific pattern. The optimal partitioning has one node at each end partitions, with the highest number of nodes in the middle partitions. The number of nodes in the first half of the partitions is non-decreasing. In contrast, the number of nodes in the second half is non-increasing, with the partitioning structure being symmetric about the middle partition. Based on this observation, we exclusively focus on the first half of the nodes and extend the results to the second half in a mirrored fashion to obtain complete partitioning.

We achieve the above-mentioned partitioning by making use of a modified geometric series as mentioned in Equation 4.1, such that each term in the series corresponds to the number of nodes in a partition.

$$\begin{aligned} 1 + r + r^2 + \cdots + r^{n\ \ 2} + Kr^{n\ \ 1} &= Z, \\ \frac{(1 - r^{n\ \ 1})}{(1 - r)} + Kr^{n\ \ 1} &= Z. \end{aligned} \tag{4.1}$$

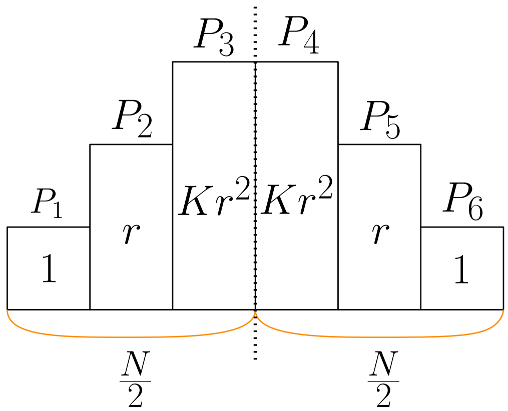
Here, $Z = \lceil \frac{N}{2} \rceil$ and $n = \lceil \frac{q}{2} \rceil$. Each term in the geometric series corresponds to the partitioning P_i as follows,

$$P_i = \begin{cases} r^{i\ \ 1} & \text{, if } 1 \le i \le n - 1 \\ Kr^{i\ \ 1} & \text{, if } i = n \ \ \& \ \ (q - 1) \text{ is odd,} \\ 2Kr^{i\ \ 1} & \text{, if } i = n \ \ \& \ \ (q - 1) \text{ is even.} \end{cases} \tag{4.2}$$

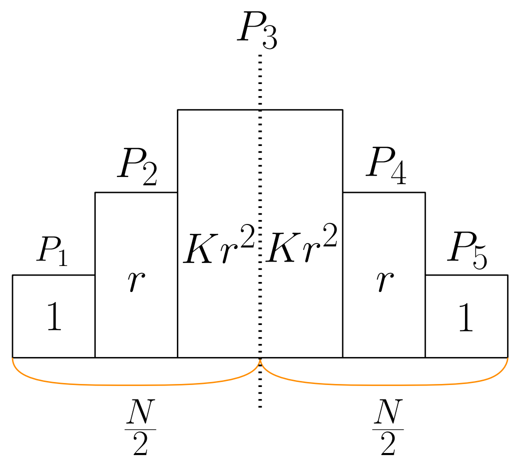
The scaling factor K in the above equations is defined as follows,

$$K = \begin{cases} \left(\frac{1}{q\ \ 1}\right) & \text{, if } (q - 1) \text{ is even.} \\ \left(1 - \frac{1}{q\ \ 1}\right) & \text{, if } (q - 1) \text{ is odd.} \end{cases}$$

Note that the partitioning approach described above differs based on whether the number of partitions q is even or odd. In the case of an even number of partitions, when $i = n$ (where n is the central partition), the central partition is split into two sub-partitions. To account for this, the term $Kr^{n\ \ 1}$ in the modified geometric series is doubled to obtain the final solution for the central partition. This solution is then mirrored onto the other half of the partitions to obtain the complete partitioning. The partitioning algorithm for obtaining the



(a) Partitioning of N nodes into 6 partitions.



(b) Partitioning of N nodes into 5 partitions.

Figure 4.1: Modified geometric partitioning.

final partitioning is presented in Algorithm 1. To improve readability for Algorithm 1, the functions to check the non-decreasing nature of the left half partition and the function to check the total sum of nodes are presented as Algorithms 2 and 3, respectively. For instance, in Figure 4.1a and 4.1b, we illustrate the partitioning with a diameter of $D = 5$ and 4 respectively. Despite their similar setups, there is a distinction in the central partition between the two cases. In the case of a diameter of 4, the central partition is split into two, while in the case of a diameter of 5, there are two separate partitions. In the following section, we will discuss the construction of network design $\mathcal{G}$ using this partitioning algorithm.

4.2 Network design $\mathcal{G}$:

Upon discussing the partitioning problem in the previous section, we can now delve into the design of the network, $\mathcal{G}$. The aim is to create a network that exhibits both strong structural controllability and maximal robustness, for specified parameters number of nodes N, number of leaders N_L, number of followers N_F, and diameter D. Let us consider a vertex set defined as follow,

$$V = V_\ell \cup V_f,$$

Algorithm 1 Clique chain partitioning

Input: N, q

Output: P

```
n ← ⌈q/2⌉
r ← solve equation 4.1
P_i ← solution of equation 4.2                                  ▷ ∀i
− − −Non − decreasing check − −−
P ← NonDecLeftNodeCount(P, n)
− − −Assign both halfs − −−
(P, rem) ← MakeSymmetricAndCheck(P, q, n)
− − −Check total number of nodes − −−
if rem < 0 then
    remL ← ⌊rem/2⌋
else
    remL ← ⌈rem/2⌉
end if
i ← n
while remL ≠ 0 do
    P_U ← P(i) + remL/|remL|
    if P_U ≥ P(i − 1) then
        P(i) ← P_U
        remL ← remL − remL/|remL|
    end if
    if i = 2 then
        i ← n
    else
        i ← i − 1
    end if
end while
(P, rem) ← MakeSymmetricAndCheck(P, q, n)
for i = n + 1 to D do
    Ri ← D + 1 − (i − n)
    if rem < 0 and (P(i) − 1 ≥ P(i + 1) or i = D) then
        P(i) ← P_UC
    else if rem > 0 and (P(Ri) + 1 ≤ P(Ri − 1) or Ri = n + 1) then
        P(Ri) ← P(Ri) + 1
    end if
    rem ← N − Σ_{j=1}^{n} P(j)
end for
```

