Networked Dynamic Systems: Identification, Controllability, and Randomness

Marzieh Nabi-Abdolyousefi

A dissertation submitted in partial fulfillment of
the requirements for the degree of

Doctor of Philosophy

University of Washington

2013

Reading Committee:

Mehran Mesbahi, Chair

Kristi Morgansen

Santosh Devasia

Maryam Fazel

Program Authorized to Offer Degree:
Aeronautics & Astronautics Engineering

University of Washington

Abstract

Networked Dynamic Systems:
Identification, Controllability, and Randomness

Marzieh Nabi-Abdolyousefi

Chair of the Supervisory Committee:
Professor Mehran Mesbahi
Aeronautics & Astronautics Engineering

The presented dissertation amis to develop a graph-centric framework for the analysis and synthesis of networked dynamic systems (NDS) consisting of multiple dynamic units that interact via an interconnection topology. We examined three categories of network problems, namely, identification, controllability, and randomness. In network identification, as a subclass of inverse problems, we made an explicit relation between the input-output behavior of an NDS and the underlying interacting network. In network controllability, we provided structural and algebraic insights into features of the network that enable external signal(s) to control the state of the nodes in the network for certain classes of interconnections, namely, path, circulant, and Cartesian networks. We also examined the relation between network controllability and the symmetry structure of the graph.

Motivated by the analysis results for the controllability and observability of deterministic networks, a natural question is whether randomness in the network layer or in the layer of inputs and outputs generically leads to favorable system theoretic properties. In this direction, we examined system theoretic properties of random networks including controllability, observability, and performance of optimal feedback controllers and estimators. We explored some of the ramifications of such an analysis framework in opinion dynamics over social networks and sensor networks in estimating the real-time position of a Seaglider from experimental data.

ACKNOWLEDGMENTS

Before starting my PhD at University of Washington, for a couple of months my imagination of UW was filled by a smiley picture of Wanda Frederick and an old picture of Prof. Mehran Mesbahi. My arrival coincided with many nice surprises; the beautiful campus, moving to a new and spacious office, and going to work on rainy days.

Now looking back at more than five years of being at UW and as a member of Distributed Space System Lab, my memory is filled with so many reminiscences of discussions, collaborations, conferences, laughs and cries, ups and downs, spending long days in the lab, hikes, every other Friday afternoon grocery shopping with Ran Dai, and many more. One common element of these memories is the presence of my supervisor, Prof. Mesbahi. Words cannot express my gratitude and respect toward him. His patience, knowledge, philosophical point of view, and in one word his personality have influenced me in different ways and made me who I am today. Working with a supervisor who considers both personal and professional growth of his students was a big fortune for me. He taught me having passions is one side of the story, but how we can have meaningful approaches toward them is another. He taught me that "enough is never enough" and there is no end to perfecting the quality of work we deliver. He taught me how to be a good teacher, a good researcher, and a better person. His lifetime lessons will stay with me forever.

I would also like to deeply thank my qualification exam, general exam, final exam, and reading committees Prof. Arthur Mattick, Prof. Soumik Pal, Prof. Eric Klavins, Prof. Uy-Loi Ly, Prof. Santosh Devasia, Prof. Kristi Morgansen, and Prof. Maryam Fazel for their time, insightful comments, and their support. In particular, I would like to sincerely acknowledge Prof. Morgansen and Prof. Fazel for their support

in different stages of my study. Their assistance taught me that at the end of the day, the academic environment is a big community of people who are aiming to help develope the world by fostering a new generation. Prof. Morgansen granted me the opportunity to work with the experimental data on the Seaglider. Her welcoming personality and unconditional support have always inspired me. Great discussions with Prof. Fazel have always been revitalizing and inspiring too. I would like to greatly acknowledge and thank the William E. Boeing Department of Aeronautics and Astronautics and, in particular, Wanda Frederick for creating a friendly environment to conduct research.

I would also like to thank my colleagues at Distributed Space Systems Lab. The lab has been the second home for me. I have learned quite a lot from each member of the lab, especially, Dr. Ran Dai, Airlie Chapman, Joshua Maximoff, and Dr. Dan Zelazo. Great discussions and fun collaborations with Airlie Chapman and Joshua Maximoff have influenced my research. Airlie's passions for research and her effort to succeed have set my professional standards and changed my definition of perfection. I would also like to deeply thank Dr. Ran Dai for making my experience in the lab unforgettable and a very pleasant one. I like to express my sincere gratitude to many good friends who made my experience in US, in Seattle, and at UW so colourful; in particular, Mr. Gregory Both and his wife Tuba who were my host family the first week of arriving to Seattle and became people who were there for me whenever I needed help.

The last and not the least is my family. I would like to thank my family for all their love and encouragements from thousands of miles away. Their pure love and their memories have enabled me to go on. I have been extremely fortunate to have a wise sister or I should say an angel such as Sareh, who can give wise advice whenever I need it the most, a responsible brother such as Aboozar, who I can trust the most to finish whatever I ask him to do, a hard working angel such as Soheila, who can play my supportive role at home when I am away, a smart sister such as Razieh,

who can make my day by just talking to her for 5 minutes, and my little angel Negin whose kindness is endless. Whatever I do in my life is an appreciation to our parents who taught us to love, be wise, responsible, supportive, kind, and have good morals.

DEDICATION

I would like to dedicate this body of work to my family members, who have supported me throughout my life.

TABLE OF CONTENTS

LIST OF FIGURES

Chapter 1

INTRODUCTION AND PRELIMINARIES

Networked dynamic systems (NDS) consist of multiple dynamic units that are connected via a network. The network can represent a sensing topology to coordinate a high-level objective or a communication topology to exchange information between agents. The research detailed in this dissertation mainly focuses on examining the relation between topological properties of the network on one hand and their system theoretical properties on the other. As such, our work is structural in its approach, and blends ideas from graph theory, combinatorics, probability theory, and control theory to examine networked dynamic systems. In the following, we briefly introduce the examined problems over networked dynamic systems.

1.1 Introduction

Controllability as a necessary condition to steer the behavior of a network or a system of agents communicating over a network to desired states is studied first. The connection between controllability and the network's topology is explored for certain classes of networks. The network dynamic system community has seen a recent surge in research focusing on the study of evolutionary networks [6, 12, 54, 40, 43, 44]. The network evolution can be viewed in different contexts, e.g., as time-varying, switching, or even as a random network. In this research, system properties of random networks are studied. In the meantime, the focus of the present document will be mainly on the Erdős-Renyi model of random networks.

Another aspect of our research has been on system identification of networked systems. This line of work has affinity with inverse problems associated with electrical networks [140], and the celebrated "Can one hear the shape of a drum?" that aims to characterize a manifold via its spectra [141], or more recently, "Can one hear the shape of a graph?" [142]. In fact, in the presented dissertation, we address the inverse problem related to consensus-type coordination algorithms. Consensus-type algorithms have recently been employed for

analysis and synthesis of a host of distributed protocols and control strategies in multi-agent systems, including flocking, formation control, rendezvous, and distributed estimation [1, 2, 3, 10, 105].

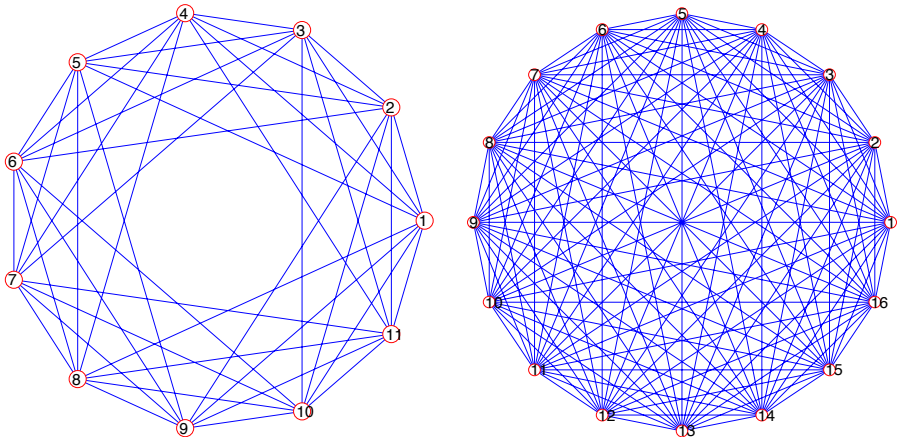
One of the key aspects of protocols studied in this report is the strong dependency between the interaction and information-exchange geometry among the multiple agents, on one hand, and the dynamic properties that these systems exhibit, on the other. Motivated by this dependency, in our work we consider the scenario where the interaction network is inside a "black box," and where only certain "boundary" nodes in the network can be influenced and subsequently observed. The "input" boundary nodes are then used to stimulate the network, whose response is subsequently observed at the "output" boundary nodes [138]. Our focus in the related sieve method for network identification, in the meantime, is to reduce the search space for the identification of the network topology by blending ideas from system identification, integer partitioning, and degree-based graph reconstruction. The implicit contribution of our analysis is its ramifications for exact identification from boundary nodes for networks that have an embedded consensus-type algorithms for their operation, including formation flying, distributed estimation, and mobile robotics.

Considering this setup in a closely related work [221], we present a node knockout procedure that aims to find the generating function of the graph Laplacian from the observed input-output data. Our focus in this approach is a *complete* identification of the network topology from input-output data that cannot be accomplished by the sieve method as proposed in [219]. This problem turns out to be non-trivial for the topology identification problem– as opposed to identifying the edge weights of an otherwise known weighted graph. However, as we show in this document, in the same spirit of the gene knockout procedure in experimental biology in order to identify interactions in gene networks [4, 144, 5], the network for consensus-type interactions can be completely identified via Laplacian based generating functions, provided that the identifier is endowed with the ability to perform node knockouts. In our context, the node knockout is essentially a grounding procedure, where the node broadcasts a zero state to its neighbors in the network without being removed. The implicit contribution of our analysis is its ramifications for exact network identification from boundary nodes of networks that have an embedded consensus-type algorithms for

their operation [3, 105].

Controllability of networked dynamic systems consisting of agents that have adopted a local coordination algorithm for their operation has recently been examined in the literature [11, 100, 101]. Network controllability is of great importance when a networked system is influenced and/or observed by external agents or inputs, for example, in the context of consensus algorithms and its extensions [105] or in quantum networks [106, 107]. It also has a direct relevance to situations where network input-output data are used for network tomography [219, 137, 139]. There is an intriguing current conjecture by Godsil [100, 106], stating that the ratio of graphs on n nodes that are *not* controllable from any set of nodes to the total number of graphs on n nodes approaches zero as $n \to \infty$. In the present report, we consider the controllability of a special class of networks characterized by circulant graphs (Figure 1.1) where agents in the network adopt a weighted consensus algorithm. In particular, we show that circulant graphs are controllable from a subset of nodes independent of their orders. Circulant graphs comprise a small fraction of graphs of order n: for a given n, there exist exactly 2^{n-1} unlabeled directed circulant graphs and $2^{\lfloor n/2 \rfloor}$ unlabeled undirected $n-$circulant graphs [108, 109]. Nevertheless, due to the wide use of circulant graphs in applications, it is expected that their generic controllability property will have applications in human-swarm interaction, network security [220], and quantum control [106].

Consider a network of n spatially distributed autonomous sensors, in which each sensor collects measurements in some modality of interest, e.g., temperature, sound, vibration, pressure, motion, or pollutant. Each sensor in the network is equipped with small storage, a radio transceiver, a micro-controller, and a battery power source on a single chip. Some engineering sensor networks consist of large numbers of sensors, from hundreds to even hundred of thousand nodes [35, 36, 37]. Sending large amounts of raw measurements to the fusion center requires a powerful processor with large bandwidth to gather the data. Significant communication delays and data loss across the network are common features of these systems. On the one hand, significant delays and packets drop have a negative impact on the performance of the designed system, as data need to arrive at their destination in time to be processed. These types of inefficiencies can be considered with stochastic models. The constraints on battery sources, on the other hand, motivate us to switch off a group



(a) (b)

Figure 1.1: Examples of circulant graphs: (a) 6-regular with 11 nodes, (b) 15-regular with 16 nodes; generated using Matgraph [178].

of sensors during some interval of measurement, which can be represented as a random communication network among the sensors or the sensors and the fusion center/node.

Battery lifetime is a major concern in all engineering applications. Specially in some cases, for instance habitat monitoring, non-rechargeable batteries are employed to observe the evolution of a particular phenomenon in nature for a couple of months [38]. In some other applications, reactive sample rates matter. For example, in soil moisture sensor networks [39], sensors observe some dynamics, which change rapidly during some intervals and slowly during other intervals. For example, soil moisture monitoring sensors need to send the measurements at higher data rates during rain events and at lower data rates between rain falls. Considering a large, wireless, multi-hop sensor network, there is too much data and too little bandwidth to pass the information along. Data traveling along unreliable communication channels, the effect of communication delays, and loss of information cannot be neglected. The stochastic nature of the communication channel and the arrival of the observations can be modeled as a random process [40, 43].

By increasing the number of sensors in a network, the need for efficient data aggregation becomes more and more evident. A centralized strategy and routing measurements to a

fusion center or node in a small network might be feasible. However, the computational and the network load both in the fusion center and in bottleneck nodes would be a major issue as the number of sensors increases in the network. Moreover, such a strategy is vulnerable to fusion center failure. Distributed estimation techniques have been studied with growing interest in the past few years by the systems community [43, 44, 45, 46, 47, 48, 49, 50, 51, 224].

In our setup, first, a partitioned information filter in a simple hierarchical estimation architecture, shown on Figure 1.2, provides a decentralized estimation algorithm. The proposed distributed filter uses the local computational capability of each sensor and uploads the processed measurements to the fusion center. Meanwhile, the proposed architecture implicitly requires that the sensors communicate with the fusion center at every time step. However, this assumption is, in general, unrealistic within the operational constraints imposed. For the sake of energy management as well as unreliable communication links and time delays, a random communication scheme between the sensors and the fusion center is considered in the dissertation. In such a setup, at every time step, the communication link between each sensor and the fusion center switches on with a probability $0 < p < 1$. In this document, we study the almost sure convergence of the estimation error in this communication setup.

We also take a probabilistic approach to design a feedback control mechanism over a consensus network evolving over a random network. Related to our work is the paper by Cao and Ren [14] who have studied optimal linear consensus algorithms for multi-vehicle systems in both continuous-time and discrete-time settings. We have been particularly inspired by a paper by Kalman in [13] which examined the linear-quadratic-regulator (LQR) for a random linear dynamic system in the discrete-time setting. In this work, Kalman proved the existence, optimality, and the stability of the proposed LQR scheme.

1.2 Preliminaries

This section introduces notational conventions and briefly reviews fundamental concepts in the areas of dynamical systems, control, combinatorics, and graph theory.

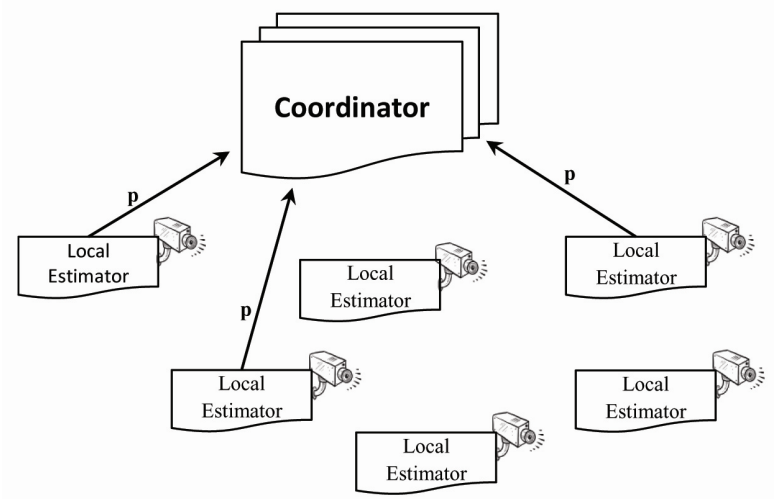


Figure 1.2: The communication link between each sensor and the coordinator switches on with a probability $0 < p < 1$.

1.2.1 Dynamical Systems and Control

This section briefly reviews a short summary of some important dynamic systems and control theory concepts that are relevant to this research. We consider a linear time-invariant (LTI) system with states $x(t) \in \mathbb{R}^{\kappa}$, control $u(t)$, exogenous state disturbance $w(t)$, measured output $y(t)$, measurement disturbance $v(t)$, and initial conditions $x(t_0)$. The LTI system, which we denote by the operator Σ, has the following state space realization

$$\Sigma : \begin{cases} \dot{x}(t) &= Ax(t) + Bu(t) + Gw(t) \\ y(t) &= Cx(t) + Du(t) + \Gamma v(t) \\ x(t_0) &= x_0. \end{cases} \tag{1.1}$$

Of primary interest is the designing of a stochastic or dynamic control law of the form $u(t) = Kx(t)$, or in the more general case $u(t) = K(t)y(t)$ to steer system Σ to some desired states. In practice, access to all states $x(t)$ is impossible. Therefore, estimating the states, $\hat{x}(t)$, through some limited measurements is unavoidable. The estimator can be designed by a linear system $A\hat{x}(t)$ to capture the main dynamics and minimize the distance between

the measurement, $y(t)$ and its estimate $C\hat{x}(t)$. The estimator assumes the form

$$\dot{\hat{x}}(t) = A\hat{x}(t) + K_f(y(t) - C\hat{x}(t)). \tag{1.2}$$

The existence of a controller or an estimator depends on the controllability (stabilizability) and observability (detectability) of the system Σ. Both controllability and observability properties of a stable system can be studied using the Gramians of the system

$$X_c = \int_0^\infty e^{At} B B^T e^{A^T t} dt \tag{1.3}$$

$$Y_o = \int_0^\infty e^{A^T t} C^T C e^{At} dt. \tag{1.4}$$

The Gramians can also be computed by solving a set of linear equations, through the following Lyapunov equations:

$$\begin{aligned} AX_c + X_c A^T + BB^T &= 0 \\ A^T Y_o + Y_o A + C^T C &= 0. \end{aligned}$$

In both cases, the Gramians are symmetric positive-semidefinite matrices. When the Gramian is positive definite, the system is controllable (observable). If this condition is not met, then each zero eigenvalue of the Gramian corresponds to an uncontrollable (unobservable) mode in the system.

In the case where the matrix is not stable, there are other tests, such as the Popov-Belevitch-Hautus (PBH) test, that can be used to determine whether the system is controllable or observable. The advantage of the Gramian is the ability to infer the relative degree of controllability or observability of different modes in the system. The relative degree of controllability or observability is accomplished by comparing eigenvalues of the Gramian.

The transpose of a vector or a matrix is denoted by the notation $(*)^T$, thus $(x^T)^T = x$.[1] The norm $||x||$ is equal to $(x^T x)^{1/2}$, and for a positive semidefinite matrix M, we use the special notation $||x||_M = (x^T M x)^{1/2}$. The eigenvalues of a matrix M are denoted by $\lambda_i(M)$. The following operations will prove to be useful throughout the report.

---

[1]The letter "T" denotes other variables such as time or transformation matrix, which will be clear from the context.

Definition 1.2.1. *Let $m_i \in \mathbb{R}^m$ denote the columns of the matrix $M \in \mathbb{R}^{m\times p}$ such that $M = [m_1, m_2, \ldots, m_p]$. Then forming the $mp-$vector by stacking the columns of M on top of each other defines the operator* $\mathbf{vec}(M)$*, i.e.,*

$$\mathbf{vec}(M) = \begin{bmatrix} m_1 \\ \vdots \\ m_p \end{bmatrix} \in \mathbb{R}^{mp};$$

we refer to the resulting vector as M_v.

The other operator **mat** is defined to map from vectors to matrices, the image of which is matrices of prescribed dimensions, i.e., $m \times p$, formed by concatenating entries of the mp-vector, read from top to bottom, from left to right and putting them in p columns; thus $\mathbf{mat}(M_v) = M = [m_1, m_2, \ldots, m_p]$.

The computation of covariances in the linear quadratic design presented in this report is facilitated by using the tensor notation. The tensor product $x \otimes y$ is defined as the outer product $\mathbf{vec}(xy^T)$. The tensor product $M \otimes N$ is equal to the Kronecker product of M and N. The tensor product of two linear transformation M and N is given by

$$(M \otimes N)(x \otimes y) = Mx \otimes Ny = \mathbf{vec}(Mxy^T N^T).$$

For any three matrices M, N, and R by which the product matrix MNR is defined, we have

$$\mathbf{vec}(MNR) = (R^T \otimes A)\mathbf{vec}(N).$$

We note that one can apply the **mat** operator on both sides of a vector identity to obtain a matrix identity.

The null and range spaces of the matrix M are denoted by $\mathcal{N}(\mathcal{M})$ and $\mathcal{R}(\mathcal{M})$, respectively. The set of positive integers is denoted by $\mathbb{Z}$, and $\mathbf{i}$ signifies $\sqrt{-1}$. When $[a]$ and $[b]$ denote two sets of indices, $M_{[a],[b]}$ denotes a submatrix of M whose rows and columns are the indices from the sets $[a]$ and $[b]$, respectively.

1.2.2 Graph Theory

An undirected (simple) graph $\mathcal{G}$ is denoted by $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with vertex set $\mathcal{V}$ and edge set $\mathcal{E}$, comprised of two-element subsets of $\mathcal{V}$; we use "nodes" or "agents" interchangeably with

"vertices." Two vertices $u, v \in \mathcal{G}$ are called adjacent if $\{u, v\} \in \mathcal{E}$. For vertex i, **deg** i and $\mathcal{N}(i)$ denote, respectively, the number of its adjacent vertices or neighbors and the set of the neighboring nodes. Graphs admit a set of convenient matrix representations. For example, the $|\mathcal{V}| \times |\mathcal{E}|$ incidence matrix $E(\mathcal{G})$ for an oriented graph $\mathcal{G}$ is a $\{0, \pm 1\}$-matrix with rows and columns indexed by vertices and edges of $\mathcal{G}$, respectively such that

$$[E(\mathcal{G})]_{ik} = \begin{cases} +1 & \text{if } i \text{ is the initial node of edge } e_k \\ -1 & \text{if } i \text{ is the terminal node of edge } e_k \\ 0 & \text{otherwise.} \end{cases}$$

Another matrix representation of a graph is in terms of its adjacency matrix. The adjacency matrix of $\mathcal{G}$ is a symmetric $|\mathcal{V}| \times |\mathcal{V}|$ matrix defined as,

$$[A(\mathcal{G})]_{ij} := \begin{cases} 1 & \text{if } \{i, j\} \in \mathcal{E} \\ 0 & \text{otherwise.} \end{cases} \tag{1.5}$$

The graph Laplacian of an oriented graph is defined as

$$L(\mathcal{G}) := E(\mathcal{G})E(\mathcal{G})^T; \tag{1.6}$$

however, the graph Laplacian is independent of a particular orientation of the graph as

$$L(\mathcal{G}) = \Delta(\mathcal{G}) - A(\mathcal{G}), \tag{1.7}$$

where the diagonal matrix $\Delta(\mathcal{G})$ is composed of the degree distribution of the nodes. The graph Laplacian of $\mathcal{G}$ is a rank deficient positive semi-definite matrix. The real spectrum of $L(\mathcal{G})$ can thereby be ordered as

$$0 = \lambda_1(L(\mathcal{G})) \leq \lambda_2(L(\mathcal{G})) \leq \ldots \leq \lambda_{|\mathcal{V}|}(L(\mathcal{G})).$$

The multiplicity of the zero eigenvalue of the graph Laplacian is equal to the number of connected components of the graph [24]. Moreover, the second smallest eigenvalue of $L(\mathcal{G})$, $\lambda_2(L(\mathcal{G}))$, also known as algebraic connectivity, turns out to be a judicious measure of graph connectivity [25]. Two graphs $\mathcal{G}_1$ and $\mathcal{G}_2$ are said to be *cospectral* if the spectrum of $A(\mathcal{G}_1)$ is the same as the spectrum of $A(\mathcal{G}_2)$. Similarly, two graphs, $\mathcal{G}_1$ and $\mathcal{G}_2$, are said to be *cospectral* with respect to the graph Laplacian if the spectrum of $L(\mathcal{G}_1)$ is the same as the

spectrum of $L(\mathcal{G}_2)$. Note that two graphs that are isomorphic are always *cospectral* with respect to the graph Laplacian, but the converse is not true. That is, two graphs that are cospectral with respect to the Laplacian need not be isomorphic.

A circulant graph of order n is a graph on n vertices in which the i-th vertex is adjacent to the $(i+j)$-th and $(i-j)$-th vertices for each j in a list l. Representation of circulant graphs in terms of adjacency and Laplacian matrices lead to circulant matrices. A circulant matrix C is a Toeplitz matrix of the form,

$$C = \begin{bmatrix} c_0 & c_1 & c_2 & c_3 & \dots & c_{n-1} \\ c_{n-1} & c_0 & c_1 & c_2 & & \vdots \\ c_{n-2} & c_{n-1} & c_0 & c_1 & \ddots & \\ \vdots & \ddots & \ddots & \ddots & & c_2 \\ & & & & & c_1 \\ c_1 & & \dots & & c_{n-1} & c_0 \end{bmatrix}. \tag{1.8}$$

This structure has a direct implication on the spectral properties of circulant matrices.

Consider the weighted consensus protocol

$$\dot{x_i}(t) = \sum_{\{i,j\} \in \mathcal{E}} w_{ij}(x_j(t) - x_i(t)),$$

adopted by n-nodes, where x_i is the state of the i-th node, e.g., its position, speed, heading, voltage, etc., evolving according to the weighted sum of the differences between the i-th node's state and its neighbors; w_{ij} represents the weight on the edge $\{i,j\}$ which for unweighted graphs is set to identity. Next, let a group of agents $\mathcal{I} \subset \mathcal{V}$ with cardinality $|\mathcal{I}| = r_{\mathcal{I}}$, "excite" the underlying coordination protocol by injecting signals to the network, with another set of agents $\mathcal{O} \in \mathcal{V}$, of cardinality $|\mathcal{O}| = r_{\mathcal{O}}$, measuring the corresponding network response. Hence, the original consensus protocol from node i's perspective assumes the form

$$\dot{x_i}(t) = \sum_{\{i,j\} \in \mathcal{E}} w_{ij}(x_j(t) - x_i(t)) + B_i u_i(t),$$

where $B_i = \beta_i \neq 0$ if $i \in \mathcal{I}$, and zero otherwise. Without loss of generality, we can always assume that $\beta_i = 1$ and modify the control signal $u_i(t)$ as $\beta_i u_i(t)$ if necessary. Adding state

observation ports to this "steered" consensus, and having $y_j(t) = x_j(t)$ when $j \in \mathcal{O}$, we arrive at the compact form of an input-output linear time-invariant system,

$$\dot{x}(t) = -L_w(\mathcal{G})x(t) + Bu(t), \quad y(t) = Cx(t), \tag{1.9}$$

where $-L_w(\mathcal{G}) \in \mathbf{R}^{n\times n}$ is the weighted Laplacian, $B \in \mathbf{R}^{n\times r_\mathcal{I}}$, and $C \in \mathbf{R}^{r_\mathcal{O}\times n}$.

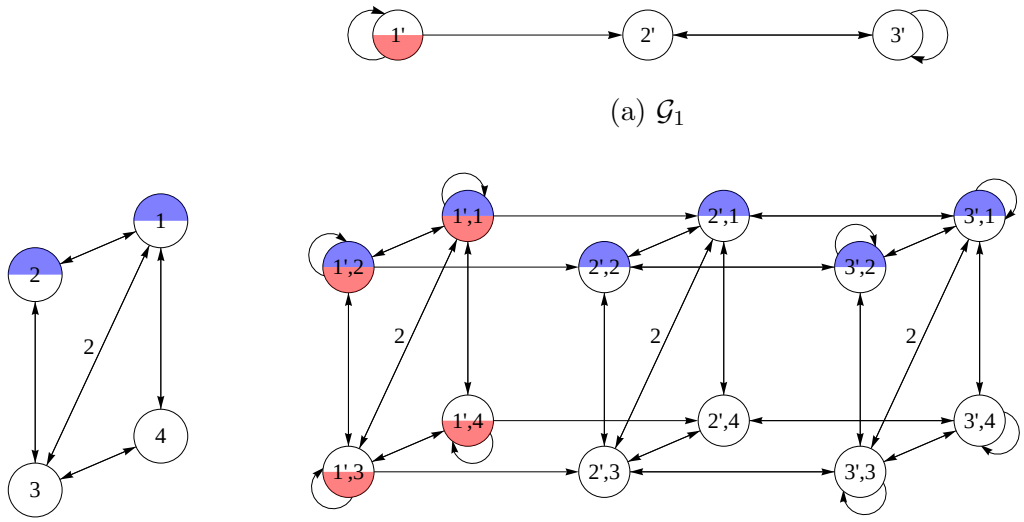
The sample space of all random graphs on n vertices is denoted by $\mathcal{G}(n,p)$ while the existence of a pair of vertices in the set $\mathcal{V}$ is determined randomly with probability $p \in (0,1]$ [15, 16], and [17]. The operators $\mathbb{E}(w)$ and $\mathbb{P}\{w = \bar{w}\}$ refer to the expected value (ensemble average) of the random variable w and the probability that the random variable w is equal to $\bar{w}$, respectively. The notation $M(w_t)$ denotes that matrix M is a function of a random variable w_t which is written as M_t for the simplicity of notation.

We define the vectors $\mathbf{1} := [1,1,\ldots,1]^T$ and $\mathbf{0} := [0,0,\ldots,0]^T$. For column vector $v \in \mathbb{R}^p$, both v_i and $[v]_i$ denote its ith element. For matrix $M \in \mathbb{R}^{p\times q}$, $[M]_{ij}$ denotes the element in its ith row and jth column. The $n \times n$ identity matrix is denoted I_n, and e_i is the column vector with all zero entries except $[e_i]_i = 1$. For $M, N \in \mathbb{R}^{n\times n}$, we write $M \succeq 0$ if M is a positive semidefinite matrix and $M \succeq N$ if $M - N \succeq 0$. The smallest and largest eigenvalues of M are denoted by $\lambda_{min}(M)$ and $\lambda_{max}(M)$, respectively. The cardinality of a set Z is denoted by $|Z|$.

1.2.3 Cartesian Product

There are many effective methods via which large-scale graphs (networks) can be synthesized from a set of smaller graphs [185]. The Cartesian product is one such method and is defined for a pair of *factor* graphs $\mathcal{G}_1 = (V_1, E_1, W_1)$ and $\mathcal{G}_2 = (V_2, E_2, W_2)$ and is denoted by $\mathcal{G} = \mathcal{G}_1 \square \mathcal{G}_2$. The product graph $\mathcal{G}$ has the vertex set $V_1 \times V_2$, and there is an edge from vertex (i,p) to (j,q) in $V_1 \times V_2$ if and only if either $i = j$ and (p,q) is an edge of E_2, or $p = q$ and (i,j) is an edge of E_1. The corresponding weight if an edge exists is $w_{((j,q),(i,p))} = w_{ji}^{\delta_{pq}} + w_{qp}^{\delta_{ij}}$ where $\delta_{ij} = 1$ if $i = j$ and 0 otherwise. An example of a Cartesian product of two factor graphs is displayed in Figure 8.1.

The Cartesian product is commutative and associative, i.e., the products $\mathcal{G}_1 \square \mathcal{G}_2$ and $\mathcal{G}_2 \square \mathcal{G}_1$ are isomorphic; similarly $(\mathcal{G}_1 \square \mathcal{G}_2) \square \mathcal{G}_3$ and $\mathcal{G}_1 \square (\mathcal{G}_2 \square \mathcal{G}_3)$ are isomorphic.



(a) $\mathcal{G}_1$

(b) $\mathcal{G}_2$ (c) $\mathcal{G} = \mathcal{G}_1 \square \mathcal{G}_2$

Figure 1.3: Factor graphs $\mathcal{G}_1$ and $\mathcal{G}_2$ and composite graph $\mathcal{G}_1 \square \mathcal{G}_2$. Edge weights of all graphs are 1 unless otherwise marked. The shading on the nodes pertains to Example 8.2.2 and 8.4.2.

A graph is called *prime* if it cannot be decomposed into the product of non-trivial graphs, otherwise a graph is referred to as composite. All graphs have a *prime factor decomposition* of the form $\mathcal{G}_1^{k_1} \square \cdots \square \mathcal{G}_m^{k_m}$, where $\mathcal{G}_i$ is prime for all i, and $\mathcal{G}_i^{k_i}$ denotes k_i Cartesian products of $\mathcal{G}_i$. Sabidussi [186] and Vizing [187] highlighted the fundamental nature of the primes noting that connected graphs decompose uniquely into primes, up to reordering. Further, Feigenbaum [188] demonstrated that a digraph can be factored into its primes in polynomial-time. Hellmuth et al. in [189] presented an algorithm to recognize original factors of approximated products. The problem of approximating graph products is considered with a wide range of applications in biology [189]. The extension of the current work to large networks approximated by Cartesian products of prime factors we leave for future work.

Many features of the composite graph's factors transfer to the composite graph itself. One such example is that if factors $\mathcal{G}_1$ and $\mathcal{G}_2$ are strongly connected then so too is $\mathcal{G}_1 \square \mathcal{G}_2$. In this section, we show that when the composite graph underlies a dynamic system many useful features of the dynamics can be revealed by examining related dynamical systems

over the factor graphs.

1.2.4 Kronecker Product

Kronecker algebra is one of the main tools in this section. The key results are presented here, and we encourage the readers to see [123] for a more rigorous survey.

Let matrix $M = [m_{ij}] \in \mathbb{R}^{r\times s}$ and $N \in \mathbb{R}^{p\times q}$. The Kronecker product $M \otimes N$ of M and N can be formed by replacing the ijth entry of M by the matrix $m_{ij}N$, for every i and j. Hence,

$$M \otimes N = \begin{bmatrix} m_{11}N & \cdots & m_{1s}N \\ \vdots & & \vdots \\ m_{r1}N & \cdots & m_{rs}N \end{bmatrix} \in \mathbb{R}^{rp\times sq}.$$

The Kronecker products $M \otimes N$ and $N \otimes M$ are permutation equivalent, i.e., there exist permutation matrices Π and Υ such that

$$M \otimes N = \Pi\,(N \otimes M)\,\Upsilon,$$

where if M and N are square, $\Pi = \Upsilon^T$. Further, the Kronecker product exhibits the mixed-product property,

$$(M \otimes N)(R \otimes S) = MR \otimes NS$$

where $M \in \mathbb{R}^{r\times s}$, $N \in \mathbb{R}^{p\times q}$, $R \in \mathbb{R}^{s\times k}$, and $S \in \mathbb{R}^{q\times l}$.

Part I

CAN ONE HEAR THE SHAPE OF COORDINATION?

Chapter 2

NETWORK IDENTIFICATION VIA NODE KNOCK-OUT

In this chapter, we examine the problem of identifying the interaction geometry among a known number of agents, adopting a consensus-type algorithm for their coordination. The proposed identification process is facilitated by introducing "ports" for stimulating a subset of network vertices via an appropriately defined interface and observing the network's response at another set of vertices.

2.1 Problem Formulation

Consider the weighted consensus protocol described in (2.1) as

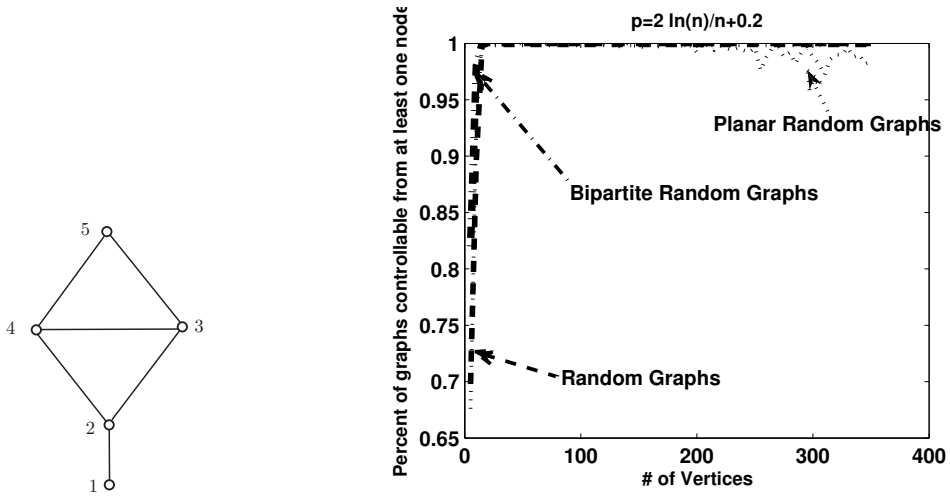
$$\dot{x}(t) = A(\mathcal{G})x(t) + Bu(t), \quad y(t) = Cx(t), \tag{2.1}$$

where $A(\mathcal{G}) = -L_w(\mathcal{G}) \in \mathbf{R}^{n\times n}$, $B \in \mathbf{R}^{n\times r_{\mathcal{I}}}$, and $C \in \mathbf{R}^{r_{\mathcal{O}}\times n}$.

Example 2.1.1. *Consider the network in Fig. 2.1; set* $\mathcal{I} = \{1, 2\}$, $\mathcal{O} = \{1, 4\}$, *and the corresponding weights equal to one on every edge. Then,*

$$A(\mathcal{G}) = -\begin{bmatrix} 1 & -1 & 0 & 0 & 0 \\ -1 & 3 & -1 & -1 & 0 \\ 0 & -1 & 3 & -1 & -1 \\ 0 & -1 & -1 & 3 & -1 \\ 0 & 0 & -1 & -1 & 2 \end{bmatrix}, \; B = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, \; C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}. \tag{2.2}$$

Even though in general, sets $\mathcal{I}$ and $\mathcal{O}$ can be distinct and contain more than one element, for the convenience of our presentation, we will assume that they are identical- and at times, we will assume that the resulting input-output system is in fact SISO. The extension of the presented results to the case when $\mathcal{I}$ and $\mathcal{O}$ are distinct will be discussed after introducing the basic setup and approach.



(a) (b)

Figure 2.1: (a) The network in Example 2.1.1, and (b) the percentage of random planar graphs that are controllable from at least one node

.

We now pose the *inverse problem* of graph-based coordination algorithms, namely, the feasibility of identifying the spectral and structural properties of the underlying network $\mathcal{G}$ via data facilitated by the input-output ports $\mathcal{I}$ and $\mathcal{O}$. In order to implement this program we will assume that: (1) the identification procedure has knowledge of the number of agents in the network, (2) if the removal of one or two nodes disconnects the underlying graph, the input/output sets $\mathcal{I}$ and $\mathcal{O}$ have non-trivial intersections with each of the resulting connected components,[1] and (3) the input/output sets $\mathcal{I}$ and $\mathcal{O}$ have been chosen such that the system described in (2.1) is controllable and observable. Although the first assumption is reasonable and the second an artifact of our approach,[2] the last assumption requires more justification which we now provide. In the trivial case when $\mathcal{I} = \mathcal{V}$ and B is equal to the identity

[1]Thus, for example, when the graph is 3-connected, the input/output sets can be chosen arbitrary to satisfy this connection.

[2]The procedure, knowing the number of nodes in the network, can identify when in fact the graph is disconnected.

matrix, the input-output weighted consensus (2.1) is clearly controllable, and by duality, observable. However, more generally, the controllability/observability of the network from a subset of its boundary nodes, is less trivial, and more to the point, not guaranteed for general graphs [105]. In the meantime, since we will need controllability and observability of the network for its identifiability, we will rely on an intriguing topical conjecture in algebraic graph theory, to the effect that for large values of n, the ratio of graphs with n nodes that are not controllable from any single node to the total number of graphs on n nodes approaches zero as $n \to \infty$ [100]. This phenomena is depicted in Fig. 2.1(b); for every node n, the percent of controllable networks from one node is calculated from 400 sample random graphs. *In the present research, we take the controllability and the observability of the underlying graph from the input and output nodes as our working assumption.* In the meantime, it is always convenient to know when the network is uncontrollable from a given node.

Lemma 2.1.2. *Let $P(s) = C(sI - A)^{-1}B$ be the input-output realization of (2.1). The uncontrollable/unobservable eigenvalues of (2.1) will not appear in the corresponding entry of $P(s)$. Specifically, $P(s)$ will be order $n - i$ polynomial for the SISO case with n agents and i uncontrollable/unobservable eigenvalues.*

Proof. Since the underlining graph is undirected, the matrix $A(\mathcal{G})$ in (2.1) is symmetric, and there exists a unitary matrix U and a real nonnegative diagonal matrix $\Lambda = \mathbf{diag}(\lambda_1, \ldots, \lambda_n)$ such that $A(\mathcal{G}) = U\Lambda U^T$. In this case, the columns of U are an orthonormal set of eigenvectors for $A(\mathcal{G})$ and the corresponding diagonal entries of Λ are its eigenvalues. Therefore,

$$P(s) = C(sI - A(\mathcal{G}))^{-1}B = C(sI - U\Lambda U^T)^{-1}B = CU(sI - \Lambda)^{-1}U^T B. \tag{2.3}$$

From the PBH test, if the system (2.1) is not controllable, there is an eigenvector that is orthogonal to B. Therefore, for an arbitrary uncontrollable eigenvalue λ_i, the i-th row of U^T is orthogonal to B, and λ_i will not appear in $(sI - \Lambda)^{-1}U^T B$. An analogous argument works for the unobservable case as well. □

2.2 System Identification

We now consider various standard system identification procedures in the context of identifying the spectra of the underlying graph Laplacian, and subsequently, gaining insights into the interconnection structure that underscores the agents' coordinated behavior.

System identification methods are implemented via sampling of the system (2.1) at discrete time instances[3] $\delta, 2\delta, \ldots, k\delta, \ldots$, with $\delta > 0$, so that the assumes the form

$$z(k+1) = A_d z(k) + B_d v(k), \quad w(k) = C_d z(k), \tag{2.4}$$

where $z(k) = x(k\delta)$, $v(k) = u(k\delta)$, $w(k) = y(k\delta)$, $A_d = e^{\delta A}$, $B_d = \left(\int_0^\delta e^{At} dt\right) B$, and $C_d = C$.[4] In fact, the system identification process leads to a realization of the model

$$\widetilde{z}(k+1) = \widetilde{A}_d \widetilde{z}(k) + \widetilde{B}_d v(k), \quad \widetilde{w}(k) = \widetilde{C}_d \widetilde{z}(k), \tag{2.5}$$

where $(\widetilde{A}_d, \widetilde{B}_d, \widetilde{C}_d)$ is the realization of (A_d, B_d, C_d) in (2.4). The identified system (2.5), on the other hand, corresponds to the continuous-time system

$$\dot{\widetilde{x}}(t) = \widetilde{A}\widetilde{x}(t) + \widetilde{B}u(k), \quad y(t) = \widetilde{C}\widetilde{x}(t), \tag{2.6}$$

with $\widetilde{A}_d = e^{\delta \widetilde{A}}$, $\widetilde{B}_d = \left(\int_0^\delta e^{\widetilde{A}t} dt\right) \widetilde{B}$, and $\widetilde{C}_d = \widetilde{C}$; in this case, $\widetilde{A} = (1/\delta) \log_M \widetilde{A}_d$ where $\log_M$ denotes the matrix logarithm. Since the system (2.5) is a realization of the system (2.4), it follows that the estimated triplet $(\widetilde{A}, \widetilde{B}, \widetilde{C})$ is a realization of (A, B, C) in (2.1). As a result, there exists a similarity transformation induced by the matrix T, such that $\widetilde{A} = TAT^{-1}$, $\widetilde{B} = TB$, and $\widetilde{C} = CT^{-1}$. In fact, in the controllable/observable case, the eigenvalues of $\widetilde{A}_d$ are precisely matched with the eigenvalues of A_d. Obtaining a zero as eigenvalue of $\widetilde{A}_d$, which is equivalent of obtaining $-\infty$ as the eigenvalue of $\widetilde{A}$, is a sign of an uncontrollable and/or unobservable mode in (2.1).[5] For example in the identification procedure called *Iterative Prediction-Error Minimization Method*, the model (2.4) for every input v_i and output w_j

[3]The system identification methods work based on data sampling from the system. Since we aimed to identify the interaction geometry of the network, we originally considered a continuous system. Therefore, we need to discretize the system (2.1).

[4]The notation e^A for a square matrix A refers to its matrix exponential.

[5]This follows from Lemma 2.1.2 since $-\infty$ will appear as zero in the corresponding entries of $P(s)$.

can be represented as $\mathbf{A}(q)w_j(k) = \mathbf{B}(q)v_i(k)$, where $\mathbf{A}(q) = 1 + a_1q^{-1} + \cdots + a_nq^{-n}$ and $\mathbf{B}(q) = b_1q^{-1} + \cdots + b_{r_\mathcal{I}}q^{-r_\mathcal{I}}$. The unknown model parameters $\theta = [a_1, \ldots, a_n, b_1, \ldots, b_{r_\mathcal{I}}]$ can then be estimated by comparing the actual output $w_j(k)$ with the predicted output $\widetilde{w}_{ji}(k|k-1)$ using the mean-square minimization. In this case, the output predictor is constructed as $\widetilde{w}_{ji}(k|k-1) = [-w_j(k-n), \ldots, -w_j(1), v_i(k-r_\mathcal{I}), \ldots, v_i(1)]$. In yet another candidate system identification procedure, namely the *Subspace Identification Method*, the system (2.4) is approximated by another system in the form (2.5) using a state trajectory of the dynamic system that has been determined from input-output observations. The Hankel matrix, which can be constructed from the gathered input-output data, plays an important role in this method. By constructing the Hankel matrix, the discrete time system matrices $\widetilde{A}_d$, $\widetilde{B}_d$, and $\widetilde{C}_d$ can then be determined. Subsequently, the continuous-time estimated matrices $\widetilde{A}$, $\widetilde{B}$, and $\widetilde{C}$ can be identified; see [28] for an extensive treatment of system identification methods.

In summary, an identification procedure such as the above two methods, implemented on a controllable and observable steered-and-observed coordination protocol (2.1), leads to a system realization whose state matrix is similar to the underlying graph Laplacian and in particular shares the same spectra and characteristic polynomial. However, a distinct and fundamental issue in our setup is that having found a matrix that is "similar" to the Laplacian of a network is far from having exact knowledge of the network structure itself [219]. This observation motivates the following question: *to what extent does the knowledge of the spectra of the graph, combined with the knowledge of the input-output matrices, reduce the search space for the underlying interaction geometry?* In this chapter, we explore this question using techniques based on integer partitioning and degree-based graph reconstruction.

Inspired by how biologists use gene knockouts for experimentally identifying genetic interaction networks in cellular organisms, we propose a node-knockout procedure for the complete characterization of the interaction geometry in consensus-type networks. In our context, the node knockout is essentially a grounding procedure– where the node broadcasts a zero state to its neighbors without being removed from the network. The proposed identification process is also facilitated by introducing "ports" for stimulating a subset of

network vertices via an appropriately defined interface and observing the network's response at another set of vertices. We then provide an example for the utility of such a network identification process in the context of fault detection for networked systems.

2.3 Characterization of the Network Topology via Node Knockout

We now explore means by which a system identification procedure as discussed in §2.2 can be used for the exact characterization of the interaction topology for consensus-type networks. Consider again the input-output LTI model (2.1). For notational simplicity, we consider unweighted Laplacian matrices; the extension of this work to the weighted Laplacian is straightforward from our analysis. In this section, we explore a method for characterizing the network by resorting to "grounding" the graph at a vertex. First, recall that via one of the system identification methods discussed previously, the characteristic equation of the system (2.1) can be found to be

$$\phi_{\mathcal{G}}(s) = \mathbf{det}(sI - A(\mathcal{G})) = \mathbf{det}(sI + L(\mathcal{G})). \tag{2.7}$$

Definition 2.3.1. *The grounded consensus at node v evolves according the dynamics*

$$\dot{x}(t) = -L_v(\mathcal{G})x(t),$$

where $L_v(\mathcal{G}) = L(\mathcal{G}\backslash v) + \Delta_v$, $\mathcal{G}\backslash v$ is the graph obtained after removing v from $\mathcal{G}$, and the diagonal matrix Δ_v is such that $[\Delta_v]_{ii} = 1$ (or $[\Delta_v]_{ii} = w_{iv}$ in the weighted consensus protocol) when $\{v, i\} \in \mathcal{E}$ and 0 otherwise. We refer to the grounding operation as node knockout.

In this section, we make the standing assumption that each vertex in $\mathcal{G}$ can be instructed to "ground" itself upon request. Now, define the polynomial

$$\phi^v_{\mathcal{G}}(s) := \mathbf{det}(sI + (L(\mathcal{G}\backslash v) + \Delta_v)) \tag{2.8}$$

which is the characteristic polynomial of the grounded consensus at node v. Following our presentation in §2.2, it then follows that the system identification procedure can be applied for the cases when one or two nodes in the graph are grounded. And in fact, provided that

the grounded consensus protocol is controllable and observable, the characteristic polynomials $\phi^v_{\mathcal{G}}(s)$ and $\phi^{uv}_{\mathcal{G}}(s)$ for situations where either vertex v has been grounded or when both vertices v and u have been grounded can be obtained. But before we proceed, let us address the controllability/observability of the influenced/observed grounded consensus.

Proposition 2.3.2. *Consider the controllable and observable steered-and-observed system (2.1) on the n-node graph $\mathcal{G}$ with $n \geq 2$. Then as long as none of the input-output vertices are identical to the grounded node(s), the grounded consensus on $L_v(\mathcal{G})$ remains controllable and observable with the corresponding reduced input and observation matrices if and only if the graph $\mathcal{G}\backslash v$ stays connected.*

Proof. Let the grounded node v be different from the input-output nodes. Without loss of generality, assume that v is the last indexed node in $L(\mathcal{G})$ and also that the input set contains just one node. In this case, since the input set does not include v, we can rewrite B as $B = [\,\hat{B}^T, 0\,]^T$. From the definition of controllability, if the grounded consensus is controllable with the pair $(L_v(G), \hat{B})$, the graph $\mathcal{G}\backslash v$ is connected. The next step is to prove that if the graph $\mathcal{G}\backslash v$ is connected, the grounded consensus is controllable, or equivalently, that if the grounded consensus is uncontrollable, the graph $\mathcal{G}\backslash v$ has to be disconnected. This is proven as follows. Since the original graph is controllable, from the PBH test there does not exist a nonzero z and λ such that $L(\mathcal{G})z = \lambda z$ and $z^T B = 0$. Thus after partitioning the matrix $L(\mathcal{G})$, one has

$$\nexists\, z \neq 0,\ \lambda \quad \text{s.t.} \quad \begin{bmatrix} L(\mathcal{G}\backslash v) + \Delta_v - \lambda I & \delta_v \\ \delta_v^T & \mathbf{deg}\ v - \lambda \\ \hat{B}^T & 0 \end{bmatrix} z = 0, \tag{2.9}$$

where I is the identity matrix with proper dimensions, and δ_v is the vector formed from the diagonal of Δ_v. Since the grounded consensus is uncontrollable, there exists a nonzero $s \in \mathbf{R}^{n-1}$ and $\hat{\lambda} \in \mathbf{R}$ such that $(L(\mathcal{G}\backslash v) + \Delta_v)s = \hat{\lambda}s$ and $s^T\hat{B} = 0$ with $\hat{\lambda} \neq 0$ since otherwise the proof is done. Therefore,

$$\begin{bmatrix} L(\mathcal{G}\backslash v) + \Delta_v - \hat{\lambda} I \\ \hat{B}^T \end{bmatrix} s = 0, \tag{2.10}$$

where I is the identity matrix with proper dimensions. Note that in order to show that $\mathcal{G}\backslash v$ is disconnected, it suffices to show that there exists a vector $r \notin \mathbf{span}\{\mathbf{1}\} \subseteq \mathbf{R}^{n-1}$ such that $L(\mathcal{G}\backslash v)r = 0$. The matrix on the left hand-side of (2.9) is full rank for all values of λ. Thus for any choice of $q \in \mathbf{R}^n$, there exists $p \in \mathbf{R}^{n+1}$ such that

$$\begin{bmatrix} L(\mathcal{G}\backslash v) + \Delta_v - \lambda I & \delta_v & \hat{B} \\ \delta_v^T & \mathbf{deg}\ v - \lambda & 0 \end{bmatrix} p = q. \tag{2.11}$$

By partitioning $p = [p_1^T,\ p_2^T,\ p_3^T]^T$ and $q = [q_1^T,\ q_2^T]^T$, we obtain

$$(L(\mathcal{G}\backslash v) + \Delta_v - \lambda I)p_1 + \delta_v p_2 + \hat{B}^T p_3 = q_1, \quad \text{and} \quad \delta_v^T p_1 + (\mathbf{deg}\ v - \lambda)p_2 = q_2. \tag{2.12}$$

Choosing $\lambda = \hat{\lambda},\ q_1 = \delta_v$, and multiplying both sides of the first identity in (2.12) by s, we obtain $p_2 = 1$. Let us choose $q_2 = \mathbf{deg}\ v - \hat{\lambda} + \alpha_1$ in (2.12) where $\alpha_1 \neq 0$. Therefore, $\delta_v^T p_1 = \alpha_1$ or $\sum_{i \in \mathcal{N}(v)} p_1(i) = \alpha_1$. In view of (2.12), we conclude that $L(\mathcal{G}\backslash v)p_1 = (\hat{\lambda}I - \Delta_v)p_1 - \hat{B}p_3$. If $(\hat{\lambda}I - \Delta_v)p_1 - \hat{B}p_3 = 0$, choosing $r = p_1$ will prove the claim if $p_1 \notin \mathbf{span}\{\mathbf{1}\}$. By choosing $p_1 = [0,\ 0,\ \ldots, \alpha_1]^T$, it can be verified that $(\hat{\lambda}I - \Delta_v)p_1 - \hat{B}p_3 = 0$ when $p_3 = \frac{\alpha_1(\hat{\lambda}-1)}{\sum_{i \in \mathcal{N}(v)} \hat{B}(i)}$. □

When the candidate grounding vertex belongs to the input set $\mathcal{I}$ in the original LTI system (2.1), the controllability and observability of the grounded consensus should be preserved by switching the control/observe channel to another vertex in the grounded consensus. The same argument holds valid when the graph $\mathcal{G}\backslash v$ is disconnected, in which case the grounded consensus looses its controllability and observability. We also note that the controllability/observability of the grounded network at two vertices follows from the same argument in the Proposition 2.3.2 and is thereby omitted. We are ready to state the main result of this section.

Theorem 2.3.3. *System identification on the steered-and-observed consensus (2.1), while allowing the grounding operation (Definition 2.3.1) at all nodes and all pairs of nodes, allows for a complete characterization of the underlying network.*

We will prove this theorem via a number of observations– and most importantly– with the help of a powerful construct in combinatorics, namely that of generating functions [9].

Definition 2.3.4. *Let $a_0, a_1, \ldots$ be a finite or infinite sequence of real numbers. Then the ordinary generating function $\chi(s)$ of the sequence is the power series*

$$\chi(s) = a_0 + a_1 s + a_2 s^2 + \cdots = \sum_{k=0}^{\infty} a_k s^k. \tag{2.13}$$

On one level, generating functions can be regarded as algebraic objects whose formal manipulation allows one to address combinatorial problems by means of algebra [8]. Yet on another level, generating functions can be considered as power series expansions of infinitely differentiable functions. Generating functions can conveniently be extended to matrices and in particular to graph Laplacians. In this venue, define the generating function $\chi_{\mathcal{G}}(s)$ with respect to the sequence of powers of the graph Laplacian as

$$\chi_{\mathcal{G}}(s) : \mathbb{C} \to \mathbb{C}^{n\times n}, \quad \chi_{\mathcal{G}}(s) := \sum_{k=0}^{\infty} s^k (-L(\mathcal{G}))^k = (I + sL(\mathcal{G}))^{-1}. \tag{2.14}$$

Generating functions based on the adjacency matrix of a graph have been studied extensively; see for example [7] (Chapter 4). In view of the resemblance of (2.14) to the transfer matrix of (2.1), our aim will be on clarifying the role of generating functions for the characterization of steered and observed consensus-type networks. In doing so, we are then able to devise a procedure for network identification for (2.1) by identifying the corresponding generating functions for the grounded and ungrounded characteristic polynomials. The ingredients of such a program are clarified by the following observations.

In order to determine the entries of the generating function of $L(\mathcal{G})$, define the matrix $\Psi_{\mathcal{G}}(s)$ as the adjugate of $sI + L(\mathcal{G})$, i.e., the complex conjugate transpose of the matrix of its cofactors. From the definition of the matrix inverse, we then have

$$\Psi_{\mathcal{G}}(s)(sI + L(\mathcal{G})) = \det(sI + L(\mathcal{G}))I = \phi_{\mathcal{G}}(s)I. \tag{2.15}$$

Lemma 2.3.5. *Let v be a vertex in the graph $\mathcal{G}$. Then*

$$s^{-1}[\chi_{\mathcal{G}}(s^{-1})]_{vv} = \frac{\phi^v_{\mathcal{G}}(s)}{\phi_{\mathcal{G}}(s)}, \tag{2.16}$$

where $[\chi_{\mathcal{G}}(s)]_{vv}$ is the v-th diagonal entry of the generating function $\chi_{\mathcal{G}}(s)$.

Proof. From (2.14), it follows that

$$s^{-1}\chi_{\mathcal{G}}(s^{-1}) = s^{-1}(I + s^{-1}L(\mathcal{G}))^{-1} = (sI + L(\mathcal{G}))^{-1} = \frac{\Psi_{\mathcal{G}}(s)}{\phi_{\mathcal{G}}(s)}; \tag{2.17}$$

the last equality in (2.17) is Cramer's rule. Then (2.16) follows immediately from (2.17); we note that $\phi^v_{\mathcal{G}}(s)$, as the v-th diagonal entry of $\Psi_{\mathcal{G}}(s)$, is the characteristic polynomials of $L(\mathcal{G}\backslash v) + \Delta_v$. On the other hand, as defined in (2.8), $\phi^v_{\mathcal{G}}(s)$ is the characteristic polynomial of the grounded consensus system matrix at node v. □

Before we can state our next result, we introduce a new notation. For $D \subseteq \mathcal{V}$, $[\chi_{\mathcal{G}}(s)]_D$ denotes the submatrix of $\chi_{\mathcal{G}}(s)$ with rows and columns indexed by the vertices in D.

Theorem 2.3.6. *(Jacobi) [8] Let D be a subset of d vertices of the graph $\mathcal{G}$. Then*

$$s^{-d}\,\mathbf{det}[\chi_{\mathcal{G}}(s^{-1})]_D = \frac{\phi^D_{\mathcal{G}}(s)}{\phi_{\mathcal{G}}(s)}; \tag{2.18}$$

note that this is an extension of (2.16).

Proof. Without loss of generality, we may assume that D consists of the first d vertices of $\mathcal{G}$. Let $\mathcal{K}$ be the matrix obtained by replacing the first d columns of the $n \times n$ identity matrix with the corresponding columns of $\Psi_{\mathcal{G}}(s)$. Consider the product $(sI + L(\mathcal{G}))\mathcal{K}$. We have

$$(sI + L(\mathcal{G}))\mathcal{K} = \begin{bmatrix} \phi_{\mathcal{G}}(s) I_d & X \\ 0 & sI + L(\mathcal{G}\backslash D) + \Delta_D \end{bmatrix},$$

where $\Delta_D = \sum_{v\in D} \Delta_v$, and the exact form of the matrix X is inconsequential. Taking the determinant of both sides of this equation yields

$$\phi_{\mathcal{G}}(s)\ \mathbf{det}\mathcal{K} = \phi_{\mathcal{G}}(s)^d\ \mathbf{det}(sI + L(\mathcal{G}\backslash D) + \Delta_D), \tag{2.19}$$

where $\mathcal{G}\backslash D$ is the graph obtained from removing the node set D from $\mathcal{G}$ (as well as all edges incident on the nodes in D). Note that $\mathbf{det}(sI + L(\mathcal{G}\backslash D) + \Delta_D) = \phi^D_{\mathcal{G}}(s)$. From (2.17) and the definition of $\mathcal{K}$, one has

$$s^{-d}\,\mathbf{det}[\chi_{\mathcal{G}}(s^{-1})]_D = \phi_{\mathcal{G}}(s)^{-d}\,\mathbf{det}\,\mathcal{K},$$

and in combination with (2.19), this yields the statement of the theorem. □

If D consists of a pair of vertices u and v, then we obtain

$$\mathbf{det}\ [\chi_{\mathcal{G}}(s^{-1})]_{D=\{u,v\}} = [\chi_{\mathcal{G}}(s^{-1})]_{uu}[\chi_{\mathcal{G}}(s^{-1})]_{vv} - [\chi_{\mathcal{G}}(s^{-1})]_{uv}[\chi_{\mathcal{G}}(s^{-1})]_{vu}. \tag{2.20}$$

Since $\mathcal{G}$ is undirected, $[\chi_{\mathcal{G}}(s^{-1})]_{uv} = [\chi_{\mathcal{G}}(s^{-1})]_{vu}$. Hence, we can determine the off-diagonal entries of the generating function of $L(\mathcal{G})$.

Lemma 2.3.7. *Let $\mathcal{G}$ be a graph and $u, v \in \mathcal{G}$. Then*

$$s^{-1}[\chi_{\mathcal{G}}(s^{-1})]_{uv} = \frac{[\Psi_{\mathcal{G}}(s)]_{uv}}{\phi_{\mathcal{G}}(s)}, \tag{2.21}$$

where $[\chi_{\mathcal{G}}(s^{-1})]_{uv}$ is the uv-entry of the generating function $\chi_{\mathcal{G}}(s)$, and

$$[\Psi_{\mathcal{G}}(s)]_{uv} = (\phi^u_{\mathcal{G}}(s)\phi^v_{\mathcal{G}}(s) - \phi_{\mathcal{G}}(s)\phi^{uv}_{\mathcal{G}}(s))^{1/2}.$$

Proof. From Theorem 2.3.6, set $D = \{u, v\}$. Thus,

$$s^{-2}\mathbf{det}[\chi_{\mathcal{G}}(s^{-1})]_{\{u,v\}} = \frac{\phi^{uv}_{\mathcal{G}}(s)}{\phi_{\mathcal{G}}(s)}. \tag{2.22}$$

From (2.20) and (2.22), it follows that

$$s^{-2}([\chi_{\mathcal{G}}(s^{-1})]_{uu}[\chi_{\mathcal{G}}(s^{-1})]_{vv} - [\chi_{\mathcal{G}}(s^{-1})]^2_{uv}) = \frac{\phi^{uv}_{\mathcal{G}}(s)}{\phi_{\mathcal{G}}(s)}.$$

From (2.16), on the other hand, it follows that

$$s^{-2}(s^2\frac{\phi^u_{\mathcal{G}}(s)}{\phi_{\mathcal{G}}(s)}\frac{\phi^v_{\mathcal{G}}(s)}{\phi_{\mathcal{G}}(s)} - [\chi_{\mathcal{G}}(s^{-1})]^2_{uv}) = \frac{\phi^{uv}_{\mathcal{G}}(s)}{\phi_{\mathcal{G}}(s)}.$$

Hence,

$$s^{-2}[\chi_{\mathcal{G}}(s^{-1})]^2_{uv} = \frac{\phi^u_{\mathcal{G}}(s)\phi^v_{\mathcal{G}}(s) - \phi_{\mathcal{G}}(s)\phi^{uv}_{\mathcal{G}}(s)}{\phi_{\mathcal{G}}(s)^2},$$

and

$$s^{-1}[\chi_{\mathcal{G}}(s^{-1})]_{uv} = \frac{(\phi^u_{\mathcal{G}}(s)\phi^v_{\mathcal{G}}(s) - \phi_{\mathcal{G}}(s)\phi^{uv}_{\mathcal{G}}(s))^{1/2}}{\phi_{\mathcal{G}}(s)}.$$

□

Since $s^{-1}[\chi_{\mathcal{G}}(s^{-1})]_{uv}$ is an entry of $s^{-1}(I + s^{-1}L(\mathcal{G}))^{-1} = (sI + L)^{-1}$, it is a rational function with $\phi_{\mathcal{G}}(s)$ as its denominator and the numerator is the $uv-$entry of $\Psi_{\mathcal{G}}(s)$.

In order to draw a direct connection between the generating function (2.14) and the steered-and-observed consensus protocol (2.1), note that $(sI+L(\mathcal{G}))^{-1}$ is defined as $s^{-1}\chi_{\mathcal{G}}(s^{-1})$. And rather conveniently, according to Lemmas 2.3.5 and 2.3.7, all entries of $s^{-1}\chi_{\mathcal{G}}(s^{-1})$ can be determined by the characteristic polynomials of the ungrounded and grounded consensus through the system identification procedure. In fact, the impulse response of $(sI + L(\mathcal{G}))^{-1}$

is the state transition matrix of the consensus system and can be calculated from $s^{-1}\chi_{\mathcal{G}}(s^{-1})$ as,[6]

$$\mathcal{L}^{-1}\{(sI + L(\mathcal{G}))^{-1}\} = e^{-L(\mathcal{G})t} = \mathcal{L}^{-1}\{s^{-1}\chi_{\mathcal{G}}(s^{-1})\}, \tag{2.23}$$

where $\mathcal{L}^{-1}$ denotes the inverse Laplace transform. Hence, the graph Laplacian $L(\mathcal{G})$ can be uniquely identified by running the system identification on the ungrounded and grounded systems corresponding to (2.1).

The next example demonstrates the proposed procedure for the network in Fig. 2.2(a). Notice that if we run system identification with the input and output set to $\mathcal{I} = \mathcal{O} = \{1\}$, we obtain the characteristic polynomial $\phi_{\mathcal{G}}(s) = s^4 + 6s^3 + 10s^2 + 4s$. In the meantime, by running the system identification on the grounded consensus at node 1 while choosing $\mathcal{I} = \mathcal{O} = \{4\}$, we obtain the characteristic polynomial $\phi_{\mathcal{G}}^{\{1\}}(s) = s^3 + 5s^2 + 6s + 1$. Notice that in this case

$$A(\mathcal{G}) = \begin{bmatrix} -2 & 1 & 0 \\ 1 & -2 & 1 \\ 0 & 1 & -1 \end{bmatrix}.$$

In the same manner $\phi_{\mathcal{G}}^{\{1\}}(s) = \phi_{\mathcal{G}}^{\{2\}}(s)$. As we ground node 2, two disconnected components will be generated, and the input and output nodes must be chosen from these distinct components of the graph. We therefore determine $\phi_{\mathcal{G}}^{\{2\}}(s)$ and $\phi_{\mathcal{G}}^{\{3\}}(s)$ as $\phi_{\mathcal{G}}^{\{2\}}(s) = s^3 + 4s^2 + 4s + 1$ and $\phi_{\mathcal{G}}^{\{3\}}(s) = \phi_{\mathcal{G}}^{\{2\}}(s)$. In the next step, in order to calculate the off-diagonal entries of $s^{-1}\chi_{\mathcal{G}}(s^{-1})$, one needs to ground a pair of nodes simultaneously. For example, if we run the system identification procedure on the consensus system grounded at nodes $\{1, 2\}$ and choose $\mathcal{I} = \mathcal{O} = \{3\}$, we obtain $\phi_{\mathcal{G}}^{\{1,2\}}(s) = s^2 + 3s + 1$. Notice that the system matrix in this case is

$$A(\mathcal{G}) = \begin{bmatrix} -2 & 1 \\ 1 & -1 \end{bmatrix}.$$

Considering the symmetry in the generating matrix $\chi_{\mathcal{G}}(s)$, by running multiple sessions of the system identification procedure, we obtain all entries of $s^{-1}\chi_{\mathcal{G}}(s^{-1})$. Thus, the Laplacian matrix, $L(\mathcal{G})$, can be explicitly found from (2.23).

---

[6]Incidentally we should mention that the impulse response of the system with the transfer matrix function as (2.14), is the state transition matrix for the Markov chain associated with the consensus protocol [105] (Chapter 3).

2.4 Edge Faults in the Network

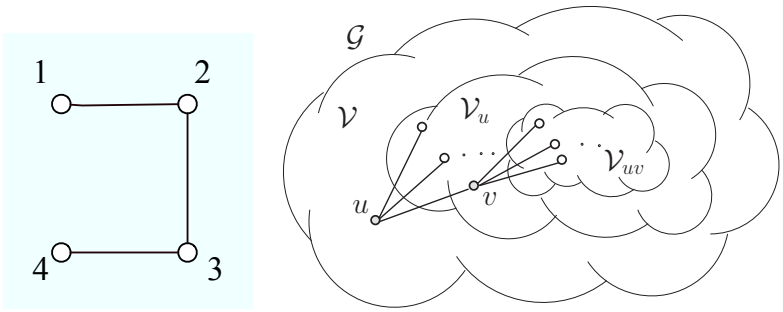
In this section, we explore the utility of the network identification procedure in the context of fault detection for networked systems. Consider a remote administrator of a network, running a protocol similar to (2.1),[7] monitoring the network's behavior by occasionally sending signals through certain boundary nodes and observing its reflection. As we discussed in previous sections, assuming the observability/controllability of the underlying network, the identification process provides us with the characteristic equation for the system and the number of edges in the network. Thus, if there is an edge failure, the characteristic equation of the modified network will reflect this failure. We would like to explore now the possibility of *identifying the broken link* from running the identification procedure on the network when the grounding operation is permissible. It is assumed that the nodes can be instructed to ground themselves, i.e., sending a zero state value to their neighbors, while not being able to independently and locally determine whether an edge in the network has been broken. In fact it is assumed that the administrator can only detect faults indirectly by observing and monitoring a few selected signals from the network. In this venue, let $E = 2|\mathcal{E}(\mathcal{G})|$ be twice the number of edges in the fault-free network $\mathcal{G}$, which is equal to the sum of the roots of the characteristic polynomial $\phi_{\mathcal{G}}(s)$. In the same vein, let us denote the sum of the roots of the characteristic equations of the grounded consensus matrices $L(\mathcal{G}_u)$ and $L(\mathcal{G}_{uv})$ as E_u and E_{uv}, respectively.

Proposition 2.4.1. *Consider the consensus protocol (2.1) over the graph $\mathcal{G}$. Let E, E_u, E_v, and E_{uv} denote, respectively, the sum of the roots of the characteristic polynomials $\phi_{\mathcal{G}}(s)$, $\phi^u_{\mathcal{G}}(s)$, $\phi^v_{\mathcal{G}}(s)$, and $\phi^{uv}_{\mathcal{G}}(s)$. Then $E - E_u - E_v + E_{uv} = 0$ if there is no edge between nodes u and v, while $E - E_u - E_v + E_{uv} = 2$ indicates that there is an edge between u and v.*

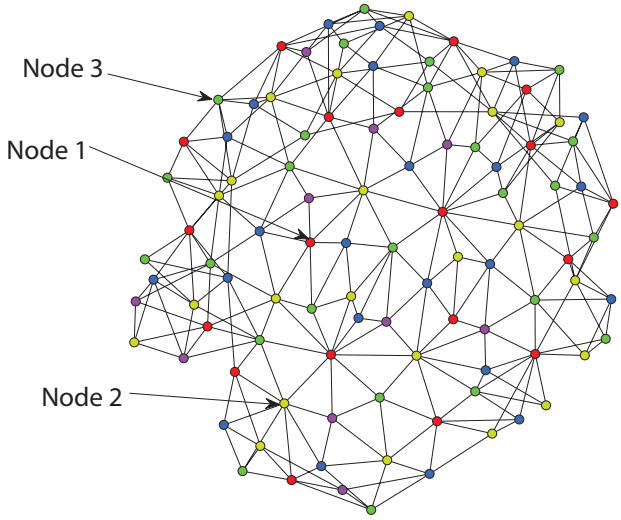
Proof. Let $\mathcal{V}_u$ denote the subset of $\mathcal{V}$ with node u excluded; see Fig. 2.2. By running the system identification on $\mathcal{G}$, we obtain the sum of the roots of the characteristic polynomial $\phi_{\mathcal{G}}(s)$ which is also equal to the sum of the degrees of all nodes in $\mathcal{V}$. In the meantime, by running the system identification on the grounded consensus at u, we obtain $E_u =$

[7] Including any class of consensus-type protocols for formation control and distributed estimation [105].

$E - \mathbf{deg}\ u$. Analogously, one obtains $E_v = E - \mathbf{deg}\ v$ when node v is grounded. Thus by



(a) (b)



(c)

Figure 2.2: (a) Graph $\mathcal{G}$, (b) grounding the consensus protocol, and (c) the graph considered for the example.

grounding the pair of nodes u and v in the consensus protocol, the sum of the roots of the characteristic polynomial $\phi_{\mathcal{G}}^{uv}(s)$ is $E_{uv} = E - \mathbf{deg}\ u - \mathbf{deg}\ v + 2I_{\{u,v\}\in\mathcal{E}}$, where $I_{\{u,v\}\in\mathcal{E}}$ is equal to one if nodes u and v are incident and zero otherwise. Thereby, $E_u + E_v - E_{uv} = E - 2I_{\{u,v\}\in\mathcal{E}}$ and the statement of the proposition now follows. □

As an example, consider the graph in Fig. 2.2 with $|\mathcal{V}| = 100$ nodes and $|\mathcal{E}| = 284$,

running the consensus protocol. Our goal is to resolve whether there is an edge between nodes 1 and 2 via the proposed network identification procedure. We note that if we run the system identification with $r_{\mathcal{I}} = r_{\mathcal{O}} = 50$, we obtain the number of edges in the network, E, as the sum of the roots of the characteristic polynomial $\phi_{\mathcal{G}}(s)$; in this case there are 568 edges in the network. Applying the system identification procedure on the grounded consensus at node 1 leads the value of $E_1 = 562$, where E_1 is the sum of the roots of the characteristic polynomial $\phi^1_{\mathcal{G}}(s)$. Analogously, the values of E_2 and $E_{1,2}$ are 560 and 554, respectively. Since $E = E_1 + E_2 - E_{1,2}$, from Proposition 2.4.1, it follows that there is no edge between nodes 1 and 2.

Chapter 3

A SIEVE METHOD FOR CONSENSUS-TYPE NETWORK TOMOGRAPHY

In the next approach, it is first noted that under the assumption of controllability and observability of the corresponding steered-and-observed network, the proposed procedure identifies a number of important features of the network using the spectrum of the graph Laplacian. We then proceed to use degree-based graph reconstruction methods to propose a sieve method for further characterization of the underlying network. Note that in this approach the weights in weighted-consensus have to be identical. An example demonstrates the application of the proposed method.

We first review qualitative characterization of the underlying interconnection topology via its identified characteristic polynomial. We then explore the possibility of reducing the search space for the underlying network via the proposed *sieve method*.

3.1 Graph Characterization via Characteristic Polynomial

Recall that via a system identification method, the characteristic equation of the system (1.9) can be found as

$$\phi_{\mathcal{G}}(s) \quad = \quad \det(sI - A(\mathcal{G})) = s^n + a_1 s^{n-1} + ... + a_{n-1}s + a_n. \tag{3.1}$$

Although the spectra of the graph Laplacian in general is insufficient to form an explicit characterization of the underlying network, it leads to a number of useful structural information about its geometry; we list a few:

1. the value $(1/n)\prod_{i=2}^{n}\lambda_i(\mathcal{G})$ is the number of spanning trees in $\mathcal{G}$,

2. one has $|\mathcal{E}| = 1/2\sum_{i=1}^{n}\lambda_i$, where $|\mathcal{E}|$ is the number of edges in the graph,

3. if $a_{n-1} = n$, the underlying interconnection is a tree. For a tree, the coefficient a_{n-2} is also called the graph Wiener index (the sum of all distances between distinct vertices of $\mathcal{G}$) [29],

4. if the associated graph is a tree, a_k is the number of k-matching in the subdivision of $\mathcal{G}$ (where each edge of $\mathcal{G}$ is replaced by a path of length 2),

5. if the eigenvalues of $L(\mathcal{G})$ are distinct, with $0, \lambda_2, \ldots, \lambda_r$ where $\lambda_r > \ldots > \lambda_2 > 0$, define $\psi_{\mathcal{G}}(x) = (x - \lambda_2) \ldots (x - \lambda_r)$. It is then well-known that $\psi_{\mathcal{G}^c} = (-1)^r \psi_{\mathcal{G}}(n - x)$, where the graph $\mathcal{G}^c$ is the complement of the graph $\mathcal{G}$. The Hoffman number of the graph, $\mu(\mathcal{G})$, can also be found as $\lambda_2 \lambda_3 \ldots \lambda_r / n$ whose properties and applications have been studied in [30],

6. let $\mathcal{T}$ be a tree with $n \geq 2$ vertices. If $\mathcal{T}$ has only one positive Laplacian eigenvalue with multiplicity one, then $\mathcal{T}$ is the star $K_{1,n-1}$ [30],

7. let $\mathcal{G}$ be a connected graph with exactly three distinct Laplacian eigenvalues. Then the algebraic connectivity (the second smallest Laplacian eigenvalue) of $\mathcal{G}$ is equal to one if and only if $\mathcal{G}$ is a star of $K_{1,n-1}$ with $n \geq 3$, and

8. if $\mathcal{G}$ is a connected graph with integer Laplacian spectra, then $d(\mathcal{G}) \leq 2\kappa(\mathcal{G})$, where $d(\mathcal{G})$ is the diameter of the graph and $\kappa(\mathcal{G}) = (1/n) \prod_{i=2}^{n} \lambda_i(\mathcal{G})$.

Although the spectra of the Laplacian provides important insights, as noted above, into the structural properties of the network, we now proceed to explore the possibility of complete identification of the underlying network using its graph spectra complemented with a sieve method.

3.2 Graph Characterization via Graph Sieve

In this section, we provide an overview of the graph sieve procedure– that in conjunction with the identified Laplacian spectra– leads to a more confined search for the network in the

black box. The essential ideas involve the judicious use of integer partitioning algorithms and degree-based graph reconstruction.

Recall that with the standing assumption of $C = B^T$ in (2.1), for the identified system matrices $(\widetilde{C}, \widetilde{A}, \widetilde{B})$, after appropriate relabeling, the product $\widetilde{C}\widetilde{A}\widetilde{B} = CAB$ leads to the first $r \times r$ block partition of the matrix $L(\mathcal{G})$. Notice that if $B \neq C^T$, we still obtain r^2 entries of the matrix $L(\mathcal{G})$ which may not contain the diagonal entries. The product $\widetilde{C}\widetilde{A}\widetilde{B}$ gives us the degree of the nodes that are in common in both input and output sets and information on the minimum degree of other nodes. Since the eigenvalues of the identified matrix $\widetilde{A}$ are identical to those of $L(\mathcal{G})$, as the result of the identification process, we have access to the sum of the degrees of all nodes in the network as well as the degrees of a subset of r-boundary nodes. Let us define $\widetilde{R}$ as the set of nodes in r-boundary nodes which appears in both $\mathcal{I}$, and $\mathcal{O}$ with $|\widetilde{R}| = \tilde{r}$. Moreover, let

$$r_d = \sum_{v \in \widetilde{R}} \mathbf{deg}\ v \quad \text{and} \quad r_m = \mathbf{trace}(\widetilde{A}) - r_d. \tag{3.2}$$

If $B = C^T$, then $r_d = \mathbf{trace}(\widetilde{C}\widetilde{A}\widetilde{B})$, and the set $\widetilde{R}$ will be equal to r-boundary nodes. We can then proceed to determine the degrees of the $n - \tilde{r}$ remaining nodes, or equivalently, partition the positive integer r_m (3.2) into $n - \tilde{r}$ integers, each assuming a value between 1 and $n - 1$, and a lower bound for the degrees of $r - \tilde{r}$ nodes [158]. The possible values for the partitioning comes from the fact that we expect the resulting graph to be connected while respecting the bounds on the maximum allowable node degrees.

Partitioned integers without constraints on the resulting partitions are often referred to as *unrestricted partitions*. Restricted partitions, on the other hand, are those with constraints on the largest value of the partition that is no greater than a value of K_U, or no smaller than K_L, or both. Algorithms that generate unrestricted partitions can often be used to generate the restricted ones by certain modifications. Several such algorithms, dealing with unrestricted and restricted integer partitioning, have been suggested in the literature. In the context of the graph realization using the proposed system identification method, we proceed to use the algorithms in [31] in order to generate different possible sets of $n - r$ integers between 1 and $n - 1$ and $r - \tilde{r}$ nodes with specified lower bounds on their degree such that their sum is s (3.2).

3.2.1 Integer Partitioning Algorithms and Complexity Analysis

Consider a degree sequence $\mathbf{b} = \{d_1, d_2, \ldots, d_{n-\tilde{r}}\}$ with $d_1 + d_2 + \cdots + d_{n-\tilde{r}} = r_m$ and a specified lower bound on $r - \tilde{r}$ of the values. Without loss of generality, assume that the first $\{d_1, \ldots, d_{r-\tilde{r}}\}$ degrees are lower bounded as

$$d_i \geq L_i \quad \text{for} \quad i = 1, \ldots, r - \tilde{r}. \tag{3.3}$$

We are interested in finding all possible partitions of r_m into $n - \tilde{r}$ integers between 1 and $n - 1$ satisfying (3.3). We have the following observation; see [21].

Lemma 3.2.1. *Let the number of partitions of r_m into $n - \tilde{r}$ integers between 1 and $n - 1$ be denoted by $P_{n-\tilde{r}}(r_m)$. Then*

$$P_{n-\tilde{r}}(r_m) = P_{n-\tilde{r}-1}(r_m - 1) + (n - \tilde{r})P_{n-\tilde{r}}(r_m - 1). \tag{3.4}$$

The following algorithm, proposed in [31], finds all partitions of r_m into $m = n - \tilde{r}$ integers between 1 and $n - 1$ satisfying (3.3). The partitioning of r_m into m components can be generated in increasing lexicographic order by starting with $d_1 = d_2 = \ldots = d_{m-1} = 1, \quad d_m = r_m - m + 1$ and continuing as follows. To obtain the next partition from the first one, scan the elements from right to left, stopping at the right most d_i such that $d_m - d_i \geq 2$. Replace d_j by $d_i + 1$ for $j = i, i + 1, \ldots, m - 1$, and then replace d_m by $s - \sum_{j=1}^{m-1} d_j$. For example, if we have $r_m = 12, \quad m = 5$, and the partition $\{1, 1, 3, 3, 4\}$, we find that 4 is greater by 2 than the rightmost 1, and so the next partition is $\{1, 2, 2, 2, 5\}$. When no element of the partition differs from the last by more than 1, we are done.

In the suggested algorithm the output size of each partitioning of r_m into some arbitrary number of integers m, $P_m(r_m)$, is $\mathcal{O}(r_m)$. This means that the total output size is $\mathcal{O}(r_m P(r_m))$. The approximate size of the number $P(r_m)$ is provided by the following asymptotic formula,

$$P(r_m) \sim \frac{1}{4\sqrt{3}} \exp\left(\pi\sqrt{\frac{2r_m}{3}}\right).$$

In other words, $P(r_m)$ grows faster than any polynomial but slower than any exponential function $Q(r_m) = c^{r_m}$. However, in our application, we are interested in a subset of $P(r_m)$ which has a specified size and satisfy certain constraints. Specifically, the integer r_m in (3.2)

Protocol 1 Integer Partitioning

$d_1 = d_2 = \ldots = d_{m-1} = 1, \quad d_m = r_m - m + 1$

$i = 1$

while $i \neq 0$ **do**

  if $\{d_1, d_2, \ldots, d_{\tilde{r}}\} \geq \{L_1, L_2, \ldots, L_{\tilde{r}}\}$ *and* $1 \leq d_i \leq n-1, \ \forall i$ **then**

    output $\{d_1, d_2, \ldots, d_m\}$

  end

  $i = m - 1$

  while $d_m - d_i < 2$ **do**

    $i = i - 1$

  end

  if $i \neq 0$ **then**

    for $j = m - 1$ *to* i *by -1* **do**

      $d_j = d_i + 1$

    end

  end

  $d_m = s - \sum_{j=1}^{m-1} d_j$

end

is approximately the number of edges in the graph. For simple graphs if $\mathcal{O}(|\mathcal{E}|) = \mathcal{O}(n)$, then the upper bound for the proposed partitioning is $\mathcal{O}(ne^{\sqrt{n}})$, and if $\mathcal{O}(|\mathcal{E}|) = \mathcal{O}(n^2)$, the upper bound for the proposed partitioning is $\mathcal{O}(n^2 e^n)$.

Although the algorithm above leads to a possible degree sequence for the underlying graph— consistent with the identification procedure– we need an additional set of conditions for ensuring that the obtained sequence in fact corresponds to that of a graph.

Definition 3.2.2. *A graphical sequence is a list of nonnegative numbers that is the degree sequence of some simple graph. A simple graph with degree sequence* $\mathbf{d}$ *is said to realize* $\mathbf{d}$.

Our next step is therefore to characterize the necessary and sufficient conditions for a set of integers to be graphical. For this, we resort to the following result.

Theorem 3.2.3. *[26, 32] For* $n > 1$, *an integer list* $\mathbf{d}$ *of size* n *is graphical if and only if* $\mathbf{d}'$

is graphical, where $\mathbf{d}'$ *is obtained from* $\mathbf{d}$ *by deleting its largest element* Δ *and subtracting* 1 *from its* Δ*-th next largest elements. The only* 1*-element graphical sequence is* $\mathbf{d}_1 = \{0\}$.

Example 3.2.4. *Consider a sequence* $\{3, 2, 2, 2, 2\}$. *Since the number of odd degree nodes is odd, the sequence is not graphical.*[1] *Let us also construct the* $\mathbf{d}'$ *sequences described above, as* $\{1, 1, 1, 2\}$ *and* $\{1\}$, *which again verifies that this sequence is not graphical.*

Since the maximum number of steps to check whether a sequence is graphical or not is n, the complexity of this algorithm is $\mathcal{O}(n)$. Given the particular algorithmic means of generating a graphical sequence for the required integer partitions, as detailed above, we now consider the problem of constructing graphs based on a graphical sequence.

3.2.2 Degree Based Graph Construction Algorithms and Complexity Analysis

Before describing the algorithm, we need to provide a few definitions.[2] Let $A(i)$ denote the adjacency set of node i defined as

$$A(i) = \{a_k \,|\, a_k \in \mathcal{V}, a_k \geq i, \text{ for all } k,\ 1 \leq k \leq d_i\}.$$

The reduced degree sequence $\mathbf{d}^{'}|_{A(i)}$ is obtained after removing node i with all its edges from $\mathcal{G}$. We now define the ordering $\leq$ between two adjacency sets of node i, $A(i) = \{\ldots, a_k, \ldots\}$ and $B(i) = \{\ldots, b_k, \ldots\}$, as $B(i) \leq A(i)$ if we have $b_k \leq a_k$ for all $1 \leq k \leq d_i$. In this case we also say that $B(i)$ is "to the left" of $A(i)$. The next lemma introduces a sufficient condition for the sequence $\mathbf{d}^{'}|_{B(i)}$ to be graphical.

Lemma 3.2.5. *[27] Let* $\mathbf{d} = \{d_1, d_2, \ldots, d_n\}$ *be a non-increasing graphical sequence, and let* $A(i)$, $B(i)$ *be two adjacency sets for some node* $i \in \mathcal{V}$, *such that* $B(i) \leq A(i)$. *If the degree sequence reduced by* $A(i)$ *(that is* $\mathbf{d}^{'}|_{A(i)}$*) is graphical, then the degree sequence reduced by* $B(i)$ *(that is* $\mathbf{d}^{'}|_{B(i)}$*) is also graphical.*

---

[1] The sum of the degree os nodes is twice the number of edges, and therefore, it is even. Hence, the number of odd degree nodes in a graphical sequence of degrees must be even.

[2] The necessary definitions and algorithms have been discussed in [27] and are briefly described here to complement the presentation.

The above lemma guarantees preservation of "graphicality" for all adjacency sets to the left of a graphical one. Now consider a graphical degree sequence $\mathbf{d}$ on n nodes obtained from previously discussed integer partitioning approach. From the identity $\widetilde{C}\widetilde{A}\widetilde{B} = CAB$, as we discussed in § 2.1, $r_{\mathcal{I}}r_{\mathcal{O}}/2$ entries of the system matrix $A(\mathcal{G})$ are known;[3] define this set of edges as being "pre-determined" in the graph which cannot be repeated again. Put these connections in the forbidden set $X(\mathbf{d})$. Algorithm 2 describes how we can construct all possible graphs avoiding the edges in $X(\mathbf{d})$. When constructing $A(\mathbf{d})$, checking graphicality is only needed for those adjacency sets which are incomparable by the ordering relationship to any of the current elements of $A(\mathbf{d})$; for the remaining sets, graphicality is guaranteed by Lemma 3.2.5.

The total number of graphs that the above algorithm produces is $\Pi_i(d_i!)$. Of course, this procedure makes sense for degree sequences $\mathbf{d}$ for which there is only a small number of labeled graphs realizing it. An upper bound on the worst case complexity $C_{\mathbf{d}}$ of the algorithm for constructing a sample from a given degree sequence $\mathbf{d}$ is $C_{\mathbf{d}} \leq \mathcal{O}(n|\mathcal{E}|)$, with $|\mathcal{E}|$ being the number of edges in the graph. For simple connected graphs, the maximum possible number of edges is $\mathcal{O}(n^2)$, and the minimum possible number is $\mathcal{O}(n)$. If $\mathcal{O}(|\mathcal{E}|) = \mathcal{O}(n)$, then $C_{\mathbf{d}} \leq \mathcal{O}(n^2)$, and if $\mathcal{O}(|\mathcal{E}|) = \mathcal{O}(n^2)$, then $C_{\mathbf{d}} \leq \mathcal{O}(n^3)$, which is an upper bound, independent of the degree sequence [23]. An algorithm to realize graphical degree sequences of directed graphs has been studied in [22] which can be used to extend these results to simple directed graphs.

Our sieve method for confining our search for the underlying graph following the system identification of §2.1, thus involves:

1. perform an integer partition on the value s (3.2), keeping the partitions that lead to a graphical sequence; let $\mathbf{G}$ denote the set of all graphs that remain after this first stage of the sieve,

2. construct a candidate connected graph in $\mathbf{G}$ that is consistent with the matrix $\widetilde{C}\widetilde{A}\widetilde{B}$ and satisfies the given degree sequence, and

[3] In the case where $C = B^T$, $r^2/2$ entries of $\widetilde{C}\widetilde{A}\widetilde{B}$ are known.

Protocol 2 Degree Based Graph Construction

Given a graphical sequence $\{d_1 \leq d_2 \leq \ldots \leq d_n \leq 1\}$.

I. Define the rightmost adjacency set $A_R(i)$ containing the d_i largest index nodes different from i. Let us also define $X(i) = \{j \in \mathcal{V}, j \neq i \text{ s.t. } \{i,j\} \notin \mathcal{E}\}$ as the forbidden neighbors of node i. Note that $X(i)$ originally might contain some nodes according to the forbidden set $X(\mathbf{d})$. Create the set $A_R(1)$ and $X(1)$ for node 1: connect node 1 to n (this never breaks graphicality). Set $X(1) = \{n\}$. Define the new sequence $\mathbf{d}' = \{d_1 - |X(1)|, d_2, \ldots, d_n\}$. Let $k = n - 1$.

   I.1. Connect another edge of 1 to k. Run the graphicality test in Theorem 3.2.3.

   I.2. If this test fails, set $k = k - 1$; repeat I.1.

   I.3. If the test passes, keep (save) the connection, add the node k to the forbidden set $X(1)$ and update the degree sequence $\mathbf{d}' = \{d_1 - |X(1)|, d_2, \ldots, d_n\}$, set $k = k-1$, and if i has edges left, repeat from I.1.

II. Create the set $A(\mathbf{d})$ of all adjacency sets of node 1 that are colexicographically smaller than $A_R(1)$ and preserve graphicality, i.e.,

$$A(\mathbf{d}) = \left\{ A(1) = \{a_1, \ldots, d_{d_1}\}, a_i \in \mathcal{V} \,|\, A(1) <_{CL} A_R(1), \mathbf{d}'|_{A(1)} \text{ is graphical} \right\},$$

   where the operation $<_{CL}$ means a colexicographic order between two sets.

III. For every $A(1) \in A(\mathbf{d})$, create all graphs from the corresponding graph realization of $\mathbf{d}'|_{A(1)}$ using this algorithm, where $\mathbf{d}'_{A(1)}$ is the sequence reduced by $A(1)$.

---

3. compare the Laplacian eigenvalues of the constructed graphs with the roots of the characteristic polynomial (3.1) and discard the inconsistent graphs.

The candidate graphs for the underlying network topology are now among the ones that remain after this three step sieve. We note that the sieve method is guaranteed to reduce

the search space for the graph structure by at least a factor of 2^n, a bound that is obtained from the bound on the number of permissible degree-based integer partitionings facilitated by the network system identification. An example for this procedure is given next. Our goal in this example is to gather information on the graph $\mathcal{G}$ shown in Figure 3.1(a) using the system identification procedure. Using nodes 1, 2, and 3 as the input-output nodes in (2.1), we obtain $\phi_{\mathcal{G}}(s) = s^6 + 220s^5 + 190s^4 + 804s^3 + 1664s^2 + 1344s$. Since the polynomial $\phi_G(s)$ has just one zero root, the underlying graph is connected. Moreover, the graph is not a tree due to the fact that $a_{n-1} \neq 6$. The graph has 11 edges and 224 spanning trees. Since the diagonal of the matrix $\widetilde{C}\widetilde{A}\widetilde{B}$ is $[\,-3, -4, -3\,]^T$, $d_1 = 3$, $d_2 = 4$, and $d_3 = 3$. In the meantime, the sum of degrees of the remaining nodes is 12. The possible integer partitions for the remaining three nodes such that the sum is 12 and each degree is less than 6 will be $\{5, 5, 2\}$, $\{5, 4, 3\}$, and $\{4, 4, 4\}$. Therefore, the set of the possible degrees sequences is comprised of $\{3, 4, 3, 5, 5, 2\}$, $\{3, 4, 3, 5, 4, 3\}$, and $\{3, 4, 3, 4, 4, 4\}$. According to the Theorem 3.2.3, three sets of integer partitioning are realizations of some graphs.

Next, we construct the graphs with the three candidate degree sequences by implementing the algorithm in [27] and constructing all connected graphs consistent with $\widetilde{C}\widetilde{A}\widetilde{B}$ and with degree sequences $\{3, 4, 3, 5, 5, 2\}$, $\{3, 4, 3, 5, 4, 3\}$, and $\{3, 4, 3, 4, 4, 4\}$. The first degree sequence, $\{3, 4, 3, 5, 5, 2\}$, is a realization of just one graph while the second degree sequence is a realization of two graphs. The third degree sequence is a realization of 12 graphs. All these graphs satisfy the constraint imposed by $\widetilde{C}\widetilde{A}\widetilde{B}$. By comparing the Laplacian spectra for the corresponding 15 graphs with the roots of the identified characteristic polynomial, we can thereby identify the original graph. Three candidates for the constructed graph are depicted in Fig. 3.1(b)-(d).

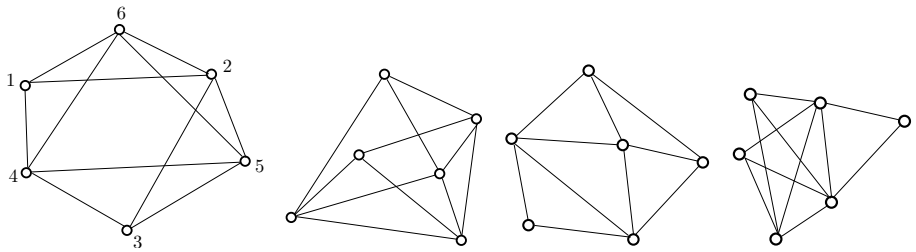


Figure 3.1: (a) A simple graph on 6 nodes, (b) a candidate graph constructed with degree sequence $\{3, 4, 3, 4, 4, 4\}$, (c) another candidate graph constructed with degree sequence $\{3, 4, 3, 4, 4, 4\}$, and (d) the graph constructed with degree sequence $\{3, 4, 3, 5, 5, 2\}$.

Chapter 4

NETWORK IDENTIFICATION VIA GRAPH REALIZATION

In this section, we examine the problem of identifying the interaction geometry among a known number of agents, adopting a consensus-type algorithm for their coordination. The proposed identification process is facilitated by introducing "ports" for steering a subset of network vertices via an appropriately defined interface and observing the network's response at another set of vertices. It is first noted that under the assumption of controllability and observability of a corresponding steered-and-observed network, the proposed procedure identifies a number of important features of the network using the spectrum of the graph Laplacian. Using the fact that system identification provides a realization of the original network, we proceed to utilize transformations to identify a graph topology that is compatible with the set of input-output data. An example demonstrates the application of the proposed method.

4.1 Review of Existing Approaches

In the first approach proposed in [219] for unweighted networks, it was first noted that the proposed identification procedure in §2.2 identifies a number of important features of the network using the spectrum of the graph Laplacian. This is in light of the fact that although the spectra of the graph Laplacian in general is insufficient to form an explicit characterization of the underlying network, it leads to a number of useful structural information about its geometry, e.g., one has $|\mathcal{E}| = 1/2 \sum_{i=1}^{n} \lambda_i$, where $|\mathcal{E}|$ is the number of edges in the graph. Since the eigenvalues of the identified matrix $\widetilde{A}$ are identical to those of $L_w(\mathcal{G})$, we have access to the number of edges in the network as well as the degrees of a subset of boundary nodes (set of input/output nodes).

The product $\widetilde{C}\widetilde{A}\widetilde{B}$ provides the degree of a subset of boundary nodes as mentioned in §2.2. Further, the product $\widetilde{C}\widetilde{A}\widetilde{B}$ encodes the degree of the nodes that are common to both

input and output sets and information on the minimum degree of the remaining input and output nodes.

Let us define $\widetilde{R}$ as the set of nodes in boundary nodes which appear in both $\mathcal{I}$ and $\mathcal{O}$ with $|\widetilde{R}| = \tilde{r}$. Moreover, let

$$r_d = \sum_{v \in \tilde{R}} \mathbf{deg}\ v \quad \text{and} \quad r_m = \mathbf{trace}(\widetilde{A}) - r_d. \tag{4.1}$$

If $B = C^T$, then $r_d = \mathbf{trace}(\widetilde{C}\widetilde{A}\widetilde{B})$, and the set $\widetilde{R}$ will be equal to the boundary nodes.

We can then proceed to determine the degrees of the $n - \tilde{r}$ remaining nodes, or equivalently, partition the positive integer r_m, defined in (4.1), into $n - \tilde{r}$ integers, each assuming a value between 1 and $n - 1$, and a lower bound for the degrees of the input and output set of nodes not common in both sets [158]. The possible values for the partitioning come from the expectation that the resulting graph is connected while simultaneously respecting the bounds on the maximum allowable node degree.

After performing integer partitioning on the value r_m in (4.1), we categorize the partitions which are feasible as a degree-sequence of a connected graph. Based on degree-sequence graph construction, we catalogue all possible graphs consistent with matrix $\widetilde{C}\widetilde{A}\widetilde{B}$ and satisfying the degree sequence. In the last sieve step, we compare the Laplacian eigenvalues of the constructed graphs with the graph spectrum calculated from the system identification procedure and discard inconsistent graphs. The candidate graphs for the underlying network topology are now among the ones that remain after this sieve step.

In order to study the complexity analysis of this approach, consider $d = \{d_1, d_2, \ldots, d_n\}$ to be a degree-sequence. An upper bound on the worst case complexity of the algorithm for constructing a sample from a given degree sequence $\mathbf{d}$, denoted by $C_{\mathbf{d}}$, is $C_{\mathbf{d}} \leq \mathcal{O}(n|\mathcal{E}|)$, with $|\mathcal{E}|$ the number of edges in the graph. For simple connected graphs, the maximum possible number of edges is $\mathcal{O}(n^2)$, and the minimum possible number is $\mathcal{O}(n)$. If $\mathcal{O}(|\mathcal{E}|) = \mathcal{O}(n)$, then $C_{\mathbf{d}} \leq \mathcal{O}(n^2)$, and if $\mathcal{O}(|\mathcal{E}|) = \mathcal{O}(n^2)$, then $C_{\mathbf{d}} \leq \mathcal{O}(n^3)$, which is an upper bound, independent of the degree sequence.

While the above approach, described in more details in [219], was applied to unweighted graphs, the second approach can be implemented for unweighted as well as weighted graphs.

In the second approach proposed in [217], inspired by how biologists use gene knockouts for experimentally identifying genetic interaction networks in cellular organisms, the authors propose a node-knockout procedure for the complete characterization of the interaction geometry in consensus-type networks.

The proposed approach in [217] and in previous chapters, despite some of its practical ramifications in fault detection, is computationally expensive. It is worth mentioning that the approaches in [219, 217] were motivated by the fact that solving the following bilinear equations in is under-constrained.

$$\begin{aligned} \tilde{A} &= TAT^{-1} \\ \tilde{B} &= TB \\ \tilde{C} &= CT^{-1}. \end{aligned} \tag{4.2}$$

In the present section, however, we investigate an approach that directly examines (4.2) in relation with the network topology identification problem. The approach utilizes an appropriate similarity transformation in order to identify a weighted graph that shares the same spectrum as the original graph with identical input/output behavior. An extension of the proposed approach for weighted graphs to unweighted graphs using mixed integer least squares will also be discussed followed by an example.

4.2 Similarity Transformation Approach

Consider the set of identities in (4.2) with known triplet $(\widetilde{A}, \widetilde{B}, \widetilde{C})$ and unknown quadruple (T, A, B, C). For a symmetric matrix A, these sets of equations are under-constrained with $n(n-1)/2$ undetermined variables; however, the special structure of the matrix A as a graph Laplacian can be considered in order to restrict degrees of freedom. The main contribution of this section is finding the transformation matrix T considering the Laplacian structure of A. The following theorem states a well-known result in matrix analysis which will be used in our approach [123, 159].

Theorem 4.2.1. *If W_1 and W_2 are similar negative semi-definite matrices, the similarity transformation between them is a unitary matrix.*

The proposed approach in this section builds on the unitary transformation between two negative-semi definite (NSD) matrices as well as the structural properties of the graph Laplacian and the Householder transformation described shortly. We note that the matrix $\widetilde{A}$ shares the same non-positive eigenvalues of the Graph Laplacian A; however, for a matrix to be NSD, it also has to be symmetric with non-positive eigenvalues [123]. Since $\widetilde{A}$ is the output of the system identification procedure, there is no guarantee that it will always be symmetric. Therefore, the next step is to find a symmetric matrix $\widetilde{A}_s$ that shares the same spectrum as $\widetilde{A}$ and has the minimum distance from $\widetilde{A}$ in some matrix norm. Choosing the Frobenius norm to induce a metric on the space of matrices, the problem can be formulated as follows:

$$\begin{cases} \min\limits_{\widetilde{A}_s} \ ||\widetilde{A} - \widetilde{A}_s||_F^2 \\ \quad \text{s.t. } \widetilde{A}_s^T = \widetilde{A}_s \qquad \text{and} \qquad \text{eig}(\widetilde{A}_s) = \text{eig}(\widetilde{A}). \end{cases} \tag{4.3}$$

Our next result provides an optimal analytic solution of the above optimization problem. This theorem constructs the nearest symmetric matrix $\widetilde{A}_s$ which shares the same spectrum as $\widetilde{A}$.

Theorem 4.2.2. *Let $\widetilde{A} \in \mathbf{R}^{n\times n}$ with non-positive set of eigenvalues. Consider the optimization problem*

$$\begin{cases} \min\limits_{\widetilde{A}_s} \ ||\widetilde{A} - \widetilde{A}_s||_F^2 = ||\widetilde{A} - X||_F^2 \\ X^T = X \quad \textit{and} \quad \textit{eig}(X) = \textit{eig}(\widetilde{A}), \end{cases} \tag{4.4}$$

where X is negative semi-definite. Then,

$$\widetilde{A}_s = U\Lambda U^T, \tag{4.5}$$

where Λ is a diagonal matrix with eigenvalues of $\widetilde{A}$ on the diagonal, $\Lambda_{i,i} = \lambda_i(\widetilde{A})$, and U is the unitary matrix in the spectral decomposition of the symmetric part of $\widetilde{A}$, i.e., $S = (\widetilde{A} + \widetilde{A}^T)/2$. The unitary matrix U appears in the spectral decomposition of S as $S = U\Lambda_S U^T$, where Λ_S is diagonal and $\Lambda_{S_{i,i}} = \lambda_i(S)$.

Proof. The proof is inspired by [160] where the nearest symmetric positive semidefinite matrix in the Frobenius norm to an arbitrary real matrix is characterized; however, the

proposed nearest symmetric positive semidefinite matrix in [160] does not share the spectrum of the original matrix.

Let X be the solution of the optimization problem (4.4). The Frobenius norm has the property that $||G+H||_F^2 = ||G||_F^2 + ||H||_F^2$, if $G = G^T$ and $H = -H^T$. Consider $\widetilde{A} = S+K$, where $S = (\widetilde{A} + \widetilde{A}^T)/2$ and $K = (\widetilde{A} - \widetilde{A}^T)/2$ are the symmetric and skew symmetric parts of $\widetilde{A}$, respectively. Then, we have

$$||\widetilde{A} - X||_F^2 = ||S - X||_F^2 + ||K||_F^2,$$

where $G = S - X$ and $H = K$. The problem of minimizing $||\widetilde{A} - X||_F^2$, therefore, reduces to minimizing $||S - X||_F^2$.

Let $S = U\Lambda_S U^T$, where U is unitary and the diagonal matrix Λ_S is such that $\Lambda_{S_{i,i}} = \lambda_i(S)$. Moreover, let $Y = U^T X U$. Then,

$$\begin{aligned} ||S - X||_F^2 &= ||U\Lambda_S U^T - X||_F^2 \\ &= ||U(\Lambda_S - U^T X U)U^T||_F^2 = ||\Lambda_S - Y||_F^2 \\ &= \sum_{i \neq j} y_{ij}^2 + \sum_i (\lambda_i - y_{ii})^2 \geq \sum_i (\lambda_i - y_{ii})^2. \end{aligned}$$

The lower bound above is attained, uniquely, for the matrix $Y = \Lambda = \mathbf{diag}(\lambda_i(\widetilde{A}))$, where $\lambda_i(\widetilde{A})$ is the i-th eigenvalue of $\widetilde{A}$ if these eigenvalues are rearranged such that $|\lambda_1(S) - \lambda_1(\widetilde{A})| \leq |\lambda_2(S) - \lambda_2(\widetilde{A})| \leq \ldots \leq |\lambda_n(S) - \lambda_n(\widetilde{A})|$. Therefore, for this choice of Y, $X = U\Lambda U^T$.

The next step is to show that Y, and consequently X, are unique. In order to prove this, it suffices to show $||\Lambda_S - \Lambda||_F^2 \leq ||\Lambda_S - W\Lambda W^T||_F^2$ for an arbitrary unitary matrix W. Let us denote $\Omega = W\Lambda W^T$. The definition of Frobenius norm now implies that

$$\begin{aligned} ||\Lambda_S - \Lambda||_F^2 &= \mathbf{trace}[(\Lambda_S - \Lambda)(\Lambda_S - \Lambda)^T] \\ &= \mathbf{trace}(\Lambda_S^2) - 2\mathbf{trace}(\Lambda_S \Lambda) + \mathbf{trace}(\Lambda^2) \\ ||\Lambda_S - \Omega||_F^2 &= \mathbf{trace}[(\Lambda_S - \Omega)(\Lambda_S - \Omega)^T] \\ &= \mathbf{trace}(\Lambda_S^2) - 2\mathbf{trace}(\Lambda_S \Omega) + \mathbf{trace}(\Omega^2) \\ &= \mathbf{trace}(\Lambda_S^2) - 2\mathbf{trace}(\Lambda_S \Omega) + \mathbf{trace}(\Lambda^2), \end{aligned}$$

where the last equality follows from $||\Lambda||_F^2 = ||\Omega||_F^2$ since W is unitary.

In order to prove the claim that Y and X are unique, we need to show $\mathbf{trace}(\Lambda_S \Lambda) \geq \mathbf{trace}(\Lambda_S \Omega)$. Note that all diagonal entries of Λ, Λ_S, and Ω are non-positive. Let us now consider $\mathbf{trace}[\Lambda_S(\Omega - \Lambda)]$. Then,

$$\begin{aligned}\mathbf{trace}[\Lambda_S(\Omega - \Lambda(\widetilde{A}))] = \sum_i \lambda_i(S)[\Omega_{i,i} - \lambda_i(\widetilde{A})] = \sum_i |\lambda_i(S)|[|\Omega_{i,i}| - |\lambda_i(\widetilde{A})|] \\ \leq |\max_i(\lambda_i(S))| \sum_i (|\Omega_{i,i}| - |\lambda_i(\widetilde{A})|) = \\ |\max_i(\lambda_i(S))| \left[\sum_i |\Omega_{i,i}| - \sum_i |\lambda_i(\widetilde{A})|\right] = 0,\end{aligned}$$

where the fact that $\Lambda(\widetilde{A})$ and Ω share the same set of eigenvalues imply the last equality. Therefore, $\mathbf{trace}[\Lambda_S(\Omega - \Lambda)] \leq 0$, and the claim is proved. □

Theorem 4.2.2 thus implies that $\widetilde{A}_s$ is the nearest symmetric approximation of $\widetilde{A}$ sharing the same spectrum. We thus proceed to find the similarity transformation between $\widetilde{A}$ and $\widetilde{A}_s$; however, it is non-trivial whether there is a similarity transformation between $(\widetilde{A}, \widetilde{B}, \widetilde{C})$ and $(\widetilde{A}_s, \widetilde{B}_s, \widetilde{C}_s)$. Therefore, first we verify whether there is a similarity transformation between $\widetilde{A}$ and $\widetilde{A}_s$ and then propose a procedure to find the transformation. The following theorem explores the existence of a similarity transformation between $\widetilde{A}$ and $\widetilde{A}_s$.

Theorem 4.2.3. *The pair of matrices $\widetilde{A}$ and $\widetilde{A}_s$ are similar.*

Proof. We note that there is a similarity transformation between $\widetilde{A}$ and A as $\widetilde{A} = TAT^{-1}$ from the system identification procedure. We also know that two symmetric matrices A and $\widetilde{A}_s$ share the same non-positive spectrum. On the other hand, Theorem 4.2.1 implies that there is a unitary transformation Q such that $A = Q^T \widetilde{A}_s Q$. Thus, there is a similarity transformation between $\widetilde{A}_s$ and $\widetilde{A}$ as $\widetilde{A} = G\widetilde{A}_s G^{-1}$ where

$$G = TQ^T. \tag{4.6}$$

Fig. 4.1 demonstrates these relationships. □

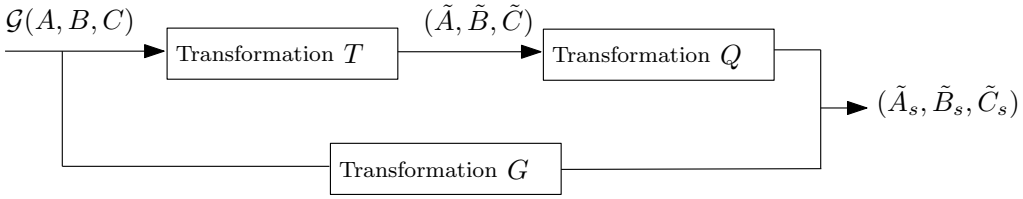


Figure 4.1: Similarity transformations between A, $\widetilde{A}$, and $\widetilde{A}_s$

As proved in Theorem 4.2.3, there is a similarity transformation between $\widetilde{A}$ and $\widetilde{A}_s$. The purpose of the next step is to find the transformation matrix G such that

$$\min_G ||\widetilde{A}G - G\widetilde{A}_s||_2, \tag{4.7}$$

where $\widetilde{A}_s$ is defined in (4.5). Consequently, a new realization $(\widetilde{A}_s, \widetilde{B}_s, \widetilde{C}_s)$ is derived as

$$\begin{aligned} \widetilde{A}_s &= G^{-1}\widetilde{A}G \\ \widetilde{B}_s &= G^{-1}\widetilde{B} \\ \widetilde{C}_s &= \widetilde{C}G. \end{aligned} \tag{4.8}$$

As we stated earlier in Theorem 4.2.1, there is a unitary transformation between two NSD matrices that have the same set of eigenvalues. The next step is to blend this fact with structural features of the graph Laplacian to find the unitary transformation Q between two NSD matrices A and $\widetilde{A}_s$, i.e., $\widetilde{A}_s Q = QA$. Since A is the Laplacian associated with an undirected weighted graph with the property that its rows sum to zero, it follows that $\widetilde{A}_s Q\mathbf{1} = QA\mathbf{1} = 0$, where $\mathbf{1}$ is an $n \times 1$ vector of ones.

The equality $\widetilde{A}_s Q\mathbf{1} = 0$ indicates that $Q\mathbf{1}$ belongs to the null space of $\widetilde{A}_s$. Let us denote the null space of $\widetilde{A}_s$ as $\widetilde{N} = \text{Null}\{\widetilde{A}_s\}$. The equality $Q\mathbf{1} = \widetilde{N}$ implies that the unitary matrix Q rotates the vector of ones to the null space $\widetilde{N}$. The well-known Householder reflection suggests a procedure to find such a unitary matrix Q.

A Householder reflection, also known as Householder transformation, is a linear transformation that describes a reflection about a plane or hyperplane containing the origin. A unit vector u orthogonal to the hyperplane defines the reflection hyperplane. Let

$$Q\frac{\mathbf{1}}{\sqrt{n}} = \frac{\widetilde{N}}{||\widetilde{N}||_2}.$$

Then, the Householder reflection suggests that the Hermitian and unitary transformation Q satisfies the following equation

$$Q = I - 2uu^T, \tag{4.9}$$

where $u = v/||v||_2$ such that $v = \widetilde{N}/||\widetilde{N}||_2 - \mathbf{1}/\sqrt{n}$.

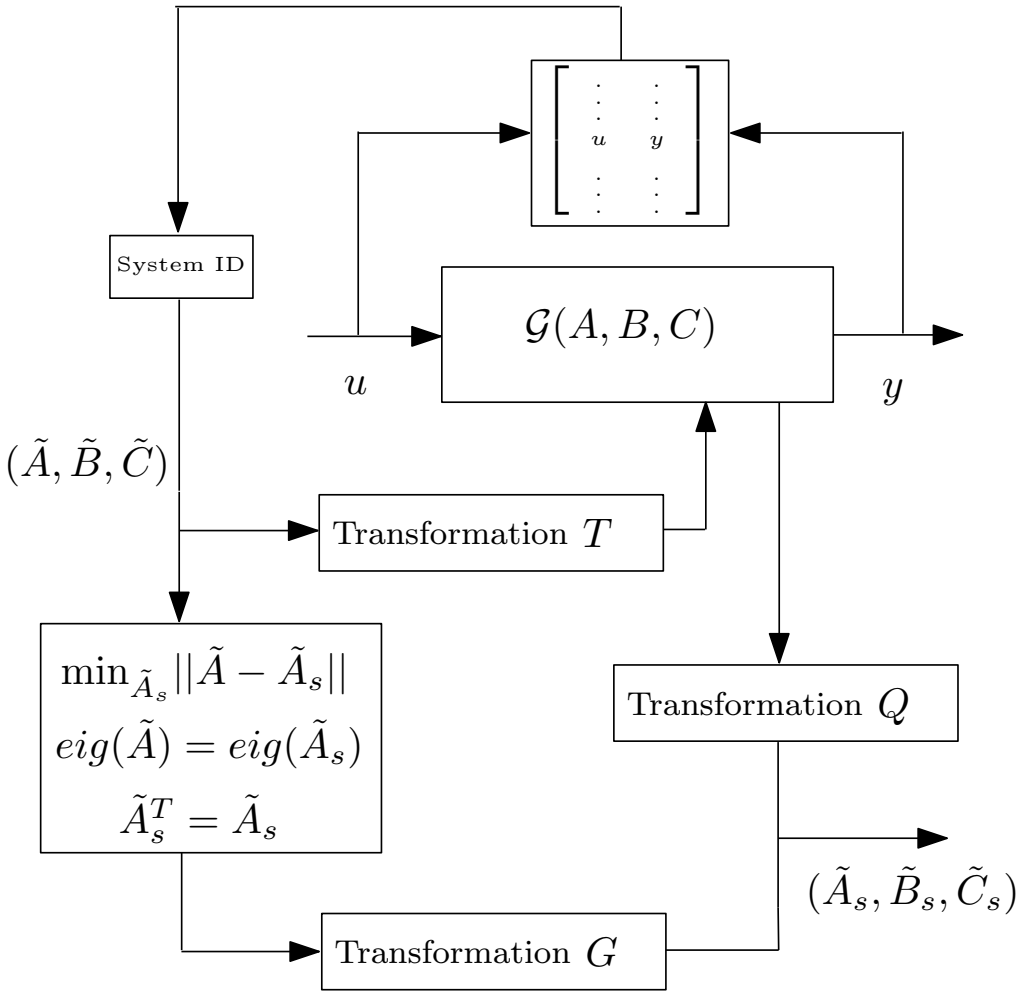


Figure 4.2: Identification procedure

4.2.1 Numerical considerations

Fig. 4.2 demonstrates a summary of the network identification procedure described above. However, there are a few numerical analysis aspects to the that will be discussed in this

section.