Algorithm 2 Ensure non-decreasing behaviour in left half

```
procedure NonDecLeftNodeCount(P, n)
    count ← 1
    while count > 0 do
        count ← 0
        for i = n to 3 do
            if P(i) < P(i − 1) and P(i − 1) > 1 then
                P(i) ← P(i) + 1
                P(i − 1) ← P(i − 1) − 1
                count ← count + 1
            else if P(i) < P(i − 1) and P(i − 1) = 1 then
                P(i) ← P(i) + 1
                count ← count + 1
            else if P(i − 1) = 0 then
                P(i − 1) ← 1
                count ← count + 1
            end if
        end for
    end while
    return P
end procedure
```

Algorithm 3 Mirror left to right and check total nodes

```
procedure MakeSymmetricAndCheck(P, q, n)
    P(q − (i − 1)) ← P(i)                                ▷ where i = 1 to n
    Sum ← Σ_{i=1}^{q} P(i)
    rem ← N − Sum
    return (P, rem)
end procedure
```

where, $V_\ell = \{\ell_{i,j}\}$ is the set of leaders and $V_f = \{u_{k,j}\}$ is the set of followers, for all $j \in \{1, 2, \ldots, D\}$, $i \in \{1, 2, \ldots, L(j)\}$ and $k \in \{1, 2, \ldots, F(j)\}$. The construction of $\mathcal{G}$ proceeds as follows:

- The leader set V_ℓ is partitioned into D partitions using the partitioning Algorithm 1

- Similarly, the follower set V_f is also partitioned into D partitions using the same Algorithm 1. Figure 4.2 shows the partitioning of the leader nodes $\{L(1)\ , L(2), \cdots ,$ $L(D)\}$ and the partitioning of follower nodes as $\{F(1),\ F(2),\ \cdots,\ F(D)\}$.

- The number of nodes in each partition of leaders is represented as $\{P_L(1),\ P_L(2),$ $\cdots, P_L(D)\}$, while for followers it is denoted as $\{P_F(1), P_F(2), \cdots, P_F(D)\}$.

- The leader nodes in each partition form a clique and are also completely connected to their adjacent leader partitions constructing a clique chain.

- The followers are connected by a path graph with the first leader and last follower as its end nodes; in the following manner,

$$\{\ell_{1,1}, u_{1,1}, \cdots, u_{F(1),1}, u_{1,2}, \cdots, u_{F(2),2}, \cdots, u_{F(D),D}\}.$$

  Figure 4.3 shows the formation of a clique chain between the leader partitions and path graph from $\ell_{1,1}$ to the $u_{F(D),D}$

- The leaders $\ell_{i,j}$ are connected to followers $\{\{u_{m,j\ \ 2}\}, \{u_{m,j\ \ 1}\}, \{u_{m,j}\}\} \forall i, m\ \ \&\ \ j > 1$.

- Additionally, all the nodes in $u_{(P_L(j)\ \ 1),j}\ \forall j > 1$ where $P_L(j) > 2$ have an edge with the nodes in $u_{m,j+1}\ \forall m$.

This network design maintains a diameter D between the first leader ($\ell_{1,1}$) to the last follower ($u_{P_F(D),D}$). It is interesting to note that, this construction provides graphs close to optimal in most cases and converges to optimal for certain inputs, we discuss this observation in the following Section 4.3 as we compare this construction to all known graphs for the given parameters and from previous experimental results from (Ellens et al., 2011).

Lemma 2.1. *For a graph $\mathcal{G}$ (as described above) with N number of nodes, the proposed leader set $\{\ell_1, \ell_2, \cdots, \ell_{N_L}\}$ is a ZFS.*

Proof. From the construction of $\mathcal{G}$, we notice that except ℓ_1, all the other leaders form a complete graph with the follower and leader partitions adjacent to them. Node ℓ_1 is only connected to the first follower and $L(2)$. This makes ℓ_1 the only leader that can initiate the zero forcing process.

Proceeding with the zero forcing process, all the other leaders would still have more than one white neighbor. Only the first follower of $F(1)$ would have a single white node in its neighborhood which it will be able to color. A similar pattern is followed through till all the nodes in $F(1)$ and also till $u_{P_F(2)\ 1,2}$ in $F(2)$. Now, $u_{P_F(2)\ 1,2}$ is unable to color $u_{P_F(2),2}$ because $u_{P_F(2)\ 1,2}$ is connected to all the nodes in $u_{i,3}$ where $i = \{1, \cdots, P_F(3)\}$. However, now all the leaders in $L(2)$ would only have one white node in their neighborhood which is $u_{P_F(2),2}$, so that allows any one of those leaders to continue the zero forcing process by coloring $u_{P_F(2),2}$. This continues till the last follower is colored and we can notice that the pattern of the zero forcing process would be unique for any given specifications of N, N_L and D. This makes the given leader set the ZFS of $\mathcal{G}$. $\square$

Lemma 2.2. *A graph $\mathcal{G}$ with number of nodes N, diameter D and given leader set $\mathcal{V}_\ell$ has maximal edge set, i.e., the inclusion of an extra edge to $\mathcal{G}$ would result in either a reduction in the graph's diameter or the speci ed leader set no longer being a ZFS of the graph $\mathcal{G}$.*

Proof. As discussed in Lemma 2.1, the Graph $\mathcal{G}$ has a unique zero forcing process. There are six types of edges that can be added to $\mathcal{G}$. They include,

- First leader to followers (non-leader) in adjacent partition edges,
  $(\ell_{1,1}, u_{k,2})$, where $k \neq 1$,

- Follower to follower in adjacent partition edge,

  $(u_{i,j}, u_{k,j^0})$, where $j^0 = \{j-1, j, j+1\}$,

- First leader to followers in non-adjacent partition edge,

  $(\ell_{1,1}, u_{k,2})$, where $k \neq 1$ for $j = 1$,

- Leader to follower in non-adjacent partition edge,

  $(\ell_{i,j}, u_{k,j^0})$, where $j^0 \neq \{j-1, j, j+1\}$,

- Leader to leader in non-adjacent partition edge,

  $(\ell_{i,j}, u_{k,j^0})$, where $j^0 \neq \{j-1, j, j+1\}$, and

- Follower to follower in non-adjacent partition edge,

  $(u_{i,j}, u_{k,j^0})$, where $j^0 \neq \{j-1, j, j+1\}$.