As discussed earlier, in order to calculate the matrix Q, it is necessary to find the null space of $\widetilde{A}_s$. Since $\widetilde{A}_s$ shares the same spectrum as A, it is singular and its null space, $\widetilde{N}$, is non-empty; however, numerically the null space $\widetilde{N}$ is generically empty. Therefore, we characterize the matrix $\widetilde{A}_s^S = \widetilde{A}_s + E$ with E as the solution of the optimization problem

$$\begin{cases} \min\limits_{E} \ \frac{||E||}{||\widetilde{A}_s||} \\ \text{s.t} \ \ \widetilde{A}_s^{S^T} = \widetilde{A}_s^S \ \ \text{and} \ \ \widetilde{A}_s + E = \widetilde{A}_s^S \text{ is singular,} \end{cases} \tag{4.10}$$

where $||\cdot||$ is the induced matrix 2-norm. This problem is equivalent to finding the nearest matrix of lower rank to a given matrix [161]. The procedure for determining the nearest singular approximation of $\tilde{A}_s$ is inspired by [162] and [161].

Suppose that $||.||$ is the induced matrix norm and let z be a normalized vector such that $||\tilde{A}_s^{-1}z|| = ||\tilde{A}_s^{-1}||$ which is a valid assumption since numerical issues generically cause $\tilde{A}_s$ to be non-singular. Note that if the vector norm is the $2-$norm, then z is the normalized eigenvector associated with the maximum eigenvalue of $\tilde{A}_s^{-1}$. Let

$$w = \frac{\tilde{A}_s^{-1}z}{||\tilde{A}_s^{-1}||},$$

and v be a vector such that $||v||_* = 1$, where $||.||_*$ is the norm dual to the vector norm $||.||$. Note that the dual norm of the $2-$norm is the 2-norm as

$$v^T w = \max_{||u||_*=1} |u^T w|.$$

It then follows that $v^T w = ||w|| = 1$ [162]; note that for the $2-$norm, it can be shown that $v = w$ [123].

Consider next the vector $2-$norm and set

$$E = -\frac{zv^T}{||\widetilde{A}_s^{-1}||} = -\frac{zw^T}{||\widetilde{A}_s^{-1}||}.$$

Let us define

$$\widetilde{A}_s^S = \widetilde{A}_s + E = \widetilde{A}_s - \frac{zw^T}{||\widetilde{A}_s^{-1}||}. \tag{4.11}$$

Then,

$$\begin{aligned} \widetilde{A}_s^S w &= \left(\widetilde{A}_s + E\right) w = \widetilde{A}_s w - \frac{zv^T}{||\widetilde{A}_s^{-1}||} w \\ &= \frac{z}{||\widetilde{A}_s^{-1}||} - \frac{z}{||\widetilde{A}_s^{-1}||} = 0 \end{aligned}$$

implies that $\widetilde{A}_s^S = \widetilde{A}_s + E$ is singular, and w is in its null space.

In order to calculate Q in (4.9), it was necessary to find the null space of $\widetilde{A}_s$, denoted by $\widetilde{N}$. We previously showed that $\widetilde{A}_s$ and $\widetilde{N}$ can be approximated by $\widetilde{A}_s^S$ and w, respectively. Therefore, $A = Q\widetilde{A}_s^S Q^T$ where Q is defined in (4.9) and $\widetilde{A}_s^S$ in (4.11).

We note that A can also be calculated in a different manner. The equality $G = TQ$ in (4.6) with known transformation G, as a solution of (4.7), implies that $T = GQ^T$. By finding the transformation T, the original matrix A is thereby determined as $A = T^{-1}\widetilde{A}T$.

The approach described in this section considers weighted consensus dynamics and provides a weighted graph that shares the same spectrum as the original weighted Laplacian with the same input/output behavior. The approach can be extended to the unweighted case, where the weights on the edges are zero or one.

In order to identify the unweighted network, we note that numerical constraints cause the calculated matrix A to be a perturbed version of a Laplacian matrix even when the underlying graph is unweighted. That is, the entries of the obtained matrix are not integer. We now propose a procedure to approximate $-A$ with the closest integer matrix L that satisfies the properties of a graph Laplacian.

The following mixed integer optimization problem determines the closest integer matrix L that minimizes the induced matrix $2-$norm of $A - L$ as

$$\min_{L} \; ||L - A||_2,$$

subject to

$$\begin{aligned}
&L_{i,i} > 0 \\
&L_{i,j} \in \{0, -1\} \quad i \neq j \\
&L = L^T \\
&L\mathbf{1} = 0 \\
&\sum_{i=1}^{n} L_{i,i} = -\text{trace } (\widetilde{A}) \\
&L \;\text{ satisfies the constraints posed by } \; \widetilde{C}\widetilde{A}\widetilde{B} = CAB \text{ described in §2.2.}
\end{aligned} \tag{4.12}$$

The proposed optimization problem can be formulated as a mixed integer least squares problem and can be solved by a powerful CPLEX software [163]. Even though there is a

subclass of mixed integer programming problems that can be solved in polynomial time, most such problems are NP-hard. This is the main pitfall associated with the application of the procedure for identification of weighted to unweighted graphs.

The implementation of the proposed methodology is discussed via an illustrative example in the next section. For this example, we study the unweighted consensus identification.

4.3 A Numerical Example

Our goal in this example is to gather information on the graph $\mathcal{G}$ shown in Fig. 4.3 running the system identification procedure. Using nodes 1, 2, and 3 as the input-output nodes in (2.1), we obtain $\phi_{\mathcal{G}}(s) = s^6 + 220s^5 + 190s^4 + 804s^3 + 1664s^2 + 1344s$. Since the polynomial $\phi_G(s)$ has just one zero root, the underlying graph is connected. As we discussed in [217], the characteristic polynomial $\phi_{\mathcal{G}}(s)$ reveals certain properties of the graph. For example, the graph is not a tree due to the fact that $a_{n-1} \neq 6$. Moreover, the graph has 11 edges and 224 spanning trees.

We also obtain the estimated matrices $\widetilde{A}$, $\widetilde{B}$, and $\widetilde{C}$ from the system identification procedure. For this example, we assume that the input and output nodes are identical. Since the diagonal of the matrix $\widetilde{C}\widetilde{A}\widetilde{B}$ is $[-3, -4, -4]^T$, $d_1 = 3$, $d_2 = 4$, and $d_3 = 3$ if we label the first three nodes as the input/output nodes. Following the proposed procedure discussed in the section, we obtain the matrix $\widetilde{A}_s$ and its nearest singular approximation $\widetilde{A}_s^S$. The transformations Q is then determined to be

$$Q = \begin{bmatrix} 0.8856 & 0.2442 & 0.2535 & 0.1640 & 0.2211 & -0.1266 \\ 0.2442 & 0.4788 & -0.5412 & -0.3500 & -0.4720 & 0.2702 \\ 0.2535 & -0.5412 & 0.4381 & -0.3634 & -0.4901 & 0.2805 \\ 0.1640 & -0.3500 & -0.3634 & 0.7650- & 0.3169 & 0.1814 \\ 0.2211 & -0.4720 & -0.4901 & -0.3169 & 0.5725 & 0.2447 \\ -0.1266 & 0.2702 & 0.2805 & 0.1814 & 0.2447 & 0.8600 \end{bmatrix}.$$

This set of data and the corresponding transformations imply that matrix A and its

integer Laplacian approximation L are as follows,

$$A = \begin{bmatrix} -3.9587 & 1.7068 & 0.4278 & 0.4031 & 0.7707 & 0.6504 \\ 1.7068 & -4.0888 & 0.2449 & 1.1947 & 0.8076 & 0.1348 \\ 0.4278 & 0.2449 & -3.7002 & 0.4628 & 0.7900 & 1.7748 \\ 0.4031 & 1.1947 & 0.4628 & -3.4354 & 0.7497 & 0.6252 \\ 0.7707 & 0.8076 & 0.7900 & 0.7497 & -3.3747 & 0.2567 \\ 0.6504 & 0.1348 & 1.7748 & 0.6252 & 0.2567 & -3.4420 \end{bmatrix}$$

and

$$-L = \begin{bmatrix} 3 & -1 & -1 & 0 & -1 & 0 \\ -1 & 4 & 0 & -1 & -1 & -1 \\ -1 & 0 & 4 & -1 & -1 & -1 \\ 0 & -1 & -1 & 4 & -1 & -1 \\ -1 & -1 & -1 & -1 & 4 & 0 \\ 0 & -1 & -1 & -1 & 0 & 3 \end{bmatrix}.$$

Fig. 4.3 depicts the original graph and the identical determined graph from the proposed approach.

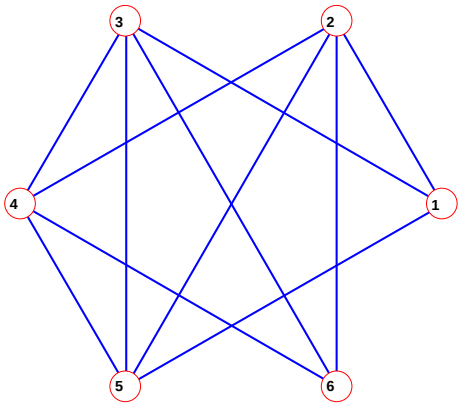


Figure 4.3: The original graph is identical to its identified twin.

Part II

CONTROLLABILITY OVER NETWORKS

Chapter 5

CONTROLLABILITY AND OBSERVABILITY OF CIRCULANT NETWORKS

This chapter examines the controllability of a group of agents with first order dynamics adopting a weighted consensus-type coordination protocol over a network, specifically, path graphs, Circulant graphs, and Cartesian products of prime networks.

It is shown that a circulant network with Laplacian eigenvalues of maximum algebraic multiplicity q is controllable from q nodes. Moreover, any set of $q-1$ input nodes leads to an uncontrollable network. Our approach leverages the Cauchy-Binet formula, which, in conjunction with the Popov-Belevitch-Hautus test, provides a new method to examine controllability properties for more general networks. The method is also utilized to fully characterize the controllability of path graphs.

We then present an analysis framework for a class of dynamic composite networks. These networks are formed from smaller factor networks via graph Cartesian products. We provide a composition method for extending the controllability and observability of the factor networks to that of the composite network.

5.1 Introduction

Recently, controllability and observability of networked dynamic systems adopting consensus-type coordination algorithm has attracted the attention of researchers in distinct disciplines [100, 101, 102, 103, 104]. Network controllability arises in situations where a networked system is influenced or observed by an external entity, a scenario that is of importance in networked robotic systems, human-swarm interaction, and network security [105, 219, 220], as well as in areas such as quantum networks [106, 107].

In [100], Godsil made an intriguing conjecture regarding the controllability of the dynamics driven by the adjacency matrix of a graph. The conjecture states that the ratio of graphs that are uncontrollable from any set of nodes to the total number of graphs of the

same order tends to zero as the order of the graph increases.[1] On another front, extensive simulations have demonstrated that it is "unlikely" that single leader-follower Laplacian based consensus networks are controllable [103]. Together these observations imply that establishing network controllability is non-generic and may be strongly dependent on the manifestation of the graph in the dynamics.

Different approaches have been adopted in the literature regarding the controllability properties of networks. In [105], the authors examine how symmetry structure of the graph, as exemplified through its automorphisms or equitable partitions, contributes to the uncontrollability of the network. Therefore, a natural question is whether certain classes of networks can be made controllable by breaking their symmetry. As graph symmetry is closely related to the multiplicity of the eigenvalues of their adjacency and Laplacian matrices,[2] it is thus natural to explore classes of graphs for which the relation between controllability and eigenvalue multiplicity can be made more explicit. Studying the controllability of circulant networks [202] and chain and multi-chain networks in [200, 201] are attempts in that direction.

The authors in [102] investigate the observability properties of a Laplacian based consensus algorithm over a path and cycle graph utilizing the PBH test and closed form eigenvector structure of path and cycle graphs; note that the maximum algebraic multiplicity of both classes of graphs is at most two. In [103], the underlying network is a Cartesian product of path graphs, called a grid. The controllability properties of the prime factors along with the PBH test are the key analysis tools here. The blend of the PBH test with the Cauchy-Binet formula, however, enables us to investigate the controllability properties of circulant networks with arbitrary maximum algebraic multiplicity.

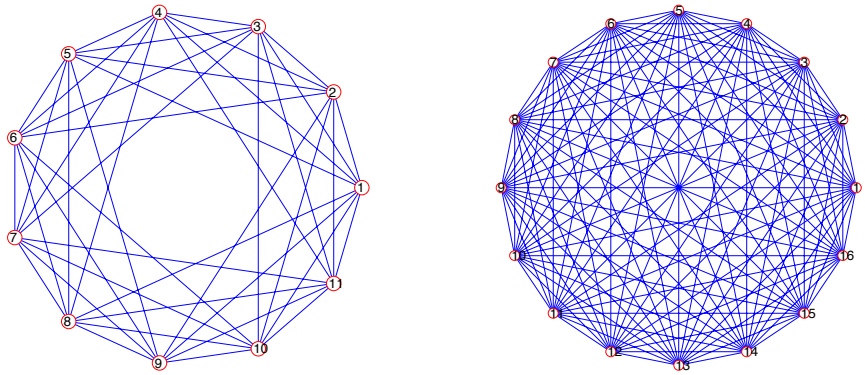
Liu et al. in [203] study the controllability of complex directed networks. They identify a subset of driver nodes that steer the network to any arbitrary state in finite time. They show that dense and homogenous networks can be controlled using a few driver nodes while the inhomogeneous networks, which appear in real networks, are the most difficult to control.

---

[1] This conjecture is stated with respect to a dynamics that is driven by the adjacency matrix of the graph.

[2] Higher algebraic multiplicity is often associated with a highly symmetric graph; the correspondence, however, is not exact for general graphs.

In the present chapter, we consider network controllability for a special class of graphs, namely those that are circulant, path, and Cartesian product of prime order networks. We begin with circulant graphs.



(a) (b)

Figure 5.1: Circulant graphs: (a) 8-regular with 11 nodes, and (b) 15-regular with 16 nodes; generated using Matgraph [178]

A circulant graph is a graph of order n in which the i-th vertex is adjacent to vertices $i \pm j \pmod{n}$, where j is in a list l; see Fig. 5.1. It is known that for an arbitrary n, there exists $2^{\lfloor n/2 \rfloor}$ undirected (unlabeled) graphs [108, 109]. On the application side, circulant graphs appear in coding theory and VLSI design, and constitute the basis for designing data alignment networks for complex memory systems [164]. Another application of circulant graphs is in quantum communication [167]. Circulant graphs, characterized by high scalability and modularity, also represent an important class of reliable interconnection networks for parallel and distributed computation [169, 170, 172, 168].

The adjacency and Laplacian matrices associated with circulant graphs are circulant, where a particular row's entries are repeated with a shift from one row to the next. As such, the elegant theory of circulant matrices can be used to examine the controllability properties of circulant networks. Such an analysis has a rich history in physics, number theory, analysis, as well as in image processing, and probability [116, 117, 118]; see also [119] and [120].

The built-in periodicity and pattern of circulant matrices provide access to the machinery of Fourier analysis and group theory for their analysis. Circulant matrices constitute a nontrivial set of algebraic objects to study as many of the corresponding matrix-theoretic questions may be resolved in "closed form," providing unprecedented insights into their properties [119]. In this chapter, we used some of these properties to derive explicit results on the controllability of circulant networks.

We examine the controllability Gramian of circulant networks and which set of input nodes generate better controllability features in the sense of the Gramian. We also show for circulant graphs of prime order, input symmetry is the necessary and sufficient condition for uncontrollability. We conjecture this property for general circulant graphs.

We then delve into the controllability of complex networks. Complex dynamic networks are an integral part of the technological world including the internet, power grids, and communication networks, as well as in nature at large such as biological and chemical systems and social networks. An explosion of research in the area of network systems has eventuated [179, 180, 181].

As we briefly mentioned, controllability for Laplacian networks has been established for special classes of graphs such as paths, circulants, grids and distance regular graphs [101, 195, 104, 218], but to our best knowledge, no other large scale networks have been investigated. In [183, 184], the controllability Gramian was used to quantify the effectiveness of control in Laplacian based networks. The authors demonstrated for the end controlled path graph, the smallest eigenvalue of the controllability Gramian rapidly decays rendering networks essentially uncontrollable for large numbers of nodes. This observation motivates the search for control inputs that not only make the network controllable, but realistically controllable.

In this chapter, we also consider network controllability for a large class of graphs, namely networks which are Cartesian products of smaller factor-networks (factors). We present two control schemes, dubbed the control product and layered control, of extending controllability of the factors to controllability of the composite network. The schemes are relevant to dynamics driven by a large class of network based matrices, including the Laplacian and adjacency matrices. The minimum number of required control inputs is addressed

by exploiting the graph automorphisms of the network. The effectiveness of the layered control is investigated by relating properties of the controllability Gramian of the factor graph dynamics to the composite graph.

Cartesian networks are common in many social networks and for this reason are ideal for the theory represented in this chapter. In political networks, for instances, there is a basic structure and membership across all liberal democratic governments with members in the category of state leader, commerce, health, defense, and trade representatives, etc. Social interactions between members vary based on the purpose of the area they represent. These interactions form an intra-government network. In turn, governments of different countries interact to differing degrees. These communication links can also be represented through an inter-government network. Between governments of different countries, representatives with the same responsibilities tend to correspond directly; for example, the Minister of International Trade for Canada corresponds with the Trade Representative for USA. For governments with closer political ties, these links between representative will be more pronounced. Consequently, the network involving representatives from all governments is a Cartesian network with factor networks; the intra-government network and inter-government network. A similar situation occurs with inter and intra-family networks; for example, when two families meet, same gender parents tend to interact more. This family interaction structure is the foundation of an opinion dynamics example analyzed in §8.7 using tools developed in this chapter.

This chapter examines the controllability of a group of first order agents, adopting a weighted consensus-type coordination protocol over a circulant network. Let us review the basic setup and the preliminaries.

5.2 Basic Setup and Preliminaries

Consider the weighted consensus protocol presented in (2.1), where $A(\mathcal{G}) = -L_w(\mathcal{G}) \in \mathbb{R}^{n\times n}$; the input matrix B has the following structure

$$B = [\delta_1, \delta_2, \cdots, \delta_r] \in \mathbb{R}^{n\times r}, \tag{5.1}$$

where $\delta_i = [0, 0, \cdots, \delta_i^j, \cdots, 0]^T$, and $\delta_i^j = 1$ if node j is directly influenced by the i-th external input, and $\delta_i^j = 0$ if otherwise.

We now consider the controllability of the system (1.9) with the aim of characterizing its unique features when $\mathcal{G}$ is circulant. In this direction, a result that provides the springboard for our subsequent analysis is the Popov-Belevitch-Hautus (PBH) test [121], stating that a linear time-invariant system $\dot{x}(t) = Ax(t) + Bu(t)$ is not controllable if and only if there exists $w \neq 0$ such that

$$w^T A = \lambda w^T \qquad \text{and} \qquad w^T B = 0.$$

We will see that this classic result in control theory, in conjunction with properties of circulant matrices, can be used to provide new insights into the controllability of circulant networks. First, we provide a quick overview on the spectral properties of circulant graphs.

Let $\mathcal{G}$ be an undirected circulant graph of degree d with n vertices, with its weighted Laplacian matrix denoted as $L_w(\mathcal{G})$. In this chapter, we assume that the underlying circulant network is connected. Note that if $\mathcal{G}$ is not connected, it is the union of isomorphic circulant components in which case we can apply our results to each component separately. As mentioned in the introduction, the Laplacian matrix of a circulant network has a special circulant structure

$$\begin{bmatrix} c_0 & c_1 & c_2 & c_3 & \cdots & c_{n-1} \\ c_{n-1} & c_0 & c_1 & c_2 & & \vdots \\ c_{n-2} & c_{n-1} & c_0 & c_1 & \ddots & \\ \vdots & \ddots & \ddots & \ddots & & c_2 \\ & & & & & c_1 \\ c_1 & & \cdots & & c_{n-1} & c_0 \end{bmatrix}; \tag{5.2}$$

in particular, a circulant matrix is Toeplitz.

Theorem 5.2.1. *[122] The eigenvalues and eigenvectors of a circulant network are, respectively,*

$$\lambda_m = \sum_{k=0}^{n-1} c_k e^{-2\pi \mathbf{i} mk/n}, \quad \textit{and} \tag{5.3}$$

$$v_m = \frac{1}{\sqrt{n}} \left[1, e^{-2\pi \mathbf{i} m/n}, \ \ldots \ , \ e^{-2\pi \mathbf{i} m(n-1)/n}\right]^T, \quad m = 0, \ \ldots, \ n-1. \tag{5.4}$$

Thus the sequence $\{\lambda_m\}_{m=0}^{n-1}$ is the discrete Fourier transform (DFT) of the sequence $\{c_k\}_{k=0}^{n-1}$. The matrix of normalized eigenvectors of the circulant graph is often called the DFT matrix; we will denote this matrix by U, where its columns are indexed from left to right by m in (5.4). Note that every circulant matrix of the same dimension has the same set of eigenvectors, and their eigenvalues (5.3) are their distinguishing spectral feature. In §III, we show that in the context of network controllability, it is in fact the algebraic multiplicities of these eigenvalues that play a crucial role. Specifically, we show that when the maximum algebraic multiplicity of the eigenvalues of the circulant matrix is q, then there is a subset of q nodes that leads the system (1.9) being controllable. The proof of this theorem utilizes the celebrated Cauchy-Binet formula in conjunction with algebraic properties of Vandermonde matrices both of which are reviewed next.

5.2.1 Cauchy-Binet formula

Let F and H be, respectively, $m \times n$ and $n \times m$ matrices, with $m \leq n$. Let $[n] = \{1, 2, \ldots, n\}$ and

$$\Gamma_m^n = \{m\text{-element subsets of } [n]\} = \{S \subseteq [n] : |S| = m\}.$$

The Cauchy-Binet formula then states that

$$\mathbf{det}(FH) = \sum_{S \in \Gamma_m^n} \mathbf{det}(F_{[m],S})\, \mathbf{det}(H_{S,[m]}), \tag{5.5}$$

where for each $S \in \Gamma_m^n$, $F_{[m],S}$ is the $m \times m$ submatrix of F with column indices in S, and $H_{S,[m]}$ is the $m \times m$ submatrix of H with row indices in S.

5.2.2 Vandermonde structure of DFT matrices

A matrix is called a Vandermonde matrix if it has the structure

$$V = \begin{bmatrix} 1 & \alpha_1 & \alpha_1^2 & \ldots & \alpha_1^{n-1} \\ 1 & \alpha_2 & \alpha_2^2 & \ldots & \alpha_2^{n-1} \\ \ldots & \ldots & \ldots & \ddots & \ldots \\ 1 & \alpha_{n-1} & \alpha_{n-1}^2 & \ldots & \alpha_{n-1}^{n-1} \end{bmatrix}, \quad \alpha_i \in \mathbb{C}; \tag{5.6}$$

it is well known that **det** $V = \prod_{i \neq j}(\alpha_i - \alpha_j)$ [123]. This means that if the elements of the vector $[\alpha_1, \alpha_2, \ldots, \alpha_n]$ are distinct, then the matrix V is full rank. It is clear that the leading principal submatrices of V in (5.6) are Vandermonde and thus nonsingular if V is nonsingular. Note that the DFT matrix U is a nonsingular Vandermonde matrix with the following structure

$$U = [v_0|v_1|\ldots|v_{n-1}] = \frac{1}{\sqrt{n}} \begin{bmatrix} w_n^{0\cdot 0} & w_n^{0\cdot 1} & \ldots & w_n^{0\cdot(n-1)} \\ w_n^{1\cdot 0} & w_n^{1\cdot 1} & \ldots & w_n^{1\cdot(n-1)} \\ \vdots & \vdots & \ddots & \vdots \\ w_n^{(n-1)\cdot 0} & w_n^{(n-1)\cdot 1} & \ldots & w_n^{(n-1)\cdot(n-1)} \end{bmatrix}, \tag{5.7}$$

where $\alpha_m = w_n^m$ and $w_n = e^{-2\pi \mathbf{i}/n}$ for $m = 0, 1, \ldots, n-1$ [134]. In this case, distinctness of the elements of the corresponding vector $[\alpha_1, \alpha_2, \ldots, \alpha_n]$ follows from the following observation: Define

$$\alpha_j = w_n^j = e^{-\frac{2\pi \mathbf{i}}{n} j},$$

for all j. It suffices to show that $\alpha_j \neq \alpha_h$ when $j, h \in \{0, 1, \ldots, n-1\}$. Assume that on the contrary, two indices $0 \leq j, h \leq n-1$ exist such that $\alpha_j/\alpha_h = 1$. Then,

$$\frac{\alpha_j}{\alpha_h} = e^{-\frac{2\pi \mathbf{i}}{n}(j-h)} = 1,$$

implying that $2\pi(j-h)/n = 2\pi k'$ for some $k' \in \mathbb{Z}$. This would in turn imply that $(j-h)/n = k' \in \mathbb{Z}$, which is a contradiction as $0 \leq j, h \leq n-1$, and the fraction $(j-h)/n$ cannot be an integer.

One of the key steps in the proof of our main result pertains to the non-singularity of submatrices of a Vandermonde matrix, referred to as generalized Vandermonde matrices. The determinants of generalized Vandermonde matrices are connected to the symmetric group and have been well-studied in the literature [125, 126, 127, 128, 129, 130].

Lemma 5.2.2. *Consider the generalized Vandermonde matrix* $\widetilde{V} = [\alpha_i^{m_j}]_{i,j=1}^{q}$*, where* $\alpha_i \in \mathbb{C}$ *and the integer set* m_j *is such that for all* $j \in \{1, 2, \ldots, q-1\}$*, the difference* $m_{j+1} - m_j$ *is a fixed integer* k*. Then, the generalized Vandermonde matrix is nonsingular if the* α_i^k*s are distinct.*

Proof. Consider the generalized Vandermonde matrix

$$\widetilde{V} = [\alpha_i^{m_j}]_{i,j=1}^{q} = \begin{bmatrix} \alpha_1^{m_1} & \alpha_1^{m_2} & \alpha_1^{m_3} & \dots & \alpha_1^{m_q} \\ \alpha_2^{m_1} & \alpha_2^{m_2} & \alpha_2^{m_3} & \dots & \alpha_2^{m_q} \\ \dots & \dots & \dots & \ddots & \dots \\ \alpha_q^{m_1} & \alpha_q^{m_2} & \alpha_q^{m_3} & \dots & \alpha_q^{m_q} \end{bmatrix};$$

we observe that

$$\mathbf{det}(\widetilde{V}) = \mathbf{det}([\alpha_i^{m_j}]_{i,j=1}^{q}) = \prod_{i=1}^{q} \alpha_i^{m_1} \ \ \mathbf{det} \begin{bmatrix} 1 & \alpha_1^{m_2-m_1} & \alpha_1^{m_3-m_1} & \dots & \alpha_1^{m_q-m_1} \\ 1 & \alpha_2^{m_2-m_1} & \alpha_2^{m_3-m_1} & \dots & \alpha_2^{m_q-m_1} \\ \dots & \dots & \dots & \ddots & \dots \\ 1 & \alpha_q^{m_2-m_1} & \alpha_q^{m_3-m_1} & \dots & \alpha_q^{m_q-m_1} \end{bmatrix}.$$

Since the sequence of integers m_j satisfies $m_{j+1} - m_j = k$, for all $j \in \{1, 2, \dots, q-1\}$ and a fixed integer k, it follows that

$$\mathbf{det}(\widetilde{V}) = \prod_{i=1}^{q} \alpha_i^{m_1} \ \ \mathbf{det} \begin{bmatrix} 1 & \beta_1 & \beta_1^2 & \dots & \beta_1^{(q-1)} \\ 1 & \beta_2 & \beta_2^2 & \dots & \beta_2^{(q-1)} \\ \dots & \dots & \dots & \ddots & \dots \\ 1 & \beta_q & \beta_q^2 & \dots & \beta_q^{(q-1)} \end{bmatrix},$$

where $\beta_i = \alpha_i^k$. The last matrix above has a Vandermonde structure; therefore, it is nonsingular if the β_is are distinct, thus proving the statement of Lemma 5.2.2. □

Recall that a symmetric polynomial in several variables is an invariant polynomial under interchanging of any two of its variables.

Theorem 5.2.3. *[132, 133] Let $\widetilde{V} = [\alpha_i^{m_j}]_{i,j=1}^{q}$, with $\alpha_i \in \mathbb{C}$ and $m_0 < m_1 < \ldots < m_q$, be a generalized Vandermonde matrix of dimension $q \times q$. Then, its determinant can be factorized as*

$$\mathbf{det}\ \widetilde{V} = \left(\prod_{i>j} (\alpha_i - \alpha_j) \right) S(x), \tag{5.8}$$

where $S(x) = \sum_k g_k x_0^{p_{k,0}} \dots x_{n-1}^{p_{k,n-1}}$ is a symmetric polynomial in $x_0, \dots, x_{n-1}$. The symmetric polynomial $S(x)$ is often called the Schur function.

The relation (5.8) implies that the zero eigenvalues of the generalized Vandermonde $\widetilde{V}$ are the non-trivial roots of the Schur function. Delvaux *et al.* [133] have used this identity to prove that for $\alpha_j = e^{-\frac{2\pi \mathbf{i} j}{n}}$, with n prime in (5.6), the Vandermonde matrix does not contain any singular submatrix.[3] Such a Vandermonde matrix is often called a Fourier matrix. Thus for n prime, the DFT matrix U (5.7) is in fact a Fourier matrix with no singular submatrix, an observation that we will use in the context of network controllability of prime ordered circulant matrices.

5.3 Controllability of Circulant Networks

It is known that for a linear time-invariant system, if the pair (A, B) is controllable with **rank** $(B) = q$, then the geometric multiplicity of each eigenvalue of A can be at most q [124]. The main result of the chapter can be interpreted as a strengthened version of the above result in the context of circulant networks. The result is as follows.

Theorem 5.3.1. *A circulant network with maximum algebraic multiplicity q is controllable from q nodes. Moreover,*

(a) when the network is of prime order, the set of q nodes can be chosen arbitrarily, and

(b) when the network has an arbitrary order, the indices of the q nodes, in the order of clockwise or counterclockwise indexing of the vertices, can be chosen as an arithmetic progression of length q with the common difference of 1.

Our approach to proving Theorem 5.3.1 starts with the PBH test, which suggests examining the determinants of submatrices of the DFT matrix associated with circulant networks. This is followed by the application of the Cauchy-Binet formula, and subsequently, utilizing the Vandermonde structure of the set of submatrices as dictated by Cauchy-Binet. A key ingredient in our proof is the following observation.

---

[3]The original result in [133] is for the case when $\alpha_j = e^{\frac{2\pi \mathbf{i} j}{n}}$. The result for n prime can be extended to the case when $\alpha_j = e^{-\frac{2\pi \mathbf{i} j}{n}}$, since the latter is a Vandermonde matrix and is the inverse of the one discussed in [133].

Proposition 5.3.2. *Let Q be the $n \times q$ submatrix of the DFT matrix (5.7) consisting of eigenvectors of the circulant matrix associated with an eigenvalue of maximum algebraic multiplicity q,*

$$Q = \left[v_{m_1}|v_{m_2}|\dots|v_{m_q}\right] = \frac{1}{\sqrt{n}} \begin{bmatrix} w_n^{0\cdot m_1} & w_n^{0\cdot m_2} & \dots & w_n^{0\cdot m_q} \\ w_n^{1\cdot m_1} & w_n^{1\cdot m_2} & \dots & w_n^{1\cdot m_q} \\ \vdots & \vdots & \ddots & \vdots \\ w_n^{(n-1)\cdot m_1} & w_n^{(n-1)\cdot m_2} & \dots & w_n^{(n-1)\cdot m_q} \end{bmatrix}. \tag{5.9}$$

Then Q contains a $q \times q$ nonsingular submatrix $Q_{[a],[q]}$ for some $[a] \in \Gamma_q^n$ that is independent of the choice of the underlying eigenvalue. Moreover, when U is of prime order, such an index set $[a] \in \Gamma_q^n$ can be chosen arbitrarily.

Proof. Let us define $\gamma_k = w_n^{m_k}$ where $k = 1, 2, \dots, q$. Then,

$$Q = \frac{1}{\sqrt{n}} \begin{bmatrix} \gamma_1^0 & \gamma_2^0 & \dots & \gamma_q^0 \\ \gamma_1^1 & \gamma_2^1 & \dots & \gamma_q^1 \\ \vdots & \vdots & \ddots & \vdots \\ \gamma_1^{(n-1)} & \gamma_2^{(n-1)} & \dots & \gamma_q^{(n-1)} \end{bmatrix}. \tag{5.10}$$

Following the set up of Lemma 5.2.2, the structure of Q (5.10) guarantees that any of its $q \times q$ submatrices, say $\widetilde{Q} = \left[\gamma_h^{p_j}\right]_{h,j=1}^{q}$, with $p_j \in \{1, 2, \dots, n-1\}$ and $p_{j+1} - p_j = \tilde{k} \in \mathbb{Z}$ for $j = 1, 2, \dots, q-1$, is nonsingular if the $\gamma_h^{\tilde{k}}$s are distinct. If $\gamma_h^{\tilde{k}}$s are not distinct, for some h and l,

$$e^{-2\pi \mathbf{i} \tilde{k} \frac{m_h}{n}} = e^{-2\pi \mathbf{i} \tilde{k} \frac{m_l}{n}} \quad \text{and} \quad \frac{\tilde{k}}{n}(m_h - m_l) \in \mathbb{Z}, \tag{5.11}$$

where $\tilde{k}, m_h, m_l \in \{1, 2, \dots, n-1\}$. From (5.11), it then follows that for n prime, the equation is not satisfied by any m_h and m_l, consistent with the result in [133] as previously mentioned.

We now observe that no two indices m_h and m_l satisfy (5.11) for $\tilde{k} = 1$ since $m_h - m_l \in \{1, \dots, n-1\}$, and the fraction $(m_h - m_l)/n$ cannot be an integer. Thus the index set $[a]$ can be chosen as an arithmetic progression of length q with common difference 1. Now consider the eigenvector matrix Q' associated with another eigenvalue λ' of the circulant

matrix with maximum algeraic multiplicity q,

$$Q' = \left[v_{m'_1} | v_{m'_2} | \dots | v_{m'_q} \right].$$

In this case, the reasoning above leading to (5.11) once again implies that no two indices m'_h and m'_l in $\{1, \dots, n-1\}$ satisfy

$$\frac{\tilde{k}}{n}(m'_h - m'_l) \in \mathbb{Z},$$

when for example $\tilde{k} = 1$. Thus, the submatrix $Q'_{[a][q]}$ is also nonsingular. □

We are now ready to prove Theorem 5.3.1.

Proof of Theorem 5.3.1. From the PBH test, the pair (A, B) is not controllable if and only if there exist $w \neq 0$ and $\lambda \in \mathbb{C}$ such that

$$w^T A(\mathcal{G}) = \lambda w^T \qquad \text{and} \qquad w^T B = 0.$$

Suppose that such a vector w does in fact exist. This vector would then be the left eigenvector of the matrix $A(\mathcal{G})$. Since $A(\mathcal{G})$ is symmetric, for each eigenvalue λ with algebraic multiplicity q', the q' eigenvectors $v_{i_1}, v_{i_2}, \dots, v_{i_{q'}}$ span the eigenspace associated with λ, where v_{i_j}'s are the corresponding columns of the DFT matrix. In this case, any eigenvector of $A(\mathcal{G})$ associated with this eigenvalue can be written as $w = \sum_{j=1}^{q'} \alpha_j v_{i_j}$. The PBH test thus implies that for some eigenvalue λ with algebraic multiplicity q',

$$(\sum_{j=1}^{q'} \alpha_j v_{i_j})^T B = (Q\alpha)^T B = \alpha^T (Q^T B) = 0,$$

where the columns of Q consist of the eigenvectors v_{i_j} $(j = 1, \cdots, q')$ corresponding to λ, and $\alpha = [\alpha_1, \cdots, \alpha_{q'}]^T$. This implies that $\mathcal{N}(B^T Q)$ is non-empty, or that equivalently, $\mathcal{R}(Q^T B)$ is a proper subspace of $\mathbb{R}^{q'}$. In the following, we consider two distinct cases: (1) the matrix Q above consists of eigenvectors corresponding to an eigenvalue (not necessary unique) with maximum algebraic multiplicity q, and (2) the matrix Q consists of eigenvectors corresponding to an eigenvalue with algebraic multiplicity less than q.

For the first case, where the input matrix B has q columns of the form (5.1) and λ has the maximum algebraic multiplicity q, having $Q^T B$ not span $\mathbb{R}^q$ implies that $\mathbf{det}(Q^T B) = 0$.

In the following, we will show that there exists a set of input nodes which results in having $\mathbf{det}(Q^T B) \neq 0$, contradicting the hypothesis that there is an eigenvector w orthogonal to the matrix B obtained from *any* q input nodes. In this direction, applying the Cauchy-Binet formula implies that

$$\mathbf{det}(Q^T B) = \sum_{S \in \Gamma_q^n} \mathbf{det}(Q^T_{[q],S}) \, \mathbf{det}(B_{S,[q]}). \tag{5.12}$$

The matrix B has a special structure (5.1), namely, $\mathbf{det}(B_{S,[q]}) = 0$ for all $S \in \Gamma_q^n$ except for the set of indices consisting of indices of the input nodes, i.e., when $S = \mathcal{I}$, in which case $\mathbf{det}(B_{\mathcal{I},[q]}) = 1$. Therefore,

$$\mathbf{det}(Q^T B) = \mathbf{det}(Q^T_{[q],\mathcal{I}}) \, \mathbf{det}(B_{\mathcal{I},[q]}) = \mathbf{det}(Q^T_{[q],\mathcal{I}}).$$

Now, in order to utilize Proposition 5.3.2, recall that a proper labeling of the circulant graph is such that the vertices are first considered to be on a circle and subsequently indexed sequentially clockwise or counterclockwise; see Fig. 5.1. Therefore, within the context of such a proper labeling, Proposition 5.3.2 implies that if the set of q input nodes is chosen as an arithmetic progression of length q with common difference 1, then independent of the choice of the underlying eigenvalue, one has $\mathbf{det}(Q^T_{[q],\mathcal{I}}) \neq 0$; moreover, for a circulant network of prime order, such a subset can be chosen arbitrarily. Therefore, $\mathbf{det}(Q^T B)$ defined in (5.12) is nonzero for these input sets, contradicting the assumption that $\mathbf{det}(Q^T B) = 0$ for all q inputs of the form (5.1) (with $r = q$).

In order to show that the network is in fact controllable from q nodes chosen as an arithmetic progression of length q with common difference 1, it remains to be shown that the corresponding matrix B is not orthogonal to *any other* eigenvector of the circulant network associated with eigenvalues with multiplicity $q' < q$. Let us denote the matrix consisting of such eigenvectors corresponding to eigenvalue λ' by $U_{\lambda'}$, i.e.,

$$U_{\lambda'} = \frac{1}{\sqrt{n}} \begin{bmatrix} \gamma_1^0 & \gamma_2^0 & \cdots & \gamma_{q'}^0 \\ \gamma_1^1 & \gamma_2^1 & \cdots & \gamma_{q'}^1 \\ \vdots & \vdots & \ddots & \vdots \\ \gamma_1^{(n-1)} & \gamma_2^{(n-1)} & \cdots & \gamma_{q'}^{(n-1)} \end{bmatrix}, \tag{5.13}$$

for some γ_js. Consider the eigenvector $w \in$ **span**$\{U_{\lambda'}\}$, i.e., $w = U_{\lambda'}\eta$ for some nonzero $\eta \in \mathbb{R}^{q'}$. If B is in fact orthogonal to w, then

$$w^T B = \eta^T U_{\lambda'}^T B = 0. \tag{5.14}$$

This would then imply that $B^T U_{\lambda'}\eta = 0$ for $\eta \neq 0$, and hence **rank** $B^T U_{\lambda'} < q'$. In the meantime, $B^T U_{\lambda'}$ consists of q consecutive rows of $U_{\lambda'}$, and therefore has a $q' \times q'$ submatrix with distinct entries and a Vandermonde structure. Hence, $B^T U_{\lambda'}$ has rank q', thus establishing a contradiction.

□

It is instructive to note that the proof of Theorem 5.3.1 provides a constructive procedure for selecting the input nodes for a controllable circulant network. This result generalizes the result of [102], where it has been shown for example that cycles–a subclass of circulant networks- with prime order can be controlled from any pair of nodes.

In the following, we introduce an application of Circulant networks in distributed computing and the ramification of the proposed analysis described in the dissertation in the context of fault detection and clock synchronization.

5.4 Fault Detection and Clock Synchronization over Distributed Computing Supercomputers

Circulant graphs appear in coding theory, VLSI design, Ramsey theory, and other areas. They constitute the basis for designing certain data alignment networks for complex memory systems [164]. Another application of Circulant graphs is in quantum communication scenarios. In general, quantum spin system can be defined as a collection of qubits on a graph, whose dynamics is governed by a suitable Hamiltonian, without external control on the system. In quantum communication scenario, circulant graphs is used in the problem of arranging N interacting qubits in a quantum spin network based on a circulant topology to obtain good communication between them [167].

Several optimizations related to the diameter minimization of degree four circulant graphs enhance their applicability to the design of efficient interconnection networks [165,

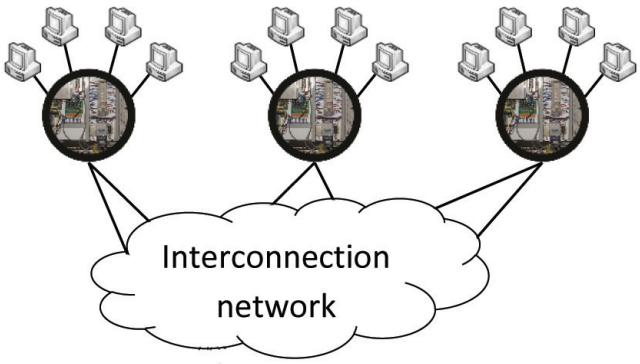
166]. Circulant graphs, characterized by high scalability and modularity, optimal fault-tolerance, and routing capabilities, represent an important class of reliable interconnection networks for the multiprocessor networks in parallel and distributed computing (MPP, Intel Paragon, Cray T3D, etc.) [167, 168, 169, 170].

In commodity parallel computing, as a method of parallel computing, a user is confronted with a machine built with mass produced fully equipped processors interconnected through a high speed, low latency network. A class of circulant graphs of degree four, with minimal topological distances, was presented in [171, 172, 173] as a basis to cluster those processing units (PU). The processing power of such a supercomputer is characterized by the number N of PUs and the number M of processors per PU. Figure 5.2(a) demonstrate the main concept in the commodity parallel computing.

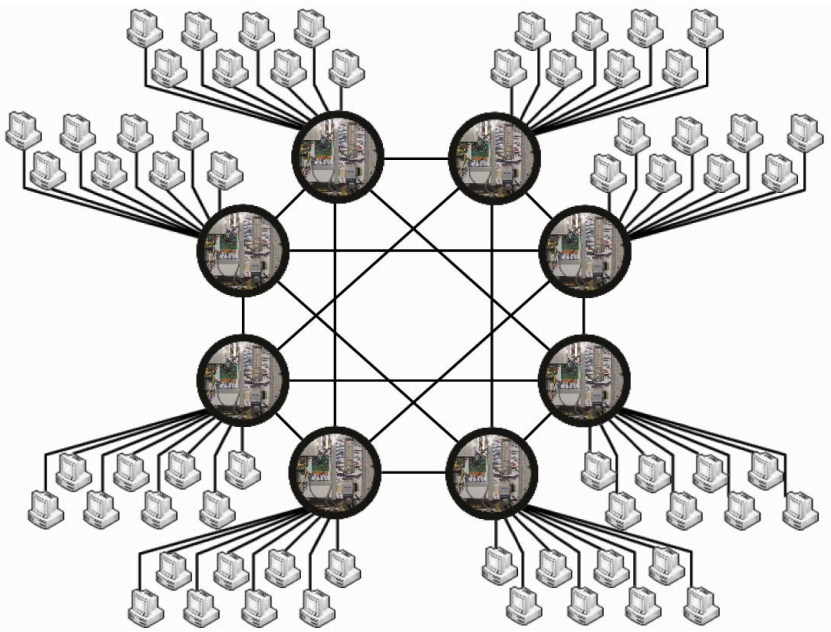
As an example of the commodity parallel computing, Figure 5.2(b) depicts the interconnection network of the Swiss-T1 cluster supercomputer. The cluster consists of 8 PUs and 4 Alpha 21264 dual-processors per PU [173, 174]. The cluster thus comprises a total of 64 computing processors. The interconnection network between the PUs is a Circulant graph of degree 4 and diameter 2. The network enables transmissions of large messages at low latencies.

Challenges in parallel and distributed computing include, and not limited to, fault detection, consensus problem, and clock synchronization over these type of large and complex systems [175]. Security and privacy preserving in parallel and distributed computing systems, e.g. in bioinformatics grids/cloud, is one of the key challenges that attract the attention of many researchers. The analysis in the current paper can be adopted to detect faults or guarantee the security of the system by observing the behavior of the observable nodes as the system is running a consensus protocol in the background. Consensus on clock time is another ramification of the proposed analysis in §5.3. Chapman et al. in [176] studied a consensus based clock synchronization algorithm in which friendly control PUs periodically connect to the network and deliver a constant correction to correct the absolute clock for the network.

Gossiping algorithms are extensively used in parallel and distributed computing [168, 177]. Gossip algorithms over a network of N agents can be summarized by $x(t+1) =$



(a)



(b)

Figure 5.2: Commodity parallel computing (a) The general representation of a commodity parallel computer [173], (b) The interconnection network of the Swiss-T1 cluster supercomputer; courtesy of `http://switzernet.com/`

$W(t)x(t)$ where $W(t)$ is a double stochastic matrix and $x(t)$ is an $N \times 1$ vector of gossiping states. The gossip dynamics can be interpreted as the discrete version of the dynamics presented in (1.9). The analysis done in the chapter can also be utilized to deal with

problems in the landscape of gossiping over distributed supercomputers; problems such as fault detection by monitoring the dynamics of observable nodes.

Chapter 6

CONTROLLABILITY GRAMIAN, SYMMETRY STRUCTURES, AND APPLICATION OF CIRCULANT NETWORKS

The previous chapter considered a group of first order agents, adopting a weighted consensus-type coordination protocol over a Circulant network. The chapter examined the controllability properties of a Circulant network. The symmetry structure of the underlying network identified as an important feature of the system that affects the controllability properties of the network. This chapter examines the controllability Gramian of the Circulant networks and delves into the necessary and sufficient controllability conditions of Circulant networks.

6.1 Introduction

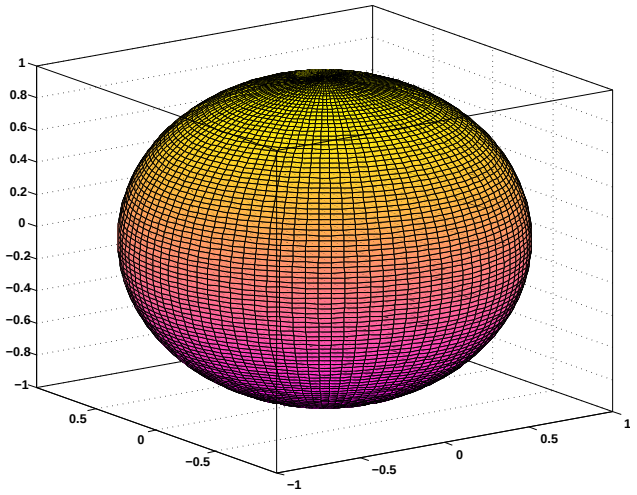
Consider the LTI system described in (2.1). The set of states reachable with an input u with norm $||u|| \leq 1$ is

$$\begin{aligned} E_c &= \{\mathcal{C}\, u\,;\, ||u|| \leq 1\} \\ &= \{\zeta \in \mathbb{R}^n \text{ and } \|\zeta\| \leq 1;\, \zeta^T P^{-1} \zeta\} \end{aligned} \tag{6.1}$$

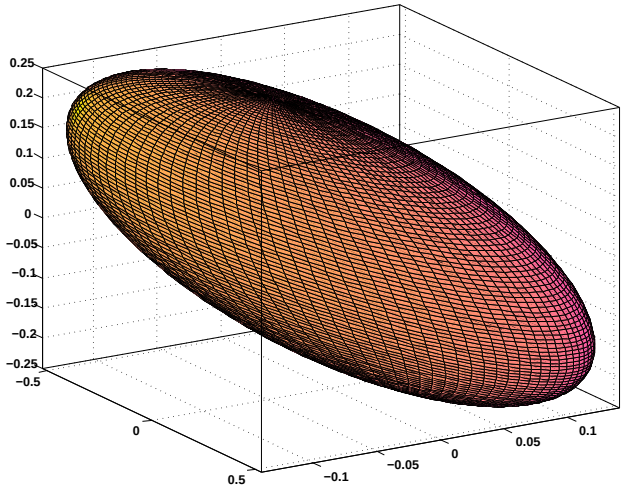
where $\mathcal{C}$ is the controllability matrix and $P = \mathcal{C}\mathcal{C}'$ is the controllability Gramian. Figure 6.1 depicts the input signal in a unit-diameter sphere and the associated controllability Gramian ellipsoid.

The length of the semi-axis of the controllability ellipsoid E_c is $\sqrt{\lambda_i}$, where λ_is are the eigenvalues of the controllability Gramian P. The semiaxis directions v_is are the eigenvectors of P. Directions v_i corresponding to large λ_i are strongly controllable while directions corresponding to small λ_i are weakly controllable. The energy required to drive the final state to $x \in \mathbb{R}^n$ is $||u_{\text{opt}}|| = x^T P^{-1} x$.

In the following, we examine the strength of the controllability of circulant graphs via their controllability Gramian.



(a)



(b)

Figure 6.1: (a) The input signal u with $||u|| \leq 1$. (b) The controllability ellipsoid for an LTI system with $A = [-2\ 1\ 0; 1\ -2\ 1; -1\ 1\ -1]$ and $B = [1\,0\,0]^T$

6.2 Controllability Gramian: the Most Controllable Node in a Circulant Network

Recall that the eigenvalues and eigenvectors of the Laplacian of Circulant network can be written as follows: where n is odd, the eigenvalues of $\mathcal{G}$, not necessarily ordered, are as

follows:

$$\lambda_0 = \sum_{k=1}^{(n-1)/2} 2c_k, \quad \lambda_j = \lambda_{n-j} = \sum_{k=1}^{(n-1)/2} 2c_k \cos\left(\frac{2jk\pi}{n}\right) \quad 1 \le j \le (n-1)/2. \tag{6.2}$$

If n is even, then

$$\lambda_0 = c_{n/2} + \sum_{k=1}^{n/2-1} 2c_k, \quad \lambda_j = \lambda_{n-j} = c_{n/2}\cos(j\pi) + \sum_{k=1}^{n/2-1} 2c_k \cos\left(\frac{2jk\pi}{n}\right) \quad 1 \le j \le n/2. \tag{6.3}$$

Note that while we do not assume that the eigenvalues are ordered, λ_0 is always the largest eigenvalue, and thus it is the spectral radius of the Laplacian.

For the rest of this chapter we fix the choice of eigenvectors. The eigenvector related to $\lambda_0 = d$ will be $u_0 = [1\ 1 \dots\ 1]^T$, and the eigenvectors related to λ_j and λ_{n-j} will be given by

$$\begin{aligned} u_j &= \left[1\ \cos(j\frac{2\pi}{n})\ \cos(2j\frac{2\pi}{n})\ \cos(3j\frac{2\pi}{n})\ \dots\ \cos((n-1)j\frac{2\pi}{n})\right]^T \\ v_j &= \left[0\ \sin(j\frac{2\pi}{n})\ \sin(2j\frac{2\pi}{n})\ \sin(3j\frac{2\pi}{n})\ \dots\ \sin((n-1)j\frac{2\pi}{n})\right]^T. \end{aligned} \tag{6.4}$$

If n is even, the eigenvector of $\lambda_{\frac{n}{2}}$ will be $u_{\frac{n}{2}} = [1\ -1\ 1\ -1\ \dots\ 1\ -1]^T$ (and $v_{\frac{n}{2}} = \mathbf{0}$).

The controllability Gramian, P, is a positive semidefinite matrix satisfying the Lyapunov equation

$$AP + PA = Q, \tag{6.5}$$

where $Q = BB^T$, and A and B are defined in (2.1). The closed form solution of the controllability Gramian P can be found as follows for $A = -L$

$$P = \int_0^\infty e^{-Lt} BB^T e^{-Lt} dt. \tag{6.6}$$

The first question of interest is how P changes as the input set $\mathcal{I}$ changes. Let $\mathcal{I}$ and $\tilde{\mathcal{I}}$ be two different sets of input nodes satisfying the condition in Theorem 5.3.1. Consider Π to be the permutation between $\mathcal{I}_1$ and $\tilde{\mathcal{I}}$ and J to be the associated permutation matrix such that $JL = LJ$ and $J^T B = \tilde{B}$ where B and $\tilde{B}$ are the associated input matrices. Let us

multiply P in (6.6) by J^T and J from left and right, respectively. Then

$$\begin{aligned} J^T P(L,B) J &= \int_0^\infty J^T e^{-Lt} B B^T e^{-Lt} J dt \\ &= \int_0^\infty e^{-Lt} J^T B B^T J e^{-Lt} dt = \int_0^\infty e^{-Lt} \tilde{B} \tilde{B}^T e^{-Lt} dt \\ &= P(L,\tilde{B}). \end{aligned} \tag{6.7}$$

The second equality is a consequence of the fact that J and L commute; therefore, J and e^{-L} commute. The equality (6.7) implies that there is a unitary transformation between $P(L,B)$ and $P(L,\tilde{B})$. Therefore, properties of the unitary transformation verify that changing the set of input nodes while the network stays the same does not alter the spectral properties of the controllability Gramian.

The following results provide bounds on both the eigenvalues and the trace of the controllability Gramian where $A = -L(\mathcal{G})$ of a Circulant graph of degree r.

Proposition 6.2.1. *Let P to be the controllability Gramian of the controllable pair (A,B), where $A = -L(\mathcal{G})$, associated with a Circulant graph on n nodes with degree equal to r and maximum algebraic multiplicity equal to q, and where B is an input matrix with rank q. Then, the following inequalities hold*

$$\mathbf{trace}(P) \geq \frac{q}{2nr} \quad \textit{and} \quad \lambda_{\max}(P) \geq \frac{1}{2\lambda_{\max}(L(\mathcal{G}))}.$$

Proof. The following upper and lower bounds for the trace of the controllability Gramian $P(A,B)$, as a function of the trace of A and B, have been shown in Wang et al. [135] to be

$$-\mathbf{trace}(P) \geq \min\{-\frac{\mathbf{trace}(BB^T)}{2\lambda_{\min}(A)},\ -\frac{\mathbf{trace}(BB^T)}{2\mathbf{trace}(A)}\}$$

which gives

$$\mathbf{trace}(P) \geq \frac{\mathbf{trace}(BB^T)}{2\mathbf{trace}(L(\mathcal{G}))}.$$

Then, the first inequalities follow since $\mathbf{trace}(BB^T) = q$ as the number of input nodes and $\mathbf{trace}(L(\mathcal{G})) = nr$ where n is the number of nodes, and r is the degree of each node.

The following upper bound is also provided for the maximum eigenvalues of $P(A,B)$ in [136]:

$$\lambda_{\max}(P(A,B)) \geq \frac{\lambda_{\max}(BB^T)}{2\lambda_{\max}^{1/2}(A^T A)}. \tag{6.8}$$

Then, it follows that as $\lambda_{\max}(BB^T) = 1$ then,

$$\lambda_{\max}(P) \quad \geq \quad \frac{1}{2\lambda_{\max}(L(\mathcal{G}))}.$$

□

The next theorem characterizes the closed form solution of the controllability Gramain for Circulant graphs.