It can be easily observed that the partitions connected with its adjacent partition form the basis for maintaining the distance between the first leader $\ell_{1,1}$ and the last follower $u_{P_F(D),D}$. Any additional edge falling into the last four categories would shorten the distance between these two nodes, ultimately leading to a reduction in the diameter of the graph. The first two categories of edges are edges between vertices that belong to a unique zero forcing process as mentioned in the design of $\mathcal{G}$ and also shown in Lemma 2.1. If an edge is added between any two non-adjacent nodes in this particular zero-forcing process, the preceding node cannot continue the zero-forcing process because it will then have more than one white neighbor. This implies that the addition of any edge other than the ones already existing in $\mathcal{G}$ will result in a graph for which the given leader set is not a ZFS or a graph with a reduced diameter. □

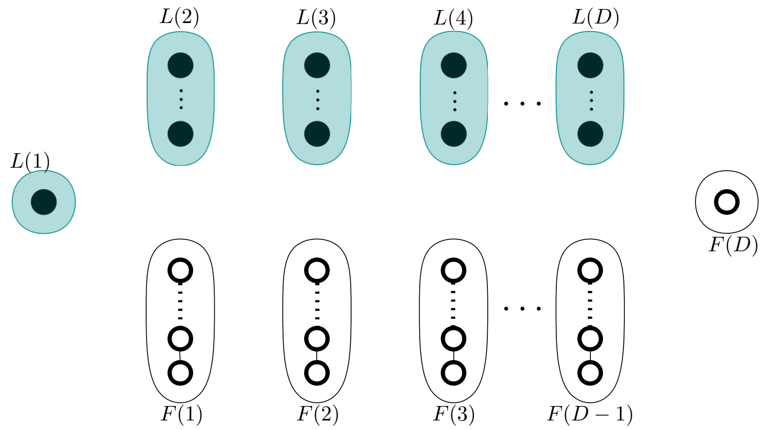


Figure 4.2: Partitioning of nodes.

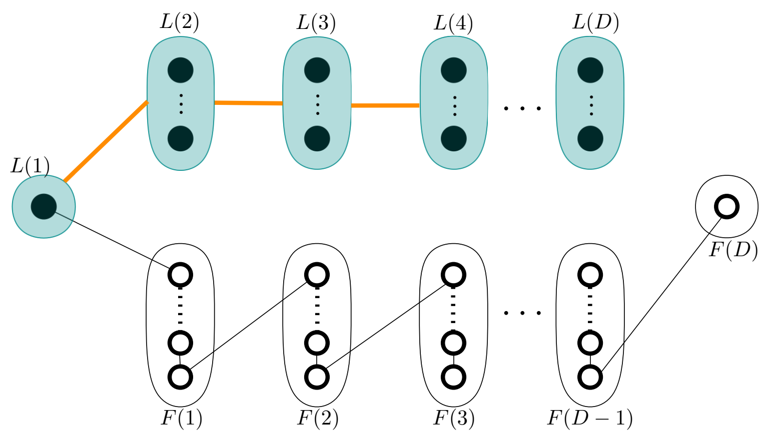


Figure 4.3: Clique chain between leaders and path graph through all followers.

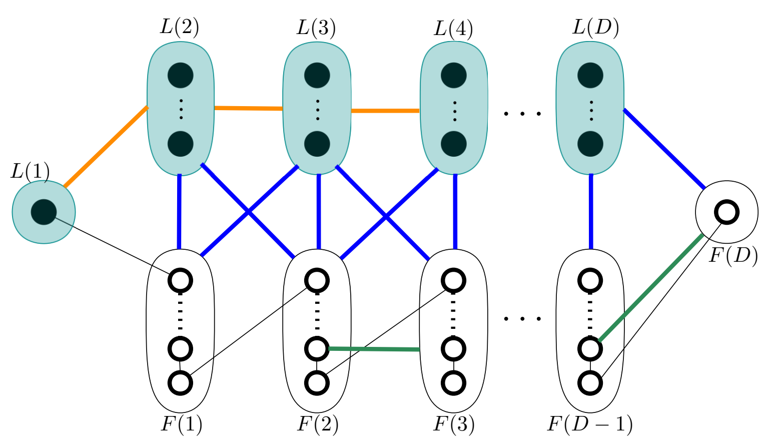


Figure 4.4: $\mathcal{G}$ with maximal edges.

4.3 Convergence to optimal clique chains

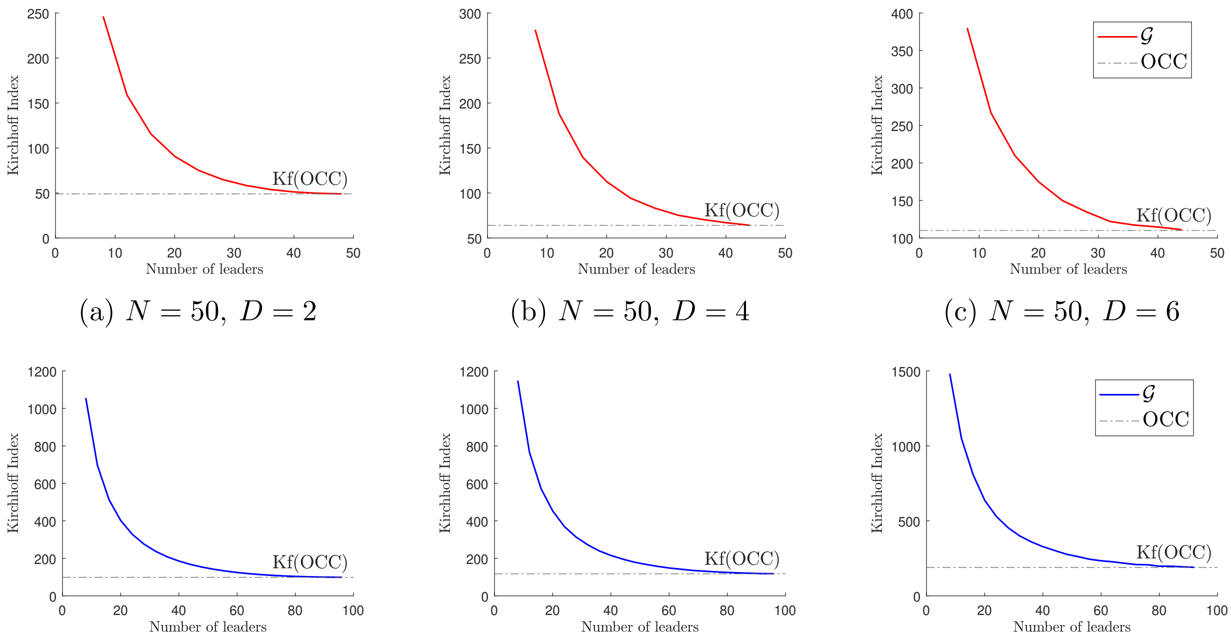
The network design $\mathcal{G}$ produces a maximally robust graph for given N, N_L and D. Now, the question arises, when the parameters N, N_L and D allow for the formation of an OCC of N nodes and D diameter, needs $N_L = N - D$ leaders), it is of interest to understand how $\mathcal{G}$ differs from OCC in such scenarios.