Theorem 6.2.2. *Consider the unitary transformation of matrix A as $A = U\Lambda U^T$ where the unitary matrix U and $\Lambda = diag[-\lambda_i]$ and λ_i is the i^{th} eigenvalue of matrix L defined in (6.2) and (6.3). The controllability Gramian P in (6.5) satisfies $P = U\tilde{P}U^T$ and $\tilde{P}_{ij} = \frac{u_i^T w_j}{\lambda_i + \lambda_j}$ where u_i is the i^{th} column of matrix U and*

$$w_j(k) = \begin{cases} u_j(k) & k \notin \mathcal{I} \\ 0 & k \in \mathcal{I}. \end{cases}$$

Proof. Since matrix A is symmetric, it is unitarily diagonizable. Consider the Lyapunov equation in (6.5) and substitute $A = U\Lambda U^T$ and $P = U\tilde{P}U^T$. Then

$$\begin{aligned} Q &= AP + PA \\ &= U\Lambda U^T U\tilde{P}U^T + U\tilde{P}U^T U\Lambda U^T \\ &= U\Lambda\tilde{P}U^T + U\tilde{P}\Lambda U^T = U\left(\Lambda\tilde{P} + \tilde{P}\Lambda\right)U^T. \end{aligned}$$

Therefore, by multiplying left and right sides of the last equality by U^T and U, respectively, and considering the fact that $UU^T = U^T U = I$, we obtain that

$$U^T Q U = \Lambda\tilde{P} + \tilde{P}\Lambda. \tag{6.9}$$

Now consider the matrix structure of U, $Q = BB^T$, Λ and $\tilde{P}$ to calculate the $\{ij\}^{\text{th}}$ entry of both sides of (6.9) as follows

$$\begin{aligned} \left(\Lambda\tilde{P} + \tilde{P}\Lambda\right)_{ij} &= (\lambda_i + \lambda_j)\tilde{P}_{ij} \\ \left(U^T Q U\right)_{ij} &= u_i^T w_j \end{aligned} \tag{6.10}$$

where u_i is the i^{th} column of matrix U and w_j is the j^{th} column of QU which is defined as

$$w_j(k) = \begin{cases} u_j(k) & k \notin \mathcal{I} \\ 0 & k \in \mathcal{I}. \end{cases}$$

Consequently, the claim is proved. □

Next step calculates the maximum and minimum eigenvalues of P or $\tilde{P}$

6.3 Symmetry Structures

Consider the following dynamics

$$\begin{aligned} \dot{x}_f(t) &= -A_f x_f(t) - B_f u(t), \\ y(t) &= -B_f^T x_f(t) \end{aligned} \tag{6.11}$$

and also consider

$$\begin{aligned} \dot{x}(t) &= -L(\mathcal{G})x(t) + Bu(t), \\ y(t) &= B^T x(t). \end{aligned} \tag{6.12}$$

In [105] it is shown that there is an intricate relationship between the controllability of (6.11) and the symmetry structure of the graph as captured by its *automorphism group.* To explore this connection for Circulant graphs with dynamics in (6.12), we start with some preliminary definitions.

Definition 6.3.1. *The system (6.12) is input symmetric with respect to the input node if there exists a nonidentity permutation J such that*

$$JL(\mathcal{G}) = L(\mathcal{G})J \quad \textit{and} \quad JB = J^T B = B. \tag{6.13}$$

We call the system asymmetric if it does not admit such a permutation for any choice of input node.

Theorem 6.3.2. *The system (6.12) is uncontrollable if it is input symmetric. Equivalently, the system (6.12) is uncontrollable if the floating graph admits a nonidentity automorphism for which the input indicator vector remains invariant under its action.*

Proof. If the system is input symmetric then there is a nonidentity permutation J such that (6.13) is satisfied. Consider an eigenvalue λ and associated eigenvector v which satisfy $L(\mathcal{G})v = \lambda v$. Therefore, for all eigenvalue-eigenvector pairs (λ, v), one has $JL(\mathcal{G})v = J\lambda v$. Using (6.13) however,

$$L(\mathcal{G})(Jv) = \lambda(Jv),$$

and as a result Jv is also an eigenvector of $L(\mathcal{G})$ corresponding to the eigenvalue λ. Note that $v \neq Jv$ is also an eigenvector of $L(\mathcal{G})$. Therefore,

$$(v - Jv)^T B = v^T B - v^T J^T B = v^T B - v^T B = 0$$

which means one of the eigenvectors of $L(\mathcal{G})$, namely, $v - Jv$, is orthogonal to B. Therefore, the PBH test implies that (6.12) is uncontrollable. □

We have already shown that Circulant graphs are controllable from q nodes where q is the maximum algebraic multiplicity of the associated Laplacian. Therefore, Theorem 6.3.2 confirms that the system is input asymmetric. The goal is to explore whether the input symmetry is a necessary condition for the uncontrollability of Circulant graphs. The next theorem shows for n even, input symmetry is not the necessary condition. Then next we prove that for n odd, the input symmetry is a necessary and sufficient condition for the uncontrollability of the system (6.12).

Theorem 6.3.3. *The system (6.11) is controllable if and only if it is input asymmetric.*

Proof. Theorem 6.3.2 implies that a controllable (6.12) is input asymmetric. The next step is to show that if it is input asymmetric, it is controllable; or equivalently if it is uncontrollable, then it is input symmetric. Assume (6.12) is uncontrollable. To show it is input symmetric, it is enough to show there is a permutation matrix J that satisfies (6.13). Let us start with a simple case where the maximum algebraic multiplicity is equal to 2, e.g., $q = 2$. In this case, the only uncontrollable scenario is the single input case. Without loss of generality assume node number one, denoted by v_1, is the input. Let us consider a permutation P that fixes the input node v_1 to satisfy $PB = B$. By switching the nodes on the right and left sides of v_1 while they have the same distance from the node v_1, a

nonidentity permutation P is obtained. The permutation preserves the edges of the original graph. Permutation P has the following structure

$$P = \begin{bmatrix} 1 & 0 & 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & 0 & 0 & \dots & 0 & 1 \\ 0 & 0 & 0 & 0 & \dots & 1 & 0 \\ \vdots & \vdots & \vdots & \vdots & \dots & \vdots & \vdots \\ 0 & 0 & 0 & 1 & \dots & 0 & 0 \\ 0 & 0 & 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & 0 & 0 & \dots & 0 & 0 \end{bmatrix}.$$

In the next step, we consider circulant graphs with prime number of nodes. The following proposition proves that the maximum algebraic multiplicity of a circulant graph of prime order p is either 2 or $p-1$ in case of complete graph. Therefore, uncontrollable non-complete circulant graphs of prime order are input symmetric.

Proposition 6.3.4. *The maximum algebraic multiplicity of a prime order circulant graph is* 2 *or* $p-1$ *in case of a complete graph.*

Proof. The spectrum of a connected complete graph of order p includes one zero and $p-1$ identical numbers. Therefore, the maximum algebraic multiplicity of a connected complete graph of order p is $p-1$. The next claim is that the maximum algebraic multiplicity of a non-complete circulant graph is 2. To prove this claim consider the closed form eigenvalues of circulant graphs in Theorem 5.2.1. Consider two identical eigenvalues λ_{m_1} and λ_{m_2}. Therefore,

$$\begin{aligned} \lambda_{m_1} &= \sum_{k=0}^{p-1} c_k e^{\frac{-2\pi i m_1 k}{p}} \\ &= \sum_{k=0}^{p-1} c_k e^{\frac{-2\pi i m_2 k}{p}} = \lambda_{m_2}. \end{aligned} \tag{6.14}$$

Since we consider undirected graphs, the imaginary parts of eigenvalues in (6.14) vanish.

Then the solutions of (6.14) are as follows

$$\exists\ k'' \in \mathbb{Z} \quad \text{s.t.} \quad \forall k, k' \in [0, n-1]$$

$$2\pi\frac{m_1 k}{p} = 2\pi\frac{m_2 k'}{p} + \pi k'' \tag{6.15}$$

$$2\pi\frac{m_1 k}{p} = 2\pi\frac{m_2 k'}{p} + 2\pi k'' \tag{6.16}$$

The equality in (6.15) is equivalent to $2(m_1 k - m_2 k') = p$ which contradicts the fact that p is prime. Since $k, k', m_1, m_2 \in [0, p-1]$ and p is prime, the only solution of (6.16) is $k = k'$ and $m_1 = p - m_2$. Therefore, $\cos(\frac{2\pi m_1 k}{p}) = \cos(\frac{2\pi(p-m_2)k}{p}) = \cos(\frac{2\pi m_2 k}{p})$. This proves the claim that the maximum algebraic multiplicity of non-complete circulant graphs is 2. □

Proposition 6.3.5. *The algebraic multiplicities of a circulant graph of order p^m for any prime p and integer m are $\{2, p-1, p^2-1, \ldots, p_m - 1\}$. The algebraic multiplicities of a circulant graph of order pq for any two distinct prime numbers p and q where $p > q$ are $\{2, p-1, pq-1\}$.*

□

Chapter 7

CONTROLLABILITY AND OBSERVABILITY OF PATH NETWORKS

In previous chapters, we presented an approach to validate and characterize the controllability properties of Circulant networks. The approach leveraged the Cauchy-Binet formula, which, in conjunction with the Popov-Belevitch-Hautus test, provided a new method to examine controllability properties for systems with Circulant underlying communication structure. This chapter demonstrates how the method is also utilized to fully characterize the controllability of path graphs.

A path $i_0 i_1 \dots i_n$ is a finite sequence of vertices such that i_{k-1} is connected to i_k. Let $\mathcal{G}$ be a path graph with n vertices. The associated Laplacian matrix has a special structure, which is called a Tridiagonal matrix. A general Tridiagonal matrix has the following structure:

$$A = \begin{bmatrix} -\alpha + b & c & 0 & 0 & \dots & 0 & 0 \\ a & b & c & 0 & \dots & 0 & 0 \\ 0 & a & b & c & \dots & 0 & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & a & -\beta + b \end{bmatrix} \tag{7.1}$$

The next theorem explores the eigenvalues and eigenvectors of this type of matrix for a special case of α and β.

Theorem 7.0.6. *[131] Suppose $\alpha = \beta = \sqrt{ac} \neq 0$. Then the eigenvalues $\lambda_1, \dots, \lambda_n$ of A in (7.1) are given by*

$$\lambda_k = b + 2\sqrt{ac}\cos\frac{k\pi}{n}, \qquad k = 1, 2, 3, \dots, n,$$

and the corresponding eigenvectors $u^k = (u_1^k, \dots, u_n^k)^T$ are given by

$$u_j^k = \rho^{j-1}\sin\frac{k(2j-1)\pi}{2n}, \qquad j = 1, 2, \dots, n$$

for $k = 1, 2, \ldots, n-1$ *and*

$$u_j^k = (-\rho)^{j-1}, \qquad j = 1, 2, \ldots, n, \qquad k = n.$$

It is trivial to see the Laplacian of a path graph with n nodes is a special case of the matrix A defined in (7.1) where $b = 2$, $c = a = -1$ and $\alpha = \beta = \sqrt{ac}$. Therefore, the eigenvalues and eigenvectors of the associated Laplacian would be

$$\begin{aligned}
\lambda_k &= b + 2\sqrt{ac}\cos\frac{k\pi}{n}, \qquad k = 1, 2, 3, \ldots, n, \\
u^k &= (u_1^k, \ldots, u_n^k)^T \quad \text{where} \\
u_j^k &= \rho^{j-1}\sin\frac{k(2j-1)\pi}{2n}, \qquad j = 1, 2, \ldots, n \quad k = 1, 2, \ldots, n-1 \\
u_j^k &= (-\rho)^{j-1}, \qquad j = 1, 2, \ldots, n, \qquad k = n.
\end{aligned} \tag{7.2}$$

The controllability of path graphs have been discussed in the literature [101, 102]. Next we explore this problem from the new perspective on the structure of the eigenvector matrix of the Laplacian. Prior to mentioning the main theorem, the next theorem proves the maximum algebraic multiplicity of the eigenvalues of all path graphs is one.

Proposition 7.0.7. *All eigenvalues of a path graph are distinct.*

Proof. Let us prove this by contradiction. Assume they are not distinct and two of them are similar, $\lambda_1 = b + 2\sqrt{ac}\cos\frac{k_1\pi}{n}$ and $\lambda_2 = b + 2\sqrt{ac}\cos\frac{k_2\pi}{n}$. Since $\lambda_1 = \lambda_2$, it implies that $\frac{k_1\pi}{n} = \frac{k_2\pi}{n} + k'\pi$ where $k_1, k_2 \in [1, 2, \ldots, n]$ and $k' \in \mathbb{Z}$. Therefore, $k_1 - k_2 = nk'$ and $k_1 - k_2 \in [0, 1, \ldots, n-1]$ which implies the equality $k_1 - k_2 = nk'$ is not satisfied for any value of k'. This contradicts the original assumption that λ_1 and λ_2 are identical. □

Theorem 7.0.8. *All path graphs are controllable from one node.*

Proof. From the PBH test, (A, B) is not controllable if and only if there exists a nonzero vector w such that

$$w^T A(\mathcal{G}) = \lambda w^T \qquad \text{and} \qquad w^T B = 0. \tag{7.3}$$

Let us prove by contradiction, and assume such a nonzero vector w exists. The vector w satisfies $w^T A(\mathcal{G}) = \lambda w^T$; therefore, it is the left eigenvector. For such a nonzero vector w,

the PBH test implies that $w^T B = 0$. This equality implies that w is a member of the left null space of B. Let us set $U = [u_1, \ldots, u_n]$ where each column u_i is the eigenvector of A, and let Q be an $n \times 1$ submatrix of U. Considering the possible cases for w, the equality is equivalent to having $Q^T B = 0$. The fact that $Q^T B$ does not span the whole $\mathbb{R}$ implies that $Q^T B = \mathbf{det}(Q^T B) = 0$. Even though Cauchy-Binet provides a formula to calculate $\mathbf{det}(Q^T B)$, this case can be managed easier as follows. Assume node k is the input node. The structure of the input matrix B in (1.9) implies that each submatrix Q is guaranteed to possess a nonzero k^{th} entry. Now we are looking for an input node, called the k^{th} node, such that the k^{th} entries of all eigenvectors in (7.2) are nonzero. From (7.2) it is trivial to see that $k = n$ always satisfies this condition. Other than the node n, any node j that satisfies

$$\sin\left(\frac{k(2j-1)\pi}{2n}\right) \neq 0 \quad \text{for all } k$$

can be chosen an input node. □

Fig. 7.1 demonstrates path graphs with $n = 4$ to 53 nodes and the controllable nodes in each graph (each column) are shown by filled dots. We already showed that a path graph is controllable. Therefore, Theorem 6.3.2 confirms that the system is input asymmetric. The goal is to explore whether the input symmetry is a necessary condition for the uncontrollability of path graphs. The next theorem shows that for path graphs, the input symmetry is not the necessary condition.

Theorem 7.0.9. *Input asymmetry is a necessary condition for the controllability of path graphs with dynamic proposed in (1.9). But it is not sufficient.*

Proof. Theorem 6.3.2 implies that a controllable path graph is input asymmetric. To show the other direction is not true, consider the following counter example. Assume $\mathcal{G}$ is a path graph with $n = 6$. Theorem 7.0.8 and also Fig. 7.1 confirm that choosing node 2 as input node does not make the dynamic in (1.9) controllable even though it is input symmetric. □

As a summary, the uncontrollable nodes of the Laplacian dynamics of a path graph with n nodes satisfy the following equations: Node j is uncontrollable if for some $k = 1, 2, \ldots, n$,

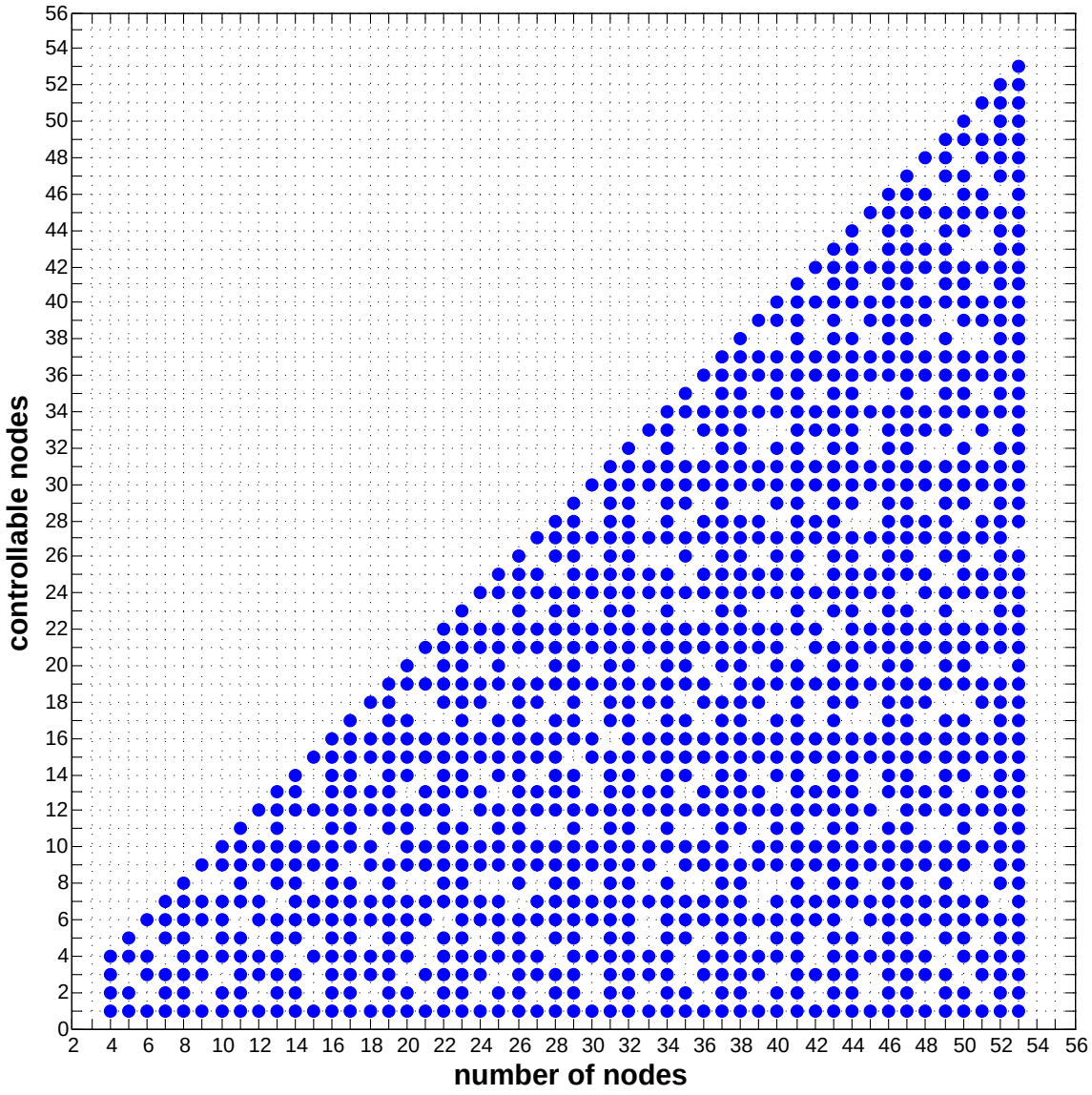


Figure 7.1: Controllable nodes in a path graph; for each node, the filled dots depict the controllable nodes in the path.

the following happens

$$\sin(\frac{k(2j-1)\pi}{2n}) = 0.$$

We can plot the eigenvectors of the path graph and study the modal analysis. As an example, consider a path with $n = 7$. From the above equation, it can be confirmed that $j = 4$ is the only uncontrollable node. Fig. 7.2 demonstrates that the 4^{th} entry of all eigenvectors is zero. Fig. 7.3 also demonstrates the modal analysis of a path of length 9.

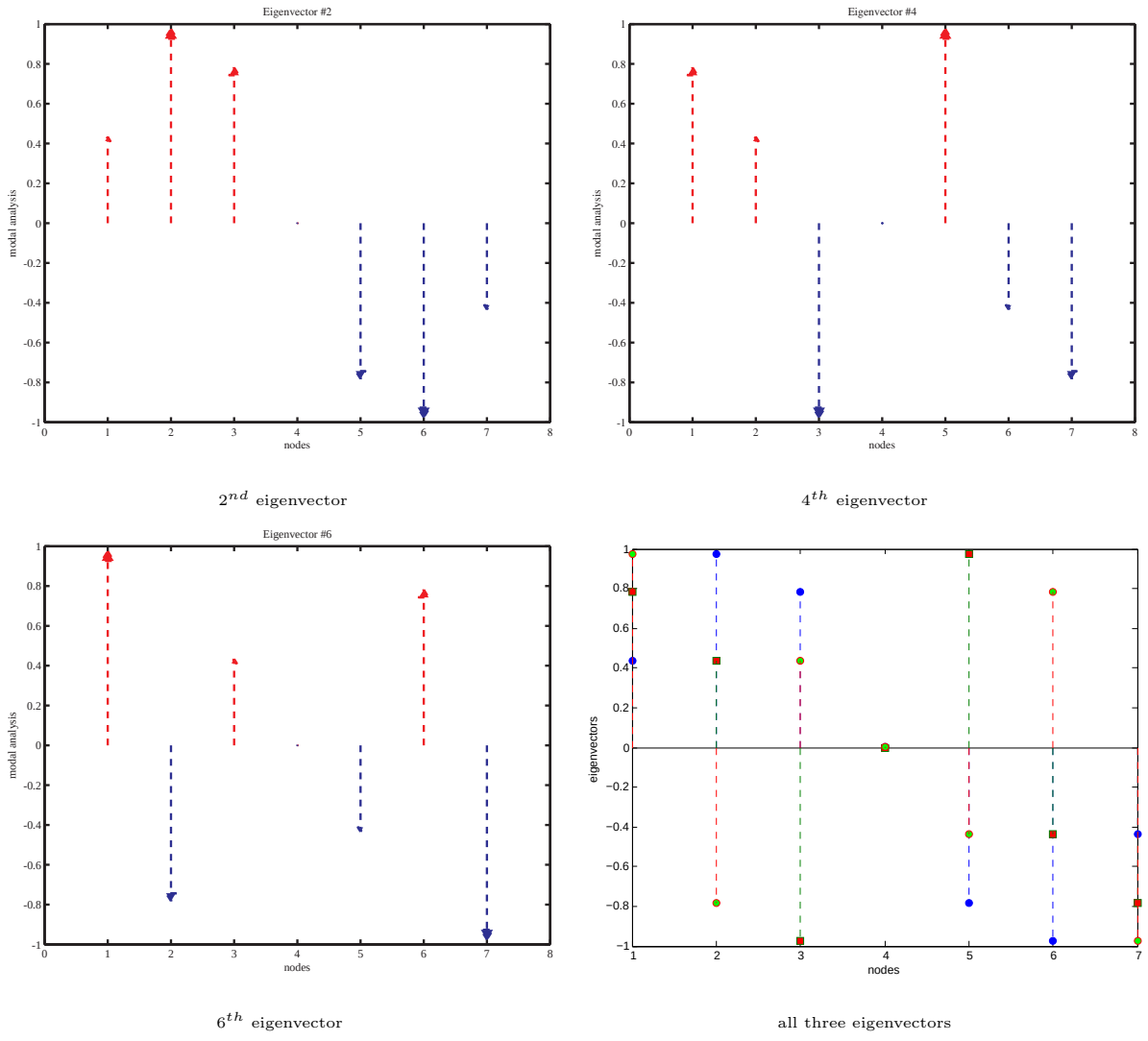


Figure 7.2: Modal analysis of a path graph with $n = 7$.

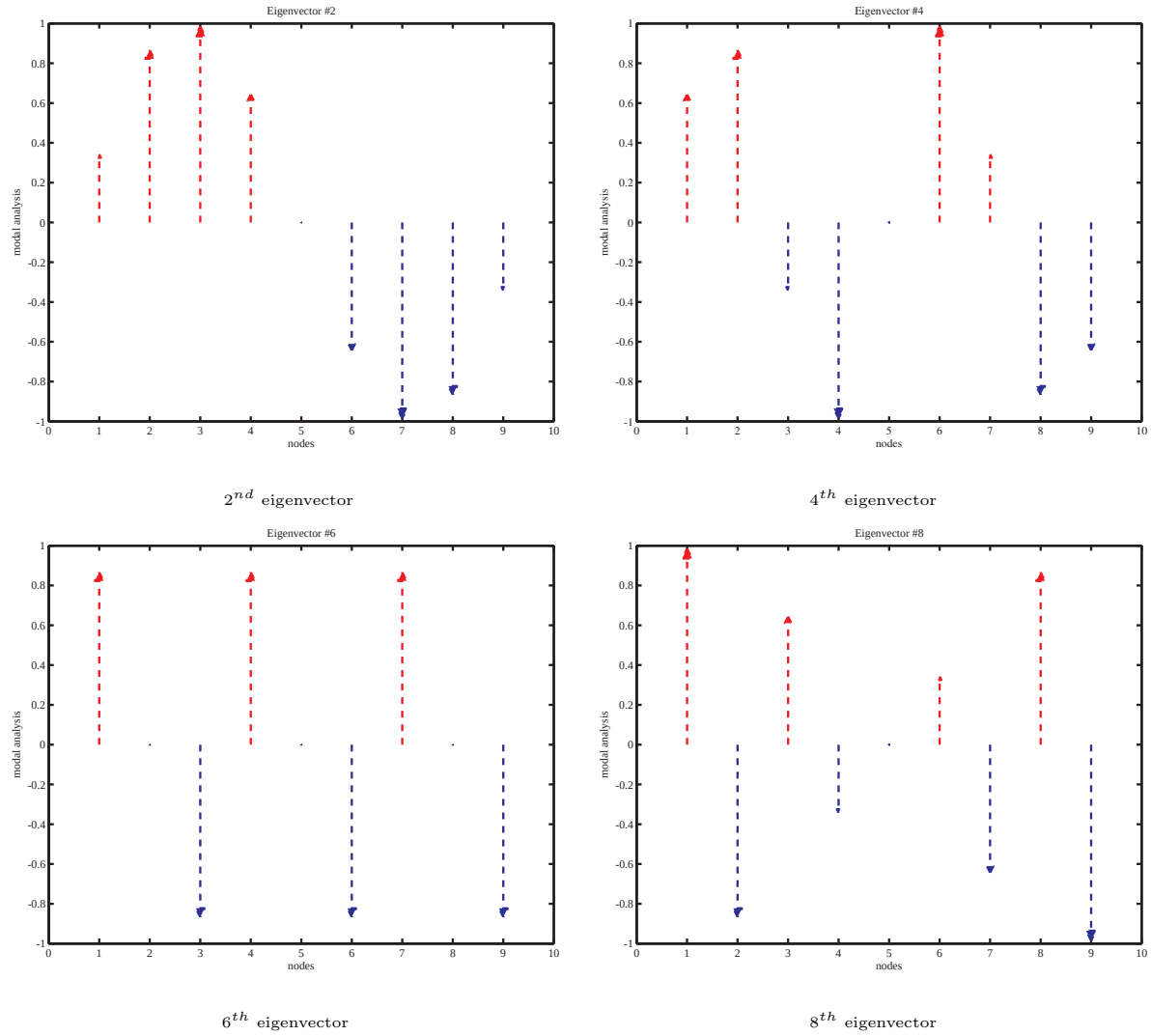


Figure 7.3: A path graph with $n = 9$. The modal analysis depicts that node 5 is the uncontrollable node.

Chapter 8

CONTROLLABILITY AND OBSERVABILITY OF CARTESIAN PRODUCT NETWORKS

This chapter presents an analysis framework for a class of dynamic composite networks. These networks are formed from smaller factor networks via graph Cartesian products. We provide a composition method for extending the controllability and observability of the factor networks to that of the composite network. We then delve into the effectiveness of designing control and estimation algorithms for the composite network via symmetry in the network and its gramian structure. Examples are provided throughout the chapter to demonstrate the results. A practical example of a Cartesian social network is provided and its opinion dynamics are analyzed using the chapter's results.

8.1 Background and Model

We provide a brief background on the constructs and models that will be used in this chapter.

Of particular interest to this chapter is the Kronecker sum defined on square matrices $M \in \mathbb{R}^{s\times s}$ and $N \in \mathbb{R}^{r\times r}$ as

$$M \oplus N := M \otimes I_m + I_n \otimes N.$$

A property of the Kronecker sum is that given the left eigenvalue-eigenvector pairs of M and N as (λ_i, u_i) for $i = 1, \ldots, n$ and (μ_j, v_j) for $j = 1, \ldots, m$, respectively, then $(\lambda_i + \mu_j, u_i \otimes v_j)$ for $i = 1, \ldots, n$ and $j = 1, \ldots, m$ are the left eigenvalue-eigenvector pairs of $M \oplus N$.

Denoting the eigenvalues of $M \oplus N$ as $\phi_{ij} := \lambda_i + \mu_j$ then ϕ_{ij} has algebraic multiplicity greater than one if $\phi_{ij} = \phi_{pq}$ where $(i, j) \neq (p, q)$. Formally, defining the set $\mathcal{W}_{ij} = \{(p, q)|\phi_{ij} = \lambda_p + \mu_q\}$, then the algebraic multiplicity of ϕ_{ij} is $|\mathcal{W}_{ij}|$. Every left eigenvector w_{ij} of ϕ_{ij} can be represented as linear combinations of eigenvectors $u_p \otimes v_q$ for $(p, q) \in \mathcal{W}_{ij}$ and some u_ps and v_qs that are left eigenvectors of λ_p and μ_q, respectively. As the geometric

multiplicity of ϕ_{ij} may be less than the algebraic multiplicity, we may not need $|\mathcal{W}_{ij}|$ vectors to represent an eigenvector of $M \oplus N$. For this purpose, we define the set $\mathcal{W}_{ij}^* \subseteq \mathcal{W}_{ij}$ composed of pairs (p, q) which correspond to linearly independent $u_p \otimes v_q$ such that $\left|\mathcal{W}_{ij}^*\right|$ is the geometric multiplicity of ϕ_{ij}.[1] Thus, (ϕ_{ij}, w_{ij}) is a left eigenvalue-eigenvector pair if and only if $w_{ij} = \sum_{(p,q)\in\mathcal{W}_{ij}^*} \alpha_{pq}(u_p \otimes v_q)$ where $\alpha_{pq} \in \mathbb{R}$ and $\sum_{(p,q)\in\mathcal{W}_{ij}^*} \alpha_{pq}^2 \neq 0$. If ϕ_{ij} is simple, then $w_{ij} = u_i \otimes v_j$ is a corresponding left eigenvector if and only if (λ_i, u_i) and (μ_i, v_j) are left eigenvalue-eigenvectors of M and N, respectively.

A remarkable feature is that the Kronecker sum satisfies the matrix exponential property

$$e^{M\oplus N} = e^M \otimes e^N. \tag{8.1}$$

We now proceed to introduce the problem dynamics and its relationship to the underlying graph.

There are many ways to construct a matrix $A(\mathcal{G}) \in \mathbb{R}^{n\times n}$ from the edges and nodes of an n node graph $\mathcal{G}$. One such method we have already touched upon; namely the adjacency matrix $\hat{E}(\mathcal{G})$. This matrix uniquely defines the graph $\mathcal{G}$, but there are many matrix representations of a graph that do not, for example the self-loop matrix $\Delta_s(\mathcal{G})$, the in-degree matrix $\Delta_{in}(\mathcal{G})$ and the out-degree matrix $\Delta_{out}(\mathcal{G})$.

One of the properties common to all of the aforementioned matrix representations is that they preserve the symmetries in the graph. By this we mean that for a representations $A(\cdot)$ if there exists a permutation matrix P such that $\hat{E}(\mathcal{G})P = P\hat{E}(\mathcal{G})$, then $A(\mathcal{G})P = PA(\mathcal{G})$. We refer to this matrix representations feature as *symmetry preserving*.

In this section, we will be considering the specific family of symmetry preserving matrix representations which also exhibit the added feature that the representation has the property

$$A(\mathcal{G}_1 \square \mathcal{G}_2) = A(\mathcal{G}_1) \oplus A(\mathcal{G}_2),$$

for all graphs $\mathcal{G}_1$ and $\mathcal{G}_2$. We denote this family of matrix representation as $\boldsymbol{A}_\oplus$.

All matrix representations so far are members of $\boldsymbol{A}_\oplus$. Another well known member of $\boldsymbol{A}_\oplus$ is the in-degree graph Laplacian (or Laplacian) matrix $\mathcal{L}(\mathcal{G})$ defined as $[\mathcal{L}(\mathcal{G})]_{ij} = -[\hat{E}(\mathcal{G})]_{ij}$

[1] We note that as **rank** $[u_1, \ldots, u_r]$ **rank** $[v_1, \ldots, v_s]$ = **rank** $([u_1, \ldots, u_r] \otimes [v_1, \ldots, v_s])$ then the left eigenvectors of M indicated by the pairs of $\mathcal{W}_{ij}^*$ are linearly independent, and similarly for the left eigenvectors of N.

for $i \neq j$ and $[\mathcal{L}(\mathcal{G})]_{ii} = [\Delta_{in}(\mathcal{G})]_{ii}$. The Laplacian matrix features in popular consensus dynamics and, with the exception of self-loops, uniquely defines the graph $\mathcal{G}$. Other noteworthy members are the out-degree graph Laplacian $\mathcal{L}_{out}(\mathcal{G})$ defined as $[\mathcal{L}_{out}(\mathcal{G})]_{ij} = -[\hat{E}(\mathcal{G})]_{ij}$ for $i \neq j$ and $[\mathcal{L}_{out}(\mathcal{G})]_{ii} = [\Delta_{out}(\mathcal{G})]_{ii}$, used in advection dynamics [190], and the M-matrix representation $M(\mathcal{G})$ where $[M(\mathcal{G})]_{ij} = -[\hat{E}(\mathcal{G})]_{ij}$ for $i \neq j$ and $[M(\mathcal{G})]_{ii} = [\hat{E}(\mathcal{G})]_{ii}$ investigated in [191]. The class $\boldsymbol{A}_{\oplus}$ of representations is by no means a small one, and many other members will be featured in examples throughout the section. It is easy to show that $\boldsymbol{A}_{\oplus}$ is closed under addition providing a simple mechanism to generate new members.

In this section, we are exploring controllability and observability of systems of the form

$$\begin{aligned} \dot{x}(t) &= A(\mathcal{G})x(t) + Bu(t) \\ y(t) &= Cx(t), \end{aligned} \tag{8.2}$$

where $A(\cdot) \in \boldsymbol{A}_{\oplus}$. For brevity, we refer to these dynamics by defining the matrix triple $(A(\mathcal{G}), B, C)$, or if only the inputs and outputs are of interest by the matrix pair $(A(\mathcal{G}), B)$ and $(A(\mathcal{G}), C)$, respectively.

It should be mentioned that due to the duality between controllability and observability for $B = C^T$, the pair $(A(\mathcal{G}), B)$ is controllable if and only if the pair $(A(\mathcal{G}), C)$ is observable. Hence, the results throughout this section will be presented in terms of controllability but are equally applicable to the observability problem. A helpful tool to establish controllability is the well known Popov-Belevitch-Hautus (PBH) test [192].

It is often of interest where the inputs and outputs of system (8.2) are in terms of the nodes of the graph $\mathcal{G}$. If the set of input nodes in the n node graph is $S = \{i_1, i_2, \ldots, i_p\}$ for $i_1 < i_2 < \cdots < i_p$; the corresponding input matrix is $B = \left[e_{i_1}, e_{i_2}, \ldots, e_{i_p}\right] \in \mathbb{R}^{n \times p}$. We uniquely define input matrices of this form as $B_n(S)$. Similarly, the output matrices are defined as $C_n(S) := B_n(S)^T$. If it is clear from the context, we remove the n for brevity.

Before proceeding to the first main result, we present a preliminary result relating the distinctness of the eigenvalues of $A(\mathcal{G})$ to its prime factor decomposition.

Proposition 8.1.1. *For $A(\cdot) \in \boldsymbol{A}_{\oplus}$, if the eigenvalues of $A(\mathcal{G})$ are simple, then the prime factor decomposition of $\mathcal{G}$ contains no powers of prime graphs.*

Proof. Let the prime factor decomposition of the n node graph $\mathcal{G}$ be $\mathcal{G} = \mathcal{G}_1^{k_1} \square \cdots \square \mathcal{G}_m^{k_m}$. If $\mathcal{G}$ is a power of a prime graph then there exists a $k_i \geq 2$ corresponding to the m node prime factor $\mathcal{G}_i$. Hence, $\mathcal{G} = \mathcal{G}_i^2 \square \mathcal{G}_b$ where $\mathcal{G}_b = \mathcal{G}_1^{k_1} \square \cdots \square \mathcal{G}_i^{k_i-2} \square \cdots \square \mathcal{G}_m^{k_m}$. Therefore $A(\mathcal{G})$ has pairs of eigenvalues with the value $\lambda_i + \lambda_j + \mu_k$ for $i, j = 1, \ldots, m$ and $k = 1, \ldots, \frac{n}{2m}$ where λ_i and λ_j are the eigenvalues of $A(\mathcal{G}_k)$ and μ_k are the eigenvalues of $A(\mathcal{G}_b)$, respectively. This follows from the eigenvalue properties of the Cartesian sum. □

Remark 8.1.2. *The discrete form of (8.2) is*

$$x(k+1) = A(\mathcal{G})x(k) + Bu(k)$$
$$y(k) = Cx(k).$$

For the discrete version, the family of matrix representations considered in the section is symmetry preserving and has the property,

$$A(\mathcal{G}_1 \square \mathcal{G}_2) = A(\mathcal{G}_1) \otimes A(\mathcal{G}_2).$$

We denote this family of matrix representations as $\boldsymbol{A}_\otimes$. The matrix exponential of the matrix representations in $\boldsymbol{A}_\oplus$ are members of $\boldsymbol{A}_\otimes$. The family $\boldsymbol{A}_\otimes$ is closed under multiplication.

8.2 Control Product

The following theorem introduces our first method of extending control of the factors of the composite graph $\mathcal{G}$ to $\mathcal{G}$ itself for the case where $A(\mathcal{G})$ has simple eigenvalues. Specifically, we examine the controllability of pair $(A(\mathcal{G}), B)$, first conjectured in [193], where $\mathcal{G} = \mathcal{G}_1 \square \mathcal{G}_2$ and $B = B_1 \otimes B_2$.

Theorem 8.2.1. *Consider $A(\cdot) \in \boldsymbol{A}_\oplus$ and graphs $\mathcal{G}_1$ and $\mathcal{G}_2$. Assume $A(\mathcal{G}) = A(\mathcal{G}_1 \square \mathcal{G}_2)$ has simple eigenvalues. Then, the pairs $(A(\mathcal{G}_1), B_1)$ and $(A(\mathcal{G}_2), B_2)$ are controllable if and only if $(A(\mathcal{G}), B)$ is controllable, where $B = B_1 \otimes B_2$.*

Proof. Assuming the eigenvalues of $A(\mathcal{G})$ are simple, w is a left eigenvectors of $A(\mathcal{G})$ if and only if $w = u \otimes v$ for some left eigenvector u and v of $A(\mathcal{G}_1)$ and $A(\mathcal{G}_2)$, respectively (see

§1.2.4). Hence,

$$w^T B \;=\; (u \otimes v)^T (B_1 \otimes B_2) = (u^T \otimes v^T)(B_1 \otimes B_2) = u^T B_1 \otimes v^T B_2.$$

Now, $w^T B = 0$ if and only if $u^T B_1 = 0$ or $v^T B_2 = 0$ (or both). Hence, by the PBH test, $(A(\mathcal{G}), B)$ is controllable if and only if $(A(\mathcal{G}_1), B_1)$ and $(A(\mathcal{G}_2), B_2)$ are controllable. □

Consider the case where $B_1 := B(S_1)$ and $B_2 := B(S_2)$, then for $S = S_1 \times S_2 := \{(i,j) | i \in S_1,\, j \in S_2\}$ we have $B(S) = B_1 \otimes B_2$. This motivates the name *control product* for the scheme.

The following example demonstrates Theorem 8.2.1 and the form of set S.

Example 8.2.2. *Consider the graphs $\mathcal{G}_1$ and $\mathcal{G}_2$ from Figure 8.1. Let the eigenvalues of $\mathcal{L}(\mathcal{G}_1)$ be $\lambda_1, \lambda_2, \lambda_3$, and let similarly the eigenvalues for $\mathcal{L}(\mathcal{G}_2)$ be $\mu_1, \mu_2, \mu_3, \mu_4$. It is a simple check that these eigenvalues are distinct. Further as $\lambda_i + \mu_j$ for $i = 1, \ldots, 3$ and $j = 1, \ldots 4$ are the eigenvalues of $\mathcal{L}(\mathcal{G})$, it is easy to note its eigenvalues are also distinct. For $S_1 = \{1'\}$ and $S_2 = \{1, 2\}$, the pairs $(-\mathcal{L}(\mathcal{G}_1), B_3(S_1))$ and $(-\mathcal{L}(\mathcal{G}_2), B_4(S_2))$ are controllable. The nodes corresponding to sets S_1 and S_2 are half shaded in Figure 8.1. Now, $B_3(S_1) \otimes B_4(S_2) = B_{12}(S)$ where $S = \{(1', 1), (1', 2)\}$, denoted by full shaded nodes in Figure 8.1. Therefore, $(-\mathcal{L}(\mathcal{G}), B_{12}(S))$ is controllable.*

We now provide an example to illustrate the requirement that the composite graph in Theorem 8.2.1 has simple eigenvalues.

Example 8.2.3. *Denote the path graph of length 2 as $\mathcal{P}_2$. It is a well known property that $\mathcal{L}(\mathcal{P}_2)$ has simple eigenvalues and for $S = \{1\}$, $(-\mathcal{L}(\mathcal{P}_2), B_2(S))$ is controllable. Further $\mathcal{P}_2 \,\square\, \mathcal{P}_2 = \mathcal{C}_4$, a length four cycle graph and $B_2(S) \otimes B_2(S) = B_4(S)$. But all cycles graphs are uncontrollable from one node (see [218]) so $(-\mathcal{L}(\mathcal{C}_4), B_4(S))$ is uncontrollable.*

8.3 Graph Automorphism

The automorphisms of a graph describe its symmetries and have been previously shown to play an important role in the controllability of $(-\mathcal{L}(\mathcal{G}), B(S))$ for simple graph $\mathcal{G}$ [101]. We will extend these results to generalized graphs and any matrix representations which are symmetry preserving.

Formally, a graph automorphism is any permutation σ of the vertex set such that $\mathcal{G}$ contains an edge $\{i, j\}$ with weight w_{ji} if and only if it also contains an edge $\{\sigma(i), \sigma(j)\}$ with weight $w_{\sigma(j)\sigma(i)}$. The set of automorphisms (which in actual fact forms a group) is denoted $\mathrm{Aut}(\mathcal{G})$. Every automorphism can be represented uniquely as a permutation matrix J which commutes with the adjacency matrix, i.e., $J\hat{E}(\mathcal{G}) = \hat{E}(\mathcal{G})J$. For all $A(\cdot)$ which are symmetry preserving then $J\hat{E}(\mathcal{G}) = \hat{E}(\mathcal{G})J$ implies that $JA(\mathcal{G}) = A(\mathcal{G})J$. As an aside, a special feature of $\hat{E}(\mathcal{G})$ is that simple eigenvalues are only present when $\mathrm{Aut}(\mathcal{G})$ has a specific structure namely:

Proposition 8.3.1. *If all the eigenvalues of $\hat{E}(\mathcal{G})$ are simple, then every automorphism of $\mathcal{G}$ has order 1 or 2, i.e., $\sigma^2(i) = i$ for all $i \in V$.*

Thus, Proposition 8.3.1 describes the graphs relevant to Theorem 8.2.1 and the Laplacian matrix. We now present the results motivating the introduction of graph automorphisms - a connection between the automorphisms and controllability.

Proposition 8.3.2. *A graph $\mathcal{G}$ is uncontrollable from any pair $(A(\mathcal{G}), B(S))$ where $A(\cdot)$ is symmetry preserving if there exists an automorphism of $\mathcal{G}$ which fixes all inputs in the set S.*

Proof. Let the permutation matrix corresponding to the automorphism be J. Therefore,

$$JA(\mathcal{G}) = A(\mathcal{G})J$$

and, if the automorphism fixes all inputs in the set S, then

$$JB(S) = J^T B(S) = B(S).$$

Consider a right eigenvector-eigenvalue pair (λ, v) of $A(\mathcal{G})$. Then as

$$A(\mathcal{G})\,(Jv) = JA(\mathcal{G})v = J\lambda v = \lambda(Jv),$$

(λ, Jv) is also an eigenvector-eigenvalue pair, and

$$(v - Jv)^T B(S) \;\; = \;\; v^T B(S) - v^T J^T B(S) = v^T B(S) - v^T B(S) = 0.$$

Therefore, by the PBH test, the pair $(A(\mathcal{G}), B(S))$ is uncontrollable. □

We note that Proposition 8.3.2 is not sufficient for controllability as discussed in [101].

Proposition 8.3.2 highlights the requirement of selecting a set of inputs that break the symmetry structure of $\mathcal{G}$. Determining sets are a useful construct to describe this process.

Definition 8.3.3. *A subset S of the vertices of a graph $\mathcal{G}$ is called a determining set if whenever $g, h \in Aut(\mathcal{G})$ so that $g(s) = h(s)$ for all $s \in S$, then $g = h$. The determining number of a graph $\mathcal{G}$, denoted $Det(\mathcal{G})$, is the smallest integer r so that $\mathcal{G}$ has a determining set of size r.*

Another term for a determining set S is the *fixing set* due to the fact that no non-trivial automorphism of $\mathcal{G}$ fixes all members in S. Formally, a set $S \subseteq V(\mathcal{G})$ is a determining set if and only if the stabilizing set $Stab(S)$ of S, defined as

$$Stab(S) := \{g \in \mathrm{Aut}(\mathcal{G}) | g(v) = v, \forall v \in S\},$$

only contains the trivial automorphism. Directly from Proposition 8.3.2 and the definition of determining sets we have the following corollary.

Corollary 8.3.4. *A necessary condition for controllability of the pair $(A(\mathcal{G}), B(S))$ is that S is a determining set. Hence, $|S| \geq Det(\mathcal{G})$.*

The automorphism group of a composite graph is intimately linked to the automorphisms of its prime factors. This link translates through to the determining set of the composite graph summarized in the following:

Theorem 8.3.5. *[194] Let $\mathcal{G} = \mathcal{G}_1^{k_1} \Box \cdots \Box \mathcal{G}_m^{k_m}$ be the prime factor decomposition for a connected graph $\mathcal{G}$. Then $Det(\mathcal{G}) = \max\{Det(\mathcal{G}_i^{k_i})\}$.*

We now have the required ground work to state a consequence of the graph automorphism structure of the graph pertaining to Theorem 8.2.1.

Theorem 8.3.6. *Under the assumptions of Theorem 8.2.1, consider the controllable pairs $(A(\mathcal{G}_1), B(S_1))$ and $(A(\mathcal{G}_2), B(S_2))$ where $|S_1| = Det(\mathcal{G}_1)$ and $|S_2| = 1$. Then $S = S_1 \otimes S_2$ is the smallest set such that $(A(\mathcal{G}_1 \Box \mathcal{G}_2), B(S))$ is controllable.*

Proof. As all eigenvalues of $A(\mathcal{G}_1 \square \mathcal{G}_2)$ are simple, then $\mathcal{G}_1$ and $\mathcal{G}_2$ has only simple eigenvalues. Hence, from Proposition 8.1.1, both $\mathcal{G}_1$ and $\mathcal{G}_2$ are not powers of prime graphs, i.e., $\mathcal{G}_1 = \mathcal{G}_{a1} \square \cdots \square \mathcal{G}_{am}$ and $\mathcal{G}_2 = \mathcal{G}_{b1} \square \cdots \square \mathcal{G}_{bn}$ are prime factor decompositions. Further as $\mathcal{G}_1 \square \mathcal{G}_2$ has distinct eigenvalues then $\mathcal{G}_1$ and $\mathcal{G}_2$ are relatively prime, i.e., $\mathcal{G}_{ai}$ is not isomorphic to $\mathcal{G}_{bj}$ for all $i = 1, \ldots, m$ and $j = 1, \ldots, n$. Therefore, $\mathcal{G} = \mathcal{G}_{a1} \square \cdots \square \mathcal{G}_{am} \square \mathcal{G}_{b1} \square \cdots \square \mathcal{G}_{bn}$ is the prime factor decomposition and $Det(\mathcal{G}) = \max(Det(\mathcal{G}_1), Det(\mathcal{G}_2))$. Further, as $(A(\mathcal{G}_2), B(S_2))$ is controllable then $1 = |S_2| \geq Det(\mathcal{G}_2) \geq 1$, so $Det(\mathcal{G}) = Det(\mathcal{G}_1)$. Now $B(S_1) \otimes B(S_2) = B(S)$ for some $S \subseteq V(\mathcal{G}_1 \square \mathcal{G}_2)$ and $|S| = |S_1|\,|S_2| = |S_1|$. As $|S| = Det(\mathcal{G})$ then the pair $(A(\mathcal{G}_1 \square \mathcal{G}_2), B(S))$ is controllable with the optimal number of inputs. $\square$

We now revisit Example 8.2.2 with Theorem 8.3.6 in mind.

Example 8.3.7. *Further examination of Example 8.2.2, the* $Aut(\mathcal{G}_2) = \{id, \sigma, \tau, \sigma\tau\}$ *where id is the identity permutation,* $\sigma(1,2,3,4) = (1,4,3,2)$ *and* $\tau(1,2,3,4) = (3,2,1,4)$. *Hence,* $Det(\mathcal{G}_2) = 2 = |S_2|$ *and* $|S_1| = 1$. *Applying Theorem 8.3.6 shows that S is the smallest controllable input set.*

8.4 Layered Control

The following theorem details our second method of extending control of the factors to the composite graph. This control scheme involves repeating the form of the control matrix B_1 to every $\mathcal{G}_1$ layer of $\mathcal{G}$, motivating the name *layered control*. As the Kronecker product exhibits permutation equivalency, these results are equivalent to extending the control matrix B_2 to every $\mathcal{G}_2$ layer of $\mathcal{G}$.

Theorem 8.4.1. *Consider* $A(\cdot) \in \boldsymbol{A}_{\oplus}$ *and graphs* $\mathcal{G}_1$ *and* $\mathcal{G}_2$ *with* n_1 *and* n_2 *nodes, respectively. The pair* $(A(\mathcal{G}_1), B_1)$ *is controllable if and only if* $(A(\mathcal{G}), B)$ *is controllable, where* $\mathcal{G} = \mathcal{G}_1 \square \mathcal{G}_2$ *and* $B = B_1 \otimes I_{n_2}$.

Proof. Denote the left eigenvalue-eigenvector pairs of $A(\mathcal{G}_1)$ and $A(\mathcal{G}_2)$ as (λ_i, u_i) for $i = 1, \ldots, n_1$ and (μ_j, v_j) for $j = 1, \ldots, n_2$, respectively. Consider the eigenvalue ϕ_{ij} of $A(\mathcal{G})$ then w_{ij} is a corresponding left eigenvector of $A(\mathcal{G})$ if and only if $w_{ij} = \sum_{(p,q) \in \mathcal{W}_{ij}^*} \alpha_{pq}(u_p \otimes v_q)$

where $\alpha_{pq} \in \mathbb{R}$, $\sum_{(p,q)\in\mathcal{W}_{ij}^*} \alpha_{pq}^2 \neq 0$, and $\mathcal{W}_{ij}^*$ are defined in §1.2.4. Hence,

$$\begin{aligned} 0 &= w_{ij}^T B = \sum_{(p,q)\in\mathcal{W}_{pq}^*} \alpha_{pq}(u_p \otimes v_q)^T (B_1 \otimes I_{n_2}) \\ &= \sum_{(p,q)\in\mathcal{W}_{pq}^*} \alpha_{pq}(u_p^T \otimes v_q^T)(B_1 \otimes I_{n_2}) \\ &= \sum_{(p,q)\in\mathcal{W}_{pq}^*} u_p^T B_1 \otimes \alpha_{pq} v_q^T \end{aligned}$$

or equivalently,

$$\sum_{(p,q)\in\mathcal{W}_{pq}^*} \left[u_p^T B_1\right]_k \left(\alpha_{pq} v_q^T\right) = 0$$

for $k = 1, \ldots, n_1$. For $\alpha_{pq} \neq 0$, this summation is zero if and only if $\left[u_p^T B_1\right]_k = 0$ for all k and p, i.e., $u_p^T B_1 = 0$ for all $(p,q) \in \mathcal{W}_{pq}^*$, since the set of eigenvectors v_q are linearly independent. Therefore as there will always exist an $\alpha_{pq} \neq 0$ corresponding to some w_{ij} then by the PBH test, $(A(\mathcal{G}), B)$ is controllable if and only if $(A(\mathcal{G}_1), B_1)$ is controllable.

□

An illustrative example follows:

Example 8.4.2. *Consider the graphs $\mathcal{G}_1$ and $\mathcal{G}_2$ from Figure 8.1. For $S_1 = \{1'\}$ and $S_2 = \{1, 2\}$, the pairs $(-\mathcal{L}(\mathcal{G}_1), B_3(S_1))$ and $(-\mathcal{L}(\mathcal{G}_2), B_4(S_2))$ are controllable. The nodes corresponding to sets S_1 and S_2 are half shaded in Figure 8.1. Now, $B_3(S_1) \otimes I = B_{12}(S_a)$ where $S_a = \{(1', 1), (1', 2), (1', 3), (1', 4)\}$, denoted by the lower half shaded nodes in Figure 8.1. Similarly, $I \otimes B_4(S_2) = B_{12}(S_b)$ and $S_b = \{(1', 1), (1', 2), (2', 1), (2', 2), (3', 1), (3', 2)\}$, denoted by the upper half shaded nodes in Figure 8.1. Therefore from Theorem 8.4.1, pairs $(-\mathcal{L}(\mathcal{G}), B_{12}(S_a))$ and $(-\mathcal{L}(\mathcal{G}), B_{12}(S_b))$ are controllable.*

Theorem 8.4.1 provides a useful tool of combining families of graphs with known controllability, such as the Laplacian of then path graphs, cycle graphs [195], and circulant graphs [218], with either graphs where controllability is hard to establish, such as random graphs, or graphs that are difficult to control, such as the complete graph.

Further, Theorem 8.4.1 can be combined with Theorem 8.2.1 to produce controllable graphs. A composite graph $\mathcal{G}$ can be decomposed into $\mathcal{G}_a \square \mathcal{G}_b$ where $\mathcal{G}_a$ is the largest factor

graph of $\mathcal{G}$ with simple eigenvalues and $\mathcal{G}_b$ has order n_b. In turn, $\mathcal{G}_a$ can be decomposed into its primes factors $\mathcal{G}_1 \square \ldots \square \mathcal{G}_k$ with corresponding controllable matrix pairs $(A(\mathcal{G}_i), B_i)$ for $i = 1, \ldots, k$ and some $A(\cdot) \in \boldsymbol{A}_{\oplus}$. Therefore by Theorem 8.2.1, $(A(\mathcal{G}_a), B_1 \otimes \cdots \otimes B_k)$ is controllable, and by Theorem 8.4.1 $(A(\mathcal{G}), B_1 \otimes \cdots \otimes B_k \otimes I_{n_b})$ is controllable. This technique is used in the following example to establish controllability of the grid $\mathcal{P}_2 \square \mathcal{P}_4 \square \mathcal{P}_5$.

Example 8.4.3. *Denote the path graphs of length two, four, and five path graph as $\mathcal{P}_2, \mathcal{P}_4$, and $\mathcal{P}_5$, respectively. Since all path graphs are controllable from either end node, designating one of the ends as the first node, the pairs $(-\mathcal{L}(\mathcal{P}_4), B_4(S))$ and $(-\mathcal{L}(\mathcal{P}_5), B_5(S))$ are controllable for $S = \{1\}$. Noting that $\mathcal{L}(\mathcal{P}_4 \square \mathcal{P}_5)$ has distinct eigenvalues using the technique as Example 8.2.2, from Theorem 8.2.1, the pair $(-\mathcal{L}(\mathcal{P}_4 \square \mathcal{P}_5), B_{20}(S))$ is controllable. Further applying Theorem 8.4.1, $(-\mathcal{L}(\mathcal{P}_2 \square \mathcal{P}_4 \square \mathcal{P}_5), B_{40}(S'))$ for $S' = \{(1,1),(2,1)\}$ is controllable as $I_2 \otimes B_{20}(S) = B_{40}(S')$. Here we have a 40 node grid controllable from 2 nodes.*

Interestingly, $\mathcal{L}(\mathcal{P}_2 \square \mathcal{P}_4 \square \mathcal{P}_5)$ has repeated eigenvalues and so, from the PBH test, at least two input nodes are required to form a controllable set. Hence, the set S' found in Example 8.4.3 is an optimal input set.

We now present a factorization lemma which represents the larger network trajectories in terms of the factors' trajectories under the control scheme described in Theorem 8.4.1. This result is an extension of the related results on Cartesian products over Laplacian-based simple networks by Nguyen and Mesbahi [196, 193].

Lemma 8.4.4. *Consider $x_1(t), x_2(t)$ to be the states of the systems*

$$\begin{aligned} \dot{x}_1(t) &= A(\mathcal{G}_1)x_1(t) + B_1 u_1(t) \\ \dot{x}_2(t) &= A(\mathcal{G}_2)x_2(t). \end{aligned}$$

The state trajectory generated by the dynamics $\dot{x}(t) = A(\mathcal{G}_1 \square \mathcal{G}_2)x(t) + Bu(t)$, where $B = B_1 \otimes I_{n_2}$, $u(t) = u_1(t) \otimes e^{A(\mathcal{G}_2)t}v$ for any $v \in \mathbb{R}^n$, is

$$x(t) = x_{1u}(t) \otimes x_2(t) + x_{1f}(t) \otimes e^{A(\mathcal{G}_2)t}v,$$

for all time t. Here, $x_{1u}(t)$ and $x_{1f}(t)$ are the unforced and forced dynamics of $x_1(t)$, respectively, i.e., $x_1(t) = x_{1u}(t) + x_{1f}(t)$, when initialized from $x(0) = x_1(0) \otimes x_2(0)$.