In previous work by (Ellens et al., 2011), they analyze the robustness of networks as the resistance of electrical circuits and exhaustively searched for networks with the lowest Kirchhoff index for specific N and D. With that notion, we want to evaluate the convergence of our network designs to the optimal Kirchhoff index. Figure 4.5 shows the variation of the Kirchhoff index with respect to the number of leaders N_L for various diameters. It can be seen that as N_L moves closer to $N - D$ for given N and D, $\mathcal{G}$ indeed converges to the Kirchhoff index value of optimal clique chains of the N and D. The dotted lines denote the Kirchhoff index of optimal clique chain for given N and D, with $N_L = N - D$.

Furthermore, we explore the robustness of our graph $\mathcal{G}$ at $N_L = N - D$ for various N, Figure 4.6, shows the variation of Kirchhoff index with respect to D, where $N_L = N - D$, for OCC and $\mathcal{G}$. It can be observed that the Kirchhoff index of $\mathcal{G}$ is almost identical to OCC. In Chapter 6, we delve deeper into the experimental setup and the results of the experiments. It is shown that $\mathcal{G}$ is indeed co-spectral to OCC for $N_L = N - D$.

Furthermore, in Chapter 6, we present experimental data to support the convergence of these networks to optimum robustness. We analyze large datasets of networks to assess the robustness of our graphs and compare their performance with other networks. This allows us to demonstrate the effectiveness of our network designs in achieving maximally robust networks for the given parameters of N, N_L and D.

The property of convergence to the optimal Kirchhoff index makes $\mathcal{G}$ a highly robust network design. However, as we observe in Chapter 6, there are cases where $\mathcal{G}$ performs sub-optimally. In some instances, we have found that a different network design, which we



(a) $N = 50,\ D = 2$ (b) $N = 50,\ D = 4$ (c) $N = 50,\ D = 6$

(d) $N = 100,\ D = 2$ (e) $N = 100,\ D = 4$ (f) $N = 100,\ D = 6$

Figure 4.5: Convergence of Kirchhoff index of $\mathcal{G}$ to that of Optimal clique chains for various N_L, given $N \in \{50, 100\}$ and $D \in \{2, 4, 6\}$.

refer to as $\mathcal{J}$, achieves optimal performance. $\mathcal{J}$ is currently a work in progress and will be explored in future research. For example, when considering $N = 8$, $N_L = 3$, and $D = 3$, $\mathcal{J}$ has a Kirchhoff index of 13.3386, while $\mathcal{G}$ has a Kirchhoff index of 15.0371. Interestingly, this construction is similar to a network design presented in (Patel et al., 2023). Future work will also investigate the deviation in Kirchhoff index between $\mathcal{J}$, $\mathcal{G}$.

Remark 2.1. It is remarkable to note that for cases where N, N_L and D are feasible to produce an optimal clique chain, both $\mathcal{G}$ and $\mathcal{J}$, converge to optimal clique chain.

However, designing a specific type of network from all possible networks for a specific set of parameters can be challenging. Graph grammars offer a promising solution for designing networks, as they provide a set of rules that can be used to generate networks in a distributed manner. These rules enable the emergence of desired network structures through local interactions among nodes, allowing for decentralized network design with specific topological