Proof. We have the unforced dynamics of $x(t)$

$$\begin{aligned} x_u(t) &:= e^{A(\mathcal{G}_1 \square \mathcal{G}_2)t} x(0) \\ &= e^{(A(\mathcal{G}_1) \oplus A(\mathcal{G}_2))t} (x_1(0) \otimes x_2(0)) \\ &= \left(e^{A(\mathcal{G}_1)t} \otimes e^{A(\mathcal{G}_2)t} \right) (x_1(0) \otimes x_2(0)) \\ &= e^{A(\mathcal{G}_1)t} x_1(0) \otimes e^{A(\mathcal{G}_2)t} x_2(0) \\ &= x_{1u}(t) \otimes x_2(t). \end{aligned}$$

and forced dynamics

$$\begin{aligned} x_f(t) &:= \int_0^t e^{A(\mathcal{G}_1 \square \mathcal{G}_2)(t-\tau)} B u(\tau) d\tau \\ &= \int_0^t \left(e^{A(\mathcal{G}_1)(t-\tau)} \otimes e^{A(\mathcal{G}_2)(t-\tau)} \right) (B_1 \otimes I) \times \left(u_1(\tau) \otimes e^{A(\mathcal{G}_2)\tau} v \right) d\tau \\ &= \int_0^t e^{A(\mathcal{G}_1)(t-\tau)} B_1 u_1(\tau) \otimes e^{A(\mathcal{G}_2)t} v d\tau = \int_0^t e^{A(\mathcal{G}_1)(t-\tau)} B_1 u_1(\tau) d\tau \otimes e^{A(\mathcal{G}_2)t} v d\tau \\ &= x_{1f}(t) \otimes e^{A(\mathcal{G}_2)t} v. \end{aligned}$$

Hence, $x(t) = x_u(t) + x_f(t) = x_{1u}(t) \otimes x_2(t) + x_{1f}(t) \otimes e^{A(\mathcal{G}_2)t} v$. □

Note that for $v = x_2(0)$, we have

$$\begin{aligned} x(t) &= x_{1u}(t) \otimes x_2(t) + x_{1f}(t) \otimes e^{A(\mathcal{G}_2)t} x_2(0) \\ &= x_{1u}(t) \otimes x_2(t) + x_{1f}(t) \otimes x_2(t) \\ &= (x_{1u}(t) + x_{1f}(t)) \otimes x_2(t) \\ &= x_1(t) \otimes x_2(t) \end{aligned} \tag{8.3}$$

which is an echo of the factorization lemma described in [193, 196]. The next corollary addresses the state signal $x(t)$ for another special case.

Corollary 8.4.5. *Consider $A(\mathcal{G}) = -\mathcal{L}(\mathcal{G})$ and the model and assumptions of Lemma 8.4.4. For $v = \mathbf{1}$, we have $u(t) = u_1(t) \otimes \mathbf{1}$ and*

$$x(t) = x_{1u}(t) \otimes x_2(t) + x_{1f}(t) \otimes \mathbf{1}. \tag{8.4}$$

Proof. The result follows directly from Lemma 8.4.4 as the vector $\mathbf{1}$ lies in the right null space of $\mathcal{L}(\mathcal{G}_2)$ so that $e^{\mathcal{L}(\mathcal{G}_2)t}\mathbf{1} = \mathbf{1}$, for all $t \geq 0$. □

The next section explores the controllability Gramian of the composite graph, under the layered control scheme.

8.5 Controllability Gramian

Consider the controllability Gramian $P(A, B)$ pertaining to dynamics (A, B) defined by the solution to the Lyapunov equation $AP(A, B) + P(A, B)A^T = -BB^T$.

The following results give bounds on both the eigenvalues and the trace of the controllability Gramian, for the layered control scheme where $A(\mathcal{G}_1 \square \mathcal{G}_2)$ is stable and symmetric. We first consider the case where the control matrix corresponds to a set of input nodes, i.e., it takes the form $B_{n_1}(S) \otimes I_{n_2}$ and the extension to the form $B_{n_1}(S) \otimes B_{n_2}(S)$ is discussed afterwards.

Proposition 8.5.1. *Let $A(\cdot) \in \boldsymbol{A}_{\oplus}$, $\mathcal{G} = \mathcal{G}_1 \square \mathcal{G}_2$ where $A(\mathcal{G})$ is stable and symmetric. Consider $P := P(A(\mathcal{G}), B)$, where $B = B_{n_1}(S) \otimes I_{n_2}$, for some node set S of $\mathcal{G}_1$. Then, the following inequalities hold*

$$-\frac{n_2\,|S|}{2f_{max}} \leq \mathbf{trace}(P) \leq -\frac{n_2\,|S|}{2f_{min}} \quad \textit{and} \quad \lambda_{max}(P) \geq -\frac{1}{2f_{max}},$$

where $f_{min} = \lambda_{min}(A(\mathcal{G}_1)) + \lambda_{min}(A(\mathcal{G}_2))$ and $f_{max} = \lambda_{max}(A(\mathcal{G}_1)) + \lambda_{max}(A(\mathcal{G}_2))$.

Proof. The following upper and lower bounds for the trace of the controllability Gramian $P(A, B)$, as a function of the trace of A and B, have been shown in Wang et al. [135] to be

$$-\frac{\mathbf{trace}(BB^T)}{2\lambda_{min}(A)} \leq \mathbf{trace}(P(A, B)) \leq -\frac{\mathbf{trace}(BB^T)}{2\lambda_{max}(A)}.$$

Then, the first inequalities follow since $\mathbf{trace}(BB^T) = n_2|S|$, $\lambda_{min}(A(\mathcal{G})) = f_{min}$ and $\lambda_{max}(A(\mathcal{G})) = f_{max}$.

The following upper bound is also provided for the maximum eigenvalues of $P(A, B)$ in [136]:

$$\lambda_{max}(P(A, B)) \geq \frac{\lambda_{max}(BB^T)}{2\lambda_{max}^{1/2}(A^T A)}. \tag{8.5}$$

Then, it follows that as $\lambda_{max}(BB^T) = 1$ then,

$$\lambda_{max}(P) \quad \geq \quad \frac{1}{2\lambda_{\max}^{1/2}(A(\mathcal{G})A(\mathcal{G})^T)} = -\frac{1}{2\lambda_{\max}(A(\mathcal{G}))} = -\frac{1}{2f_{\max}}.$$

□

Proposition 8.5.2. *Let $A(\cdot) \in \boldsymbol{A}_{\oplus}$, $\mathcal{G} = \mathcal{G}_1 \square \mathcal{G}_2$ where $A(\mathcal{G})$ is stable and symmetric. Consider $P := P(A(\mathcal{G}), B)$, where $B = B_{n_1}(S_1) \otimes B_{n_2}(S_2)$, for some node set S_1 of $\mathcal{G}_1$ and S_2 of $\mathcal{G}_2$. Then, the following inequalities hold*

$$-\frac{|S_1|\,|S_2|}{2f_{max}} \leq \mathbf{trace}(P) \leq -\frac{|S_1|\,|S_2|}{2f_{min}}$$

and

$$\lambda_{max}(P) \geq -\frac{1}{2f_{max}},$$

where $f_{min} = \lambda_{min}(A(\mathcal{G}_1)) + \lambda_{min}(A(\mathcal{G}_2))$ and $f_{max} = \lambda_{max}(A(\mathcal{G}_1)) + \lambda_{max}(A(\mathcal{G}_2))$.

Proof. The proof follows from the proof presented for Proposition 8.5.1 using the fact that $\mathbf{trace}(B_{n_1}(S_1) \otimes B_{n_2}(S_2)) = \mathbf{trace}(B_{n_1}(S_1)) \cdot \mathbf{trace}(B_{n_1}(S_2)) = |S_1|\,|S_2|\,.$ □

Proposition 8.5.1 and 8.5.2 provide insight into generating large composite networks with effective control inputs, specifically via maximizing the trace and largest eigenvalue of the controllability Gramian.

The next theorem studies the relation between the controllability Gramian of the factor graphs and the composite graph, under the control scheme introduced in Theorem 8.4.1, for generalized control matrices B_1.

Theorem 8.5.3. *Let $\mathcal{G} = \mathcal{G}_1 \square \mathcal{G}_2$ where $A(\mathcal{G})$ and $A(\mathcal{G}_1)$ are stable. Then, $P(A(\mathcal{G}), B_1 \otimes I_{n_2})$ satisfies the equality*

$$W^T\ P(A(\mathcal{G}_1 \square \mathcal{G}_2), B_1 \otimes I_{n_2})\ W = \mathbf{D}\,(P_i)\,,$$

where $W = \Pi(V \otimes I_{n_2})$, V is a unitary matrix with columns corresponding to the left singular vectors of $A(\mathcal{G}_1) + A(\mathcal{G}_1)^T$ and $\Pi \in \mathbb{R}^{nm}$ is a permutation matrix defined through block matrix construct as $\Pi = \left[E_{ij}^T\right]_{\substack{i=1,\ldots,n \\ j=1,\ldots,m}}$. Here, $E_{ij} \in \mathbb{R}^{n \times m}$ has entry 1 in position i, j and zero otherwise. Also $P_i = P(A(\mathcal{G}_1) + \mu_i I_{n_1}, B_1)$, and $\mu_1, \ldots, \mu_{n_2}$ are the eigenvalues of

$A(\mathcal{G}_2)$, and the operator $D(M_i)$ for M_i forms a block diagonal matrix with diagonal blocks $M_1, M_2, \ldots, M_{n_2}$.

Proof. Let us recall the definition of the controllability Gramian as

$$\begin{aligned}
&P(A(\mathcal{G}), B_1 \otimes I_{n_2}) = \int_0^\infty e^{A(\mathcal{G})t} \left(B_1 \otimes I_{n_2}\right) \left(B_1 \otimes I_{n_2}\right)^T e^{A(\mathcal{G})^T t} dt \\
&= \int_0^\infty e^{(A(\mathcal{G}_1) \oplus A(\mathcal{G}_2))t} \left(B_1 \otimes I_{n_2}\right) \left(B_1 \otimes I_{n_2}\right)^T \times e^{(A(\mathcal{G}_1) \oplus A(\mathcal{G}_2))^T t} dt \\
&= \int_0^\infty \left(e^{A(\mathcal{G}_1)t} \otimes e^{A(\mathcal{G}_2)t}\right) \left(B_1 \otimes I_{n_2}\right) \left(B_1^T \otimes I_{n_2}\right) \times \left(e^{A(\mathcal{G}_1)^T t} \otimes e^{A(\mathcal{G}_2)^T t}\right) dt \\
&= \int_0^\infty \left(e^{A(\mathcal{G}_1)t} B_1 B_1^T e^{A(\mathcal{G}_1)^T t}\right) \otimes e^{\left(A(\mathcal{G}_2) + A(\mathcal{G}_2)^T\right)t} dt
\end{aligned}$$

which from the permutation equivalency of the Kronecker sum [123] is equivalent to

$$\begin{aligned}
\Pi^T P(A(\mathcal{G}), B_1 \otimes I_{n_2}) \Pi &= \int_0^\infty e^{\left(A(\mathcal{G}_2) + A(\mathcal{G}_2)^T\right)t} \otimes \left(e^{A(\mathcal{G}_1)t} B_1 B_1^T e^{A(\mathcal{G}_1)^T t}\right) dt \\
&= \int_0^\infty V \Sigma(t) V^T \otimes \left(e^{A(\mathcal{G}_1)t} B_1 B_1^T e^{A(\mathcal{G}_1)^T t}\right) dt, \qquad (8.6)
\end{aligned}$$

where Π is a permutation matrix and $e^{\left(A(\mathcal{G}_2) + A(\mathcal{G}_2)^T\right)t} = V \Sigma(t) V^T$ is the singular value decomposition as $\Sigma(t)$ is a diagonal matrix with $\Sigma(t)_{jj} = e^{2\mu_j t}$. Then, it follows that

$$\begin{aligned}
W^T P(A(\mathcal{G}), B_1 \otimes I_{n_2}) W &= \int_0^\infty \mathbf{D}\left[e^{2\mu_i t} \left(e^{A(\mathcal{G}_1)t} B_1 B_1^T e^{A(\mathcal{G}_1)^T t}\right)\right] dt \\
&= \mathbf{D}\left[\int_0^\infty e^{2\mu_i t} \left(e^{A(\mathcal{G}_1)t} B_1 B_1^T e^{A(\mathcal{G}_1)^T t}\right) dt\right] \\
&= \mathbf{D}\left[\int_0^\infty e^{\left(A(\mathcal{G}_1) + \mu_i I_{n_1}\right)t} B_1 B_1^T e^{\left(A(\mathcal{G}_1)^T + \mu_i I_{n_1}\right)t} dt\right] \\
&= \mathbf{D}\left[P(A(\mathcal{G}_1) + \mu_i I_{n_1}, B_1)\right]. \qquad (8.7)
\end{aligned}$$

where the operator $D(M_i)$ for M_i is defined in the theorem statement. The equation in (8.7) captures the relationship between the controllability Gramian of the composite graph and its factor graphs as

$$W^T \; P(A(\mathcal{G}_1 \square \mathcal{G}_2), B_1 \otimes I_{n_2}) \; W = \mathbf{D}\left(P_i\right),$$

where $W = \Pi(V \otimes I_{n_2})$, $P_i = P(A(\mathcal{G}_1) + \mu_i I_{n_1}, B_1)$, and $\mu_1, \ldots, \mu_{n_2}$ are the eigenvalues of $A(\mathcal{G}_2)$ as claimed in the statement. □

Next we discuss the ramifications of Equation (8.7) on generating large scale controllable graphs. The equation implies that

- when $A(\mathcal{G}_2)$ is stable with $\mu_i \leq 0$ for all i, we have $e^{2\mu_i t} \leq 1$, and therefore

$$P(A(\mathcal{G}_1) + \mu_i I_{n_1}, B_1) \preceq P(A(\mathcal{G}_1), B_1) \quad \forall\ i. \tag{8.8}$$

- when $A(\mathcal{G}_2)$ is unstable with $\mu_i \geq 0$ for all i, we have $e^{2\mu_i t} \geq 1$, and therefore

$$P(A(\mathcal{G}_1) + \mu_i I_{n_1}, B_1) \succeq P(A(\mathcal{G}_1), B_1) \quad \forall\ i. \tag{8.9}$$

The following is an example pertaining to the time-continuous results.

Example 8.5.4. *Define the matrix representation*

$$[A(\mathcal{G})]_{ij} = \begin{cases} -w_{ji} & \text{for } i \neq j \\ -20\sum_{i=1}^{n} w_{ii} + 11\sum_{i\neq j} w_{ji} & \text{otherwise} \\ \quad -10\sum_{i\neq j} w_{ji} & \end{cases}$$

In a more compact form $A(\mathcal{G}) = \mathcal{L}(\mathcal{G}) - 20\Delta_s(\mathcal{G}) + 10(\Delta_{in}(\mathcal{G}) - \Delta_{out}(\mathcal{G}))$. *Hence, as* $\boldsymbol{A}_{\oplus}$ *is closed under addition,* $A(\cdot) \in \boldsymbol{A}_{\oplus}$. *Consider the graphs* $\mathcal{G}_1$ *and* $\mathcal{G}_2$ *from Figure 8.1. Then* $A(\mathcal{G}_2) = \mathcal{L}(\mathcal{G}_2)$ *and so it has eigenvalues* $0 = \mu_1 < \mu_2 \leq \mu_3 \leq \mu_4$ *and*

$$A(\mathcal{G}_1) = \begin{bmatrix} -15 & -1 & 0 \\ -1 & -6 & 0 \\ 0 & -1 & -8 \end{bmatrix},$$

which has all strictly negative eigenvalues. From Equation (8.8) then $\lambda_{min}(P(A(\mathcal{G})), B_1 \otimes I) = \lambda_{min}(P(A(\mathcal{G}_1), B_1))$. *Therefore, the optimal single node input set* S_1 *in terms of the smallest controllability Gramian for the graph* $\mathcal{G}_1$ *generates the optimal set* S *for the graph* $\mathcal{G}$ *of the form* $B_1(S_1)\square\, I_4$. *The smallest eigenvalues of* $P(A(\mathcal{G}_1), B_1)$ *from single input nodes* $1'$, $2'$ *and* $3'$ *are* 0.0218×10^{-5}, 0.3653×10^{-5}, *and 0, respectively. Thus the optimal set is* $S_1 = \{2'\}$ *for* $\mathcal{G}_1$ *and* $S = \{\{1, 2'\}, \{2, 2'\}, \{3, 2'\}, \{4, 2'\}\}$ *for* $\mathcal{G}$.

As mentioned in §8.1, the discete version of our results follow directly from the continuous proofs. As an example, the discrete version of Theorem 8.5.3 for the observability Gramian is proved here and will subsequently be used in the social network example in §8.7.

Theorem 8.5.5. *Let $\mathcal{G} = \mathcal{G}_1 \square \mathcal{G}_2$ where $A(\mathcal{G})$ and $A(\mathcal{G}_1)$ are stable. Then, $P(A(\mathcal{G}), C_1 \otimes I_{n_2})$ satisfies the equality*

$$W^T \; P(A(\mathcal{G}_1 \square \mathcal{G}_2), C_1 \otimes I_{n_2}) \; W = \mathbf{D}\left(P_i\right),$$

where $W = \Pi(V \otimes I_{n_2})$, V is a unitary matrix with columns corresponding to the left singular vectors of $A(\mathcal{G}_1) + A(\mathcal{G}_1)^T$ and $\Pi \in \mathbb{R}^{nm}$ is a permutation matrix defined through block matrix construct as $\Pi = \left[E_{ij}^T\right]_{\substack{i=1,\ldots,n \\ j=1,\ldots,m}}$. Here, $E_{ij} \in \mathbb{R}^{n \times m}$ has entry 1 in position i, j and zero otherwise. Also $P_i = P(A(\mathcal{G}_1) + \mu_i I_{n_1}, C_1)$, and $\mu_1, \ldots, \mu_{n_2}$ are the eigenvalues of $A(\mathcal{G}_2)$, and the operator $D(M_i)$ for M_i forms a block diagonal matrix with diagonal blocks $M_1, M_2, \ldots, M_{n_2}$.

Proof. Let us recall the definition of the observability Gramian as

$$P(A(\mathcal{G}), C) = \sum_k (A(\mathcal{G})^T)^k C^T C A(\mathcal{G})^k,$$

where $C = C_1 \otimes I_{n_2}$. Then the proof follows from the same steps as Theorem 8.5.3. □

8.6 Layered Output Feedback

An attraction of composite networks is that they exhibit repeated layers of the factors. Theorem 8.2.1 takes advantage of this by extending the controllable inputs in one factor layer to many. The same can be done to the observable outputs. The next proposition shows that the control signal can similarly be designed for a factor and extended to the composite network with the effect of generating distributed output feedback stabilization.

Proposition 8.6.1. *Consider $A(\cdot) \in \boldsymbol{A}_{\oplus}$ and the n_1 node and n_2 node graphs $\mathcal{G}_1$ and $\mathcal{G}_2$, where the matrix $A(\mathcal{G}_2)$ is semistable. If the dynamics $(A(\mathcal{G}_2), B_2, C_2)$ is stabilizable with output feedback $u_a = K y_a$ for inputs u_a and outputs y_b, then the dynamics $(A(\mathcal{G}_1 \square \mathcal{G}_2), I_{n_1} \otimes B_2, I_{n_1} \otimes C_2)$ is stabilizable with the output feedback $u = (K \otimes I_{n_2}) y$ for inputs u and outputs y. Further, the control law can be realized with local layer feedback across the layers of $\mathcal{G}_1$.*

Proof. For convenience, we present the equivalent result in terms of the layers of $\mathcal{G}_2$. As K is a stabilizing feedback for the system described by the matrices $(A(\mathcal{G}_2), B_2, C_2)$, then $A(\mathcal{G}_2)+B_2KC_2$ is stable. Consider the dynamics of the system $(A(\mathcal{G}_1 \square \mathcal{G}_2), I_{n_1} \otimes B_2, I_{n_1} \otimes C_2)$ with output feedback $u = (I_{n_1} \otimes K)y$. Then,

$$\begin{aligned} \dot{x}(t) &= \left(A(\mathcal{G}_1 \square \mathcal{G}_2) + (I_{n_1} \otimes B_2)\,(I_{n_1} \otimes K)\,(I_{n_1} \otimes C_2)\right) x(t) \\ &= \left(A(\mathcal{G}_1) \otimes I_{n_2} + I_{n_1} \otimes A(\mathcal{G}_2) + I_{n_1} \otimes B_2KC_2\right) x(t) \\ &= \left(A(\mathcal{G}_1) \oplus (A(\mathcal{G}_2) + B_2KC_2)\right) x(t). \end{aligned}$$

As the Cartesian sum of semistable and stable matrices is stable, then $I_{n_1} \otimes K$ is stabilizing.[2] Further, as $I_{n_1} \otimes K$ is block diagonal the feedback loop can be broken into the inputs and outputs of each layer of $\mathcal{G}_2$. Specifically distributing the inputs and outputs, we have $u = \left[u_1^T, \ldots, u_{n_1}^T\right]^T$ and $y = \left[y_1^T, \ldots, y_{n_1}^T\right]^T$, where u_i and y_i are the inputs and outputs of the ith layer of $\mathcal{G}_2$. Hence, the feedback can be written as $u_i = Ky_i$ for $i = 1, \ldots, n_1$, i.e., local layer feedback. □

Proposition 8.6.1 describes a setup where we have a stabilizing distributed feedback on each factor layer requiring only local sensors and actuators placed on that layer. The following is an example illustrating this layered output feedback stabilization.

Example 8.6.2. *Define the matrix representation*

$$[A(\mathcal{G})]_{ij} = \begin{cases} w_{ji} & \text{for } i \neq j \\ \frac{1}{2}w_{ii} - \sum_{i \neq j} w_{ji} & \text{otherwise.} \end{cases}$$

An equivalent form is $A(\mathcal{G}) := -L(\mathcal{G}) + \frac{1}{2}\Delta_s$, *and so* $A(\cdot) \in \boldsymbol{A}_{\oplus}$. *For the graphs* $\mathcal{G}_1$ *and* $\mathcal{G}_2$ *described in Figure 8.1,* $A(\mathcal{G}_2) = -L(\mathcal{G}_2)$ *which is a semistable matrix and*

$$A(\mathcal{G}_1) = \begin{bmatrix} -\frac{1}{2} & 1 & 0 \\ 1 & -1 & 0 \\ 0 & 1 & -\frac{1}{2} \end{bmatrix}.$$

[2]Since each eigenvalue of the composite matrix lies on the left half plane.

These dynamics are unstable. Consider the dynamics of the system described by the matrices $(A(\mathcal{G}_1), B(S_1), C(S_2))$ *where* $S_1 = \{1'\}$ *and* $S_2 = \{2'\}$*, then the output feedback* $u = ky$ *is stabilizing for* $k < -\frac{1}{2}$*. Therefore, from Proposition 8.6.1, the output feedback* $k \otimes I_4$ *is stabilizing for the composite system, which is realized by the distributed feedback* $u_{(1',i)} = ky_{(2',i)}$ *for* $i = 1, \ldots, 4$*, where* $u_{(1',i)}$ *is the input applied to node* $(1', i)$ *and* $y_{(2',i)}$ *is the output measured from node* $(2', i)$*.*

8.7 Social Network Example

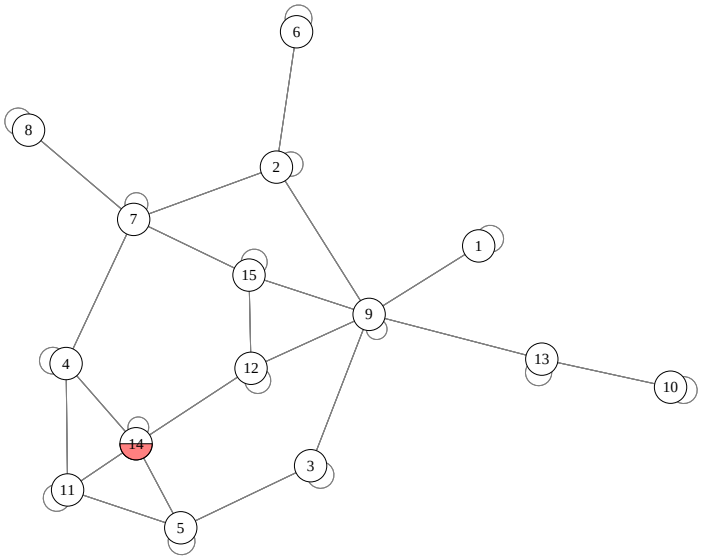
The goal in this section is to estimate the internal state of a social influence network in a neighborhood utilizing the observability properties of the Cartesian product networks. The model is adopted from [72, 73] and it has been under development by social psychologists and mathematicians since the 1950's [72]. The time invariant model described in [72] and the time varying model in [73] postulate simple recursive definitions for the influence process in a group of N agents. For the time invariant case,

$$x(t+1) = WF(\mathcal{G})x(t) + (I - W)x(0) \quad t = 1, 2, \ldots, \tag{8.10}$$

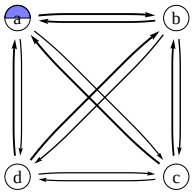
where $x(0)$ is an $N \times 1$ vectors of agents' initial opinions on an issue, $x(t)$ is an N vector of agents' opinions at time t. The interpersonal influences are captured in $F(\mathcal{G}) \in \mathbf{R}^{N \times N}$ where $0 \leq F_{ij} \leq 1$, $\sum_j^N F_{ij} = 1$ and $\mathcal{G}$ is the underlying communication network. The diagonal matrix $W = \mathbf{diag}(w_{11}, w_{22}, \cdots, w_{NN})$ is the agents' susceptibilities to interpersonal influence on the issue ($0 \leq w_{ij} \leq 1$).

A Cartesian network structure of dynamics is not uncommon in social networks due to the layered structure of society. For example, consider a network of nuclear families where each family interacts with other families. Interactions between families involves common gender parents interacting with same gender parents and similarly children with same aged children. This network can be realized through a Cartesian product of a family member graph and inter-family graph.

This example was analyzed for fifteen, four member families. The family member graph $\mathcal{G}_2$ with nodes a, b, c and d corresponding to a father, mother, older child and younger child is depicted with corresponding $A(\mathcal{G}_2)$ matrix in Figure 8.1b. Similarly, the family interaction

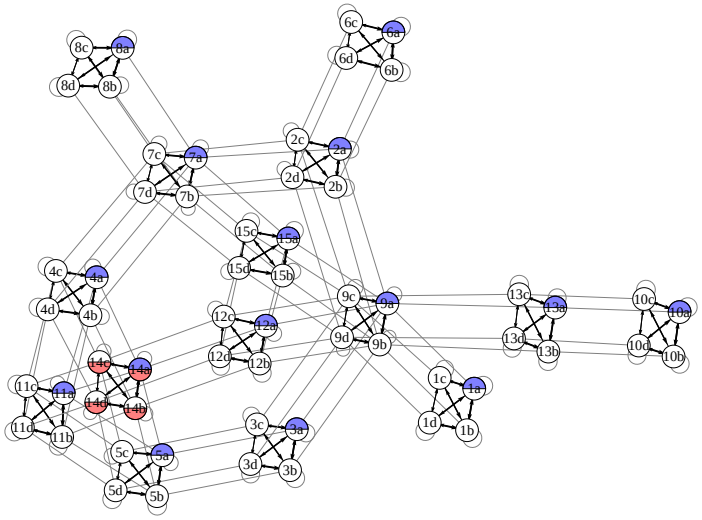


(a) $\mathcal{G}_1$



$$\hat{E}(\mathcal{G}_2) = \begin{bmatrix} 0 & 0.4 & 0.1 & 0.1 \\ 0.3 & 0 & 0.2 & 0.2 \\ 0.4 & 0.3 & 0 & 0.1 \\ 0.35 & 0.45 & 0.1 & 0 \end{bmatrix}$$

(b) $\mathcal{G}_2$



(c) $\mathcal{G}_1 \square \mathcal{G}_2$

Figure 8.1: Left: Factor graphs $\mathcal{G}_1$ and $\mathcal{G}_2$ and composite graph $\mathcal{G}_1 \square \mathcal{G}_2$. For image clarity in $\mathcal{G}_1$, all edge weights are 1, and in $\mathcal{G}_2$ and $\mathcal{G}_1 \square \mathcal{G}_2$ the edge weights can be divulged through the definition of $\hat{E}(\mathcal{G}_2)$ and $\mathcal{G}_1$.

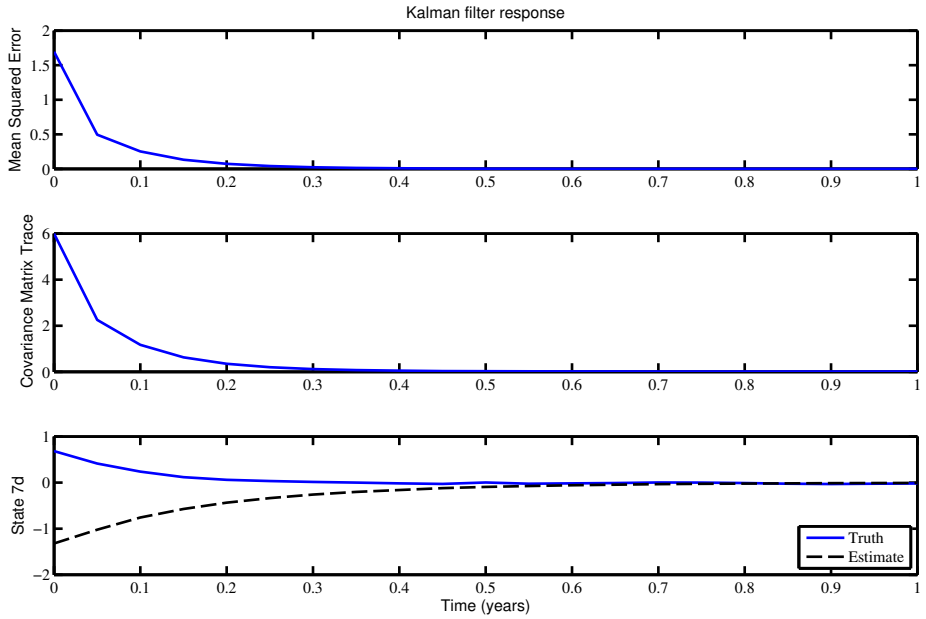


Figure 8.2: The mean squared error, covariance matrix trace and state of a random node (node 7d) over time for the discrete Kalman filter pertaining to Figure 8.1.

graph $\mathcal{G}_1$, based on the famous Florentine family graph [87], is depicted in Figure 8.1a with nodes representing the fifteen families. The composite graph $\mathcal{G}_1 \square \mathcal{G}_2$ corresponds to the 60 members' social interactions. The discrete time opinion dynamics of the interaction have a state matrix $A(\mathcal{G}) = \exp(-\delta(\mathcal{L}(\mathcal{G}) + \max(\Delta_s(\mathcal{G}))I))$ and $\boldsymbol{A}_\otimes$ is closed under multiplication $A(\cdot) \in \boldsymbol{A}_\otimes$.[3] This is an adaptation of the traditional opinion dynamics in (12.1) when $W = I$ as $\exp(-\delta\mathcal{L}(\mathcal{G}))$ is of the form of $F(\mathcal{G})$ with the addition of the self loop term, increasing a state's dependence on its own opinion.

It is often unrealistic or overly expensive to make a census of the full population of a social group. An alternative is to sample the network and subsequently estimate its state dynamics. A requirement for an estimator to converge is observability. If S is the input

[3]The matrix representations $A_1(\mathcal{G}) = \exp(-\delta\mathcal{L}(\mathcal{G}))$ and $A_2(\mathcal{G}) = \exp(-\delta \max(\Delta_s(\mathcal{G}))I)$ are both in $\boldsymbol{A}_\otimes$ and $A(\mathcal{G}) = A_1(\mathcal{G})A_2(\mathcal{G})$ as $\mathcal{L}(\mathcal{G})$ and $\max(\Delta_s(\mathcal{G}))I$ commute [123].

nodes of the graph, the input-output discrete social interaction dynamics are

$$x(k+1) = A(\mathcal{G}_1 \square \mathcal{G}_2)x(k) \tag{8.11}$$
$$y(k) = C_{60}(S)x(k),$$

where $t = k\delta$ for time step $\delta = 0.05$ year.

Assuming that all the members of one demographic can be observed, e.g., all mothers, then Theorem 8.5.5 provides the optimal selection. Specifically, the dynamics are observable and, under one demographic, optimal for $C_{60}(S) = I \otimes C_4(S_2)$ where $S_2 = \{a\}$ and $S = \{1a, 2a, \ldots, 15a\}$. This information would be attractive to an advertiser, as the opinions of all 60 people can be efficiently divulged by surveying the fathers in the society. If instead an advertiser was interested in the best family to survey, then Theorem 8.5.5 can be applied to $\mathcal{G}_1$ giving the observable dynamics with $C_{60}(S) = C_{15}(S_1) \otimes I$ where $S_1 = \{14\}$ and $S = \{14a, 14b, 14c, 14d\}$, i.e., every member of family 14 is observed directly. As $A(\mathcal{G}_1 \square \mathcal{G}_2)$ has simple eigenvalues, using Theorem 8.2.1 the observation matrix $C_{60}(S) = C_{15}(S_1) \otimes C_4(S_2)$ where $S = \{14a\}$ is observable. Therefore, in this scenario, surveying the father of family 14 would provide the opinion of all members of the society.

A discrete Kalman filter[4] was applied to dynamics (8.11) with $\mathcal{G}_1$ and $\mathcal{G}_2$ depicted in Figure 8.1 and $S = \{1a, 2a, \ldots, 15a\}$, i.e., all the fathers in the society are observed. The resultant mean squared error, covariance matrix trace, and a sample state estimate over time is provided in Figure 8.2, supporting the observability of the pair $(A(\mathcal{G}_1 \square \mathcal{G}_2), C_{60}(S))$.

[4]For a detailed description of the discrete Kalman filter see [197].

Part III

SYSTEM PROPERTIES OF STOCHASTIC NETWORKS

Chapter 9

SYSTEM PROPERTIES OF STOCHASTIC NETWORKS: CONTROLLABILITY AND OPTIMALITY

In this chapter, we first examine the pertinent necessary and sufficient conditions for controllability and observability of protocols over random networks . The randomness could be presented in the network topology, e.g., whether two nodes are connected based on a random distribution. The randomness could also be contributed by the randomly selected input and/or output nodes. We, then, present a framework for designing estimator and feedback controllers for observing and directing a group of agents adopting a consensus-based protocol over random networks. We will discuss the adopted protocols as they are introduced.

In order to design the feedback controllers, we need to first explore the controllability of the network. We proceed to explore conditions on the underlying distributions of the random network to guarantee the controllability; designing optimal infinite horizon linear quadratic regulators in the random setup then follows. The rate of convergence of the designed feedback controller is then examined. A practical example on opinion dynamics and optimal marketing strategies over social networks is used to support the analysis presented in this section.

On the estimation side, the stability and convergence properties coordinated decentralized scheme in the random setting are explored utilizing contractive properties of random Riccati equation. The performance of the estimator is evaluated as a function of the probability distribution of the random network. Two examples on coordinated decentralized estimation of opinion dynamics over social networks and real-time underwater acoustic localization for the *Seaglider*[1] vehicle are presented.

---

[1]The example is based on the experimental data courtesy of Prof. Morgansen's Nonlinear Dynamics and Control Lab at the University of Washington, `http://www.aa.washington.edu/research/ndcl/`.

9.1 Introduction

Random networks appear in many applications, e.g., in modeling complex networks [61, 62, 63, 65, 66, 76], modeling social networks [64, 76], and sensor networks [41, 42, 35, 36, 37, 44].

As an illustration consider social networks. The literature reveals a broad class of research that investigate real-world social networks in different scenarios such as friendship networks, the collaboration network of film actors, cooperation between companies, network of directors of fortune 1000 companies, and scientific collaboration networks of engineers and biologists, among many other examples. The results include a wide range of models, some of which are Erdős-Rényi model [16], and small-world model proposed by Watts and Strogatz [61]; *small-world* networks lie somewhere between regular and completely random networks.

Scale-free networks, as another model of real world social networks, studied in [63] are based on two ingredients: (1) continuous expansion of large networks by the addition of new vertices and (2) preferential attachment of new vertices to already well-connected vertices. A model of large networks based on these two ingredients causes a stationary scale-free power-law distributions.

In another model studied in [64], the goal is to model a large network for which only the degree distribution is known. In this model, the normalized probability p_k that a randomly chosen vertex in the network has degree k is given. In order to model a network with the pre-defined degree distribution, Newman et al. [64] take a number N of vertices and assign to each a random number k of "stubs" chosen independently from the distribution p_k for each vertex. Then, the stubs are randomly selected in pairs and joined by edges.

As another illustration of random networks, consider a network of N spatially distributed autonomous sensors, in which each sensor collects measurements in some modality of interest, e.g., temperature, sound, vibration, pressure, motion, or pollutant. Each sensor in the network is equipped with small storage, a radio transceiver, a micro-controller, and a battery power source on a single chip. Engineering sensor networks consist of different number of sensors; from few numbers to large numbers of sensors, hundreds to even hundred thousand nodes [35, 36, 37].

Sending large amounts of raw measurements to the fusion center requires a very powerful processor with large bandwidth to gather the data. Significant communication delays and data loss across the network are common features of these systems. On the one hand, significant delays and packets drop have a negative impact on the performance of the designed control, as data need to arrive at their destination in time to be used for control or estimation purposes. These types of inefficiencies can be considered with stochastic models.

Battery lifetime is a major concern in all engineering applications. Specially in some cases, for instance habitat monitoring, non-rechargeable batteries are employed to observe the evolution of a particular phenomenon in nature for a couple of months [38]. In some other applications, reactive sample rates matter. For example in soil moisture sensor networks [39], sensors observe some dynamics, which change rapidly during some intervals and slowly during other intervals. Consequently, soil moisture monitoring sensors need to send the measurements at higher rates during rain events and at lower rates between rain falls. The constraints on battery sources, on the other hand, motivate us to switch off a group of sensors during some interval of measurement or communication with the fusion center, which can be demonstrated by a random communication network between the sensors or the sensors and the fusion center/node.

Considering a large, wireless, multi-hop sensor network, there are too much data and too little bandwidth to pass the information around. When data travels along unreliable communication channels, the effect of communication delays, and loss of information cannot be neglected. The stochastic nature of the communication channel and the arrival of the observations can be modeled as a random process [40, 43].

9.2 Problem Formulation

In a random network, the existence of an edge between a pair of vertices in the set $\mathcal{V}$ is determined based on a random distribution, e.g., based on Erdős-RT̂enyi distribution [16] an edge exists with probability $p \in (0, 1]$ and independent of other edges. The sample space of all such random graphs will be denoted by $\mathcal{G}(n, p)$. Note that the value of edge probability can be the same or distinct for all potential edges. This probability can also be fixed, or in more interesting scenarios, a function of the order of the graph, $p(n)$.

We consider a random interactive network with associated Laplacian matrix $L(w_t)$, where w_k's are sequence of mutually independent random events. Having embed a random network in the dynamic system, it is convenient to consider an arbitrary sampling of the time axis at intervals $\Delta > 0$. Therefore, the dynamic can be expressed in the sampled-data form

$$x(t+1) = A(w_t)x(t) + B(w_t)u(t), \tag{9.1}$$

where

$$A(w_t) = e^{-L(w_t)\Delta}, \quad x(t) := x(\Delta t), \tag{9.2}$$

and $B(w_t) \in \mathbb{R}^{n\times m}$ is the input matrix with m input signals applied to m nodes.

A sample sequence of a random network on 8 nodes and a sample of random network in a swarm of ground robots are depicted in Figure 9.1 and 9.2, respectively.

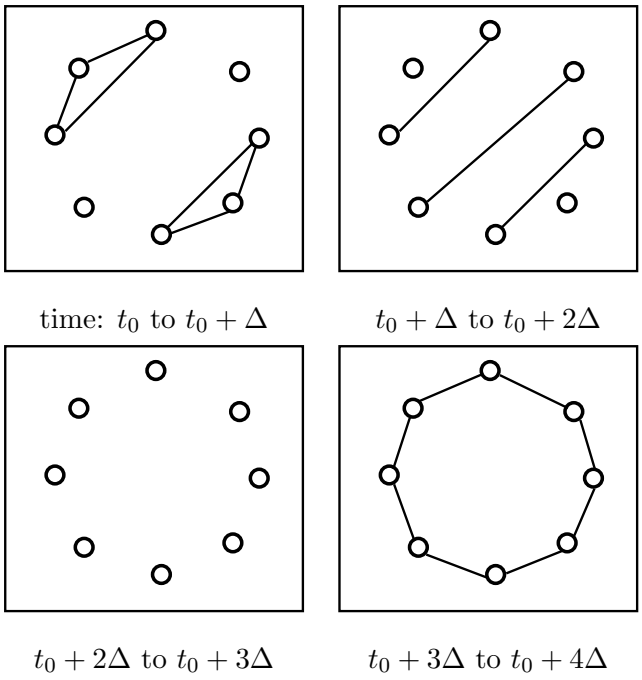
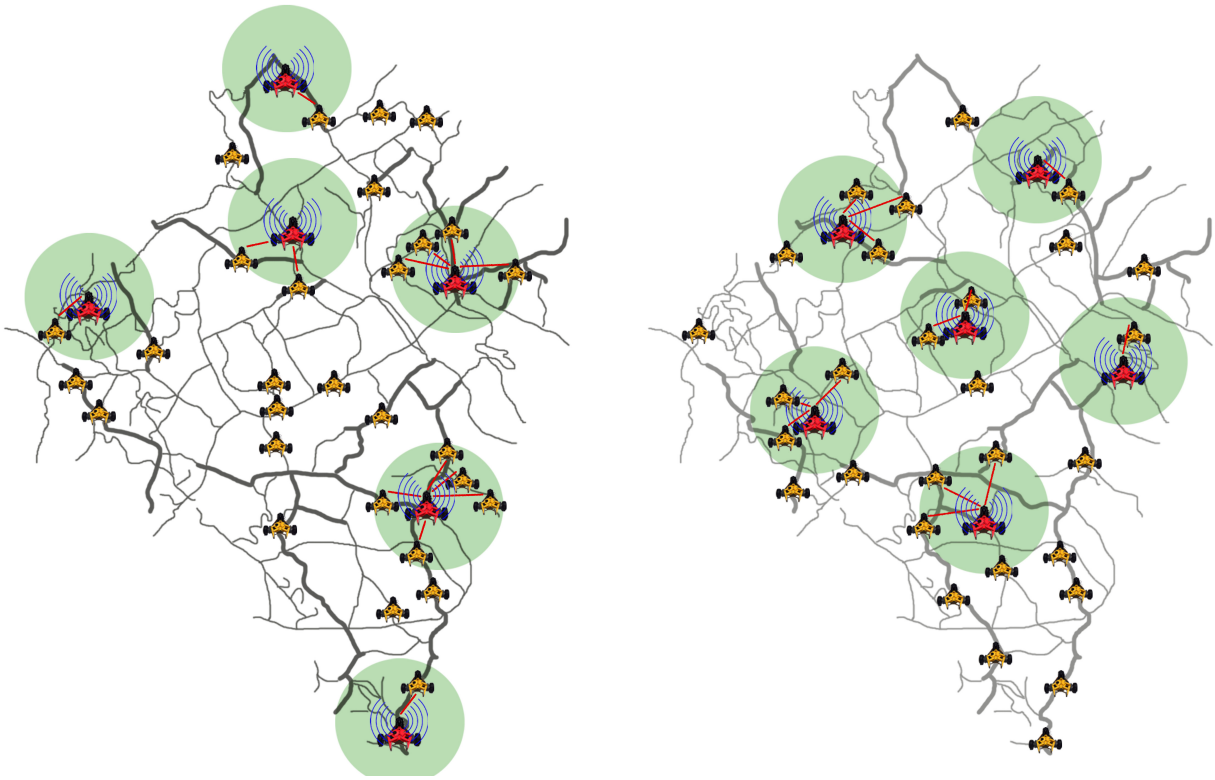


Figure 9.1: Behavior of a random network over a time interval

In order to consider the sampled-data setting for the Dirichlet dynamics, described in chapter 1, we need to partition the matrix exponential $e^{-L(w_t)\Delta}$. If the m input nodes are



time: t_0 to $t_0 + \Delta$ $\quad$ $t_0 + \Delta$ to $t_0 + 2\Delta$

Figure 9.2: Behavior of random communication in a swarm of ground robots over two time intervals

the first set of m nodes, $e^{-L(w_t)\Delta}$ can be then partitioned as

$$e^{-L(w_t)\Delta} = \begin{bmatrix} * & B^T(w_t) \\ B(w_t) & A(w_t) \end{bmatrix}. \tag{9.3}$$

In a more general scenario, when the input nodes are not the first m nodes, the partitioning can be done in the similar manner as (9.3). Therefore, the dynamic system can also be expressed as (9.1) with matrices described in (9.3). It is assumed that (9.1) is stationary, i.e., the probability distributions of $A(w_t)$ and $B(w_t)$ do not depend on t.

Whether we consider the consensus or Dirichlet dynamics, we have a dynamic system described by random matrices $A(w_t)$ and $B(w_t)$, altering at every time interval based on the independent random events w_t. For the simplicity of notation, $A(w_t)$ and $B(w_t)$ will be subsequently referred to as A_t, and B_t, respectively.

The objective of this chapter is to study the system properties of a group of agents adopting consensus-type algorithms over random networks. In order to achieve this goal, controllability and observability are the fundamental properties to examine/study first.

9.3 Controllability/Observability of Stochastic Systems

Controllability for networked systems is of fundamental importance among the many topics of interest in networked systems. The basic issue, e.g., in controllability for protocols evolving on random networks, is whether it is possible to transfer any initial state to the desired state, in some stochastic sense, by applying a judicious control input. Note that the observability is the dual notion of the controllability. Therefore, the results of the controllability can be applied equivalently to the observability problem. The controllability issue comprises the focus of this section.

In [223] and [18], it has been proven that a weak observability condition is necessary for Kalman filter over the random networks to converge asymptotically surely (a.s.). Following the duality principle, we now introduce the weak controllability condition as follows.

Definition 9.3.1. *Let $S_t = B_t B_t^T$. Then, the linear system (9.1), or equivalently, the sequence $\{(A_t, B_t),\ t \in \mathbb{Z}\}$, is called weakly controllable if for some $t \geq 1$,*

$$\mathbb{P}\{\mathbf{det}\{S_t + A_t S_{t-1} A_t^T + \ldots + (A_t \ldots A_2) S_1 (A_2^T \ldots A_t^T)\} \neq 0\} \neq 0. \tag{9.4}$$

The probability constraint in (9.4) implies that the controllability Gramian of the discrete and random system in (9.1) should be full rank. It is theoretically convenient to consider the system characteristics as elements of appropriate Hilbert space L_2.

In order to gain more intuition on the controllability of the random consensus and Dirichlet dynamics in (9.1) and (9.3), first, let us provide general definitions on stochastic controllaility that have been introduced in the jump-parameter systems literature [19, 20].

Definition 9.3.2. *A stochastic system is considered to be:*

- *Weakly state controllable if for all $x_0 \in \mathbb{R}^n$, $x_1 \in \mathbb{R}^n$, and all $\epsilon \geq 0$, there exists a random time T a.s. finite and a control law u defined on $[0, T]$ such that*

$$\mathbb{P}\{||x(T) - x_1|| \leq \epsilon\} > 0, \tag{9.5}$$

where $x(T)$ denotes the value at $t = T$ of the trajectory starting in x_o at $t = 0$ under the control u.

- *State controllable if the probability defined in (9.5) can be made equal to one.*

- *Strongly state controllable if the hitting time $T_H = \inf\{t > 0; ||x(t) - x_1|| \leq \epsilon\}$ has finite expectation, i.e., $\mathbb{E}\{T_H\} < +\infty$.*

The above notions of controllability are referred to as "weakly controllable", "controllable", and "strongly controllable", respectively, in the jump-parameter systems literature [19, 20]. The labeling convention is partially adopted to avoid confusion with Definition 9.3.1.

We will prove in more details that the condition in Definition 9.3.1 is the appropriate notion to guarantee the error defined as $x(T) - x_1$ converges asymptotically surely, i.e., for all $x_0 \in \mathbb{R}^n$, $x_1 \in \mathbb{R}^n$, and all $\epsilon \geq 0$, there exists T such that

$$\mathbb{P}\{||x(T) - x_1|| > \epsilon\} = 0, \quad \text{or} \quad \mathbb{P}\{||x(T) - x_1|| \leq \epsilon\} = 1. \tag{9.6}$$

Thus, if (9.4) is valid, the system is state controllable. In the next step, we show that weakly state controllability is not indeed the appropriate notion for the development of a feedback regulator theory over random networks.

In order to examine the necessary condition for designing a feedback regulator over random networks, we first notice that the state controllability, strong state controllability, and weak state controllability are analogous to the controlled version of recurrence, positive recurrence, and non degeneracy (or weak recurrence), respectively, of the Markov chain defined by (9.1). To clarify the analogy, let us start with recalling the classification of states in Markov chain.

A Markov chain on the finite or infinite state space domain $\mathcal{D}$ is called

- recurrent if for all $i \in \mathcal{D}$ and for all t_0, there exists an almost surely finite random time $t > t_0$ such that $\mathbb{P}\{x(t) = i | x(t_0) = i\} = 1$; or equivalently the state i is recurrent if

$$\mathbb{P}\{x(t) = i \text{ for infinitely many } t\} = 1. \tag{9.7}$$

Then, the Markov chain is called recurrent if (9.7) holds for all $i \in \mathcal{D}$.

- positive recurrent if the probability that $x(t) = i$ for infinitely many t is equal to 1 and in addition the return time $T_R = \inf\{t > 0, x(t) = i \,|\, x(0) = i\}$ has a finite expectation.

- weak recurrent or nondegenerate if $\mathbb{P}\{x(t) = i | x(t_0) = i\} > 0$; or equivalently the state i is weak recurrent if

$$\mathbb{P}\{x(t) = i \text{ for infinitely many } t\} > 0. \tag{9.8}$$

  Then, the Markov chain is called weakly recurrent if (9.8) holds for all $i \in \mathcal{D}$.

- transient if $\mathbb{P}\{x(t) = i | x(t_0) = i\} = 0$; or equivalently the state i is transient if

$$\mathbb{P}\{x(t) = i \text{ for infinitely many } t\} = 0.$$

  Then, the Markov chain is called transient if this holds for all $i \in \mathcal{D}$.

The purpose of the following section is to demonstrate that every state is either recurrent or transient. In this direction, let us specify the random variable T_i as the first passage time to state i as

$$T_i(w_t) = T_i = \inf\{t \geq 1 : x(t) = i\}. \tag{9.9}$$

We now define inductively the rth passage time to state i by $T_i^{(r)}$ as

$$T_i^{(r+1)}(w_t) = \inf\{n \geq T_i^{(r)}(w_t) + 1 : x(t) = i\} \quad \text{for} \quad r = 0, 1, 2, \ldots, \tag{9.10}$$

where $T_i^{(0)}(w_t) = 0$, and $T_i^{(1)}(w_t) = T_i(w_t)$.

Let us also introduce the *number of visits* V_i to the state i, which can be written in terms of the indicator function $1_{\{x(t)=i\}}$ as

$$V_i = \sum_{t=0}^{\infty} 1_{\{x(t)=i\}} \tag{9.11}$$

and it follows that

$$\mathbb{E}(V_i) = \sum_{t=0}^{\infty} \mathbb{P}\{x(t) = i\}. \tag{9.12}$$

Now we are ready to state the next theorem to point out that every state of a discrete Markov chain in a finite or an infinite state space $\mathcal{D}$ is either recurrent or transient.

Theorem 9.3.3. *[13] The following dichotomy holds:*

- *if* $\mathbb{P}\{T_i < \infty\} = 1$, *then the state* i *is recurrent,*
- *if* $\mathbb{P}\{T_i < \infty\} < 1$, *then the state* i *is transient,*

where T_i *is defined in (9.9). In particular, every state is either transient or recurrent.*

Proof. If $f_i = \mathbb{P}\{T_i < \infty\}$, then it can be shown by induction that $\mathbb{P}\{V_i > r\} = f_i^r$. Assuming this, if $\mathbb{P}\{T_i < \infty\} = 1$, then

$$\mathbb{P}\{V_i = \infty\} = \lim_{r\to\infty} \mathbb{P}\{V_i > r\} = 1$$

so state i is recurrent and $\mathbb{E}(V_i) = \infty$.

On the other hand, if $\mathbb{P}\{T_i < \infty\} < 1$, then

$$\mathbb{E}(V_i) = \sum_{r=0}^{\infty} \mathbb{P}\{V_i > r\} = \sum_{r=0}^{\infty} f_i^r = 1/(1 - f_i) < \infty, \tag{9.13}$$

so $\mathbb{P}\{V_i = \infty\} = 0$ and i is transient. □

Considering the result in Theorem 9.3.3, if the state i is weakly recurrent and

$$\mathbb{P}\{x(t) = i \ \text{ for infinitely many } t\} \neq 1,$$

the analogous controlled Markov chain built on (9.1) does not hit the desired state infinitely often. Now the question is whether weak state controllability or state controllability is the necessary condition for designing a feedback controller.

In order to examine the weak state or state controllability as the necessary condition for designing a feedback controller, let us restate the control problem of interest. The control problem is in fact a regulator problem where we consider choosing a suitable control function $u(t)$ to ensure that every initial state $x(t_0) = x_0$ is derived to the reference signal $x = 0$ such that a performance index is minimized and the origin is stable.

The reference signal $x = 0$ of (9.1) is asymptotically stable in the mean square (or in the norm) if

$$\mathbb{E}(||x(t, x_0)||^2) < \infty \quad \text{and} \quad \lim_{t\to\infty} \mathbb{E}(||x(t, x_0)||^2) = 0 \quad \text{for all } x_0. \tag{9.14}$$

Similarly, we have asymptotic stability with probability one if

$$||x(t, x_0)|| < \infty \quad \text{and} \quad \lim_{t\to\infty} ||x(t, x_0)||^2 = 0 \tag{9.15}$$

hold with probability one.

In the next step, following the same line of reasoning used in classification of Markov chains, we prove that states from the weak state controllability subset, denoted by $\mathfrak{C}_w$, do not hit the weak controllable subset infinitely often.

Theorem 9.3.4. *Let $\mathfrak{C}_{\mathfrak{w}} \subseteq \mathbb{R}^n$ consists of states that satisfy the weak state controllability in Definition 9.3.2. If $x_1 \in \mathfrak{C}_w$, then there is no T a.s. finite, such that the state $x(T)$ hits the subset $\mathfrak{C}_w$ infinitely often.*

Proof. Let us introduce $T_1 = \inf\{t \geq 1 : x(t) = x_1\}$. A disk with radius ϵ centered at x_1 is indicated by $D_\epsilon(x_1)$. The number of times $x(T)$ hits the disk $D_\epsilon(x_1)$ while $||x(T+\delta)-x_1|| \leq ||x(T) - x_1||$ for $\delta > 0$ is denoted by V_1. Therefore, the definition of V_1 is as follows

$$V_1 = \sum_{t=0}^{\infty} 1_{\{||x(T)-x_1||\leq\epsilon\}} \tag{9.16}$$

and note that its expectation is

$$\mathbb{E}(V_1) = \sum_{t=0}^{\infty} \mathbb{P}\{||x(T) - x_1|| \leq \epsilon\}. \tag{9.17}$$

Now the claim is that $\mathbb{P}\{T_1 < \infty\} < 1$ and it holds since $x_1 \in \mathfrak{C}_w$, and therefore, $\mathbb{P}\{||x(T) - x_1|| \leq \epsilon\} > 0$. Then, the definition of T_1 implies the claim.

Let us define $f_1 = \mathbb{P}\{T_1 < \infty\} < 1$. Hence, the expectation of V_1 is formalized as

$$\mathbb{E}(V_1) = \sum_{t=0}^{\infty} \mathbb{P}\{V_1 > r\} = \sum_{t=0}^{\infty} f_1^r = 1/(1 - f_1) < \infty.$$

The last inequality implies that $\mathbb{P}\{V_1 = \infty\} = 0$, and therefore, the system does not hit $D_\epsilon(x_1)$ infinitely often. □

Theorem 9.3.4 indicates that for the controllability purposes one should require all states be in the state controllability set and not in the weak controllability set. The next theorem gives an algebraic condition, in the probabilistic sense, to check the weak controllability criteria defined in Definition 9.3.1.

Theorem 9.3.5. *The system (9.1) is weakly controllable if and only if for some* $t \geq 1$,

$$\mathbb{P}\{\textbf{rank } (B_t, A_tB_{t-1}, A_tA_{t-1}B_{t-2}, \ldots, A_t \ldots A_1B_0) = n\} \neq 0. \tag{9.18}$$

Proof. Sufficiency: The stochastic linear system (9.1) can be written as

$$\begin{aligned} x(t+1) &= A_tx(t) + B_tu(t) \\ &= A_tA_{t-1} \ldots A_0x(0) + A_t \ldots A_1B_0u(0) + A_t \ldots A_2B_1u(1) + \ldots + B_tu(t) \\ &= A_tA_{t-1} \ldots A_0x(0) + \begin{bmatrix} A_t \ldots A_1B_0, & A_t \ldots A_2B_1, & \ldots, & B_t \end{bmatrix} \begin{bmatrix} u(0) \\ u(1) \\ \vdots \\ u(t) \end{bmatrix}. \end{aligned} \tag{9.19}$$

If $\mathbb{P}\{\textbf{rank } (B_t, A_tB_{t-1}, A_tA_{t-1}B_{t-2}, \ldots A_t \ldots A_1B_0) = n\}$, for some $t \geq 1$, is not identically zero, then for that explicit t, the input vector $\begin{bmatrix} u(0) & u(1) & \ldots & u(t) \end{bmatrix}^T$ can be chosen such that the system (9.19) is controllable.

Necessity: From (9.19), one may conclude that the controllable subspace is the subspace of the range of $C = \begin{bmatrix} B_t & A_tB_{t-1} & A_tA_{t-1}B_{t-2} & \ldots & A_t \ldots A_1B_0 \end{bmatrix}$. The selected $u(i)$ cannot generate a subspace larger than the range of the controllability matrix C. Therefore, **rank** $C \leq n$. Consider the case where **rank** $C < n$. Then, a state x_1 exists that does not belong to the range of C and, for that x_1, one has

$$\mathbb{P}(||x(T) - x_1|| \leq \epsilon) = 0, \text{ for any } u, T, \text{ and } x_0,$$

which contradicts the assumption. Therefore, **rank** $C = n$. Then, for some t the condition (9.18) holds. □

9.4 Linear Quadratic Regulator over random networks

The control problem considered here is included selecting a suitable control function $u(t)$. The control law $u(t)$ ensures that every initial state $x(t_0) = x_0$ is derived to the reference signal $x = 0$ such that the performance index

$$\mathbb{E}\{\rho^T ||x(T)||_S^2 + \sum_{t=t_0}^{T-1} \rho^t ||x(t)||_Q^2\}, \quad \rho > 0,\, Q > 0, \text{ and } S \geq 0 \tag{9.20}$$

is minimized. The performance index above is minimized while the states are stable, in some probabilistic sense, as previously discussed in (9.14) and (9.15). Since the random events w_t's are independent, the control law $u(t)$ depends only on $x(t)$ and the knowledge of $x(t-1), x(t-2), \ldots$ carries no additional information about the future evolution of (9.1). Let the control law be of the feedback form

$$u(t) = -K(t)x(t), \tag{9.21}$$

leading to the closed loop matrix A_{c_t} defined as

$$A_{c_t} = A_t - B_t K(t). \tag{9.22}$$

Given any control law, e.g., the feedback law in (9.21), the dynamics defined in (9.1) generates the random sequence $x(t_0), x(t_0+1), \ldots$ referred to as the "motion" of (9.1) and denoted by $x(t \geq t_0, x_0)$. For the simplicity of the notation, $x(t \geq t_0, x_0)$ will be denoted by $x(t)$.