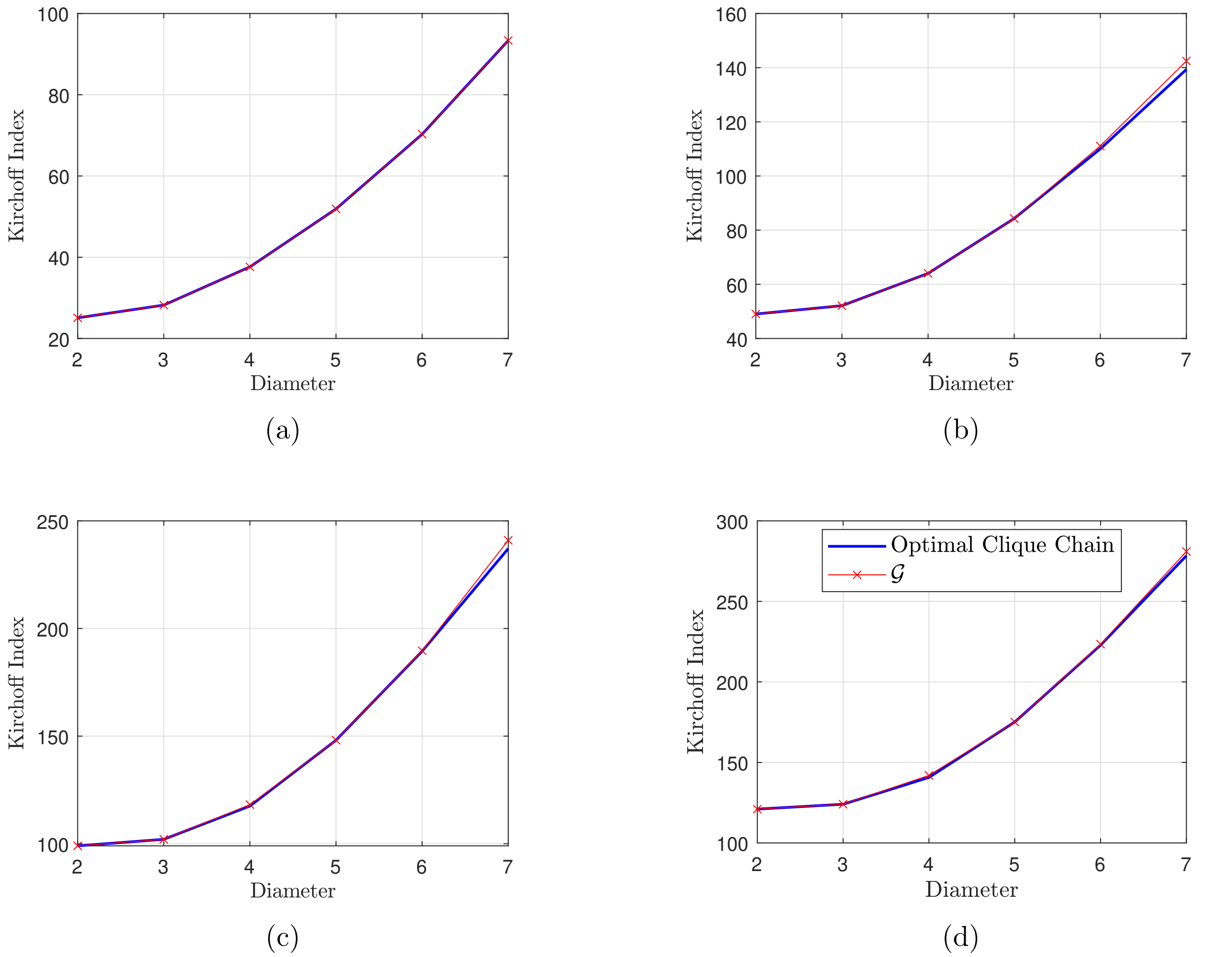


Figure 4.6: Comparison of Kirchhoff index of $\mathcal{G}$ to that of Optimal clique chains for N, D, and $N_L = N - D$.

characteristics. The upcoming chapter delves into graph grammars, wherein we introduce one specifically for $\mathcal{G}$ and showcase its practical implementation with an illustrative example.

CHAPTER 5

DISTRIBUTED DESIGN OF NETWORKS USING GRAPH GRAMMARS

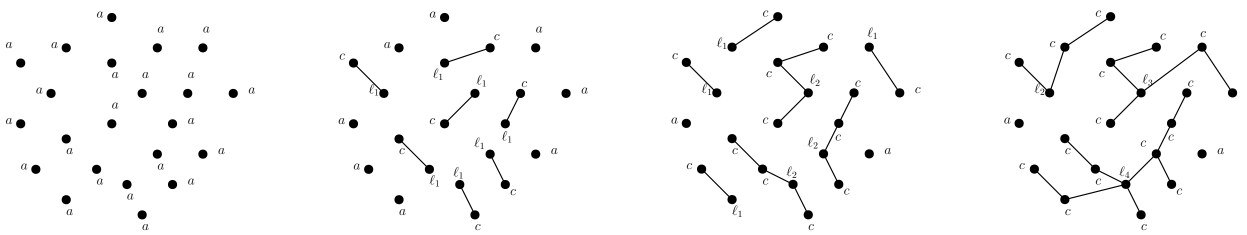
In graph theory, graph grammars are a set of rules that govern how graphs can be constructed through a set of graph transformation rules. Graph grammars allow for the generation of complex graphs by iteratively applying transformation rules to a starting graph. The rules specify how the graph can be modified by adding or removing vertices and edges. For a graph $\mathcal{G}$ with a set of initially labeled nodes, we define a set of rules $\mathcal{R} = \{r_1, r_2, \cdots, r_n\}$ using which the nodes modify connections with other nodes and update their labels (Klavins, 2007). A rule of the form $r_i : \mathcal{G}_t \longrightarrow \mathcal{G}_{t+1}$, is applicable at time t if there exist nodes in $\mathcal{G}_t$ with desirable labels to apply the rule r_i, then a transformed graph $\mathcal{G}_{t+1}$ is formed at the next time step $t + 1$. For example, consider the following simple example with 2 rules,

$$(r_0) \quad a \qquad a \quad \rightharpoonup \quad \ell_1 \quad \textbf{—} \quad c$$

$$(r_1) \quad \ell_i \qquad \ell_i \quad \rightharpoonup \quad \ell_{i+1} \quad \textbf{—} \quad c \qquad \forall 1 \leq i \leq 4.$$

These rules can be used to generate graph structures known as crystals, as described in (Abbas and Egerstedt, 2011). An illustrative example of this process is shown in Figure 5.1. Consider a set of nodes labeled 'a' at time $t = 0$, as depicted in Figure 5.1a. By applying various rules at each time step, Figure 5.1d shows a possible structure of the network after 4 evolutions.

It is important to note that even though the graph grammars are split into multiple rules, its application is only based on the availability of nodes suitably labeled to apply a rule (Patel et al., 2023), these rules can be applied concurrently with each other. In the following section, we present the graph grammars to generate the networks presented in Chapter 4 and demonstrate its application through an example.



(a) (b) (c) (d)

Figure 5.1: Example of application of graph grammars to form crystals.