It is well known that if the control law is stationary, the motions of (9.1) are asymptotically stable in the mean square if and only if

$$|\lambda_i[\mathbb{E}(A_{c_t}^T \otimes A_{c_t}^T)]| < 1 \quad \text{for} \quad i = 1, \ldots, n^2. \tag{9.23}$$

In [13], it has been shown that, for the dynamics (9.1), the condition (9.23) also implies asymptotic stability with probability one.

Consider the performance index defined in (9.20). Let the minimum value of (9.20) to be $V(x_0, t_0, T)$ and $x(t_1) = x_1$ where $t_1 = t_0 + 1$. The principle of optimality [34] and the

mutual independence of w_t imply that

$$\begin{aligned} V(x_0, t_0, T) &= \min_u \{\rho \mathbb{E}[V(x_1, t_1, T] + ||x_0||_Q^2\}, \\ &= \min_u \{\rho \mathbb{E}[V(Ax_0 + Bu, t_1, T)] + ||x_0||_Q^2\}. \end{aligned} \tag{9.24}$$

It can be observed from (9.20) that $V(x, T, T) = ||x||_S^2$. Assuming by induction that

$$V(x, t, T) = ||x||_{P(t,T)}^2 = \min_u \{\rho \mathbb{E}[||Ax + Bu||_{P(t+1,T)}^2] + ||x||_Q^2\},$$

where $P(T, T) = S$. The solution of the optimization problem in (9.20) for $T < \infty$ has been introduced in [13] as

$$\begin{aligned} u(t) &= -K(t)x(t) = -\mathbb{E}[B^T P(t+1, T)B]^\dagger \mathbb{E}[B^T P(t+1, T)A]x(t) \\ P(t, T) &= \rho \mathbb{E}[A_{c_t}^T P(t+1, T) A_{c_t}] + Q, \end{aligned} \tag{9.25}$$

where A_{c_t} is defined in (9.22) and $\dagger$ is the pseudoinverse operator. Considering the stationarity of w_k we can now write

$$P(t, T) \equiv P(0, T - t) \equiv P(T - t).$$

Hence, equations in (9.25) can be written as

$$P(t+1) = \rho \mathbb{E}[A_{c_t}^T P(T) A_{c_t}] + Q, \text{ and } P(0) = S. \tag{9.26}$$

The more interesting case for us is the infinite horizon scenario, $T = \infty$. Let us also set $S = 0$ in (9.20). The next theorem describes designing the optimal feedback controller for the infinite horizon scenario.

Theorem 9.4.1. *[13] Equation (9.26) has a fixed point P_* if and only if the regulator problem has an optimal solution in the limit $T = \infty$. The fixed point P_* is necessarily unique and all iterates of $P(t)$ of (9.26) starting at $P_0 \geq 0$ converge to P_*. Moreover, $||x_0||_{P_*}^2$ is the optimal index. The optimal control law is constant and given by*

$$K_* = (\mathbb{E}[B_t^T P_* B_t])^{-1} \mathbb{E}[B_t^T P_* A_t], \tag{9.27}$$

where

$$P_* = \rho \left[\mathbb{E}(A_{c_t}^T P_* A_{c_t}) - \mathbb{E}(A_{c_t}^T P_* B_t) \left(\mathbb{E}(B_t^T P_* B_t) \right)^{-1} \mathbb{E}(B_t^T P_* A_{c_t}) \right]. \tag{9.28}$$

The existence of the fixed point P_* in (9.28) and the optimality of the solution (9.27) have been discussed in [13]. Now the essential question is whether there exists a control law which guaranteed the stability. The next theorem provides the necessary and sufficient condition to answer this question. It also gives necessary and sufficient condition on when the regulator problem has an optimal solution.

Theorem 9.4.2. *The optimal infinite horizon regulator problem in the limit $T = \infty$, defined in (9.27) and (9.28), has a unique solution if and only if $\rho < \rho_{\max}$. The system (9.1) can be made asymptotically stable in the mean square if and only if $\rho_{\max} > 1$. Any optimal system with $1 \leq \rho < \rho_{\max}$ is asymptotically stable in the mean square and is therefore asymptotically stable with probability* 1.

Proof. The proof has been discussed in [13]. □

We note that $\rho_{\max}$ in Theorem 9.4.2 can be evaluated by successive iterations. Furstenburg and Kesten [33] have also suggested an expression for calculating $\rho_{\max}$. In this direction, let us define $\mu(K)$ as

$$\mu(K) = \lim_{t\to\infty} t^{-1}\{\mathbb{E}\log||A_{c_t}\dots A_{c_1}||\},$$

which can also be defined as

$$\lim_{t\to\infty} t^{-1}\log||A_{c_t}\dots A_{c_1}||\}$$

with probability one. Given K, there is a corresponding $\mu(K)$, such that one can determine $\rho_{\max}$ as

$$\rho_{\max}^{-1/2} = \inf_K\{e^{\mu(K)}\}. \tag{9.29}$$

Let us now proceed to get a better understanding of the parameter $\rho_{\max}$. In order to estimate $\rho_{\max}$, the first step is estimating $||A_{c_t}|| = ||A_t - B_tK(t)||$. Let us consider the case where $K(t) = K_*$ and $B_t = I$ where I is $n \times n$ identity matrix. Therefore,

$$\begin{aligned} B_tK_* &= B_t[\mathbb{E}(B_t^TP_*B_t)]^{-1}\mathbb{E}(B_t^TP_*A_t) \\ &= [\mathbb{E}(P_*)]^{-1}\mathbb{E}(P_*)\mathbb{E}(A_t) = \mathbb{E}(A_t). \end{aligned}$$

Consequently,

$$||A_{c_t} - B_t K_*||_2 = ||A_t - \mathbb{E}(A_t)||_2 = \lambda_{\max}\{A_t - \mathbb{E}(A_t)\}.$$

The next step is to estimate $\lambda_{\max}\{A_t - \mathbb{E}(A_t)\}$. For the analysis purposes, the considered dynamics $A(w_t)$ in (9.1) is a consensus-based protocol associated with a normalized Laplacian; following we briefly define this dynamics.

Consider the graph $\mathcal{G}_t = (\mathcal{V}, \mathcal{E}(w_t))$ with $|\mathcal{V}| = n$, the edge set $\mathcal{E}(w_t)$ is constructed based on a random distribution at time t, and the weighted consensus protocol

$$\dot{x_i}(t) = \sum_{\{i,j\}\in\mathcal{E}(w_t)} \frac{1}{\sqrt{\mathbf{deg}\ i\ \mathbf{deg}\ j}}(x_j(t) - x_i(t)), \tag{9.30}$$

adopted by n-nodes, where $\mathbf{deg}\ i$ is the degree of node i and x_i is the state of the i-th node, e.g., its position, speed, heading, voltage, etc., evolving according to the weighted sum of the differences between the i-th node's state and its neighbors. Next, let a group of agents $\mathcal{I} \subset \mathcal{V}$ with cardinality $|\mathcal{I}| = r_{\mathcal{I}}$, selected deterministically or based on a distribution, "excite" the underlying coordination protocol by injecting signals to the network. Hence, the original consensus protocol from node i's perspective assumes the form

$$\dot{x_i}(t) = \sum_{\{i,j\}\in\mathcal{E}} \frac{1}{\sqrt{\mathbf{deg}\ i\ \mathbf{deg}\ j}}(x_j(t) - x_i(t)) + B_i u_i(t), \tag{9.31}$$

where $B_i = \beta_i$ if $i \in \mathcal{I}$, and zero otherwise. Without loss of generality, we can always assume that $\beta_i = 1$ and modify the control signal $u_i(t)$ as $\beta_i u_i(t)$ if necessary. The weights are defined to be $\frac{1}{\sqrt{\mathbf{deg}\ i\ \mathbf{deg}\ j}}$ for each neighbor node j. We then arrive at the compact form of a linear time-invariant system,

$$\dot{x}(t) = -L_t x(t) + B_t u(t), \tag{9.32}$$

where $L_t = L(\mathcal{G}_t) \in \mathbb{R}^{n\times n}$ is the normalized Laplacian matrix. The input matrix $B_t \in \mathbb{R}^{n\times r_{\mathcal{I}}}$ where the j^{th} column has 1 at $\{ij\}$ entry if $i \in \mathcal{I}$ and zero otherwise and $u(t) = [u_1^T, u_2^T, \dots u_{r_{\mathcal{I}}}^T]^T$. The discrete version of (9.32) has $A(w_t) = e^{-L_t\delta}$.

Assume $0 < \delta \ll 1$ and estimate $A_t \approx I - \delta L_t$. Therefore, $\mathbb{E}(A_t)$ can be estimated as $I - \delta\mathbb{E}(L_i)$. Now let us estimate $||A_t - B_t K_*||_2$ where $B_t = I$ as

$$\begin{aligned} ||A_t - B_t K_*||_2 &= \lambda_{\max}(A_t - \mathbb{E}(A_t)) \\ &= \lambda_{\max}(I - \delta L_t - I + \delta\mathbb{E}(L_t)) \\ &= -\delta\lambda_{\max}(L_t - \mathbb{E}(L_t)). \end{aligned}$$

Chung in [92] has shown that if $pn \gg \ln n$, then with probability at least $1 - 1/n$, one has

$$|\lambda_k(L_t) - \lambda_k(\mathbb{E}(L_t))| \leq 3\sqrt{\frac{6\ln 2n}{pn}},$$

where L_t is the normalized Laplacian, n is the number of nodes, and p is the probability in the Erdős-RT̂enyi distribution. Then,

$$\begin{aligned} -3\sqrt{\frac{6\ln 2n}{pn}} &\leq \lambda_{\max}(L_t) - \lambda_{\max}(\mathbb{E}(L_t)) \\ &\leq \lambda_{\max}\left(L_t - \mathbb{E}(L_t)\right). \end{aligned} \tag{9.33}$$

Now by multiplying (9.33) by $-\delta$ we get

$$3\delta\sqrt{\frac{6\ln 2n}{pn}} \geq -\delta\lambda_{\max}(L_t - \mathbb{E}L_t) = ||A_t - B_tK_*||_2.$$

Let us estimate $t^{-1}\mathbb{E}\{\log||A_{c_t}\dots A_{c_1}||\}$ as

$$\begin{aligned} t^{-1} \quad & \mathbb{E}\{\log||A_{c_t}\dots A_{c_1}||\} \leq t^{-1}\mathbb{E}\{\sum_{t=1}^{t}\log||A_{c_t}||\} \leq \\ t^{-1} \quad & \{t\log\left(3\delta\sqrt{\frac{6\ln 2n}{pn}}\right)\} = \log\left(3\delta\sqrt{\frac{6\ln 2n}{pn}}\right). \end{aligned}$$

For the problem set-up in (9.32) with normalized Laplacian, we can show $\mu(K) < 0$ where the control gain $K = 0$. Therefore, $\log\left(3\delta\sqrt{\frac{6\ln 2n}{pn}}\right) < 0$ and $3\delta\sqrt{\frac{6\ln 2n}{pn}} < 1$ which imply that

$$p > \frac{54\delta\ln 2n}{n} \quad \text{and} \quad p > \frac{\ln n}{n}.$$

Briefly, if $p > \max\{\frac{54\delta\ln 2n}{n}, \frac{\ln n}{n}\}$, then $\mu(K_*) \leq \log\left(3\delta\sqrt{\frac{6\ln 2n}{pn}}\right)$. Therefore,

$$\frac{1}{\sqrt{\rho_{\max}}} = \inf_K e^{\mu(K)} \leq e^{\mu(K_*)} \leq 3\delta\sqrt{\frac{6\ln 2n}{pn}},$$

which provides an upper bound for $\rho_{\max} \geq \frac{pn}{54\delta^2\ln 2n}$ for the normalized consensus dynamics defined in (9.32).

9.5 A Numerical Example

In the next example, we implement the proposed controller for a randomly evolving network according to the Erdős-Renyi distribution.

Consider a group of seven agents, coordinating their respective orientations to achieve a particular alignment over a random information-exchange network. Fig. 9.3 shows the evolution of the information graph with the edge probability $p = 0.2$ for the first 12 seconds of the simulation. The red crossed node at each interval acts as the input to the network, which is selected uniformly from the set of possible input nodes. Figure 9.4 demonstrates the convergence for the states to the reference signal $x = 0$ when the random network is running the consensus protocol.

Fig. 9.5 also demonstrates the convergence of the case when the random network is running the Dirichlet protocol. Fig. 9.6 demonstrates the convergence of the states to the reference signal $x = 0$ when the probability of the existence of an edge in the random network is $p = 0.7$ and the network is running the consensus protocol.

These figures also show the different convergence rates. Note that the convergence rate is a function of the probability p.

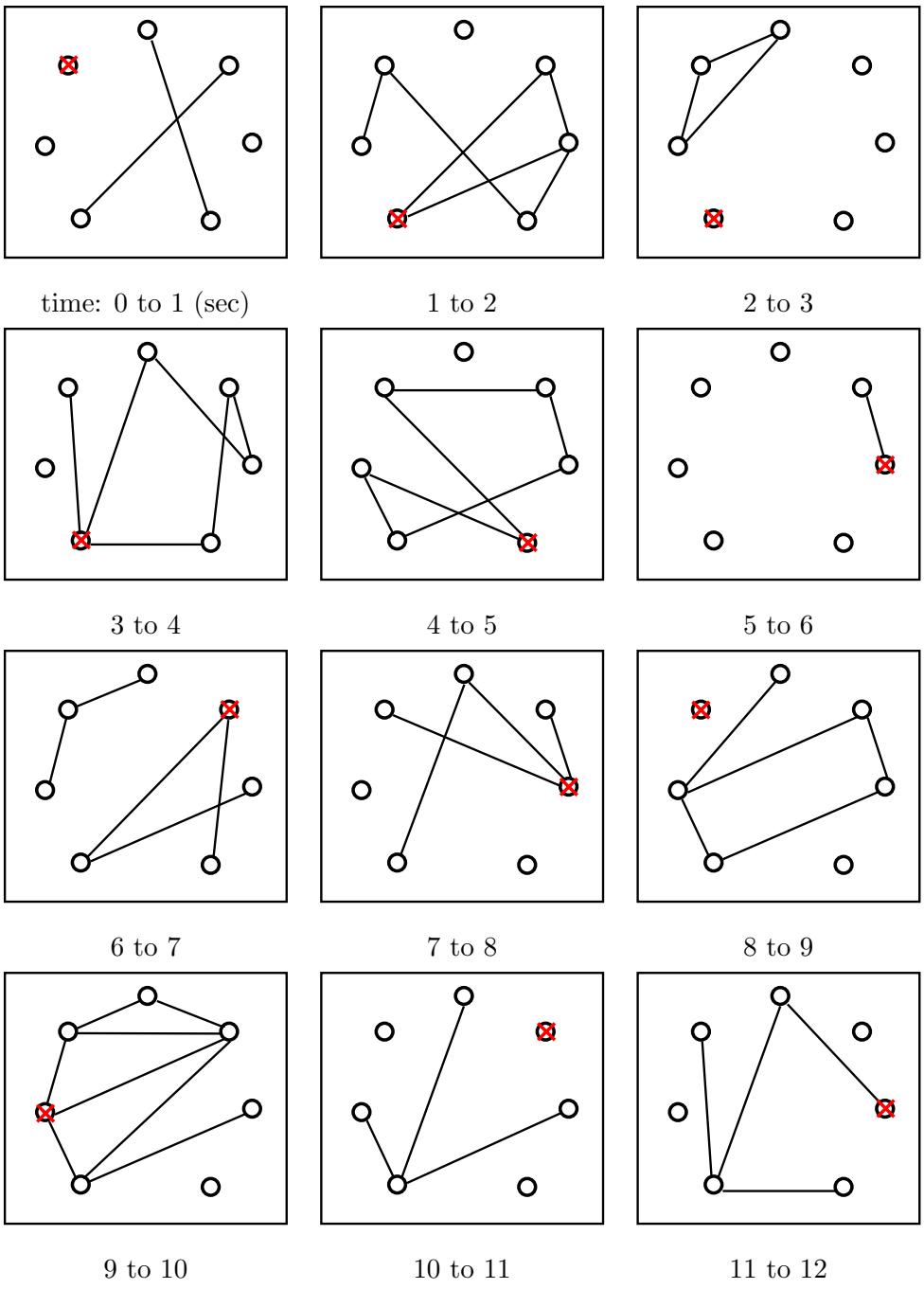


Figure 9.3: Behavior of the random network evolving based on Erdős- RT̂enyi distribution with $p = 0.2$ in the first 12 intervals. The red crossed nodes are the input nodes.

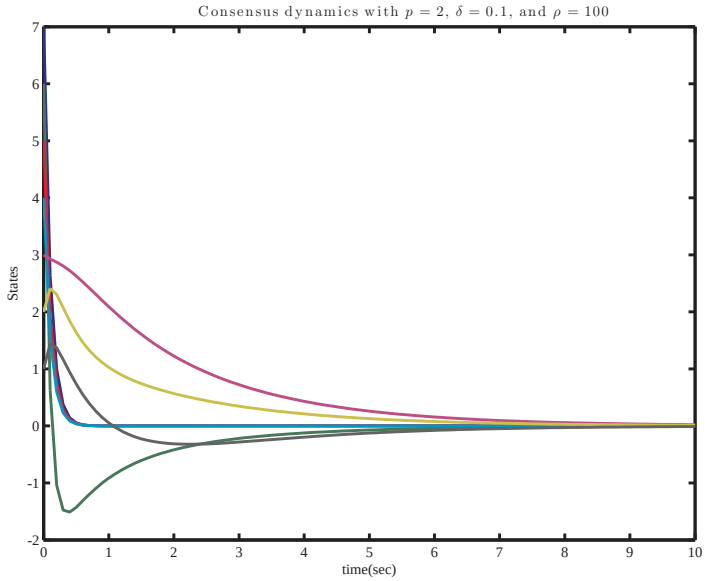


Figure 9.4: Convergence of the states to the reference signal $x = 0$ in the consensus dynamics over a random network with $p = 0.2$.

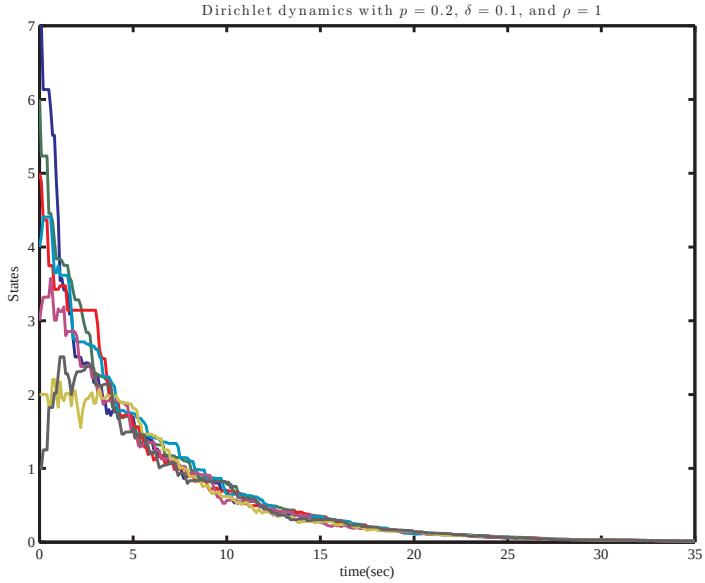


Figure 9.5: Convergence of the states to the reference signal $x = 0$ in the Dirichlet dynamics over a random network with $p = 0.2$.

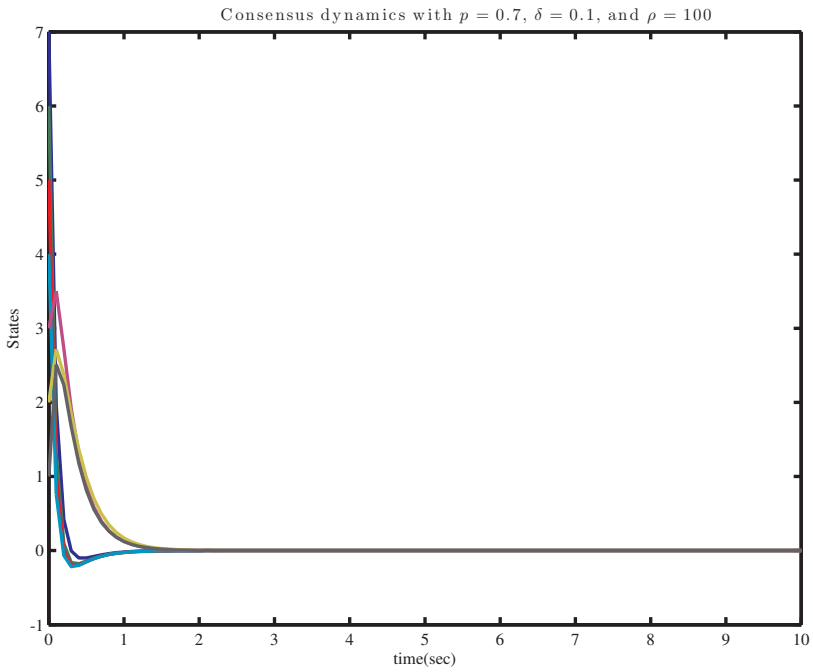


Figure 9.6: Convergence of the states to the reference signal $x = 0$ in the consensus dynamics over a random network with $p = 0.7$.

Chapter 10

COORDINATED DECENTRALIZED ESTIMATION OVER RANDOM NETWORKS

In this chapter, we consider three representative problems on observability and estimation over networks in the presence of randomness. The aim of this work is to highlight that these problems, among many others, can be approached via a unified formulation. The unified formulation can subsequently be utilized for proving almost sure stability and convergence of filtering algorithms over distinct classes of random networks. More specifically, we show stability and convergence properties of random variations on the coordinated decentralized estimation using this approach. We then proceed to study the effect of the random network structure on the performance of the estimation algorithms.

10.1 Considered Problems

In this section, we examine coordinated decentralized estimator design over networks in three different scenarios involving randomness, [55, 56, 57]:

(a) A coordinated decentralized filter is considered that uses the local computational capability of each sensor while also benefits from the presence of a coordinator. A sensor network is observing a dynamic process while each sensor calculates, based on the partial information available to it, an estimate of the state of the dynamic process, e.g., an application demonstrated in Figure 10.1. A partitioned information filter in a simple hierarchical estimation architecture, shown on Fig. 10.2, provides a decentralized estimation architecture for this problem [45].

Meanwhile, the proposed architecture implicitly requires that the sensors communicate with the coordinator at every time step. However, this assumption might be unrealistic or overly expensive within the operational constraints. For more economical energy usage and/or in presence of unreliable communication links and packet drop-outs, a

random communication scheme between the sensors and the coordinator is thus assumed. Hence, at every time step, the link between each sensor and the coordinator becomes active according to a random distribution, e.g., with probability $p \in (0, 1)$. A natural question is thereby whether this randomized coordinated decentralized estimation still has convergence properties that parallel the original deterministic coordinated decentralized estimation scheme.

(b) A group of dynamic agents are running a formation task based on the consensus protocol over a static network. The consensus algorithm is corrupted by Gaussian noise on the interaction/communication links between these agents. Estimating the states of agents in this setup is facilitated by interfacing with a group of nodes called "output ports" and observing their states over time; see Figure 10.3. The observability condition requires us to find a set of nodes that led to an "observable" process. Finding the set of output nodes for guaranteeing an observable network, however, is challenging; in Rahmani *et al.* [101] it is shown that, for example, the symmetric structure of network with respect to the observation ports has to be broken to avoid an unobservable dynamics. In our setup, we assume that at every time step based on a random distribution, e.g., with probability $p \in (0, 1]$, each node is selected as the observation port; see Figure 10.4. It is of interest to determine whether the subsequent coordinated decentralized estimation on such a process has the desired (albeit, probabilistic) stability and convergence properties.

(c) In this case, the static formation network described in the previous scenario also randomly changes at every time step, according to a random distribution, e.g., the Erdős-RT̂enyi distribution is demonstrated in Figure 10.5. Thus, both the underlying dynamics and the measurement scheme evolve randomly over time.

The main focus of this research is designing stable decentralized estimator in the stochastic setup. In this direction, let us start with recalling the deterministic setup.

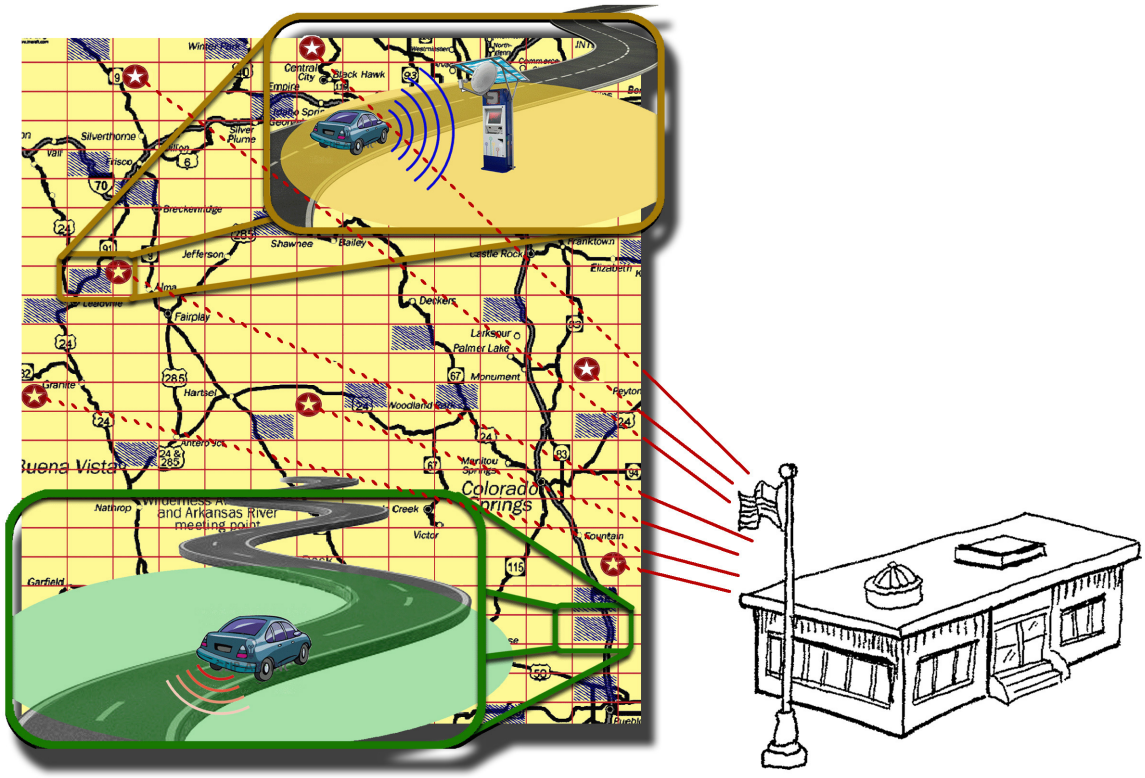


Figure 10.1: Based on statistics from U.S. Department of Transportation the total length of roads in United States including interstate, other principal and minor arterials, major and minor collector, and local in 2009 was $4,050,717$ miles while it increased over $5,500$ miles in four years. It is vitally important to be able to consistently monitor the road performance to guarantee the highest level of safety to motorway users with as little discomfort as possible. In this context, the introduction of an automated mechanisms seems indispensable. The proposed mechanisms is that volunteer cars with installed cameras underneath their cars monitor some random section of a road and transmit the collected data to a local control centre located somewhere on the route to their destination. The local control centers transmit the pre-processed data to the global coordinator. Autostrada del Brennero has developed a mobile monitoring system installed on a series of service vehicles, capable of acquiring, elaborating, and transmitting data in real time. The instrumented vehicles were initially designed to capture the road surface temperature by the use of contact-less IR-sensors [99]. Randomness is presented in part of the roads that are monitored by the volunteer drivers, data transmitting between cars and the local control centers, and also between the local control centers and the global coordinator.

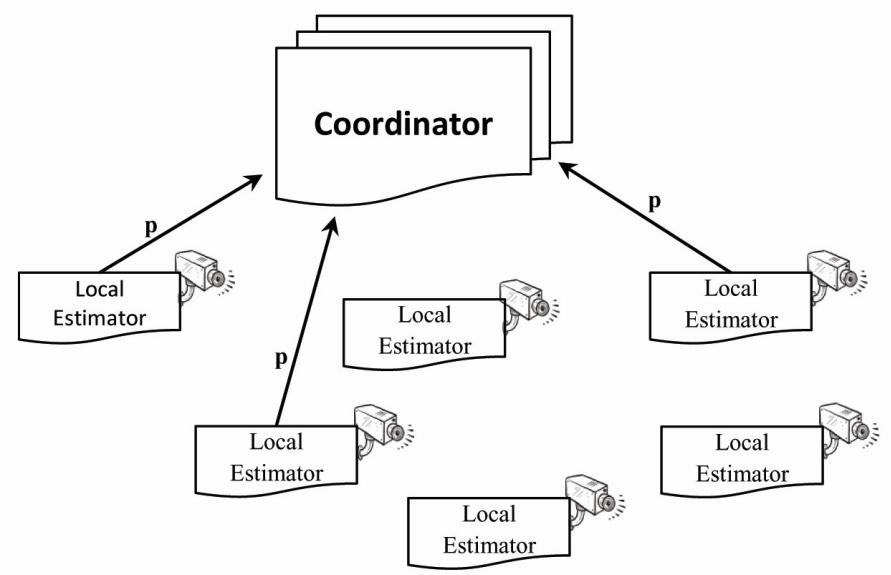


Figure 10.2: The communication link between each sensor and the coordinator switches on with a probability $0 < p < 1$.

10.2 Deterministic Setup

Consider a sensor network that is employed to observe the evolution of the dynamical process described by

$$x(t+1) = Ax(t) + w(t), \tag{10.1}$$

where $x(t) \in \mathbb{R}^n$ is the state vector with initial condition $x(0)$ distributed as zero mean Gaussian with covariance Σ_0 and $w(t)$ is an uncorrelated zero mean Gaussian sequence with covariance Ξ. Each one of the N sensors has a measurement map of the form $z_i(t) = H_i x(t) + v_i(t), \quad i = 1, 2, \ldots, N$, when in compact form, can be represented as

$$z(t) = H(t)x(t) + v(t), \tag{10.2}$$

$$H = \begin{bmatrix} H_1 \\ \vdots \\ H_N \end{bmatrix}, \quad \text{and} \quad v(t) = \begin{bmatrix} v_1(t) \\ \vdots \\ v_N(t) \end{bmatrix}. \tag{10.3}$$

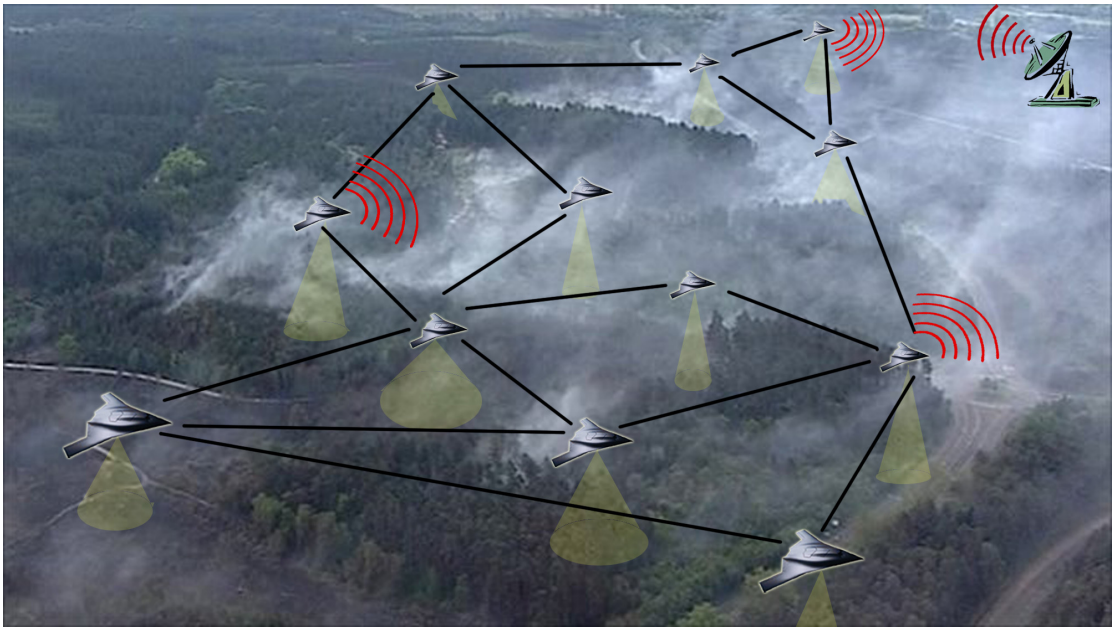


Figure 10.3: Multiple Unmanned Air Vehicles (UAVs) are emerging as a promising technology for monitoring large forest fires since a forest fire is typically inaccessible by ground vehicles. Frequent updates concerning the progress of a forest fire are essential for controlling the fire. On the other hand, multiple UAVs are being used for monitoring rivers for flood, and volcanos, as well as many other scenarios. Even individual alpine hikers canbenefit from a protective swarm of UAVs overhead. UAVs can be utilized to detect the illegal dumping of waste or toxic materials and also the progression of the desert and other uninhabitable areas. Among many other applications of UAVs are monitoring wild animal poaching, illegal fishing, tree blow-down in a forest after a hurricane, following a heavy snowfall, or land temperature monitoring. All these scenarios often involve limited bandwidth for inter-UAV communication. Natural causes can also induce data dropping that can be modeled as random. Sensor occlusion which are also probable due to the limited battery-lifetime, can also be dealt with by randomly turning the sensors on and off.

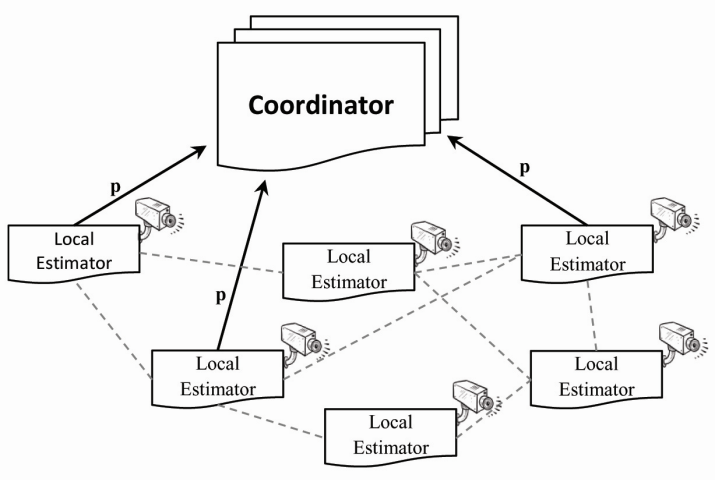
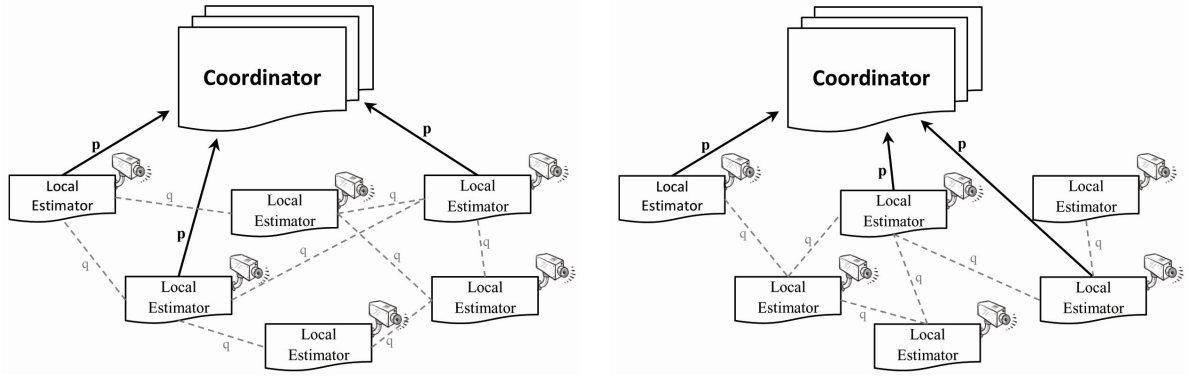


Figure 10.4: The communication architecture; each link switches on with a probability $0 < p < 1$.



time t time $t + \Delta_t$

Figure 10.5: Behavior of a random communication network in two time intervals

Under the hypothesis of the stabilizability of the pair (A, Ξ) and detectability of the pair (A, H), the estimation error covariance, $\Sigma(t|t)$, utilizing a Kalman filter, converges to a unique value from any initial condition [52].

The Kalman filter updates the states according to

$$\hat{x}(t|t) = \hat{x}(t|t-1) + K(z(t) - H\hat{x}(t|t-1)), \tag{10.4}$$

with Kalman gain $K = \Sigma(t|t)H$, where $\Sigma(t|t)$ is the covariance matrix of the error vector $\hat{x}(t|t) - x(t)$. Thus,

$$\Sigma(t|t) = \mathbb{E}\{(x(t) - \hat{x}(t|t))(x(t) - \hat{x}^T(t|t))\} = (\Sigma(t|t-1)^{-1} + H^T H)^{-1}, \tag{10.5}$$

where $\Sigma(t|t-1) = A\Sigma(k-1|k-1)A^T + \Xi$. The information filter, which is proven to be advantagous in distributed formulation of Kalman filtering, is an alternative representation of the filter in terms of the information state vector and information matrix defined via $y(t) = H^T z(t)$ and $Y = H^T H$ [52]. In this venue, the information state estimation and predicted vectors can be calculated as

$$\hat{y}(t|t) = I(t|t)\hat{x}(t|t) \quad \text{and} \quad \hat{y}(t|t-1) = I(t|t-1)\hat{x}(t|t-1),$$

where the information matrix $I(t|t) = \Sigma^{-1}(t|t)$ and $I(t|t-1) = \Sigma^{-1}(t|t-1)$. The linear Kalman filter may now be written in terms of the information state vector and the information matrix as

$$\begin{aligned} I(t|t) &= I(t|t-1) + Y(t) \\ \hat{y}(t|t) &= \hat{y}(t|t-1) + y(t) \end{aligned} \tag{10.6}$$

or in additive form,

$$\begin{aligned} I(t|t) &= I(t|t-1) + \sum_{i=1}^{N} Y_i(t) \\ \hat{y}(t|t) &= \hat{y}(t|t-1) + \sum_{i=1}^{N} y_i(t), \end{aligned} \tag{10.7}$$

where $Y_i = H_i^T H_i$ and $y_i(t) = H_i^T z_i(t)$. The interpretation of the additive form of the information filter in (10.7) is as follows: each sensor performs a local Kalman filter based on local sensor measurements; in this case we have $H_i^T H_i = I_i(t|t) - I_i(t|t-1)$. The information matrix can then be updated at each sensor node by receiving the difference

$I_i(t|t) - I_i(t|t-1)$, summing them up across all sensors, and then adding them to $I(t|t-1)$ to obtain $I(t|t)$. Similarly, the information vector can be updated by summing up the received $y_i(t)$ from each sensor. The above scheme can then be considered in terms of the state and covariance update by including a coordinator. The global update assumes the form $\hat{x}(t|t) = (I - K(t)H)\hat{x}(t|t-1) + K(t)z(t)$, with $K(t)z(t) = \Sigma(t|t)H^T z(t) = \Sigma(t|t)\sum_i H_i^T z_i$, and $I - K(t)H = \Sigma(t|t)\Sigma(t|t-1)^{-1}$. In the meantime, $H_i^T z_i = I_i(t|t)\hat{x}_i(t|t) - I_i(t|t)(I - K_i H_i)\hat{x}_i(t|t-1)$, and therefore,

$$\hat{x}(t|t) = \Sigma(t|t)(I(t|t-1)\hat{x}(t|t-1) + \sum_i I_i(t|t)\hat{x}_i(t|t-1) - I_i(t|t-1)\hat{x}_i(t|t-1)). \tag{10.8}$$

More details on this section are presented in [45, 105]. We note that the error covariance matrix update in (10.5) can be written as

$$\Sigma(t|t) = (A\Sigma(k-1|k-1)A^T + Q)(I + R(t)Q + R(t)A\Sigma(k-1|k-1)A^T)^{-1}, \tag{10.9}$$

where $R(t) = H^T H$.

10.3 Random setup

Now consider the sensor network and the coordinator shown in Figures 10.2, 10.4, and 10.5. Note that in reference to the deterministic setup, in all these cases the measurement matrix H in (10.3) is a random matrix. That is when the i^{th} sensor sends its estimate to the coordinator, H_i becomes "active" in the i^{th} row. Activation of the i^{th} sensor happens with probability $0 < p < 1$ at every time step. Note that $\{H(t), t \geq 1\}$ is a strictly stationary ergodic process and that there is a σ-algebra, $\mathscr{F}$, generated by all measurements $z_1, z_2, \ldots, z_t$ for all $t \geq 1$.

The sensor-coordinator network can be modeled by a star graph with $N+1$ nodes that has the coordinator at its center. Let us denote the coordinator as node with label 1. Hence, at every time step, sensor i communicates with the coordinator with probability p, i.e., the edge set $\mathcal{E}$ contains the edge $(i, 1)$ with probability p. The graph, $\mathcal{G}_t(N+1, \mathcal{E}, p)$, at the t^{th} time step can then be represented in terms of its $(N+1) \times (N+1)$ symmetric adjacency

matrix $\mathcal{A}$ as

$$\mathcal{A}_{i1} = \begin{cases} 1 & \text{if } (i,1) \in \mathcal{E} \\ 0 & \text{otherwise} \end{cases} \tag{10.10}$$

Define the set $\mathcal{M}$ as the resulting set of all possible adjacency matrices

$$\bar{\mathcal{A}} = \mathbb{E}[\mathcal{A}(t)] = \int_{\mathcal{M}} \mathcal{A}\, d\mathcal{D}(\mathcal{A}), \tag{10.11}$$

where $\mathcal{D}$ is the probability distribution on the space $\mathcal{M}$. It follows that $\bar{\mathcal{A}}$ for the considered random graph model is irreducible.

We now assume that the following conditions hold:

Hypothesis 10.3.1. *(1) The matrix A in the dynamic process (10.1) is invertible, and (2)* $\log^+ ||A||$, $\log^+ ||A^{-1}||$, *and* $\log^+ ||\Xi||$ *are integrable, where* $\log^+ x = \max\{0, \log x\}$.

We now to explore conditions on the pair (A, H) in the scenarios (a), (b), and (c) discussed in §10.1 that would allow a randomized estimation scheme with guaranteed (probabilistic) stability and convergence. A key technical construct that proves to be instrumental in this direction is that of weak detectability [43, 53, 18]. To get a general model for cases (a), (b), and (c), assume the matrix in (10.1) is time varying and is denoted by $A(t)$.

Definition 10.3.2. *Let $R(t) = H^T(t)H(t)$. The sequence $\{(A(t), H(t)), t \in \mathbb{N}\}$ is said to be weakly observable if, for some $t \geq 1$, $\mathbb{P}\{\mathbf{det}[\Omega(t)] \neq 0\} \neq 0$, where*

$$\begin{aligned} \Omega(t) &= R(1) + A^T(1)R(2)A(1) + \ldots + A^T(t-1)\ldots A^T(1)R(t)A(t-1)\ldots A(1) \\ &= \sum_{i=1}^{t} A^T(i-1)\ldots A^T(1)R(i)A(i-1)\ldots A(1). \end{aligned} \tag{10.12}$$

The weak observability condition holds if the matrices $A(t)$ and $\sum_{i=1}^{t} R(i)$ are invertible. The key observations is now that *weak detectability is a generic property for all three scenarios discussed in §10.1.* As such, the main results of [53, 18], in the context of estimation over random networks become directly applicable. We now briefly review relevant results that are instrumental for proving stability and convergence of the corresponding coordinated distributed estimators over random networks pertaining to these three scenarios.

Let us start by first describing the asymptotic behavior of the error covariance matrix $\Sigma(t|t)$ in (10.9). In this venue, let $\mathcal{P}$ (respectively, $\mathcal{P}_0$) denote the set of $N \times N$ nonnegative (respectively, positive) symmetric matrices. It can be shown that almost surely, for any solution $\Sigma(t|t)$ of (10.5), there is a constant covariance matrix $\bar{\Sigma}$ such that $||\Sigma(t|t) - \bar{\Sigma}||$ converges to zero as $t \to \infty$. The main result in the next section is that the error covariance matrices $\Sigma(t|t)$ are contractions on $\mathcal{P}_0$ with respect to the Riemannian metric to be defined shortly.

Define the symplectic group $Sp(N, \mathbb{R})$ as the set of all the matrices M of order $2N$ such that $M^T J M = J$ where $J = \left(\begin{smallmatrix} 0 & I \\ -I & 0 \end{smallmatrix}\right)$. If we now write

$$M = \begin{pmatrix} A & B \\ C & D \end{pmatrix} \in Sp(N, \mathbb{R}), \tag{10.13}$$

where the entries are $N \times N$, then BA^T and $A^T C$ are symmetric and $A^T D - C^T B = I$. We associate to the system (10.1) and (10.2) the so-called Hamiltonian matrices $M(t)$ of order $2N$ written in block form as

$$M(t) = \begin{pmatrix} A(t) & \Xi\, A(t)^{-T} \\ \Theta\, A(t) & (I + \Theta\, \Xi) A(t)^{-T}) \end{pmatrix}. \tag{10.14}$$

Therefore, the set of all Hamiltonian matrices can be represented as

$$\mathscr{H} = \left\{ \left(\begin{smallmatrix} A & B \\ C & D \end{smallmatrix}\right) \in Sp(d, \mathbb{R}); A \text{ is invertible; } BA^T \in \mathcal{P},\ A^T C \in \mathcal{P} \right\}. \tag{10.15}$$

We define three subsets $\mathscr{H}_1$, $\mathscr{H}_2$, and $\mathscr{H}_0$ of $\mathscr{H}$ by

$$\begin{aligned} \mathscr{H}_1 &= \left\{ \left(\begin{smallmatrix} A & B \\ C & D \end{smallmatrix}\right) \in Sp(d, \mathbb{R});\ BA^T \in \mathcal{P},\ A^T C \in \mathcal{P}_0 \right\}, \\ \mathscr{H}_2 &= \left\{ \left(\begin{smallmatrix} A & B \\ C & D \end{smallmatrix}\right) \in Sp(d, \mathbb{R});\ BA^T \in \mathcal{P}_0,\ A^T C \in \mathcal{P} \right\}, \\ \mathscr{H}_0 &= \mathscr{H}_1 \cap \mathscr{H}_2. \end{aligned} \tag{10.16}$$

The Riemannian metric δ and the Euclidean norm $||.||$ are used in this chapter with the following definitions.

Definition 10.3.3. *The Riemannian metric δ on $\mathcal{P}_0$ is defined, for any $P,\ Q \in \mathcal{P}_0$, as*

$$\delta(P, Q) = \left\{ \sum_{i=1}^{N} \log^2 \lambda_i \right\}^{1/2}, \tag{10.17}$$

where $\lambda_1, \ldots, \lambda_N$ are the eigenvalues of the matrix PQ^{-1}. On the other hand, the Euclidean norm on $\mathbb{R}^{N\times N}$ is defined as $||M|| = \sup\{||Mx||; x \in \mathbb{R}^N, ||x|| = 1\}$.

For any matrix $M \in \mathscr{H}$ defined in (10.13), we associate the map $\Phi_M : \mathcal{P}_0 \to \mathcal{P}_0$ by

$$\Phi_M(T) = (AT + B)(CT + D)^{-1}, \quad T \in \mathcal{P}_0. \tag{10.18}$$

Moreover, let us define $\Phi(T)$ for the Hamiltonian matrices (10.14) as

$$\Phi(T(t)) = (A(t)T(t) + \Xi\, A(t)^{-T})(\Theta\, A(t)T(t) + (I + \Theta\, \Xi\,)A(t)^{-T})^{-1}.$$

Therefore, by a straightforward modification of (10.9), the error covariance matrix $\Sigma(t|t)$ satisfies

$$\Sigma(t + 1|t + 1) = \Phi_{M(t)}(\Sigma(t|t)), \tag{10.19}$$

which is the *discrete Riccati equation.*

The following properties hold for deterministic system matrices in (10.1) and (10.2):

(a) For any M in $\mathscr{H}$, and $T,\ S$ in $\mathcal{P}_0$,

$$\delta(\Phi_M(T), \Phi_M(S)) \leq \delta(T, S).$$

(b) For any M in $\mathscr{H}_1$ or in $\mathscr{H}_2$, and $T,\ S$ in $\mathcal{P}_0$,

$$\delta(\Phi_M(T), \Phi_M(S)) < \delta(T, S).$$

(c) For any M in $\mathscr{H}_0$, there exists $\rho(M),\ 0 < \rho(M) < 1$, such that, for all $T,\ S$ in $\mathcal{P}_0$,

$$\delta(\Phi_M(T), \Phi_M(S)) \leq \rho(M)\delta(T, S).$$

In the random setup, $M(t), t \in \mathbb{N}$, as defined in (10.14), is the sequence of Hamiltonian matrices associated to the linear system (10.1) and (10.2). If the system is weakly detectable (respectively, weakly controllable), then, almost surely, $M(t) \ldots M(1)$ is in $\mathscr{H}_1$ (respectively, $\mathscr{H}_2$) for large enough values $t \in \mathbb{N}$. The main result of this section implies that the conditional error covariance matrix $\Sigma(t|t)$ does not diverge. In fact, the error is asymptotically

stationary and is independent of the initial conditions. To prove this statement, let (E, δ) be a complete separable metric space. A Lipschitz map $\Psi : E \to E$ is one for which

$$\rho(\Psi) := \sup\{\frac{\delta(\Psi(x), \Psi(y))}{\delta(x, y)}; x, y \in E, x \neq y\} \tag{10.20}$$

is finite. Now consider the process $\{X(t), t \in \mathbb{N}\}$ generated by the following difference equation

$$X(t) = \Psi(X(t-1)). \tag{10.21}$$

The following theorem explores the ergodic stationary solution of (10.21) and its almost sure convergence properties.

Theorem 10.3.4. *[18] Let $\{\Psi(t), t \in \mathbb{N}\}$ be a stationary ergodic sequence of Lipschitz maps from E into E. Suppose the following conditions hold:*

(a) For some x in E, $\mathbb{E}[\log^+ \delta(\Psi(x), x)]$ is finite.

(b) The random variable $\log^+ \rho(\Psi(1))$ is integrable, and for some integer $\bar{t} > 0$, the real number

$$\alpha = \frac{1}{\bar{t}}\mathbb{E}[\log \rho(\Psi(\bar{t}) \; o \ldots \; o\Psi(1))]$$

is strictly negative.

Then there exists an ergodic stationary process $\{\bar{X}(t), t \in \mathbb{N}\}$ with values in E, generated by (10.21), such that almost surely,

$$\lim_{t\to\infty} \frac{1}{t} \log \delta(X(t), \bar{X}(t)) \leq \alpha < 0.$$

The next theorem proves the almost surely contraction property of the discrete Riccati equation defined in (10.19).

Theorem 10.3.5. *Consider the linear system (10.1) and (10.2) with stochastic measurement matrix $H(t)$, that is weakly detectable and controllable. Then there exists an ergodic stationary $\mathcal{P}_0$-valued process $\{\bar{\Sigma}(t|t), t \in \mathbb{N}\}$ that is the solution of (10.19). Furthermore, there is a negative real number $\alpha < 0$ such that, almost surely, for any solution of (10.19) for which the initial covariance error is in $\mathcal{P}_0$,*

$$\lim_{t\to\infty} \frac{1}{t} \log \delta(\Sigma(t|t), \bar{\Sigma}(t|t)) \leq \alpha < 0. \tag{10.22}$$

Proof. Consider the random contractions $\{\Phi(t), t \in \mathbb{N}\}$ on the metric space $(\mathcal{P}_0, \delta)$ defined in (10.19). To apply Theorem 10.3.4, we first need to check the conditions of this theorem. Ccondition (a) is that for some $P \in \mathcal{P}_0$, $\mathbb{E}[\log \delta(\Phi(P), P)]$ is finite. Let us choose P to be the identity matrix and define $T = AA^T + Q$. Therefore, we obtain $\Phi(I) = T(I + R(1)T)^{-1} = (T^{-1} + R(1))^{-1}$. Consider the definition of the Riemannian metric δ and the smallest and the largest eigenvalues of the positive definite matrix $\Phi(I)$ as $\lambda_1 = 1/||\Phi(I)^{-1}||$ and $\lambda_N = ||\Phi(I)||$. Hence,

$$\delta(\Phi(I), I)^2 \leq N \max(\log^2 ||\Phi(I)||, \log^2 ||\Phi^{-1}(I)||) \tag{10.23}$$

where

$$||\Phi^{-1}(I)|| = ||(T^{-1} + R(1))^{-1}|| \leq ||T|| \leq ||A||^2 + ||Q||. \tag{10.24}$$

The first inequality in (10.24) comes from the fact that $T - (T^{-1} + R(1))^{-1}$ is positive definite and the second inequality is the conseqence of the definition of T. We also have

$$\begin{aligned} ||\Phi^{-1}(I)|| &\leq ||(T^{-1} + R(1))|| \leq ||T^{-1}|| + ||R(1)|| \\ &\leq ||A^{-1}||^2 + ||H(1)||^2. \end{aligned} \tag{10.25}$$

The assumptions of the Theorem 10.3.5 and inequalities in (10.24) and (10.25) imply that $\mathbb{E}[\log \delta(\Phi(I), I)]$ is finite. Regarding condition (b) in Theorem 10.3.4, it is known that $\Phi(t)$ is a contraction.Thus, it suffices to show that $\rho(\Phi(t)\ o\ \ldots\ o\ \Phi(1))$ is smaller than one for some $k > 0$ with positive probability.

Let $M(t)$ be the sequence of Hamiltonian matrices associated to the system (10.1) and (10.2). Then, it follows that for all $t \in \mathbb{N}$ large enough $\mathbb{P}\{M(t) \ldots M(1) \in \mathscr{H}_0\} \neq 0$ since $\mathscr{H}_0 = \mathscr{H}_1 \cap \mathscr{H}_2$. And therefore, $\mathbb{P}\{\rho(\Phi(t)\ o\ \ldots\ o\ \Phi(1)) < 1\} \neq 0$. Since both conditions of Theorem 10.3.4 satisfy, the theorem implies the result in (10.22). □

We now show that the estimation error defined as $x(t) - \hat{x}(t)$ is exponentially stable by the stochastic Lyapunov theory. In order to prove this, introducing some preliminary results and a few new variables are required. First, define $B(t) = A(t) - K(t)H(t)$, $T(t) = K(t)K^T(t) + Q(t)$, and let $\Sigma(t|t) = B(t)\Sigma(k-1|k-1)B^T(t) + T(t)$. It is proved in [18] that if $G(t) = T(t) + B(t)T(k-1)B^T(t) + \ldots + B(t) \ldots B(2)T(1)B^T(2) \ldots B^T(t)$, under the main Hypothesis 10.3.1, there exists $t \in \mathbb{N}$ such that $\mathbb{P}\{\mathbf{det}(G(t)) \neq 0\} > 0$.

Theorem 10.3.6. *Consider the system (10.1) and (10.2) with the stochastic parameters for which the system is weakly detectable and controllable while Hypothesis (10.3.1) holds. Then, the estimation error $x(t) - \hat{x}(t)$ is almost surely asymptotically stable.*

Proof. For notational simplicity suppose that $\mathbb{P}\{\mathbf{det}(G(1)) \neq 0\} > 0$. To prove the exponential stability of the error $x(t)-\hat{x}(t)$, it suffices to show that there is a real number $\gamma > 0$, such that almost surely

$$\lim_{t\to\infty} \frac{1}{t} \log ||(A(t) - K(t)H(t)), \ldots, (A(t) - K(1)H(1))|| \leq -\gamma$$

for any solution $\{\Sigma(t|t),\ t \in \mathbb{N}\}$ of (10.9) while the initial error covariance is in $\mathcal{P}_0$.