5.1 Distributed design of network $\mathcal{G}$

We define the set of rules $\mathcal{R}$ that encode the rules to construct the graph $\mathcal{G}$ for a given number of nodes N, number of leaders N_L and diameter D. The number of followers can be calculated as $N_F = N - N_L$. For the given parameters we first run the partitioning Algorithm 1 to generate the vectors P_l and P_f whose elements comprise the number of nodes in each partition of leaders and followers respectively. Consider a set of $N-1$ number of nodes labeled a and one node labeled $S_{1,1}$ which is the starter node. With this information, we present the graph grammar as follows,

(r_0) $\quad a \quad S_{1,1} \quad a \rightharpoonup y_{1,1} — \ell_{1,1} — S_{1,2}$

(r_1) $\quad S_{1,i} \quad a \rightharpoonup \ell_{1,i} — S_{1,i+1} \qquad 2 \leq i < D$

(r_2) $\quad y_{1,i} \quad a \rightharpoonup f_{1,i} — y_{1,i+1} \qquad 1 \leq i < D$

(r_3) $\quad S_{1,i} \rightharpoonup \ell_{1,i} \qquad i = D$

(r_4) $\quad y_{1,i} \rightharpoonup f_{1,i} \qquad i = D$

(r_5) $\quad \ell_{j,i} \quad a \rightharpoonup \ell_{j,i} — \ell_{j+1,i} \qquad 1 \leq j < P_\ell(i)$

(r_6) $\quad f_{j,i} \quad a \rightharpoonup f_{j,i} — f_{j+1,i} \qquad 1 \leq j < P_f(i)$

(r_7) $\quad f_{j,i} \quad f_{1,i+1} \rightharpoonup f_{j,i} — f_{1,i+1} \qquad \forall i \;\&\; j = P_f(i)$

(r_8) $\quad f_{1,i} — f_{1,i+1} \rightharpoonup f_{1,i} \quad f_{1,i+1} \qquad \forall\, i; \begin{cases} i \geq 2 & \text{where } P_f(i) > 2 \\ i = 1 & \text{where } P_f(i) > 1 \end{cases}$

(r_9) $\quad f_{j,i} \quad f_{m,i+1} \rightharpoonup f_{j,i} — f_{m,i+1} \qquad \forall\, m,\; j = P_f(i) - 1 \;\;\&\;\; 2 \leq i < D$

$(r_{10})\quad \ell_{i,j} \qquad \ell_{m,n} \rightharpoonup \ell_{i,j}\text{—}\ell_{m,n} \qquad\qquad \forall\, m,\ i,\ \text{ where } j-1 \leq n \leq j+1; 1 < j < D.$

$(r_{11})\quad \ell_{i,j} \qquad f_{m,n} \rightharpoonup \ell_{i,j}\text{—}f_{m,n} \qquad\qquad \forall\, m,\ i;\ j-2 \leq n \leq j,\ 1 < j \leq D$

The set of rules r_0 to r_7 serves as the backbone for the intended graph. These rules connect two paths of length D at the starting node $S_{1,1}$ - one path comprising of leader nodes and the other consisting of follower nodes. Each node i in the path generates a branch of length $P_\ell(i)$ if it is a leader or $P_f(i)$ if it is a follower. Rules r_8 to r_{11} maximize the number of edges possible while preserving SSC, ensuring that robustness is maximized, as assessed by Kirchhoff index as mentioned in Section 2.4.

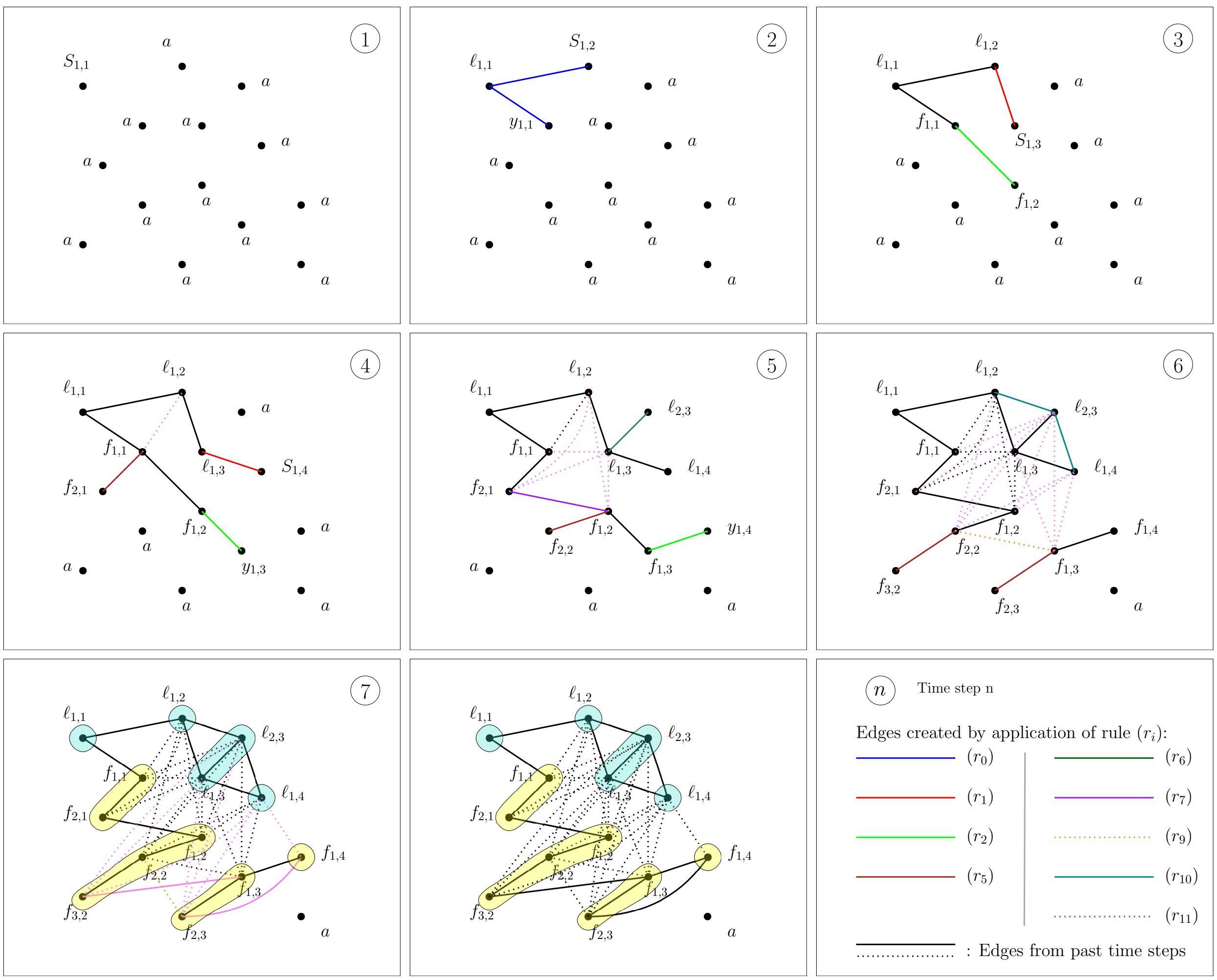


Figure 5.2: An illustrative example of constructing $\mathcal{G}$ using rules $\mathcal{R}$.

CHAPTER 6

NUMERICAL EVALUATION

The primary objective of this work is to design networks that exhibit both SSC and maximal robustness in networks for given parameters N, N_L and D. We propose that these networks perform well in terms of robustness and can even achieve optimal robustness for certain network parameters. To numerically evaluate these results from Section 4.3, we conduct a series of experiments using an exhaustive dataset of connected undirected networks(Mckay, 2023). In this chapter, we describe the experimental setup and the tests conducted to evaluate the performance of network design $\mathcal{G}$.

In Chapter 3, we demonstrated that the network design $\mathcal{G}$ is both strong structurally controllable and maximally robust. To evaluate the optimality of the network design we conduct an exhaustive search on a large database of networks, aiming to identify networks that exhibit higher robustness performance than $\mathcal{G}$.

In order to conduct the experiment, we generate all possible graphs with 8 and 9 nodes using $\mathcal{G}$, considering a various number of leaders and diameters. However, due to the combinatorial nature of network formation, the number of possible networks increases exponentially with the number of nodes, making it computationally expensive to search through all possible graphs with 10 or more nodes. Nevertheless, the database of graphs with 8 and 9 nodes is small enough to conduct the study and large enough to produce graphs with a wide range of diameters and numbers of leaders. The results of this experiment are presented in Table 6.1 for 8 and 9 number of nodes respectively. The total number of connected undirected networks with 8 nodes in (Mckay, 2023) is 11117 and of 9 nodes is 261080. The table presented also displays the number of graphs available for a given N, N_L, and D from the extensive dataset of N nodes. This serves to highlight the rarity of the solution set.

As the number of nodes increases, the number of networks grows exponentially, making exhaustive searching through such a large dataset becomes challenging. To overcome this,

Table 6.1: Experimental results for graphs with 8 and 9 nodes.

N	D	N_L	$K_f(\mathcal{G})$	Total number of graphs with N, N_L, D	Better than $\mathcal{G}$
8	2	2	21.2440	3 (0.03%)	0
		3	13.1897	424 (3.81%)	0
		4	9.8240	2678 (24.09%)	1
		5	8.1569	987 (8.88%)	0
		6	7.3333	61 (0.55%)	0
	3	3	15.0371	2820 (25.37%)	15
		4	12.3260	2692 (24.22%)	4
		5	11.0159	196 (1.76%)	0
	4	4	19.0333	203 (1.83%)	0
9	2	2	27.7994	3 (0.001%)	0
		3	17.1865	1152 (0.44%)	0
		4	12.6409	36693 (14.05%)	3
		5	10.2675	48666 (18.64%)	1
		6	8.9666	4902 (1.88%)	0
		7	8.2857	102 (0.04%)	0
	3	3	19.2594	26832 (10.28%)	84
		4	15.4066	97227 (37.24%)	155
		5	12.4155	23548 (9.02%)	0
		6	11.9583	491 (0.19%)	0
	4	4	20.713	7702 (2.95%)	0
		5	19.5714	730 (0.28%)	0

Table 6.2: Evaluation of robustness of $\mathcal{G}$ with random graphs.

N	D	N_L	Number of graphs	$K_f(\mathcal{H})$	$K_f(\mathcal{G})$
65	3	38	424	303.4176	85.6981
70	3	39	352	349.0156	96.0770

we utilize a database of random graphs with $N \in \{65, 70\}$ and diameter 3, generated by (Ahmed and Abbas, 2023). Let us call a network from this dataset as $\mathcal{H}$. We identify the ZFS of the network $\mathcal{H}$ with the highest robustness, i.e., lowest Kirchhoff index and compute $\mathcal{G}$ for those parameters. We observe that the Kirchhoff index of $\mathcal{G}$ is at least three times better than the random graphs, as shown in Table 6.2.

CHAPTER 7

CONCLUSIONS

The distributed design of network systems that are both strong structurally controllable and maximally robust is a key focus of this work. We emphasize that since controllability and robustness are opposing properties of networks, designing networks that excel in both aspects is a challenging problem. Furthermore, the task of designing networks with given parameters number of nodes N, and number of leaders N_L, that are simultaneously strong structurally controllable and maximally robust poses even greater challenges. To tackle this problem, we provide network designs $\bar{\mathcal{G}}_1$, $\bar{\mathcal{G}}_2$, and $\bar{\mathcal{G}}_3$ that take user inputs N and N_L and provide networks that are strong structurally controllable and maximally robust. But their diameters are limited as $2 \leq D \leq \lceil \frac{N}{N_L} \rceil$. To provide more flexibility in diameter, we further improve the problem to take diameter D also as a user input to provide a network that is strong structurally controllable and maximally robust. We propose a novel network design $\mathcal{G}$ that achieves strong structural controllability in terms of zero forcing process and maximal robustness in terms of Kirchhoff index for given N, N_L and D. The construction of our proposed network $\mathcal{G}$, involves solving a number partitioning, which is a combinatorial optimization problem. We use a modified geometric series to arrive at a solution for number partitioning. We extensively evaluate our graphs using a large database of networks, and our findings indicate that $\mathcal{G}$ is close to optimal and even achieves optimality in some cases. To further verify the optimality of $\mathcal{G}$, we compare it with networks that are known to have optimum robustness. Additionally, we propose graph grammars, which are sets of rules that govern the interactions between nodes, as a distributed method for constructing $\mathcal{G}$. We believe that the results of this work have potential applications in scenarios where the need to control a network system in the most resilient manner with minimal inputs is critical. A potential direction for future research could be to investigate the resilience of networks constructed using $\mathcal{G}$ in the presence of adversarial conditions.