Assume that $\lambda(t) = ||T(t)^{-1}||^{-1},\ \sigma(t) = ||\Sigma(t|t)||$, and $\alpha = ||\Sigma_0^{-1}||$. For positive integer $\bar{t}$, set $x_{\bar{t}} \in \mathbb{R}^N$ by the backward recursion to be $x(t) = B^T(t+1)x(t+1)$. Consider the Lyapunov function $V(t) = x^T(t)\Sigma(t|t)x(t)$. Then, it follows that

$$\begin{aligned} V(t+1) - V(t) &= x^T(t+1)\Sigma(t+1|t+1)x(t+1) - x(t)^T\Sigma(t|t)x(t) \\ &= x^T(t+1)(\Sigma(t+1|t+1) - B(t+1)\Sigma(t|t)B^T(t+1))x(t+1) \\ &= x^T(t+1)T(t+1)x(t+1) \\ &\geq \lambda(t+1)||x(t+1)||^2 \geq \frac{\lambda(t+1)}{\sigma(t+1)}V(t+1). \end{aligned}$$

Consequently, $V(t+1)(1 - \frac{\lambda(t+1)}{\sigma(t+1)}) \geq V(t)$. Define $\tau(t+1) = 1 - \frac{\lambda(t+1)}{\sigma(t+1)}$. We also have

$$||x(0)||^2 \leq ||\Sigma_0^{-1}|| \leq ||\Sigma_0^{-1}||\tau(1)\tau(2)\ldots\tau(\bar{t})V(\bar{t}) \leq ||\Sigma_0^{-1}||\tau(1)\tau(2)\ldots\tau(\bar{t})||\Sigma(\bar{t}|\bar{t})||||x(\bar{t})||^2.$$

Since $x(0) = B^T(1)\ldots B^T(\bar{t})x(\bar{t})$, this implies that

$$||x(0)|| = ||B^T(1)\ldots B^T(p)||^2||x(\bar{t})||^2 \leq \alpha\tau(1)\ldots\tau(\bar{t})\sigma(\bar{t}).$$

Therefore,

$$\frac{1}{\bar{t}} \log ||B^T(1)\ldots B^T(\bar{t})||^2 \leq \frac{1}{\bar{t}} \log \alpha + \frac{1}{\bar{t}} \log \sigma(t) + \frac{1}{\bar{t}} \sum_{i=1}^{\bar{t}} \log \tau(i).$$

From the contraction property of $\Sigma(t+1|t+1)$ we know that $\lim_{\bar{t}\to\infty} \frac{1}{\bar{t}} \log \sigma(\bar{t}) \leq 0$. From Birkhoff's ergodic theorem it thus follows that

$$\lim_{\bar{t}\to\infty} \frac{1}{\bar{t}} \log ||B^T(\bar{t})\ldots B^T(1)||^2 \leq \mathbb{E}(\log \tau(1)).$$

As by assumption $\mathbb{P}\{\mathbf{det}(G(1)) \neq 0\} > 0$, $\mathbb{E}(\log \tau(1)) < 0$, thus proving the statement of the theorem. □

We conclude this section with a direct consequence of the above framework in the context of the three scenarios introduced in §10.1.

Corollary 10.3.7. *The coordinated decentralized estimators, following the stochastic version of (10.28), for the models described by (a)-(b) §10.1 converge almost surely.*

10.4 Random Communication and Estimation Performance

Now let us reformulate the coordinated decentralized estimation over random sensor networks and investigate the effect of the underlying probability on estimation performance.

The arrival of the observation at time t for the i^{th} sensor is defined as a binary random variable γ_{i_t} with probability distribution $p_{\gamma_{i_t}}(1) = \hat{\lambda}_t$ where γ_{i_t} is independent of γ_{i_s} if $t \neq s$. For the sake of the simplicity of the analysis, we assume that $\hat{\lambda}_t$ is identical across all sensors in all time steps; denoted this value by $\hat{\lambda}$. Now let us define

$$\begin{aligned} \hat{x}(t|t) &\equiv \mathbb{E}[x(t)\,|\bar{z}(t), \gamma_t] \\ \Sigma(t|t) &\equiv \mathbb{E}[(x(t) - \hat{x}(t)(x(t) - \hat{x}(t)^T\,|\bar{z}(t), \gamma_t], \end{aligned}$$

where $\bar{z}(t) \equiv [z(0), \ldots, z(t)]^T$ and $\gamma_t \equiv \begin{bmatrix} \gamma_{1_0} & \cdots & \gamma_{1_t} \\ \vdots & \ldots & \vdots \\ \gamma_{N_0} & \cdots & \gamma_{N_t} \end{bmatrix}$.

Consider the discrete time linear system in (10.1) that is observed by N sensors

$$z_i(t) = H_i(t)x(t) + v_i(t) \quad i = 1, 2, \ldots, n,$$

each with its own time-varying observation matrix H_i that is corrupted by zero-mean Gaussian noise v_i with covariance V_i. The centralized Kalman filter can be implemented for the system

$$x(t+1) = Ax(t) + w(t), \quad z(t) = H(t)x(t) + v(k),$$

where H and v are defined in (10.3). Provided that the noise vector on each sensor is independent zero mean Gaussian both in time and across sensors, the covariance matrix for

noise on the fused measurement assumes the form

$$V = \begin{bmatrix} V_1 & 0 & \dots & 0 & 0 \\ 0 & V_2 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & V_N \end{bmatrix}.$$

However, in the random setup the i^{th} sensor at time t observes the state with probability distribution $p_{\gamma_{i_t}}(1) = \hat{\lambda}$, the output noise covariance V_i's are defined as follows

$$\text{cov}(v_i(t)\,|\gamma_{i_t}) = \begin{cases} \mathcal{N}(0, V_i), & \gamma_{i_t} = 1 \\ \mathcal{N}(0, \sigma_i^2 I), & \gamma_{i_t} = 0 \end{cases} \tag{10.26}$$

for some σ_i^2. Therefore, the variance of the observation at time t is V_i if $\gamma_{i_t} = 1$, and $\sigma_i^2 I$ otherwise. In reality, the absence of observation corresponds to a "fictitious" observation with variance $\sigma_i^2 I = \sigma I$ and $\sigma \to \infty$ when the real observation does not arrive. Therefore, the output noise covariance V follows to be

$$V = \begin{bmatrix} \begin{cases} \mathcal{N}(0, V_1), & \gamma_{1_t} = 1 \\ \mathcal{N}(0, \sigma^2 I), & \gamma_{1_t} = 0 \end{cases} & 0 & \dots & 0 & 0 \\ 0 & \begin{cases} \mathcal{N}(0, V_2), & \gamma_{2_t} = 1 \\ \mathcal{N}(0, \sigma^2 I), & \gamma_{2_t} = 0 \end{cases} & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & \ddots \end{bmatrix}$$

for some σ^2.

Now, let us rewrite the information filter as follows. Consider the equations in (10.6) where $Y_i(t) = \gamma_{i_t} H_i^T(t) V_i^{-1} H_i(t)$ and $y_i(t) = \gamma_{i_t} H_i^T(t) V_i^{-1} z_i(t)$. As we mentioned earlier, the summation formulas (10.6) suggest a direct method for making recursive steps of the Kalman filtering over the sensor network-well-more distributed. This is done by letting each sensor to keep a local copy of the information matrix $I(t|t-1)$ and the information vector $\hat{y}(t|t-1)$.

Then, when each sensor performs a local Kalman filter based on local sensor measurements, one has $\gamma_{i_t} H_i^T(t) V_i^{-1} H_i(t) = I_i(t|t) - I_i(t|t-1)$. Therefore, the information matrix

can be updated at the coordinator by receiving the difference $I_i(t|t) - I_i(t|t-1)$, summing them up across all sensors, and then adding them to $I(t|t-1)$ to obtain $I(t|t)$. Similarly, the information vector can be updated by summing up the received $y_i(t)$ from each sensor. The above scheme can be considered in terms of the state and covariance update by including a coordinator. The global update assumes the form

$$\hat{x}(t|t) = (I - K(t)H)\hat{x}(t|t-1) + K(t)z(t) \tag{10.27}$$

with

$$K(t)z(t) = \Sigma(t|t)H^T V^{-1} z(t) = \Sigma(t|t) \sum_i \gamma_{i_t} \beta_{i_t} H_i^T V_i^{-1} z_i,$$

and

$$I - K(t)H = \Sigma(t|t)\Sigma(t|t-1)^{-1},$$

where β_{i_t} is a binary random variable that defines the arrival of the estimation of the i^{th} sensor at the coordinator with the probability distribution $p_{\beta_{i_t}}(1) = \tilde{\lambda}$ identical for all sensors.

In the meantime, $\gamma_{i_t} H_i^T V_i^{-1} z_i = I_i(t|t)\hat{x}_i(t|t) - I_i(t|t)(I - K_i H_i)\hat{x}_i(t|t-1)$ and therefore

$$\hat{x}(t|t) = \Sigma(t|t)(I(t|t-1)\hat{x}(t|t-1) + \sum_i \beta_{i_t} \left[I_i(t|t)\hat{x}_i(t|t-1) - I_i(t|t-1)\hat{x}_i(t|t-1) \right]). \tag{10.28}$$

The information matrix can then be updated as

$$I(t|t) = I(t|t-1) + \sum_i \gamma_{i_t} \beta_{i_t} H_i^T V_i^{-1} H_i = I(t|t-1) + \sum_i \beta_{i_t} \left[I_i(t|t) - I_i(t|t-1) \right].$$

Sinopoli et al. in [40] proved the existence of a critical value for the arrival rate of the observations which are modeled as a random process. They discuss that beyond the critical value of the rate of the observations, a transition to an unbounded state error covariance occurs. They also specify lower and upper bounds on the critical rate of arrivals of the observation. Inspired by the results in [40] we arrive at the following theorem.

Theorem 10.4.1. *If the system (10.1) is weakly controllable and weakly observable and $A(t)$ is unstable, then there exists a $\lambda_c \in [0, 1)$ such that $\lim_{t\to\infty} \mathbb{E}(\Sigma(t|t)) = +\infty$ for $0 \leq \lambda \leq \lambda_c$.*

Moreover, if $\lambda_c \leq \lambda \leq 1$ *and for all* $\Sigma_0 \in \mathcal{P}$*, there exists* $\tilde{M} \in \mathcal{P}$ *and depends on* Σ_o *such that* $\mathbb{E}(\Sigma(t|t)) \leq \tilde{M}$*.*

The probability λ *is defined as* $\lambda = \hat{\lambda}\,\bar{\lambda}$ *where* $\hat{\lambda}$ *is the probability of masuring new data by the sensors and* $\bar{\lambda}$ *is the probbaility of communication between the sensor and the coordinator. Furthermore, there exist* $\underline{\lambda}$ *and* $\bar{\lambda}$ *such that* $\underline{\lambda} \leq \lambda_c \leq \bar{\lambda}$ *where*

$$\begin{aligned} \underline{\lambda} &= 1 - \frac{1}{\alpha^2}, \\ \bar{\lambda} &= \underset{\lambda}{arginf} \left[\exists(\hat{K}, \hat{X}) \,|\, \hat{X} > \phi(\hat{K}, \hat{X}) \right], \\ \phi(K, X) &= (1-\lambda)(AXA^T + \Xi) + \lambda(FXF^T + V), \end{aligned} \tag{10.29}$$

$\alpha = \max |\alpha_i|$*,* α_i*'s are the eigenvalues of* A*,* $F = A + KH$*, and* $V = \Xi + K\Theta K^T$*.*

Proof. The Kalman filter process discussed in [40] is equivalent to the information filter mentioned above. Therefore, the lower and upper bounds in (10.29) follows directly from [40]. □

As an example, consider a network of four sensors in the coordinated decentralized estimator setup. The communication network between the sensors and the coordinator is shown in Fig. 10.6. The dynamics of the system, and the corresponding observation network are as follows:

$$\begin{aligned} x(t+1) = & \begin{bmatrix} 0.5 & 0 & 0 & 0 & 0 & 0 \\ 0.2 & -0.5 & 0.1 & 0 & 0 & 0 \\ 0.5 & 0.6 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.5 & 0 & 0 \\ 0 & 0 & 0 & 0.2 & -0.5 & 0.1 \\ 0 & 0 & 0 & 0 & 0.5 & 0.6 \end{bmatrix} x(t) \\ & + [1\ 1\ 1\ 1\ 1\ 1]^T w(t) \\ z(t) = & \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} x(t) + \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} v(t). \end{aligned} \tag{10.30}$$

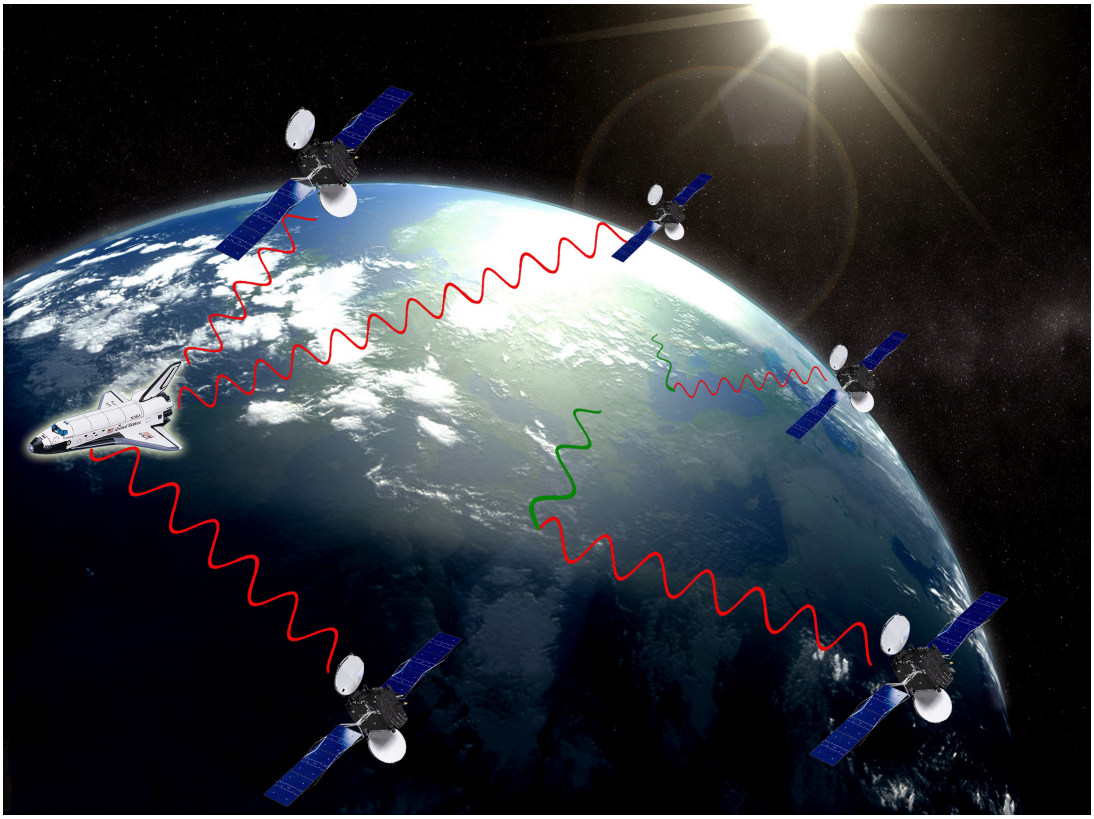


Figure 10.6: The star communication network. The dashed line represents lack of communication between the sensor and the coordinator which happens with probability $1-p$ at every time step.

It is assumed that the system and the measurement noise signals are independent zero-mean Gaussian with covariances $\Xi = 10^{-2}$ and the identity matrix, respectively.

In the proposed estimation setup, the first sensor estimates the states of the system (10.30) as it observes the signal

$$z^1(t) = [0 \quad 1 \quad 0 \quad 0 \quad 0 \quad 0]x(t) + v(t).$$

Analogously, for example, the second sensor estimates the states of the system (10.30) as it observes the vector

$$z^2(t) = [0 \quad 0 \quad 0 \quad 0 \quad 1 \quad 0]x(t) + v(t).$$

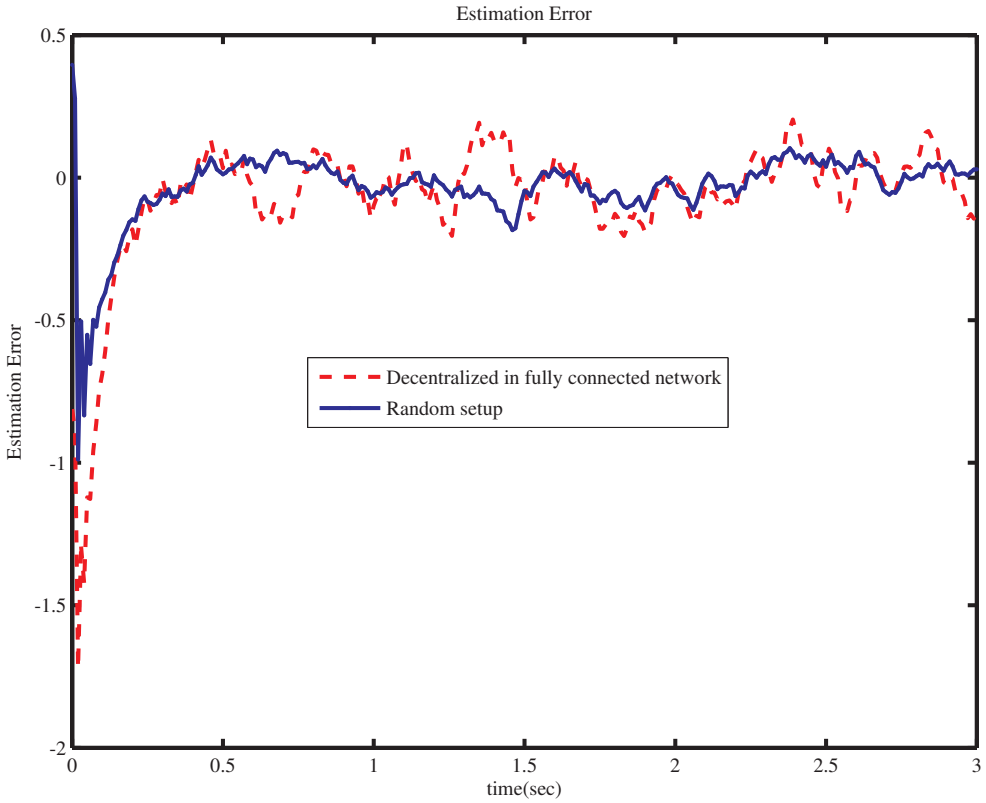


Figure 10.7: Estimation of the sixth state in different distributed estimation setups with $p = 0.6$.

In the random setup, at each time step, each sensor sends its estimate to the coordinator with probability $p = 0.6$. Fig. 10.7 demonstrates the estimation error of the sixth state in the centralized, decentralized, and the random estimation setup. Note that the system matrix in (10.30) is not observable by considering each measurement separately, while the weak observability condition holds in the random setup.

10.5 Coordinated Distributed Estimation over Opinion Dynamics

It is often unrealistic or overly expensive to make a census of the full population of a social group. An alternative is to sample the network and subsequently estimate its state dynamics. The goal in this section is to estimate the internal state of a social influence network in a neighborhood in a coordinated distributed manner.

Consider the opinion dynamics (12.3) with $\lambda_t = 1$ and the fifteen Florentine family graph depicted in Figure 12.2. At every time step a random family is chosen based on the uniform distribution to fill out a survey. The observability discussion of the setup is similar to that presented in §12.3.

The resultant estimation error is provided in Figure 12.4 for decentralized coordinator estimation setup where 1) always m members of the society are observed, referred to as the deterministic setup 2) at every time step one member is selected randomly to be observed, referred to as the random setup. Let us define the efficiency coefficient as the estimation error multiplied by the number of individuals who are observed. Figure 12.4 also depicts the defined efficiency coefficient at each time step of the simulation.

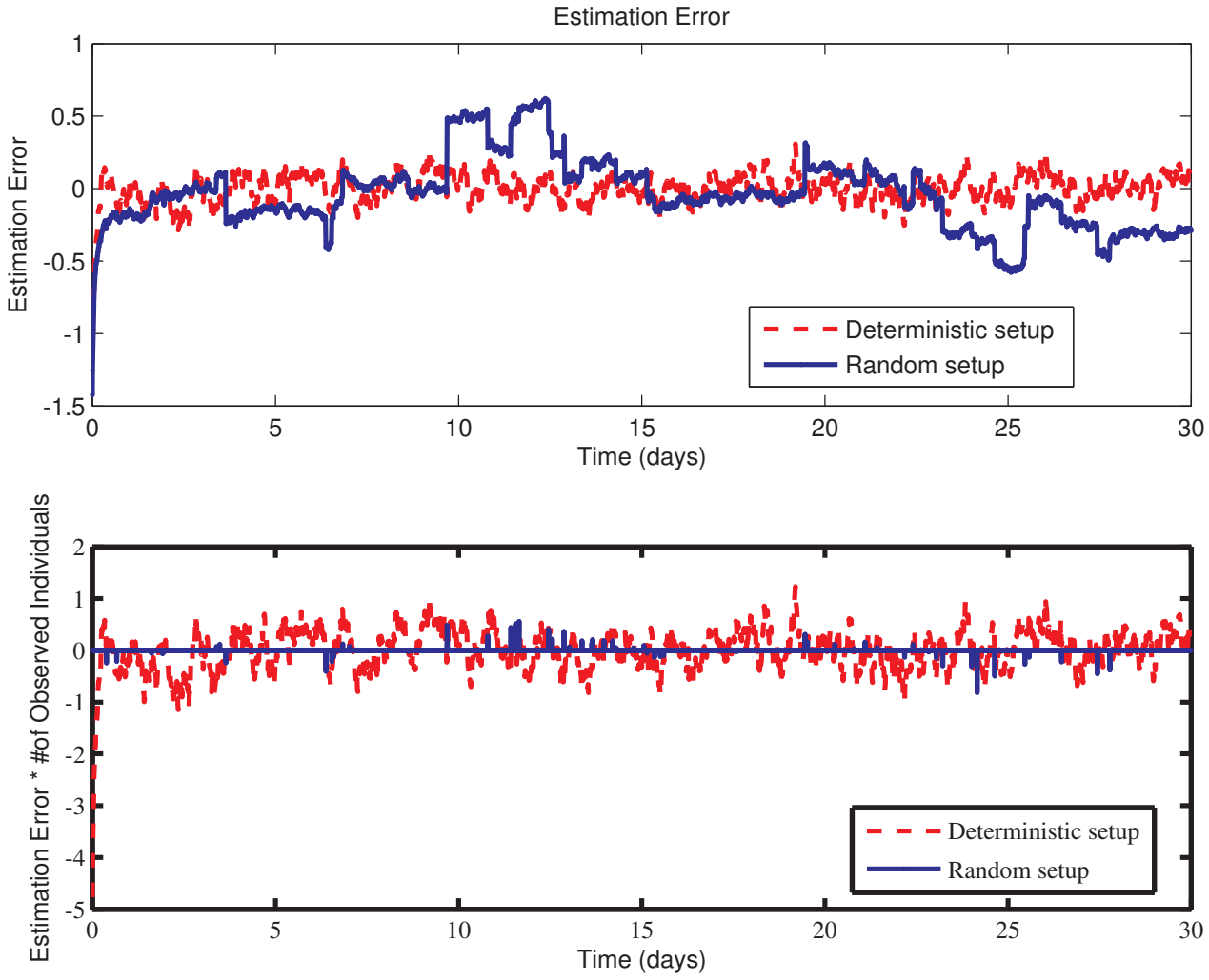


Figure 10.8: Estimation error and the efficiency coefficient of the estimation error in 1) deterministic setup 2) random communication between the observers and the coordinator. The covariance of the estimation error multiplied by the number of the observed individuals during the simulation is 121.21 for the deterministic setup while the value is 3.50 for the random scenario.

Chapter 11

ONLINE COORDINATED DECENTRALIZED LOCALIZATION OF THE SEAGLIDER WITH INTERMITTENT OBSERVATIONS

As we discussed in the previous section, packet drops in sensor networks can be modeled by random networks. We then developed a coordinated decentralized estimation procedure in the presence of packet drops. In this section, we describe an online long-baseline underwater acoustic localization system to estimation the three-dimensional position of the *Seaglider* underwater vehicle that was developed that was developed at University of Washington as a collaboration between the Applied Physics Lab and School of Oceanography [204]; depicted during a surface maneuver in Figure 11.1.

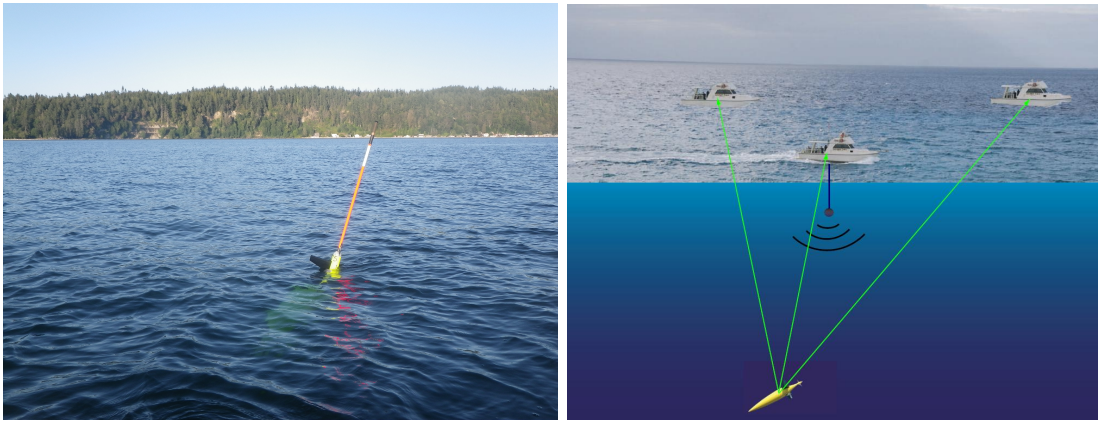


Figure 11.1: Seaglider underwater glider used in the localization experiments, [206, 207].

The system consists of three acoustic transponders that are placed at known locations at the surface of the water. The measured round-trip travel time of acoustic signals is used to calculate the slant ranges between the surface units and the glider. The range measurements from the three transponders can be used to locate the position of the vehicle using dynamic estimation algorithms.

Each transponder may fail to measure the round-trip travel time. In the experiment, approximately fifty percentile of the attempts to communicate between the transponders and the *Seaglider* failed in the shallow water. At each round of communication, the transponder updates its estimation if it receives new measurement and interact with two others to send them the new update. Packet drop-out is probable at this step as well. In another scenario, each transponder may attempt to send the updated estimation to a coordinator with some probability of failure.

11.1 Introduction

Buoyancy-driven underwater gliders are highly efficient marine vehicles that were originally developed for performing oceanographic data collection missions [204, 205]. Underwater gliders locomote by balancing between the buoyancy force and gravity via changing their net weight and the location of center of gravity relative to the center of buoyancy [208, 209, 210]. These vehicles are extremely efficient because they spend most of their time in steady trim flight condition, where they don't actively expend energy.

Theoretical study of underwater glider dynamics has revealed the potential value of novel motion control algorithms in improving efficiency and performance [208, 212, 211]. The focus of the research done in Nonlinear Dynamics and Control Lab at the University of Washington was to devise a set of field experiments and develop the corresponding field equipment and data processing tools to be able to validate underwater glider motion control algorithms[1].

The glider used in the experiment has the following properties. The glider's mass is 51 kg; the same as the mass of the displaced water when the glider is perfectly balanced. The vehicle outer skin has a streamlined and low-drag profile in order to maximize the efficiency. The maximum diameter of the hull is approximately 30 cm. The wings are attached at the maximum diameter giving a total wing-span of 1 m. The vehicle length is 1.8 m without the antenna mast.

The various navigation instruments on the vehicle include: 1) a GPS unit to obtain a

[1]Courtesy of Prof. Morgansen's Nonlinear Dynamics and Control Lab at the University of Washington, `http://www.aa.washington.edu/research/ndcl/`.

position update at the times it surfaces, 2) a 3D compass to measure heading and tilt angle, and 3) an acoustic pinger/altimeter to obtain depth and respond to acoustic interrogations. The range measurements between the glider and the surface acoustic transponders is obtained by the low-frequency acoustic pinger. The pinger has programmable interrogation and reply frequencies in the 7-15 kHz frequency band. As the glider receives an interrogation, by sending a reply allows the surface units to measure round-trip travel times.

The speed of the vehicle is approximately $V_a = 0.3$ m/s, which can be influenced by a set of programmable parameters. The pilot has control over the duration of the dive and the desired depth. The two parameters together specify the vertical descent rate.

Following we present an overview of the Long-BaseLine(LBL) system developed as part of the work done in Nonlinear Dynamics and Control Lab to estimate the position of the seaglider.

The LBL system consists of three identical beacons that were placed at known locations at the surface of the water. The beacons were placed on flotation devices to keep them above the water surface and anchored to the sea-floor to prevent them from drifting away from their deployment locations. Anchoring the beacons to the sea-floor made the geometry of the beacon network unaltered during the field-experiment, meanwhile, prevent the loss of the equipments. The bottom topography could obscure straight line-of-sight between the glider and the beacons, which required several days of area surveying in order to find optimal locations to implement the experiment; Figure 11.2 depicts the geometry that was used in the field experiments discussed in this section.

Each of the beacons consists of an acoustic ranging unit, GPS antenna, a computer to synchronize the ranging measurements and to log data, and batteries to power all the components. The acoustic ranging unit consists of an AT-440 LF transponder and UDB-9000 deck box, made by Benthos. The deck box features a serial communication port over which commands can be received from a computer. The batteries are 12V 5000 mAh NiMh battery packs with the life cycle of close to 8 hours. Three of these packs are used in parallel to power the laptop and an additional unit provides power for the GPS unit separately. The computer issues requests for an acoustic ping over the serial port, and the response received from the *Seaglider* is reported back to the computer upon receipt. The GPS antenna is

Figure 11.2: The geometry of ranging units that was used in the ranging field experiments, [206, 207].

a Garmin 17xHVS unit that reports position information in NMEA 0183 compatible data format. GPS updates are reported every second over the serial port. The updates are synchronized to GPS time, which is exploited to orchestrate the sequence of interrogations from different beacons.

All the electronics are housed inside Pelican case enclosures that were fitted with waterproof connectors for communication between the computer and the UDB-9000 deck box. The system also contained Freewave radio modems that were intended for communication between the beacons and an operator station at the beach.

Each beacon was allocated a time window, outside which it was not allowed to ping; see Figure 11.3. The pings were issued in a round-robin fashion, where each node pinged the glider exactly once within a ranging cycle. The time period of the ranging cycle is mission dependent, and should be as frequent as possible. We chose 4 second window size (12 seconds period for an entire ranging cycle).

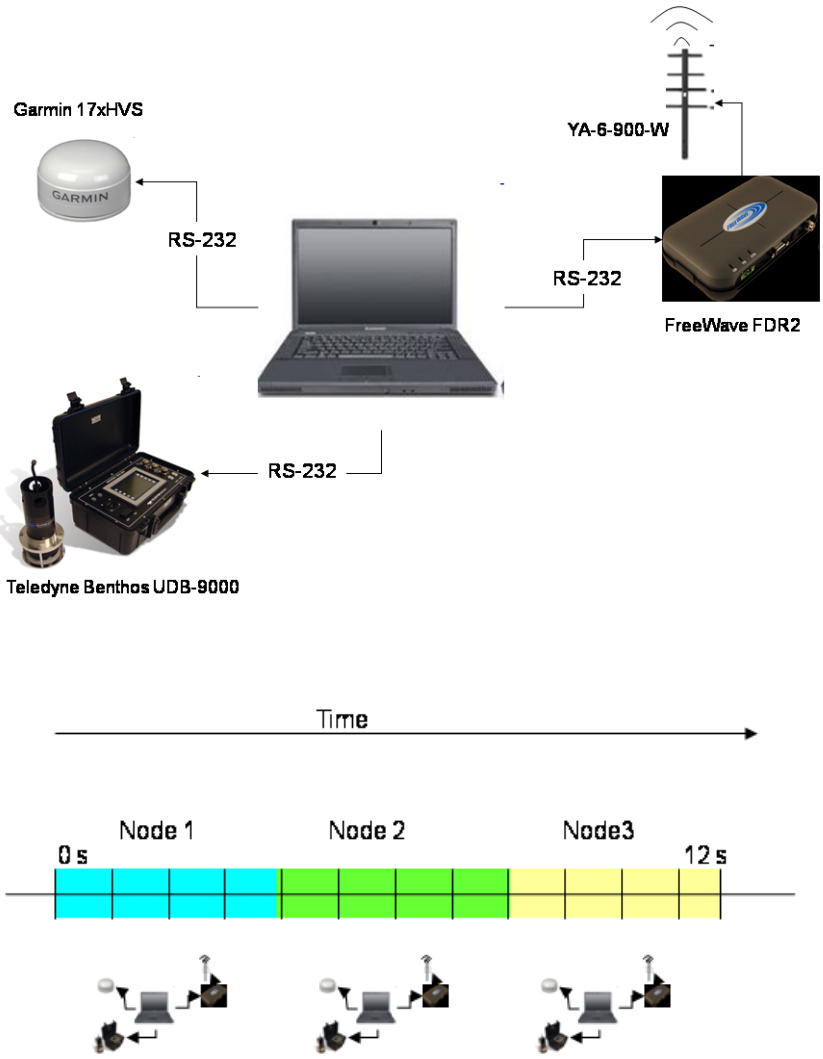


Figure 11.3: (Left) The components in each node share information via serial communication. (Right) A ranging cycle is divided into three 4 second time-windows. Each node is allowed to ping only in its allocated time-window, [206, 207].

The ping is issued at the beginning of the time-window, and the algorithm waits for the response not more than 4 seconds. If a response is received, the data are recorded in the log file, with corresponding timing and GPS information. The beacon updates the estimation of the glider position based on the new measurement and attempts to communicate with the two other beacons and send them the updated estimation. The beacon then goes to sleep until the beginning of the next time-window. If no response is received within 4 seconds following the ping request, the algorithm registers it as "no response received", and the beacon sends its prediction of the glider position to its neighbors if it is able to communicate with them.

The length of the time window was selected to give maximal resolution, while avoiding multiple beacons pinging at the same time. In order not to register a response to the previous node's request as a valid return, the time-window has to be long enough that the pings are mutually exclusive. Based on this idea, the time-window effectively determines the maximum range of the beacon network; for a 4 second window size, the glider has to be within approximately 3 km to each of the beacons.

11.2 Underwater Acoustic Ranging

n underwater applications, acoustic signals are primarily used for communication rather than radio signals due to poor propagation characteristics of radio signals in water [213]. Acoustic signals can spread over long distances depending on the frequency of the signal and the physical properties of the medium (pressure, salinity, temperature).

The measured round-trip travel time of an acoustic signal is given by the equation

$$t_{RTT} = t_{TAT} + \frac{2R^{3D}}{c} + \epsilon, \tag{11.1}$$

where t_{TAT} is the turn-around time (the amount of time the electronics needs to detect the signal and send a response), R^{3D} is the slant range to the vehicle, and c is the speed of sound in water. The term ϵ represents the error between the true and measured round-trip travel time. The error term in equation (11.1) can be written as the sum of two components: $\epsilon = \epsilon_s + v$, where ϵ_s represents the systematic errors and v is the random noise component. While the random noise component is assumed to be white Gaussian zero-mean with known

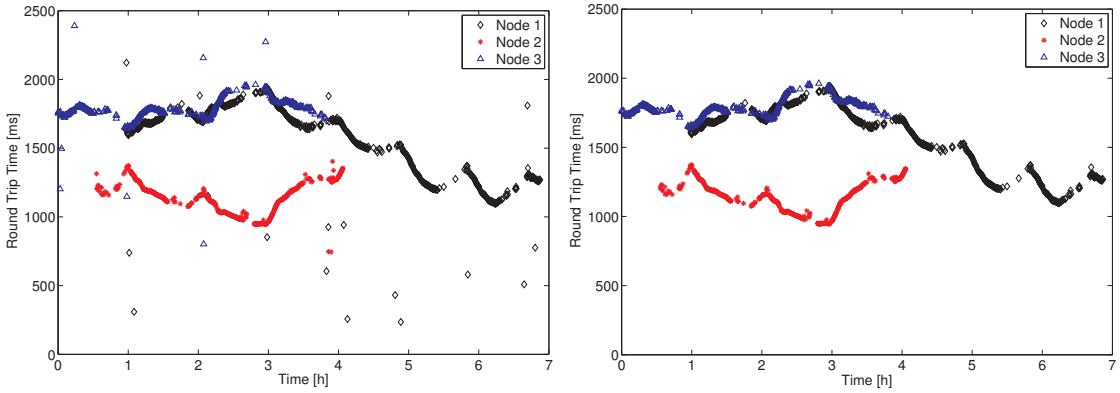


Figure 11.4: Round-trip travel times to the *Seaglider* on July 15, 2010. The figure on the left shows the raw measurements, the figure on the right shows the data after the outliers have been removed using a difference filter, [206, 207].

variance, the major systematic error components need to be eliminated.

One class of systematic error is cause by *Multipath*. Multipath is the phenomenon when an acoustic signal is reflected off of a sound barrier — such as the ocean bottom, the water surface, a thermocline or a vertical wall — and is received at the beacon with a time delay and phase shift. The error due to multipath often has significant magnitude, which results in sudden jumps in the position estimation. These "outliers" may be removed manually, or by use of methods based on plausibility validation in spatial or temporal domains [214]. In case they appear singularly, a simple differentiation of the consecutive measurements successfully reveals outliers, as they appear as distinctive spikes; as depicted in Figure 11.4.

Now we briefly review the seaglider dynamics and the hybrid extended Kalman filter.

11.3 Dynamic Filtering

In §11.3.1, we briefly review a simple unicycle model for the seaglider. The implementation we present in Section 11.3.2 is hybrid EKF adopted from [215]. In addition to improving the estimation accuracy, the EKF can also serve as an adaptive parameter estimator for unknown model parameters - constant or slowly varying. The current velocity and the vehicle's flow-relative velocity are treated as unknown parameters.

11.3.1 Vehicle Model

A simple unicycle model is employed in the glider localization problem. The model was extended with trivial dynamics of the velocities to allow the filter to adaptively estimate these unknown parameters:

$$\dot{x}_N = V_a \cos\psi + V_x \tag{11.2}$$

$$\dot{y}_E = V_a \sin\psi + V_y \tag{11.3}$$

$$\dot{\psi} = u \tag{11.4}$$

$$\dot{V}_a = 0 \tag{11.5}$$

$$\dot{V}_x = 0 \tag{11.6}$$

$$\dot{V}_y = 0. \tag{11.7}$$

In the above equations V_a is the flow-relative speed of the glider, V_x and V_y are the North and East components of the current velocity vector, and ψ is the heading angle measured from North. The only external input in this model is the turn rate of the vehicle, u, which is determined by the vehicle roll angle. Notice that the velocities V_x, V_y, and V_a are treated as states in the model with trivial dynamics. In reality they are constant or slowly varying parameters, the values of which are to be estimated. The EKF discussed in the next section provides estimates of these parameters along with the vehicle position information.

11.3.2 Hybrid Extended Kalman Filter

The EKF is an extension of the Kalman filter for nonlinear processes. The algorithm relies on the linearization of the nonlinear dynamics. As long as the nominal point about which the linearization is performed approximates the real state closely, the linearized model for propagating the state and covariance will be valid, and the Kalman update equations will yield the minimum variance estimate of the states.

To estimate the position of the underwater glider, one may use a hybrid EKF. The discussion below follows from Chapter 5 of [215]. Since the states are propagated in continuous time while the measurements are obtained at discrete time-steps, whenever a new slant-range measurement is available, the filter will be hybrid. Let us define the state vector

as $x = (x_N, y_E, \psi, V_a, V_x, V_y)^T$, and then, equations (11.2)-(11.7) can then be written in the form

$$\dot{x}(t) = f(x(t), u(t)) + w(t), \tag{11.8}$$

where $w(t) \sim \mathcal{N}(0, \Xi(t))$ is the white Gaussian zero-mean process noise. At discrete time intervals $t = k\,\Delta t$ measurements are available by the output equation

$$y(k\,\Delta t) = h(x(k\,\Delta t)) + v(k\,\Delta t), \tag{11.9}$$

where Δt is the sampling period, $k \in \mathbb{Z}$, and $v(k\,\Delta t) \sim \mathcal{N}(0, \Theta_k)$. The measurement vector for the underwater localization problem is

$$h(x(k\,\Delta t)) = \begin{pmatrix} R(k\,\Delta t) \\ \psi(k\,\Delta t) \end{pmatrix}, \tag{11.10}$$

where $R(k\,\Delta t)$ is the in-plane range measurement from a corresponding beacon, and $\psi(k\,\Delta t)$ is the vehicle heading angle obtained from the compass. Between the measurements the state estimate is propagated in continuous time using the assumed nonlinear dynamic model without noise as

$$\dot{\hat{x}}(t) = f(\hat{x}(t), u(t)). \tag{11.11}$$

The state covariance matrix is propagated using the linearized state matrix and using the continuous-time Kalman filter equations

$$\dot{\Sigma}(t) = A(\hat{x}(t))\Sigma(t) + \Sigma(t)A^T(\hat{x}(t)) + \Xi(t), \tag{11.12}$$

where

$$A(\hat{x}(t)) \equiv \left.\frac{\partial f}{\partial x}\right|_{\hat{x}(t)} = \left.\begin{pmatrix} 0 & 0 & -V_a \sin\psi & \cos\psi & 1 & 0 \\ 0 & 0 & V_a \cos\psi & \sin\psi & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}\right|_{\hat{x}(t)}. \tag{11.13}$$

As a new measurement arrives at time step k, the Kalman gain matrix K_k is computed at the measuring beacon and corrections are applied to the mean and covariance estimate of the state vector as follows,

$$K_k = \Sigma(k|k-1)H_k^T(\hat{x}_k^-)\left(H_k(\hat{x}_k^-)\Sigma(k|k-1)H_k^T(\hat{x}_k^-) + \Theta_k\right)^{-1}. \tag{11.14}$$

In the above equation

$$H_k(\hat{x}_k^-) \equiv \left.\frac{\partial h}{\partial x}\right|_{\hat{x}_k^-} = \left.\begin{pmatrix} \frac{\hat{x}_{N_k}^- - x_0}{\hat{R}_k} & \frac{\hat{y}_{E_k}^- - y_0}{\hat{R}_k} & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix}\right|_{\hat{x}_k^-}, \tag{11.15}$$

where $(x_0, y_0)^T$ are the corresponding node locations that provided the range information, and

$$\hat{R}_k = \sqrt{(\hat{x}_{N_k}^- - x_0)^2 + (\hat{y}_{E_k}^- - y_0)^2}. \tag{11.16}$$

Then, the position estimation and the error covariance are updated as

$$\hat{x}_k^+ \;=\; \hat{x}_k^- + K_k\left(y_k - h(\hat{x}_k^-)\right)$$

and

$$\Sigma(k|k) \;=\; \left(I - K_k H_k(\hat{x}_k^-)\right)\Sigma(k|k-1). \tag{11.17}$$

The vector $\hat{x}_k^-$ is the *a priori* estimate of the state that is based on the process model (11.11). Equation (11.11) is implemented using numerical integration between the measurement updates. The output of this integration routine is $\hat{x}_k^-$ which is the prior estimate of the state before the measurement arrives. The vector $\hat{x}_k^+$ is the *a posteriori* estimate of the state vector, i.e., the estimate after the measurement has been obtained and processed at time step k.

11.4 Experimental Results

The field experiments presented in this section took place in Port Susan on July 15-16, 2010. The location for the experiments was chosen because of its vicinity to University of Washington campus and because of the relatively minimal boat traffic compared to other locations in the Puget Sound. Port Susan is approximately 110 meters deep in the middle of the bay and tidal currents are typically strong. The water features high salinity stratification which, however, is confined to a very shallow region close to the water surface. This is due to the large amounts of freshwater entering the bay, creating a freshwater lens on the top of the water column. This layer of freshwater causes deterioration of the acoustic signal

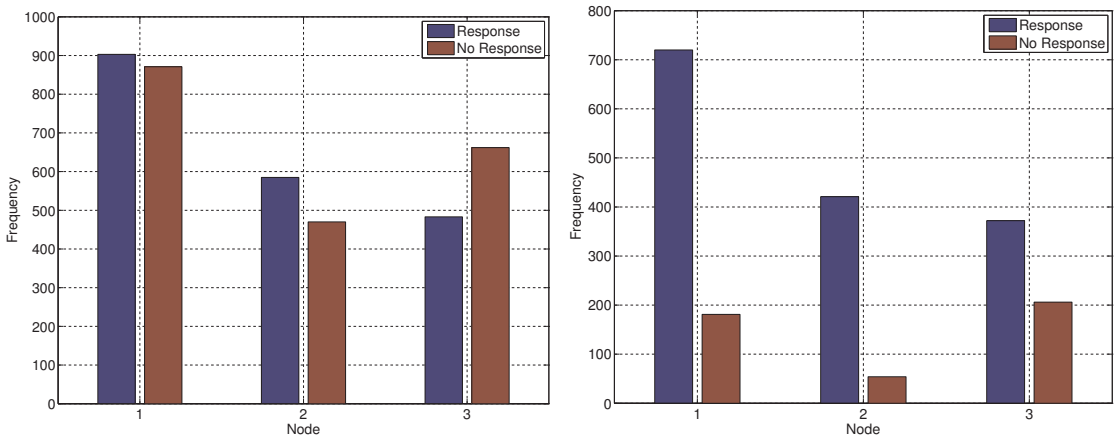


Figure 11.5: Ranging experiments performed in Port Susan on July 15, 2010. Relative frequency of valid responses versus no responses, [206, 207].

quality, and many responses are lost. For this reason, the transponders were lowered below this freshwater layer to 20 meters depth.

A *Seaglider* was deployed in the afternoon on July 14, 2010, and recovered on July 16, 2010. The glider was programmed to perform 50 minute dives to the bottom. Figure 11.5 shows histograms of "valid responses" versus "no responses received" from July 15. Comparison of Figure 11.5 left versus right illustrates the deterioration of acoustic responses when the glider is close to the water surface. An approximately four hour portion of the entire data set represents a time window, where all three nodes were simultaneously active.

The underwater localization methods described in Section 11.3 were used to estimate the underwater position of the *Seaglider* from experimental data collected in Port Susan on July 15, 2010. The results from dive number 18 can be seen in Figure 11.6. The dive started at 11:56 a.m. PDT and lasted approximately 50 minutes.

The setup of the experiment is as follows. Every 4 seconds one node pings to the glider. Updated estimation of the seaglider position would be attempted to be sent to other nodes if the new measurement is available; otherwise, the predicted estimation based on the previous measurement is sent to other nodes. Note that with some probability the attempt to communicate with other nodes would fail.

The data were reduced to include only the 30 minute portion that the glider spent below 20 m depth, due to deterioration of acoustic responses near the water surface, as mentioned in Section 11.2.

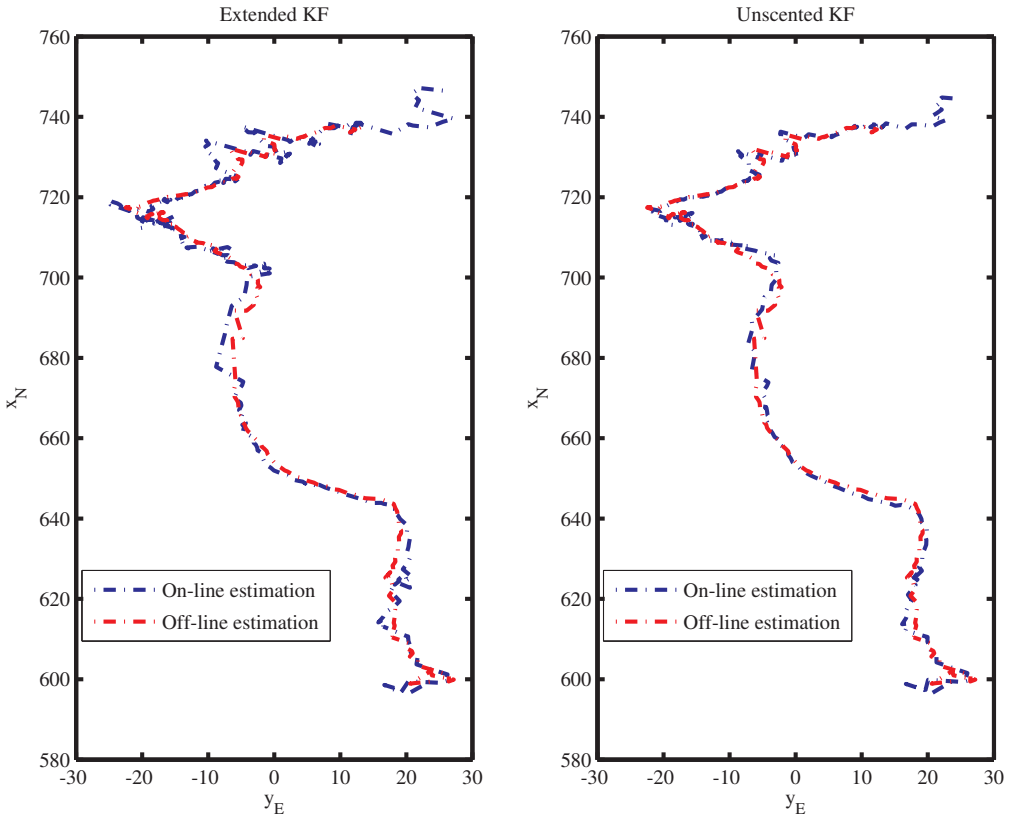


Figure 11.6: Estimation of the position in the off-line and on-line setting with extended and unscended Kalman filter while the probability of each beacon sends its estimation ot the neighbor beacons are $p_1 = p_2 = p_3 = 0.8$.

Figure 11.6 demonstrates the position estimation of the seaglider in real-time scenario compared to the case that the measurements are taken and the estimation is implemented off-line.

Chapter 12

SOCIAL CONTROL AND OPTIMAL MARKETING

This chapter presents a framework to design optimal marketing strategy over an opinion dynamics; utilizing the results presented in chapter 9.

12.1 Introduction

Studying the opinion dynamics has been the center of attention of researchers studying social networks [73, 71, 72, 79, 80, 91]. To describe the problem, consider a group of interacting agents among whom some process of opinion formation takes place. The group may be a small one or an entire society in which the individuals influenced by their different social networks develop a wide spectrum of opinions. In each case, there is a process of opinion dynamics that may lead to a consensus among individuals or more generally a more stratified pattern of opinions. As an example, understanding how opinions form and then diffuse through a population is a central theme in investigating voters behaviour and effectiveness of the propaganda.

The pertinent literature covers a wide set of research topics and applications. For example, in[71] and [73] the authors study the effect of social positions in influence networks and describe how a network of interpersonal influences enters into the formation of agents' settled opinions. Yildiz et al. [80] study discrete opinion dynamics in a social network with "stubborn agents" who influence others but do not change their opinions. The importance of the social networks and opinion dynamics in analyzing diseases correlated with behaviours such as tobacco or alcohol use, poor nutrition, and lack of physical activities is studied in [75]. Kuzoki in [83] applies the influence mechanisms to an application of freight mode choice to investigate the effect that opinions have on choice set considerations, attribute perceptions, and the market adoption of a new rail freight service.

Acemogluy and Ozdaglar in [81] provide an overview of recent research on belief and

opinion dynamics in social networks. They discuss both Bayesian and non-Bayesian models. We now review two consensus-based non-Bayesian models and choose one model for the rest of the analysis.

12.2 Opinion Dynamics and Social Control Models

The first model is adopted from [72, 73] and it has been under development by social psychologists and mathematicians since the 1950's [72]. The time invariant model described in [72] and the time varying model in [73] postulate simple recursive definitions for the influence process in a group of N agents. For the time invariant case this model assumes the form

$$x(t+1) = WF(\mathcal{G})x(t) + (I - W)x(0) \quad t = 1, 2, \cdots, \tag{12.1}$$

where $x(0)$ is an $N \times 1$ vectors of agents' initial opinions on an issue and $x(t)$ is an N vector of agents' opinions at time t. The interpersonal influences are captured by $F(\mathcal{G}) \in \mathbb{R}^{N\times N}$ where $0 \leq F_{ij} \leq 1$, $\sum_j^N F_{ij} = 1$ and $\mathcal{G}$ is the underlined interaction network. And the diagonal matrix $W = \mathbf{diag}(w_{11}, w_{22}, \cdots, w_{NN})$ is the agents' susceptibilities to interpersonal influence on the issue ($0 \leq w_{ij} \leq 1$).

The second model is adopted from [78]. In this work, De Marzo et al. study a model of opinion formation in which individuals are subject to "persuasion bias"; that is, they fail to account for possible repetition in the information they receive. According to this research, persuasion bias implies that the phenomenon of social influence depends not only on accuracy, but also on how well-connected one is in the social network of interactions. In order to formalize this setup, let the beliefs of the i^{th} agent at time t is denoted by $x_i(t)$. The persuasion assumption is that agents ignore repetitions of information. Rather, agents treat the information they hear in each round as new and independent, and ascribe the same relative weights to those they listen to at each time step. Agents are allowed, however, to vary the weight they give to their own beliefs relative to outsiders over time as they update their opinions.

Agent i's beliefs at time $t+1$ is updated based on

$$x_i(t+1) = (1-\lambda_t)x_i(t) + \lambda_t \sum_{j \in \mathcal{N}_i} \frac{\pi_{ij}^t}{\pi_{ii}^{t+1}} x_j^t, \tag{12.2}$$

where $\lambda_t \in (0,1]$ is the relative weight that agent i gives to its own belief and is identical for all agents; the set $\mathcal{N}_i$ refers to the listening set of the i^{th} agent including the agent i; agent i assigns the precision π_{ij}^t at time t to agent j (π_{ij}^t is the weight that indicates how much the i^{th} agent is influenced by the j^{th} agent). And the precision of agent i to her belief at time t is denoted by $\pi_{ii}^{t+1} = \sum_{j \in \mathcal{N}_i} \pi_{ij}^t$. Thus, in the compact form, agents beliefs after round $t+1$ is expressed as

$$x(t+1) = T_t(\mathcal{G})x(t), \tag{12.3}$$

where $T_t(\mathcal{G}) = (I-\lambda_t)I + \lambda_t T(\mathcal{G})$ and $T(\mathcal{G})$ denotes the listening matrix with element $T_{ij}(\mathcal{G}) = \frac{\pi_{ij}^t}{\pi_{ii}^{t+1}}$ if i listens to j, and zero if otherwise.

Notice that the dynamics (12.3) is an adaptation of the traditional opinion dynamics in (12.1) when $W = I$ and $F(\mathcal{G}) = T_t(\mathcal{G})$. The two models mentioned above are consensus-based and non-Baysian and describe how an individual's opinion is influenced by her neighbors in the social network. In practice, however, we are affected by external signals such as media, advertisements, etc. The goal in this section is to consider the external signals as exogenous inputs and systematically study the behavior of the social opinion in presence of the input signal. From control engineers perspective, the term *social control* appears reasonable for the mentioned problem.

In sociology, the concept of *social control* has undergone various transformations. In contemporary sociology, social control is primarily understood in the context of the enforcement of law and/or the control of crime and deviance [82]. *Social control* can be viewed as social order. In this direction, from the late nineteenth century onward, social control was used primarily in American sociology to refer to a society's capacity to regulate itself without resource to force. An important theoretical shift in the sociology of *social control* came into the existence in the period following World War II, where the concept of social control was employed to refer to the more repressive and coercive forms of control that are instituted, not by socialization into norms, but on the basis of power and force. The

emphasis in this concept of *social control* is thus on control. Another view of *social control* is on control of the crime and deviance. From the 1950s onward, *social control* refers to those institutions and mechanisms that define and functionally respond to crime and/or deviance [82, 84, 85].

The *social control* can be used to alter, modify, deconstruct an existed behavior/habit in the social network, or construct a new one. As an illustration of such utility, Craig in [86] demonstrates how businesses employ their advertising to socially control and thus censor viewpoints they do not like. Other examples can also be found in the literature that exhibit the use of media, advertisements, and other exogenous inputs to socially control and influence the opinions and behaviors of the individuals. The goal in this section is to quantify the effect of the exogenous input signals to control the opinion dynamics in an arbitrary social network as modeled in (12.3).

Utilizing an exogenous input signal as a method to model external influencing factors, to the best knowledge of the authors, is not common in modeling opinion dynamics over social networks. Qian et al. in [88] present an efficient opinion control strategy for complex networks, in particular, for social networks. They propose an adaptive bridge control (ABC) strategy that calls for controlling special individuals referred to as "bridge" and requires no knowledge of the bridges' local or global communication links. The efficiency of the proposed ABC strategy has been examined by implementing the strategy on random networks, small-world networks, scale-free networks, and the random networks adjusted by the edge exchanging method.

In [88], bridges describe special individuals with possibly small degree but connecting different groups within the network. The opinion dynamics of the bridges is augmented by a value $r(t)$ which expresses the control strength of the evolution process of the opinion. Different kind of individuals who can be selected to be influenced by an exogenous input signal are discussed in more details later in the section.

The purpose of this section is to control the internal states of a social influence network in a neighborhood utilizing an optimal marketing strategy. Optimal marketing strategies over social networks have been the center of attention of researchers [89, 90] as well as companies such as Microsoft and Google. As an example, Dayama et al. in [90] study optimal mix

of incentive strategies for product marketing on social networks. Hartline et al. in [89] also discuss the use of social networks in implementing viral marketing strategies. In this section, we propose an optimal marketing strategy by utilizing linear quadratic regulator design on networks.

Let us consider the model defined in (12.3). The opinions and behaviors of the individuals are influenced by their personal social networks as well as exogenous components such as media and advertisements. We assume the advertising happens every k steps while the opinion dynamics is evolving every one step. We also assume a noisy observation of the opinion dynamics with the same rate that the advertisements are arriving. Therefore, the modified dynamics in (12.3), for $\lambda_t = 1$, can be represented as

$$\begin{aligned} x(t+1) &= T(\mathcal{G})^k x(t) + Bu(t) + w(t) \\ z(t) &= Hx(t) + v(t), \end{aligned} \tag{12.4}$$

where $t \in \mathbb{Z}$ is the time when the advertisements and the observations are carried out; the integer k is the difference between the time step of the opinion dynamics and that of advertisements/observations. The term $u(t)$ captures the effect of exogenous inputs such as media and advertisements that influence the individuals; the observation of the internal states $x(t)$ provided to the outside world is denoted by $z(t)$. Terms $w(t)$ and $v(t)$ model the dynamics and observation disturbances and are assumed to be independent Gaussian signals with covariances Ξ and Θ, respectively.

The input and output matrices B and H in (12.4) indicate the randomly chosen individuals to be influenced and observed, respectively. If the set of input nodes in the N node graph is $S_{\mathcal{I}} = \{i_1, i_2, \ldots, i_p\}$, the corresponding input matrix is $B = [e_{i_1}, e_{i_2}, \ldots, e_{i_p}] \in \mathbb{R}^{n \times p}$, where $e(i)$ is the column vector with all zero entries except $[e_i]_i = 1$. The set of output nodes $S_{\mathcal{O}}$ and the corresponding output matrix can be identical to or distinct from the input set of nodes and the input matrix, respectively.