BIBLIOGRAPHY

Abbas, W. and M. Egerstedt (2011). Hierarchical assembly of leader-asymmetric, single-leader networks. In *American Control Conference (ACC)*, pp. 1082–1087.

Abbas, W. and M. Egerstedt (2012). Robust graph topologies for networked systems. In *3rd IFAC Workshop on Distributed Estimation and Control in Networked Systems (NecSys)*, pp. 85–90.

Abbas, W., M. Shabbir, A. Y. Yazıcıoğlu, and A. Akber (2020). Tradeoff between controllability and robustness in diffusively coupled networks. *IEEE Transactions on Control of Network Systems 7*(4), 1891–1902.

Ahmed, O. U. and W. Abbas (2023). Random graph data. `https://github.com/ObaidUllah-2022/ZFS_Data`. Accessed: 2023-04-20.

AIM Minimum Rank Special Graphs Work Group (2008). Zero forcing sets and the minimum rank of graphs. *Linear Algebra and its Applications 428*(7), 1628–1648.

Chapman, A. and M. Mesbahi (2013). On strong structural controllability of networked systems: A constrained matching approach. In *American control conference (ACC)*, pp. 6126–6131.

Ellens, W., F. M. Spieksma, P. Van Mieghem, A. Jamakovic, and R. E. Kooij (2011). Effective graph resistance. *Linear Algebra and its Applications 435*(10), 2491–2506.

Jia, J., H. L. Trentelman, W. Baar, and M. K. Camlibel (2019). Strong structural controllability of systems on colored graphs. *IEEE Transactions on Automatic Control 65*(10), 3977–3990.

Klavins, E. (2007). Programmable self-assembly. *IEEE Control Systems Magazine 27*(4), 43–56.

Mckay, B. (2023). Combinatorial data. `http://users.cecs.anu.edu.au/~bdm/data/graphs.html`. Accessed: 2023-04-17.

Menara, T., G. Bianchin, M. Innocenti, and F. Pasqualetti (2017). On the number of strongly structurally controllable networks. In *American Control Conference (ACC)*, pp. 340–345.

Mertens, S. (2006). Number partitioning. *Computational complexity and statistical physics* (125).

Monshizadeh, N., S. Zhang, and M. K. Camlibel (2014). Zero forcing sets and controllability of dynamical systems defined on graphs. *IEEE Transactions on Automatic Control 59*, 2562–2567.

Mousavi, S. S., M. Haeri, and M. Mesbahi (2017). On the structural and strong structural controllability of undirected networks. *IEEE Transactions on Automatic Control 63*(7), 2234–2241.

Pasqualetti, F., S. Zampieri, and F. Bullo (2014). Controllability metrics, limitations and algorithms for complex networks. *IEEE Transactions on Control of Network Systems 1*(1), 40–52.

Pasqualetti, F., S. Zhao, C. Favaretto, and S. Zampieri (2020). Fragility limits performance in complex networks. *Scienti c Reports 10*(1), 1–9.

Patel, P., J. Suresh, and W. Abbas (2023). Distributed design of controllable and robust networks using zero forcing and graph grammars. *https://arxiv.org/abs/2303.05596*.

Shabbir, M., W. Abbas, Y. Yazicioglu, and X. Koutsoukos (2022). Computation of the distance-based bound on strong structural controllability in networks. *IEEE Transactions on Automatic Control*.

Siami, M. and N. Motee (2013). Fundamental limits on robustness measures in networks of interconnected systems. In *52nd IEEE Conference on Decision and Control*, pp. 67–72.

Wang, H., R. Kooij, and P. Van Mieghem (2010). Graphs with given diameter maximizing the algebraic connectivity. *Linear Algebra and its Applications 433*(11-12), 1889–1908.

Yim, M. (2007). Modular self-reconfigurable robot systems: Challenges and opportunities for the future. *IEEE Robotics and Automation Magazine 10*, 2–11.

Young, G. F., L. Scardovi, and N. E. Leonard (2016). A new notion of effective resistance for directed graphs—part i: Definition and properties. *IEEE Transactions on Automatic Control 61*, 1727–1736.

Zelazo, D. and M. Bürger (2015). On the robustness of uncertain consensus networks. *IEEE Transactions on Control of Network Systems 4*(2), 170–178.

BIOGRAPHICAL SKETCH

Johir Suresh was born in Coimbatore, Tamil Nadu, India. He was the school topper in 12th grade at PSG Public Schools, Coimbatore. He pursued a Bachelor of Technology in Automobile Engineering at Manipal Institute of Technology, Manipal, where he worked on modeling vehicle dynamics for Project Manas. He graduated in June 2020 and is currently graduating with a Master of Science in Mechanical Engineering at The University of Texas at Dallas, with a concentration in dynamic systems and control. He conducted research on the distributed design of controllable and robust networks under the guidance of Dr. Waseem Abbas. He will be pursuing his PhD in Automotive Engineering at Clemson University starting Fall 2023.

CURRICULUM VITAE

Johir Suresh

May 9, 2023

Educational History:

BTech, Automobile Engineering, Manipal Institute of Technology, 2020
MS, Mechanical Engineering, The University of Texas at Dallas, 2023

Employment History:

Grader: MECH 3305: CAD, The University of Texas at Dallas, Spring 2023
Grader: MECH 4340: Mechanical Vibrations, The University of Texas at Dallas, Fall 2022
Co-founder, IndoPure, Coimbatore, India, March 2020 – Present
Intern, Driven By You Mobility LLP, Hyderabad, India, May 2019 – June 2019
Racecar Driver, Volkswagen Motorsport India, 2017

Professional Recognitions and Honors:

All India Senior School Certificate Examination, School 1st, PSG Public Schools, 2016

Publications:

Patel, P., Suresh, J., Abbas, W., "Distributed Design of Controllable and Robust Networks using Zero Forcing and Graph Grammars", American Controls Conference 2023.

Professional Memberships:

Institute of Electrical and Electronics Engineers (IEEE), 2023 - present
American Society of Mechanical Engineers (ASME), 2023 - present