Controlling the spread of rumors or idea, which seem nonsignificant at first but they may even break out is essential over social networks. Researchers have proposed lots of immunization strategies [93], including random immunization, targeted immunization [94, 95], and acquaintance immunization [96, 97], and also the pinning control of dynamics in

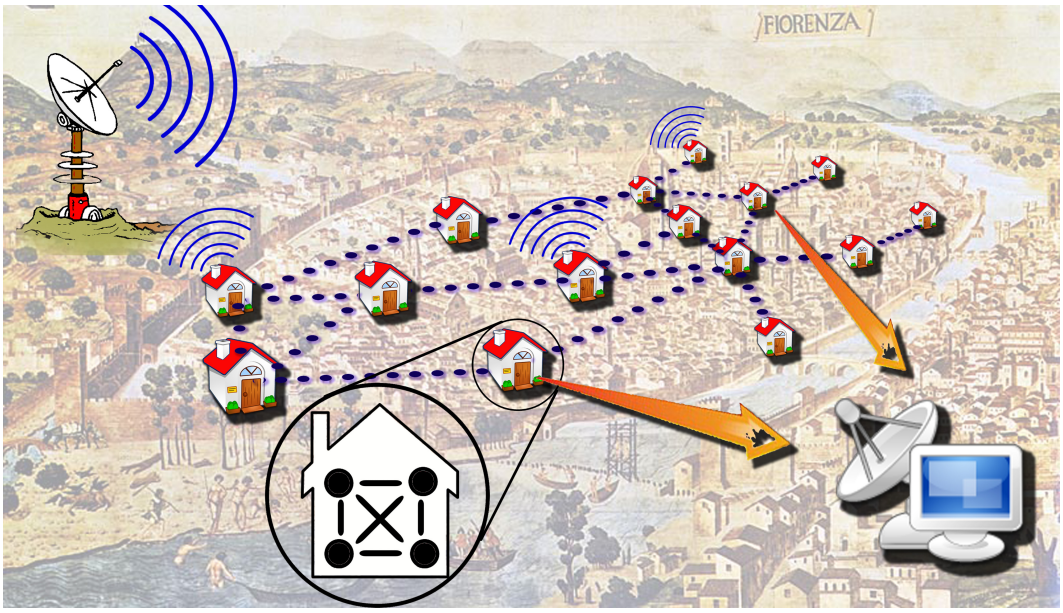


Figure 12.1: The picture depicts fifteen Florentine family graph to analyse the social control and optimal marketing. Randomly selected individuals get to see some advertisements or are influced by media. Another group, also, randomly selected, get to fill out a survey. The combination of these set of influenced and observed individuals provide data to promote a new idea, a product, or a behavior in the community.

complex networks [98]. The influenced and observed individuals, in our setup, are selected from a specific random distributions. Before, any further analysis, we explore and justify why and how a group of randomly chosen influenced and observed individuals would control the opinion dynamics and the social behaviors.

There are two types of ties in the society; strong and weak ties. Personal communication between closer and stronger links are within a personal group (strong ties) and weaker and less personal communications are a set of other acquaintances, colleagues, or random individuals on street (weak ties) [76, 77]. Granovetter in [76] mentions that most network models focus, implicitly, on the strong ties. In his well-known theory, Granovetter claims that the cohesive power of weak ties, however, should be stressed.

Summary of the research done regarding the strength of weak ties is described as follows: the strength of weak ties is at least as strong as the influence of the strong ties. Despite the relative inferiority of the weak tie parameter in the model's assumption, their effect approximate or exceeds that of strong ties, in all stages of the opinion life cycle [77]. In this research, the randomly chosen influenced and observed individuals are considered as sample of weak ties in the main social network. The influencing and observing of the selected individuals can be carried out through different tools such as internet marketing and internet polling. The analysis delineates the effect of random input and output agents in the control of opinion dynamics and social behavior.

The following sections present the simulations that validate the mentioned analysis in previous sections. The simulations are based on machine-generating data. Testing the studied theory on machine-generated data, as opposed to empirical data only, allows us to conduct repeatable tests that stress and explore certain aspects of the theory.

The example was analyzed for fifteen, four member families. The family interaction graph $\mathcal{G}$, based on the famous Florentine family graph [87], is depicted in Figure 12.1 with nodes representing the fifteen families. The goal is to design an optimal linear-quadratic-Guassian feedback control to regularize the opinions of the fifteen families regarding a new product or behavior. The deigned LQR feedback control law in §9.4 required access to all the internal states which is unrealistic or overly expensive; therefore, an estimator coupled with the LQR controller will be designed.

12.3 Controllability and Observability

The example is analyzed for fifteen, four member families. The family member graph is assumed to be a complete graph where all four members assumed to have identical influential on each other. On the other hand, the family interaction graph $\mathcal{G}$ is based on the famous Florentine family graph [87] and depicted in Figures 12.1 and 12.2.

For the sake of the simplicity of the simulation, it is assumed that when families i and j interact, all four members of each family also interact and influence each other in identical

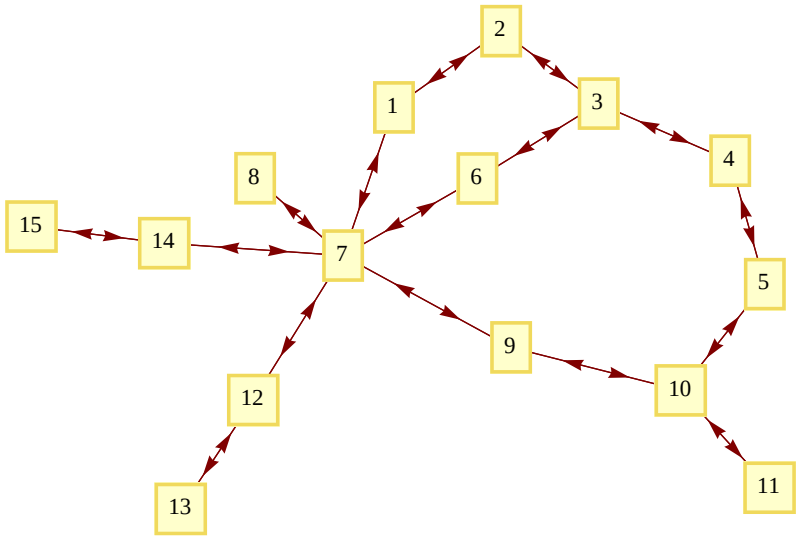


Figure 12.2: The fifteen Florentine family graph.

manner. [1] It is sufficient to study the evolution of opinion dynamics over the interaction between the fifteen families, depicted in graph $\mathcal{G}$. Consider a communication network with 15 nodes, instead of 60 nodes, is justified for two reasons: (1) the complete and identical influence dynamics of the inter-family network (2) the assumption that all four members of families i and j influence each other identically as long as they communicate in the intra-family network G.

The discrete time opinion dynamics of the interaction have a state matrix $A(\mathcal{G}) = \exp(-\delta\, L(\mathcal{G}))$. Consider the time step to be equivalent to one day. Every k days, m individuals are selected uniformly to see an advertisement which is equivalent to the signal $-K^* X(T)$. The weak controllability defined in (9.3.1) holds if $A(t)$ is controllable and there exists a time T such that $\sum_{t=1}^{T} B(t)B'(t)$ is also invertible. There exists a time T such that $\sum_{t=1}^{T} B(t)B'(t)$ is full rank since detecting the influenced individuals (the input nodes in matrix B) happens uniformly and the probability that one individual is not selected at some

[1] Another type of communications between these fifteen, four member, families is studied in [216]. The setup in [216] analyzes a scenario that as two families meet, same gender parents and same age children tend to interact. This family inetraction structure is a Cartesian network with factor networks; the inter-family network and intra-family network.

point approaches zero as time goes to infinity. Therefore, the system in (12.4) is weakly controllable.

12.4 Estimation and Linear Quadratic Regulators

The input signal is $u = -K^*x(t)$ and requires access to all internal states $x(t)$. It is often unrealistic or overly expensive to make a census of the full population of a social group in order to obtain $x(t)$. An alternative is to sample the network and subsequently estimate its state dynamics. In the same manner as the control part, m' uniformly selected individuals are observed every k days. The separation principle [34] allows that the control and estimation algorithms can be designed separately. Then, it follows that

$$u = -K^*\hat{x}(t), \tag{12.5}$$

where $\hat{x}(t)$ is the result of a discrete Kalman filter that was applied to the dynamics in (12.4). Therefore, we arrive at the following closed-form

$$\begin{aligned} x(t+1) &= Ax(t) - BK^*\hat{x}(t) + w(t), \\ \hat{x}(t) &= (A - BK^*)\hat{x}(t) + K_e(z(t) - H\hat{x}(t)), \end{aligned} \tag{12.6}$$

where K_e is the estimation gain obtained from the Kalman filter.

The resultant mean squared error, covariance matrix trace, and a sample state estimate over time is provided in Figure 12.4, supporting the controllability of the pair $(A(\mathcal{G}), B(t))$ and the observability of $(A(\mathcal{G}), H(t))$. The required input signal $u(t)$ and K^* are also depicted in Figure 12.4.

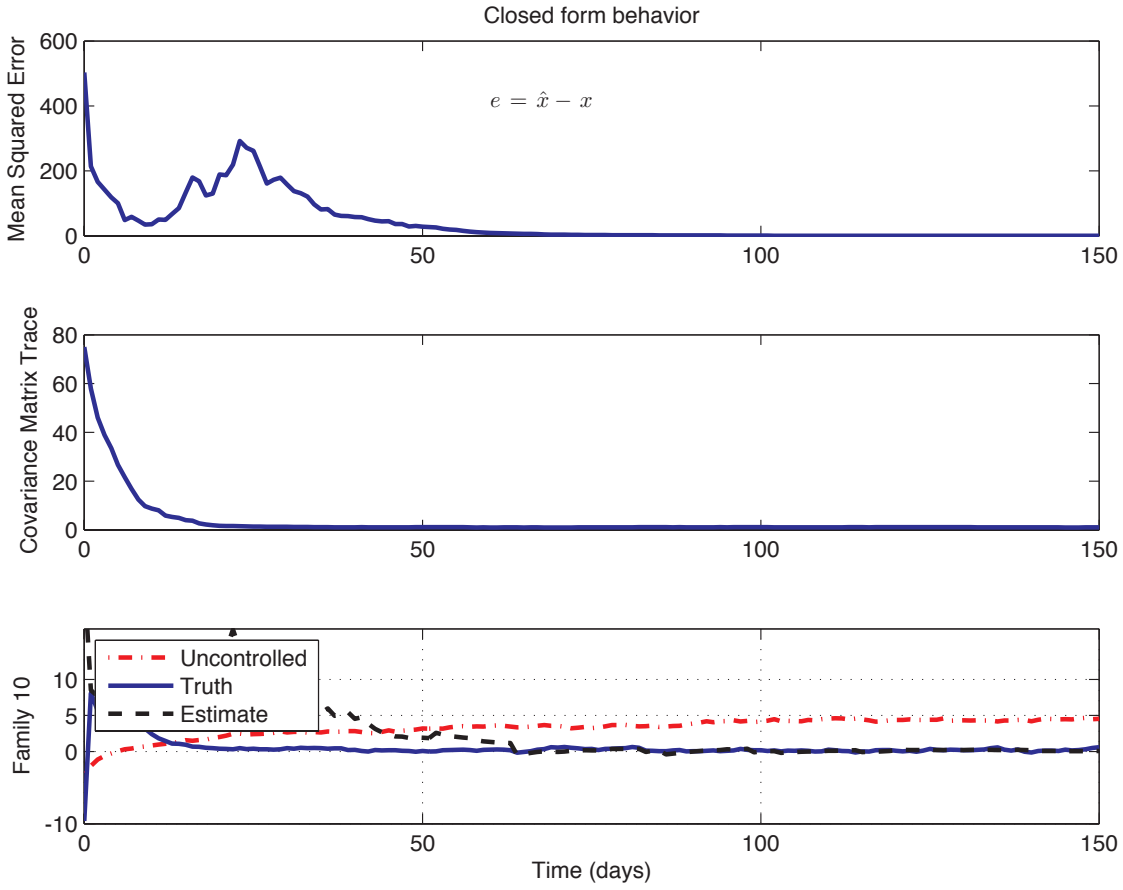


Figure 12.3: The mean squared error, covariance matrix trace and state of a random family (family 10) over time for the closed form dynamics in 12.6.

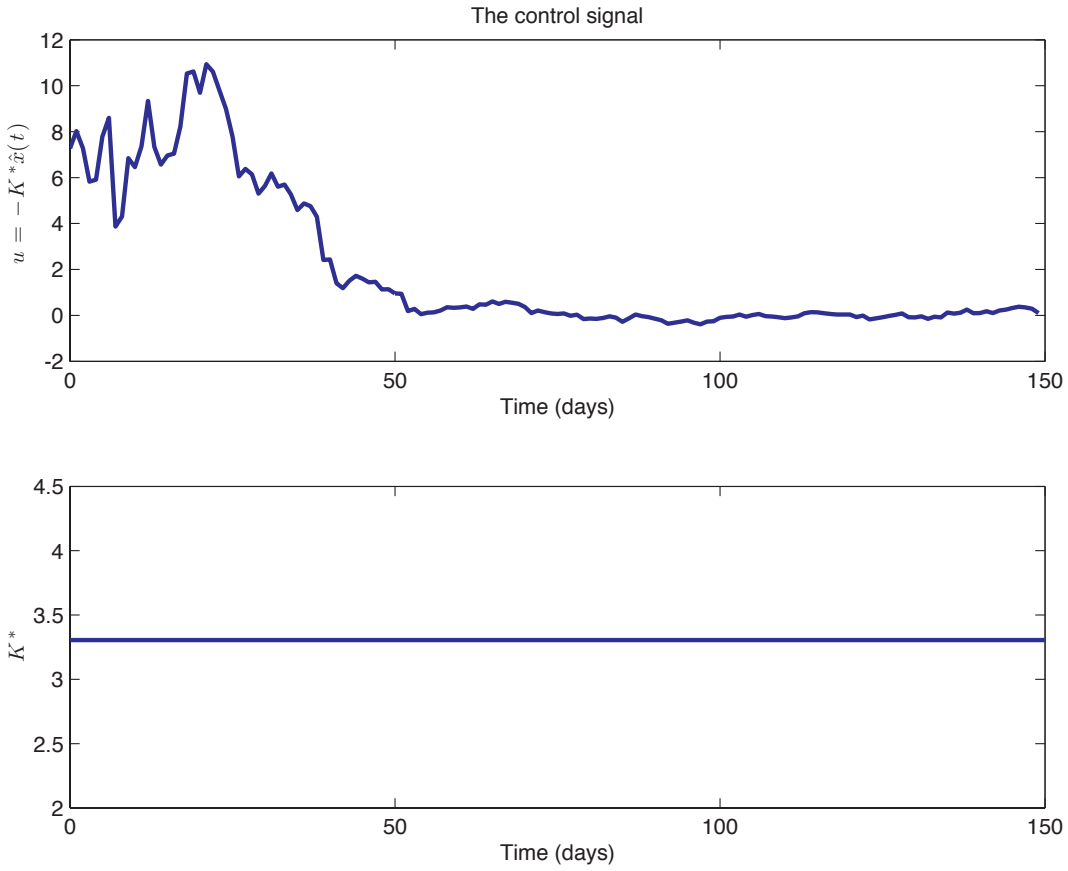


Figure 12.4: The required input signal $u(t) = -K^*\hat{x}(t)$ and the constant feedback gain K^*

Chapter 13

CONCLUDING REMARKS AND FUTURE DIRECTIONS

This dissertation aimed to develop a graph-centric framework for the analysis and synthesis of networked dynamic systems (NDS) consisting of multiple dynamic units that interact via an interconnection topology. Three categories of network problems, namely, identification, controllability, and randomness, are examined. Following we mention the concluding remarks, open problems, and future directions.

13.1 Conclusions

In the first part, we introduced a network identification scheme that involves the excitation and observation of nodes running consensus-type protocols. Starting with the number of vertices in the network as a known parameter as well as the controllability and observability of the resulting steered-and-observed network, the proposed procedure strives to collect pertinent information on the topology of the underlying graph. In this direction, we explored the ramification of grounding the consensus protocol at various vertices for the exact characterization of the network topology from the input-output data. The ramification of the proposed procedure for fault-detection and isolation for networked systems was also explored. We point out that this work has direct implications for a host of consensus-based distributed algorithms used for flocking and distributed estimation. In the second approach, we examined the applications of spectral characterization of graphs as well as a sieve method that is based on integer partitioning algorithms and feasible graphical sequences. In the third approach, however, the spectral characterization of graphs as well as similarity transformation tools are utilized to find the original graph up to an isomorphism compatible with the input-output data.

In the second part, the network controllability is studied. Network controllability aims to provide structural and algebraic insights into features of the network that enable ex-

ternal signal(s) to control the state of the nodes in the network. It is known that the symmetry structure of the graph, as exemplified through its automorphisms or equitable partitions [105], contributes to the uncontrollability of the network. Therefore, a natural question is whether certain classes of networks can be made controllable by breaking their symmetry. As graph symmetry is closely related to the multiplicity of the eigenvalues of their adjacency and Laplacian matrices,[1] it is thus natural to explore classes of graphs for which the relation between controllability and eigenvalue multiplicity can be made more explicit. Study the controllability of circulant networks is an attempt in that direction.

We believe that techniques that have been employed in the dissertation, blending PBH test with the celebrated Cauchy-Binet formula to reduce network controllability to examining non-singularity of submatrices of its (Laplacian) eigenvectors, will prove instrumental in shedding light on the general problem of network controllability over undirected as well as directed graphs. For example, this methodology provides a full characterization of controllability of path graphs.

This dissertation also presents an analysis of the controllability and observability of dynamics over composite networks formed by the graph Cartesian product of its factors. We explored the composition of control matrices of the factor networks to form the control matrix of the composite networks. We provided new insight into the effectiveness of two composite control schemes using graph symmetry and the controllability Gramian.

Motivated by the analysis results for the controllability and observability of deterministic networks, a natural question is whether randomness in the network layer or in the layer of inputs and outputs generically lead to favorable system theoretic properties. In the third part of the dissertation, we examine system theoretic properties of random networks including controllability, observability, performance of optimal feedback controller, and estimator design, and explore some of the ramifications of such an analysis framework. In this direction, the necessary and sufficient conditions for the controllability of random consensus networks generated based on Erdős-Renyi distribution are examined. Based on the controllability condition, the stability properties of the optimal linear quadratic regulator

[1]Higher algebraic multiplicity is often associated with a highly symmetric graph; the correspondance however is not exact for general graphs.

are also discussed.

In the dissertation a decentralized information filter has been introduced. The proposed distributed filter uses the local computational capability of each sensor and uploads the processed measurements to the fusion center. For the sake of energy management as well as unreliable communication links and time delays, a random communication scheme between the sensors and the fusion center is assumed. We then proceeded to employ a Lyapunov method for proving almost sure stability of the proposed random decentralized algorithm based on almost sure contraction property of the corresponding discrete Riccati equation when the system matrix and the noise signals are bounded.

13.2 Future Directions and Open Problems

In this section, I provide insights, comments, open problems, and potential future directions in three domains of network identification, network controllability, and system properties of stochastic networks.

One of the proposed network identification algorithms leveraged on the similarity transformation properties of diffusion type protocols. A question of interest, and yet open problem, is to make explicit statement about the relation between two similar interconnection networks. In other words, we are interested to know whether two similar networks with similarity transformation between their associated Laplacian matrices are isomorphic or just cospectral. The approach toward this problem is either finding a counter example with two similar cospectral networks or proofing the statement that two networks are isomorphic if and only if there is a similarity transformation between the associated matrices. The statement that two Laplacian matrices associated with two isomorphic networks are similar is straightforward to prove.

The next direction of extension is regarding the underlying dynamics. The underlying dynamics in network identification must capture the underlying communication network. On the other hand, our proposed network identification schemes were applied to consensus protocols. One natural extension of the presented approaches is to consider different type of protocols. In the following, one method to extend the network identification setup is suggested and a more general framework as the underlying dynamics.

Consider an LTI system as

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t), \end{aligned}$$

where $x = [x_1, x_2, \ldots, x_n]^T$, the system matrix $A \in \mathbb{R}^{n \times n}$, the input matrix $B \in \mathbb{R}^{n \times r_{\mathcal{I}}}$ with $r_{\mathcal{I}}$ inputs, and the output matrix $C \in \mathbb{R}^{r_{\mathcal{O}} \times n}$ with $r_{\mathcal{O}}$ outputs. The set of states x composes the set of nodes $\mathcal{V} = \{x_1, x_2, \ldots, x_n\}$. The set of links $\mathcal{E}$ consists of the interaction between the states, e.g., there is a weighted directed link with weight A_{ij} from x_i to x_j if $A_{ij} \neq 0$. The described procedure generates a weighted directed network $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. The input signal u_j influences node x_i if $B_{ij} \neq 0$. The observation y_i at time t is the summation of the states of nodes with indices $j \in \{k | \, C_{ik} \neq 0\}$. The purpose of network identification now is to identify the system matrices A, B, and C, or in another word identify the underlying communication network $\mathcal{G}$.

Network identification is a sub-class of inverse problems. Inverse problems have wide range of applications in geophysics, medical imaging, remote sensing, ocean acoustic tomography, among many others. One potential research direction is to investigate real-world problems, which requires identifying an underlying network. Identifying the underlying communication network in biological networks or social networks seems a natural extension of our proposed algorithms. The interpretation of stimulating and observing the behavior of these type of networks, however, require fundamental modifications.

Fault detection is necessary and essential part of a system design. One promising and applied extension of the network identification is to exaamine how the proposed techniques leverage identifying the faulty node and/or links.

In the following, I describe few open problems and provide few potential extensions in the domain of network controllability. Controllability of a single system could sometimes be challenging. In networked systems, the question is more challenging from different perspectives, the dimension of the system is larger and the coupling between the systems adds another layer of complexity. Understanding the effect of coupling between the systems on the controllability of the networked systems is fundamental. Whether symmetry in the communication between agents plays an essential role in the controllability or there are other

features of the networked systems, which provide necessary and sufficient condition for the controllability of the system is critical to be answered.

Roughly, the concept of controllability denotes the ability to move a system from any initial state to any other desired states using only certain admissible manipulations. In some scenarios, the system is required to move around in a limited configuration space. Therefore, modifying the definitions and the requirements of network controllability may relax the problem. Identifying the proper constrains of the controllability or its dual observability and their potential applications are among potential directions of extensions. Aligned with the same line of elongation is to examine different type of weak and strong structural controllability.

In recent years, large scale networks are becoming more ubiquitous in many areas of engineering. Controllability, observability, and security analysis of large scale networks are among the substantial facets that needed to be examined. We investigated the controllability properties of large networks composed of smaller factor networks via graph Cartesian. Future work of particular interest involves extending these results to other types of graph products such as the direct product. Another extension of interest is controllability features of approximated products. Another research direction is exploring potential applications of controllability and observability analysis of large scale networks.

Selecting a subset of nodes to create a static controllable interconnected network is often challenging. One method to overcome this limitation is to switch the input nodes in real-time based on random distributions or deterministic algorithms that minimize a cost function. The analysis in this dissertation showed that randomness in the network layer or in the layer of inputs and outputs generically leads to favorable system theoretic properties. Along this direction, the underlying communication network can change randomly based on some distributions, or the interconnection network can be rewired to minimize a cost or regret function associated to the controllability properties of the system.

As part of the current dissertation, we studied the system properties of stochastic systems including controllability, observability, and performance of optimal feedback controllers and estimators. We explored the ramifications of such an analysis framework in opinion dynamics over social networks and sensor networks in estimating the real-time position of a

Seaglider from experimental data. Providing explicit relation between the distributions of the underlying stochastic networks and the system properties of the networks is yet another interesting direction of extension.

BIBLIOGRAPHY

[1] A. Jadbabaie, J. Lin, and A. S. Morse. Coordination of groups of mobile autonomous agents using nearest neighbor rules, *IEEE Transactions on Automatic Control*, vol. 48, no. 6, pp. 988-1001, 2003.

[2] R. Olfati-Saber, J. A. Fax, and R. M. Murray. Consensus and cooperation in networked multi-agent systems, *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215-233, January 2007.

[3] F. Bullo and J. Cortes and S. Martinez. *Distributed Control of Robotic Networks*, Princeton, 2009.

[4] R. Kuhn and W. Wurst. *Gene Knockout Protocols (Methods in Molecular Biology)*. Humana Press, 2009.

[5] M. J. Tymms and I. Kola. *Gene Knockout Protocols*, Humana Press, 2001.

[6] S. Janson, T. Luczak, and A. Rucinski. *Random Graphs*, John Wiley and Sons, 2000.

[7] C. D. Godsil. *Algebraic Combinatorics*, Chapman and Hall Mathematics, 1993.

[8] R. A. Brualdi. *Introductory Combinatorics*, Prentice-Hall, 1999.

[9] H. Wilf. *Generatingfunctionology,* Academic Press, 1994.

[10] A. Fax and R. M. Murray, Information flow and cooperative control of vehicle formations, *IEEE Transaction on Automatic Control*, vol. 49, no. 9, pp.1465-1476, 2004.

[11] M. Mesbahi. State-dependent graphs and their controllability properties, *IEEE Transaction on Automatic Control*, vol. 50, no. 3, pp. 387-392, March 2005.

[12] R. Olfati-Saber and R. M. Murray. Consensus problems in networks of agents with switching topology and time delays, *IEEE Transaction on Automatic Control*, vol. 49, no. 9, pp. 1520-1533, 2004.

[13] R. E. Kalman. Control of randomly varying linear dynamical systems, *Procceding of Applied Mathematics: American Mathematical Society*, vol. XIII, pp. 287-298, 1962.

[14] Y. Cao and W. Ren. Optimal linear-consensus algorithms: an LQR perspective, *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 40, Issue 3, pp 819-830, 2010.

[15] E. N. Gilbert. Random Graphs, *Annals of Mathematical Statistics*, vol. 30, no. 4, pp. 1141-1144, 1959.

[16] P. Erdős and A. Rényi. On the evolution of random graphs, *Mathematical Institute of the Hungarian Academy of Sciences*, no. 5, pp. 17-61, 1960.

[17] B. Bollobás. *Random Graphs*, Cambridge University Press, 2001.

[18] P. Bougerol. Kalman filtering with random coefficients and contraction, *SIAM Journal of Control and Optimization*, vol. 31, no. 4, pp. 942 - 959, 1993.

[19] M. Mariton. Stochastic controllability of linear systems with Markovian jumps, *Automatica*, vol. 23, issue 6, pp 783-785, 1987.

[20] M. Mariton. *Jump Linear Systems in Automatic Control*, CRC Press, 1990.

[21] M. Bona. *A Walk Through Combinatorics*, World Scientific Publishing, 2002.

[22] P. L. Erdős, I. Miklős, and Z. Toroczkai. A simple Havel-Hakimi type algorithm to realize graphical degree sequences of directed graphs, arXiv:0905.4913v2, 2010.

[23] C. I. D. Genio, H. Kim, Z. Toroczkai, and K. E. Bassler. Efficient and exact sampling of simple graphs with given arbitrary degree sequence, arXiv:1002.2975v1, 2010.

[24] C. Godsil and G. Royle. *Algebraic Graph Theory*, Springer-Verlag, 2001.

[25] M. Fiedler. Algebraic connectivity of graphs, *Czechoslovak Mathematical Journal*, vol. 23 , no. 98, pp. 298-305, 1973.

[26] S. L. Hakimi. On realizability of a set of integers as degrees of the vertices of a linear graph, *Journal of the Society for Industrial and Applied Mathematics*, vol. 10, no. 3, pp. 496–506, 1962.

[27] H. Kim, Z. Toroczkai, P. L. Erdos, I. Miklos, and L. A. Szekely. Degree-based graph construction, arXiv:0905.4892v2, 2009.

[28] L. Ljung. *System Identification*, Prentice-Hall, 1999.

[29] B. Mohar and S Poljak. Eigenvalues in combinatorial optimization, *Combinatorial and Graph-Theoretical Problems in Linear Algebra*, vol. 50, pp. pp. 107-151, 1993.

[30] Y. Teranishi. The Hoffman number of a graph, *Discrete Mathematics*, vol.260, pp. 255–265, 2003.

[31] E. M. Reingold, J. Nievergelt, and N. Deo. *Combinatorial Algorithms: Theory and Practice*, Prentice Hall, 1977.

[32] D. B. West. *Introduction to Graph Theory*, Prentice Hall, 2000.

[33] H. Furstenberg and R. Kesten. Products of random matrices, *The Annals of Mathematical Statistics*, vol. 31, pp. 457-469, 1960.

[34] P. Dorato, C. T. Abdallah, and V. Cerone, Linear Quadratic Control, An Introduction, Princeton-Hall, 2000.

[35] Smart Dust Project Home Page. Univ. California, Berkeley. [On- line] http:// robotics.eecs.berkeley.edu/~pister/SmartDust/.

[36] NEST Project at Berkeley Home Page. Univ. California, Berkeley. [On- line] http: //webs.cs.berkeley.edu/nest-index.html.

[37] Dust Network Inc. Home Page, [On- line] http://www.dustnetworks.com/about/ careers.

[38] A. Mainwaring, D. Culler, J. Polstre, R. Szewczyk, and J. Anderson, Wireless sensor networks for habitat monitoring, *Proceedings of the 1st ACM international workshop on wireless sensor networks and applications-WSNA*, Atlanta, GA, USA, 2002.

[39] R. Cardell-Oliver, M. Kranz, and K. Mayer. A reactive soil moisture sensor network: design and field evaluation, *International Journal of Distributed Sensor Networks*, vol. 1, pp. 149-162, 2005.

[40] B. Sinopoli, L. Schenato, M. Franceschetti, K. Poolla, M. Jordan, and S. Sastry. Kalman filtering with intermittent observations, *IEEE Transaction on Automatic Control*, vol. 49, pp. 1453-1464, 2004.

[41] L. Doherty. *Algorithms for position and data recovery in wireless sensor networks*, Master's thesis, UC Berkeley, 2000.

[42] S. Simic and S. Sastry. Distributed environmental monitoring using random sensor networks, *Proceedings of Workshop on Information Processing in Sensor Networks*, Palo Alto, CA, April 2003

[43] S. Kar, and J. M. F. Moura, Gossip and distributed Kalman filtering: weak consensus under weak detectability, http://arxiv.org/abs/1004.0381v1

[44] S. Kar, B. Sinopoli, and J. M. F. Moura, A random dynamical systems approach to filtering in large-scale networks, http://arxiv.org/abs/0910.0918v1

[45] S. Grime, and H.F. Durraat-Whyte, Data fusion in decentralized sensor networks, *Control Engineering Practice*,vol. 2, no.5, pp. 849-863, 1994.

[46] A. Simonetto, T. Keviczky, and R. Babuska, Distributed nonlinear estimation for robot localization using weighted consensus, *IEEE International Conference on Robotics and Automation*, Anchorage, AK, USA, 2010.

[47] P. Alriksson and A. Rantzer. Model based information fusion in sensor networks, *Proceedings of the 17th IFAC World Congress*, Seoul, Korea, 2008.

[48] R. Carli, A. Chiuso, L. Schenato, and S. Zampieri. Distributed Kalman filtering using consensus strategies, *Proceedings of the 46th Conference on Decision and Control*, New Orleans, CA, USA, 2007.

[49] R. Olfati-Saber. Distributed Kalman filtering for sensor networks, *Proceedings of the 46th IEEE Conference on Decision and Control*, New Orleans, CA, USA, 2007.

[50] R. Olfati-Saber. Kalman-consensus filter: optimality, stability, and performance, *Proceedings of the 48th IEEE Conference on Decision and Control*, Shanghai, China, 2009.

[51] A. Speranzon, C. Fischione, and K. H. Johansson. Distributed and collaborative estimation over wireless sensor networks, *Proceedings of the 45th IEEE Conference on Decision and Control*, San Diego, CA, USA, 2006.

[52] P.S. Maybeck. *Stochastic Models, Estimation and Control, Vol. I.* Academic Press, 1979.

[53] P. Bougerol. Almost sure stabilizability and Riccati equation of linear systems with random parameters, *SIAM Journal on Control and Optimization*, vol. 33, no. 3, pp. 702-717, 1995.

[54] Y. Hatano and M. Mesbahi. Agreement Over Random Networks, *IEEE Transaction on Automatic Control*, vol. 50, no. 11, pp. 1867-1872, 2005.

[55] A.S. Willsky, M.G. Bello, D.A. Castanon, B.C. Levy, and G.C. Verghese. Combining and updating of local estimates and regional maps along sets of one-dimensional tracks, *IEEE Transactions on Automatic Control*, vol. 27, no. 4, 1982.

[56] V. Gupta, N. C. Martins, and J. S. Baras. Stabilization over erasure channels using multiple sensors, *IEEE Transactions on Automatic Automatic Control,* vol. 57, no.7, 2009.

[57] A. Chiuso and L. Schenato. Information fusion strategies and performance bounds in packet-drop networks, *Automatica*, vol. 47, no. 7, pp. 13041316, 2011.

[58] S. Kar and J. Moura. Moderate deviations of the random Riccati equation, http://arxiv.org/pdf/0910.4686v2.

[59] P. Diaconis and D. Freedman. Iterated random functions, *SIAM REVIEW*, vol. 41, no. 1, pp. 45-76, 1999.

[60] S. Milgram. The small-world problems, *Psychology Today*, vol. 2, pp. 60-67, 1967.

[61] D. J. Watts and S. H. Strogatz. Collective dynamics of small world networks, *Nature*, vol. 393, pp. 440-442, 1998.

[62] S. H. Strogatz. Exploring complex networks, *Nature*, vol. 410, pp. 268-276, 2001.

[63] R. Albert and A-L. Barabasi. Statistical mechanics of complex networks, *Review of Modern Physics*, vol. 74, pp. 47-91, 2002.

[64] M. E. J. Newman, D. J. Watts, and S. H. Strogatz. Random graph models of social networks, *Proceeding of the National Academy of sciences*, vol. 99, issue 1, 2002.

[65] S. A. Pandit and R. E. Amritkar. Characterization and control of small-world networks, http://arxiv.org/pdf/chao-dyn/9901017v1.

[66] X. Fan Wang and G. Chen. Complex networks: small-world, scale-free and beyond, *IEEE Circuit and System Magazine*, vol. 3, issue 1, pp. 6-20, 2003

[67] F. Gregoretti, V. Belcastro, D. di Bernardo, and G. Oliva. A parallel implementation of the network identification by multiple regression (NIR) algorithm to reverse-engineer regulatory gene networks, *PLoS ONE* vol. 5, no. 4, 2010.

[68] F. M. Lopes, R. M. Cesar, and L.D. Costa. Gene expression complex networks: synthesis, identification, and analysis, *Journal of Computational Biology*, vol. 18, no.10, pp. 1353-1367, 2011.

[69] J. T. Mettetal, D. Muzzey, C. Gomez-Uribe, and A. van Oudenaarden. The Frequency dependence of Osmo-adaptation in Saccharomyces cerevisiae, *Science*, vol. 319, no. 5862, pp. 482-484, 2008.

[70] D. Muzzey, C. A. Gomez-Uribe, J. T. Mettetal, and A. van Oudenaarden. A systems-level analysis of perfect adaptation in yeast osmoregulation, *Cell* vol. 138, no. 1, pp. 160-171, 2009.

[71] N. E. Friedkin and E. C. Johnsen. Social positions in influence networks, *Social Networks*, vol. 19, no. 3, pp. 209-222, 1997.

[72] N. E. Friedkin and E. C. Johnson. Influence networks and opinion change, *Advances in Group Processes*, vol. 16, no. 1, pp. 1-29, 1999.

[73] R. Hegselmann and U. Krause. Opinion dynamics and bounded confidence models, *Journal of Artificial Societies and Social Simulation (JASSS)*, vol.5, no. 3, 2002.

[74] S. P. Borgatti and P. C. Foster. The network paradigm in organizational research: a review and typology, *Journal of Management*, vol. 29, no. 6, pp. 991-1013, 2003.

[75] T. W. Moore, P. D. Finley, J. M. Linebarger, A. V. Outkin, S. J. Verzi, N. S. Brodsky, D. C. Cannon, A. A. Zagonel, and R. J. Glass. Extending opinion dynamics to model public health problems and analyze public policy interventions, *NECSI ICCS* 2011, in press.

[76] M. Granovetter. The strength of the weak ties, *American journal of sociology*, vol.78, issue 6, pp. 1360-1380, 1973.

[77] J. Goldenberg, Barak Libai, and Eitan Moller. Talk of the network: a complex systems look at the underlying process of word-of-mouth, *Marketing Letters*, vol.12, no.3 , pp. 211-223, 2001.

[78] P. M. DeMarzo, D. Vayanos, and J. Zwiebel. Persuasion bias, social influence, and unidimensional opinions, *The Quarterly Journal of Economics*, vol. 118, no. 3, pp. 909-968, 2003.

[79] F. Amblard and G. Deffuant. The role of network topology on extremism propagation with the relative agreement opinion dynamics, *Physica A: Statistical Mechanics and its Applications*, vol. 343, pp. 725-738, Nov. 2004.

[80] E. Yildiz, D. Acemoglu, A. Ozdaglar, and A. Saberi. Discrete opinion dynamics with stubborn agents, *Computer Engineering*, pp. 1-48, 2011.

[81] D. Acemoglu and A. Ozdaglar. Opinion dynamics and learning in social networks, LIDS report 2851, to appear in the inaugural issue of Dynamic Games and Applications, 2010.

[82] M. Deflem. The concept of social control: theories and applications, *the International Conference on Charities as Instruments of Social Control in Nineteenth-Century Britain,* Rennes, France, 2007.

[83] A. T. C. Y. Kozuki. Modelling the dynamics of opinion formation and propagation: an application to market adoption of transportation services. *M.Sc. Thesis*, University of Maryland, College Park, 2007.

[84] E. A. Ross. Social control, *The American Journal of Sociology*, vol. 1, no. 5, 1896.

[85] E.A. Ross. Social Control. IV. Suggestion, *The American Journal of Sociology*, vol. 2, no. 2, pp. 255-263, 1896.

[86] R. L. Craig. Business, advertising, and the social control of news, *Journal of Communication Inquiry*, vol. 28, no. 233, 2004.

[87] J. F. Padgett. Marriage and Elite structure in Renaissance Florence, 1282-1500, *Paper delivered to the Social Science History Association*, 1994.

[88] C. Qian, J. Cao, J. Lu, and J. Kurths. Adaptive bridge control strategy for opinion evolution on social networks, *Chaos*, vol. 21, no. 2, p. 025116-1:7, 2011.

[89] J. Hartline, V. Mirrokni, and M. Sundararajan. Optimal marketing strategies over social networks, *Proceeding of the 17th international conference on World Wide Web - WWW*, Beijing, China , p. 189, 2008.

[90] P. Dayama, A. Karnik, and Y. Narahari. Optimal mix of incentive strategies for product marketing on social networks, http://arxiv.org/pdf/1203.0135v1.pdf, 2012.

[91] E. Rogers. *Diffusion of Innovations,* 5th Edition. Free Press, 2003.

[92] F. Chung and M. Dadcliffe. On the Spectra of General Random Graphs, http://www.math.ucsd.edu/~fan/wp/randomsp.pdf.

[93] Y. Chen, G. Paul, S. Havlin, F. Liljeros, and H. E. Stanley. Finding a better immunization strategy, *Physical Review Letters*, vol. 101, pp. 058701, 2008.

[94] P. Holme, B. J. Kim, C. N. Yoon, and S. K. Han. Attack vulnerability of complex networks, *Physical Review Letters*, vol. 65, pp. 56109, 2002.

[95] R. Pastor-Satorras and A. Vespignani. Dynamical and correlation properties of the Internet, *Physical Review Letters* vol. 65, pp. 66130, 2002.

[96] R. Cohen, S. Havlin, and D. Ben-Avraham. Efficient immunization strategies for computer networks and populations, *Physical Review Letters*, vol. 91, pp. 247901, 2003.

[97] L. K. Gallos, F. Liljeros, P. Argyrakis, A. Bunde, and S. Havlin. Improving immunization strategies, *Physical Review E Rapid Communications*, vol. 75, pp. 045104, 2007.

[98] J. Q. Lu, D. W. C. Ho, and Z. D. Wang. Pinning stabilization of linearly coupled stochastic neural networks via minimum number of controllers, *IEEE Transaction on Neural Networks* vol.20, no. 10, pp. 1617-1629, 2009.

[99] K. Bergmeister. Road surface monitoring and pavement analysis, *SECAP: Pula, Croatia*, pp. 283-288, 2006.

[100] C. D. Godsil. *Controllable Subsets in Graphs,* http://arxiv.org/abs/1010.3231v1.

[101] A. Rahmani, M. Ji, M. Egerstedt, and M. Mesbahi. Controllability of multi-agent systems from a graph-theoretic perspective, *SIAM Journal on Control and Optimization*, vol. 48, no. 1, pp. 162-186, 2009.

[102] G. Parlangeli and G. Notarstefano. On the reachability and observability of path and cycle graphs, *49th IEEE Conference on Decision and Control*, Atlanta, GA, USA, 2010.

[103] G. Notarstefano and G. Parlangeli. Observability and reachability of grid graphs via reduction and symmetries, *50th IEEE Conference on Decision and Control and European Control Conference*, Orlando, FL, USA, 2011.

[104] S. Zhang, K. Camlibel, and M. Cao. Controllability of diffusively-coupled multi-agent systems with general and distance regular coupling topologies, *50th IEEE Conference on Decision and Control and European Control Conference*, Orlando, FL, USA, 2011.

[105] M. Mesbahi and M. Egerstedt. *Graph Theoretic Methods for Multiagent Networks*, Princeton University Press, 2010.

[106] C. D. Godsil and S. Severini. Control by quantum dynamics on graphs, *Physical Review A*, vol. 81, pp. 052316-1:5, 2010.

[107] C. D. Godsil. Quantum physics and graph spectra, http://quoll.uwaterloo.ca/mine/Talks/bristol.pdf.

[108] M. Klin, V. Liskovets, and R. Poschel. Analytical enumeration of circulant graphs with prime-squared number of vertices, *Seminaire Lotharingien de Combinatoire*, vol. 36, p. B36d, 1996.

[109] V. Liskovets. Some identities for enumerators of circulant graphs, *Journal of Algebraic Combinatorics*, vol. 18, pp. 189-209, 2003.

[110] F. T. Leighton. *Introduction to Parallel Algorithms and Architectures: Arrays, Trees, Hypercubes*, Morgan Kaufmann Publication, 1991.

[111] B. Litow and B. Mans. On isomorphic chordal rings, *Proceedings of the 7th Australasian Workshop on Combinatorial Algorithms*, Sydney, Australia, pp. 108-111, 1996.

[112] B. Litow and B. Mans. On the Adam property for Circulant graphs, *Technical Report of the Department of Computer Science, James Cook University*, 1996.

[113] B. Mans. Optimal distributed algorithms in unlabeled tori and chordal rings, *Journal on Parallel and Distributed Computing*, vol. 46, no. 1, pp. 80-90, 1997.

[114] P. Erdős and D.F. Hsu. Distributed loop networks with minimum transmission delay, *Theoretical Computer Science*, vol. 100, pp. 223-241, 1992.

[115] M.T. Liu. Distributed loop computer networks, *Advanced in Computers*, vol. 17, pp. 163-221, 1978.

[116] S. Georgiou and C. Koukouvinos. Multi-level k-circulant supersaturated designs, *Metrika*, vol. 64, pp. 209-220, 2006.

[117] V. V. Strok. Circulant matrices and the spectra of de Bruijn graphs, *Ukrainian Math. Journal*, vol. 44, pp. 1446-1454, 1992.

[118] Y. K. Wu and R.Z. Jia, and Q. Li. g-Circulant solutions to the $(0, 1)$ matrix equation $A^m = J_n$, *Linear Algebra and its Applications*, vol. 345, pp. 195-224, 2002.

[119] P. J. Davis. *Circulant Matrices*, American Mathematical Society, 1994.

[120] D. S. G. Pollock. Circulant matrices and time-series analysis, *International Journal of Mathematical Education in Science and Technology*, vol. 33, pp. 213-230, 2002.

[121] T. Kailath. *Linear Systems*, Prentice-Hall Information and System Science Series. Prentice Hall, 1979.

[122] R. M. Gray. *Toeplitz and Circulant Matrices: A Review,* Boston: Now Publishers, 2006.

[123] R. A. Horn and C. R. Johnson. *Matrix Analysis*, Cambridge University Press, 1994.

[124] E. D. Sontag. *Mathematical Control Theory: Deterministic Finite Dimensional Systems*, Springer, New York, 1998.

[125] E. R. Heineman. Generalized Vandermonde determinants, *Transactions of the American Mathematical Society*, vol. 31, no. 3, pp. 464-476, 1929.

[126] R. C. King. Generalized Vandermonde determinants and Schur functions, *Proceedings of the American Mathematical Society*, vol. 48, no. 1, pp. 53-56, 1975.

[127] I. G. Macdonald. *Symmetric Functions and Hall Polynomials*, Oxford, 1995.

[128] T. Ernst. Generalized Vandermonde determinants, *U. U. D. M. Report*, 2000.

[129] H. P. Schlickewei and C. Viola. Generalized Vandermonde determinants, *Acta Arithmetica*, XCV.2, pp. 123-137, 2000.

[130] G. Sobczyk. Generalized Vandermonde determinants and applications, *Aportacion Matematicas*, vol. 30, pp. 41-53, 2002.

[131] W. C. Yueh. Eigenvalues of several Tridiagonal matrices, *Applied Mathematics E-Notes*, vol. 5, p. 66-74, 2005.

[132] O. H. Mitchell. Note on determinants of powers, *American Journal of Mathematics*, vol. 4, no. 1, pp. 341-344, 1881.

[133] S. Delvaux and M. Van Barel. Rank-deficient submatrices of Fourier matrices, *Linear Algebra and its Applications*, vol. 429, issue 7, pp. 1587-1605, 2008.

[134] G. Strang. Wavelet transforms versus Fourier transforms, *Bulletin of the American Mathematical Society*, vol. 28, pp. 288-305, 1993.

[135] S. Wang, T. Kuo, and C. Hsu. Trace bounds on the solution of the algebraic matrix Riccati and Lyapunov equation, *IEEE Transactions on Automatic Control*, vol. 31, no. 7, pp. 654-656, 1986.

[136] Z. Gajic and M. T. J. Qureshi. *Lyapunov Matrix Equation in System Stability and Control*, W. F. Ames, Ed. Academic Press, San Diego, CA, 1995.

[137] S. Babaeizadeh, D. Brooks, and D. Isaacson. 3-d electrical impedance tomography for piecewise constant domains with known internal boundaries, *IEEE Transactions on Biomedical Engineering*, vol. 54, no. 1, pp. 2-10, 2007.

[138] O. Marques, T. Drummond, and D. Vasco. A computational strategy for the solution of large linear inverse problems in geophysics, *International Proceedings Parallel and Distributed Processing Symposium*, pp. 22-26, 2003.

[139] J. Mueller, S. Siltanen, and D. Isaacson. A direct reconstruction algorithm for electrical impedance tomography, *IEEE Transactions on Medical Imaging*, vol. 21, no. 6, pp. 555-559, 2002.

[140] E. B. Curtis and J. A. Morrow. *Inverse Problems for Electrical Networks*, World Scientific, 2000.

[141] M. Kac, Can one hear the shape of a drum? *The American Mathematical Monthly*, vol. 73, no. 4, 1966.

[142] B. Gutkin and U. Smilansky. Can one hear the shape of a graph? *Journal of Physics A: Mathematical and General*, vol. 34, pp. 6061-6068, 2001.

[143] A. Fujita, J. R. Sato, H. M. Garay-Malpartida, R. Yamaguchi, S. Miyano, M. C. Sogayar, and C. E. Ferreira. Modeling gene expression regulatory networks with the sparse vector autoregressive model, *BMC Systems Biology*, vol. 1, 2007.

[144] E. August and A. Papachristodoulou, Efficient, sparse biological network determination, *BMC Systems Biology*, vol. 3, no. 1, pp. 1-13, 2009.

[145] D. Yu, M. Righero, and L. Kocarev. Estimating topology of networks, *Physical Review Letters*, vol. 97, p. 188701, 2006.

[146] D. Napoletani and T. Sauer. Reconstructing the topology of sparsely connected dynamical networks, *Physical Review Letter*, vol. 77, p. 026103, 2008.

[147] E. D. Kolaczyk. *Statistical Analysis of Network Data: Methods and Models*, Spring-Verlag, 2009.

[148] C. D. Michener and R. R. Sokal. A quantitative approach to a problem in classification, *Evolution*, vol. 11, no. 2, pp. 130-162, 1957.

[149] R. N. Mantegna and H. E. Stanley. *Introduction to Econophysics: Correlations and Complexity in Finance*, Cambridge University Press, 2000.

[150] Y. Yuan and J. Goncalves. Minimal-time network reconstruction for DTLTI systems, *49th IEEE Conference on Decision and Control*, Atlanta, GA, USA, 2010.

[151] W. E. Moustafa, G. M. Namata, A. Deshpande, and L. Getoor. Declarative analysis of noisy information networks, *ICDE Workshop on Graph Data Management: Techniques and Applications*, 2011.

[152] S. Kok, G. M. Namata and L. Getoor. Collective graph identification, *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2011.

[153] E. T. E. Zheleva and L. Getoor. *Privacy in Social Networks, Synthesis Lectures on Data Mining Series*, Morgan and Claypool Publishers, 2012.

[154] D. Materassi and G. Innocenti. Topological identification in networks of dynamical systems, *IEEE Transactions on Automatic Control*, vol. 55, no. 8, pp. 1860-1871, 2010.

[155] ——. Unveiling the connectivity structure of financial networks via high-frequency analysis, *Physica A: Statistical Mechanics and its Applications*, vol. 388, no. 18, pp. 3866-3878, 2009.

[156] M. Ayazoglu, M. Sznaier, and N. Ozay. Blind identification of sparse dynamic networks and applications, *50th IEEE Conference on Decision and Control*, Orlando, Fl, USA, 2011.

[157] B. M. Sanandaji, T. L. Vincent, and M. B. Wakin. Exact topology identification of large-scale interconnected dynamical systems from compressive observations, *American Control Conference*, San Francisco, CA, USA, 2011.

[158] L. E. Dickson. *History of the theory of the numbers, Volume II, Diophantine Analysis*, Chelsea Publishing Company, 1971.

[159] B. Craven. Complex symmetric matrices, *Journal of the Australian Mathematical Society*, vol. 10, no. 3-4, pp. 341-354, 1969.

[160] N. J. Higham. Computing a nearest symmetric positive semidefinite matrix, *Linear Algebra and its Applications*, vol. 103, no. 6, pp. 103-118, 1988.

[161] ——. *Matrix Nearness Problems and Applications*, Oxford U. Press, 1989.

[162] G. W. Stewart and J. Sun. *Matrix Perturbation Theory*, Academic Press Inc, 1990.

[163] IBM ilog cplex optimization studio, http://www-01.ibm.com/software/integration/optimization/cplex-optimization-studio/.

[164] C.K. Wong and D. Coppersmith. A combinatorial problem related to multimodule memory organizations, *Journal of the ACM*, vol. 21, no.3, pp. 392-402. 1974.

[165] F.T. Boesch and J.F.Wang. Reliable Circulant networks with minimum transmission delay, *IEEE Transactions on Circuits and Systems*, vol.33, no.12, pp.1286-1291. 1985.

[166] .-C. Bermond, F. Comellas, and D.F. Hsu. Distributed loop computer networks: A survey, *Journal of Parallel and Distributed Computing*, vol. 24, pp. 2-10, 1995.

[167] A. Ilic and M. Basic. New results on the energy of integral circulant graphs, http://arxiv.org/abs/1104.4261, 2011.

[168] E. A. Monakhova. Algorithms and lower bounds for p-gossiping in circulant networks, *Third International Symposium on Parallel Architectures, Algorithms, and Networks*, pp.132-137, 1997.

[169] R. S. Wilkov. Analysis and design of reliable computer networks, *IEEE Trans. Communications*, vol. 20, pp. 660-678, 1972.

[170] W. J. Bouknight, S. A. Denenberg, D. E. McIntyre, J.M. Randall, A.H. Sameh, and D.L. Slotnick. The Illiac IV system, *Proc. IEEE*, vol. 60, no. 4, pp. 369-388. 1972.

[171] C. Martinez, R. Beivide, C. Izu, and J. Gutierrez. Distance-hereditary embedding of Circulant graphs, *Proceeding of IEEE International Conference on Information Technology: Coding and Computing (ITCC-2003)*, pp. 320-324. Las Vegas, Nevada, 2003.

[172] R. Beivide, E. Herrada, J. L. Balcazar, and A. Arruabarrena. Optimal distance networks of low degree for parallel computers, *IEEE Transactions on Computers,* vol. C-40, no. 10, pp. 1109-1124, 1991.

[173] P. Kuonen and R. Bruber. Parallel computer architectures for commodity computing and the Swiss-T1 machine, *EPFL Supercomputer Review*, pp.3-11 ,1999.

[174] R. Gruber, A. De Vita, M. Stengel, and T. Minih-Tran. Application dedicated clustering, *EPFL Supercomputer Review*, pp. 37-40, 2002.

[175] Wikipedia contributors, Distributed computing, *Wikipedia, The Free Encyclopedia*, http://en.wikipedia.org/wiki/Distributed_computing.

[176] Airlie Chapman, Eric Schoof, and Mehran Mesbahi. Semi-Autonomous Networks: Theory and Decentralized Protocols, *Proceeding of the IEEE International Conference on Robotics and Automation*, Anchorage, AK, pp.1958-1963, 2010.

[177] A. G. Dimakis, S. Kar, J. M. F. Moura, M. G. Rabbat, and A. Scaglionw. Gossip Algorithms for Distributed Signal Processing, *Proceedings of the IEEE*, vol. 98, issue 11, pp. 1847-1864, 2010.

[178] http://www.ams.jhu.edu/~ers/matgraph/

[179] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, and D. Hwang. Complex networks: Structure and dynamics, *Physics Reports*, vol. 424, no. 4-5, pp. 175–308, 2006.

[180] N. Ganguly, A. Deutsch, and A. Mukherjee. *Dynamics on and of complex networks: applications to biology, computer science, and the social sciences*, Birkhauser, Boston, 2009.

[181] L. Kocarev and G. Vattay. *Complex dynamics in communication networks*, Springer, Berlin, 2005.

[182] A. Chapman and M. Mesbahi. UAV Swarms: Models and Effective Interfaces, in *Handbook of Unmanned Aerial Vehicles (to appear)*, Springer, 2012.

[183] A. Chapman and M. Mesbahi. System theoretic aspects of influenced consensus: Single input case, *IEEE Transactions on Automatic Control*, vol. 57, no. 6, pp. 1505-1511, 2012.

[184] ——. Semi-autonomous consensus: network measures and adaptive trees, *IEEE Transactions on Automatic Control (accepted)*, 2013.

[185] W. Imrich and S. Klavzar. *Product Graphs: Structure and Recognition*, Wiley, 2000.

[186] G. Sabidussi. Graph multiplication, *Mathematische Zeitschrift*, vol. 72, pp. 446-457, 1960.

[187] V. G. Vizing. The Cartesian product of graphs, *Vycisl. Sistemy*, pp. 30-43, 1963.

[188] J. Feigenbaum. Directed graphs have unique Cartesian factorizations that can be found in polynomial time, *Discrete Applied Mathematics*, vol. 15, pp. 105-110, 1986.

[189] M. Hellmuth, W. Imrich, W. Klockl, and P. F. Stadler. Approximate graph products, *European Journal of Combinatorics*, vol. 30, no. 5, pp. 1119-1133, 2009.

[190] A. Chapman and M. Mesbahi, Advection on graphs, *50th IEEE Conference on Decision and Control*, Orlando, FL, pp. 1461-1466, 2011.

[191] ——. Stability analysis of nonlinear networks via M-matrix theory: Beyond linear consensus, *American Control Conference*, Montreal, Canada, 2012.

[192] T. Kaliath. *Linear Systems*, Prentice, 1979.

[193] M. Mesbahi. On the factorization, observability, and identification of the agreement protocol, *16th Mediterranean Conference on Control and Automation*, vol. 1, pp. 1610-1615, 2008.

[194] D. L. Boutin. The determining number of a Cartesian product, *Journal of Graph Theory*, vol. 61, no. 2, pp. 77-87, 2009.

[195] G. Parlangeli and G. Notarstefano. On the reachability and observability of path and cycle graphs, *IEEE Transactions on Automatic Control*, vol. 57, no. 3, pp. 743-748, 2012.

[196] A.-t. Nguyen and M. Mesbahi. A factorization lemma for the agreement dynamics, *46th IEEE Conference on Decision and Control*, no. 1, pp. 288-293, 2007.

[197] J. L. Crassidis and J. L. Junkins. *Optimal Estimation of Dynamic Systems*, CRC Press LLC, 2004.

[198] G. Luenberger. Observers for Multivariable Systems, *IEEE Transaction on Automatic Control*, vol. AC-11, no. 2, pp. 190-197, 1966.

[199] P. Benner. Factorized Solution of Sylvester Equations with Applications in Control, *Proceeding of International Symposium on Mathematics Theory Networks and System (MTNS)*, vol. 1, 2004.

[200] M. Egerstedt, S. Martini, M. Cao, K. Camlibel, and A. Bicchi. Graph-Based Controllability of Networked Systems, *IEEE Control Systems Magazine*, To appear.

[201] M. Egerstedt, S. Martini, M. Cao, K. Camlibel, and A. Bicchi. Interacting with Networks: How Does Structure Relate to Controllability in Single-Leader Consensus Networks?, *IEEE Control Systems Magazine*, To appear.

[202] M. Nabi-Abdolyousefi and M. Mesbahi. On the controllability properties of Circulant networks, *IEEE Transactions on Automatic Control*, Accepted.

[203] Y.-Y. Liu, J.-J. Slotine, A.-L. Barabasi. Controllability of complex networks, *Nature*, vol. 473, 2011.

[204] C. C. Eriksen, T. J. Osse, R. D. Light, T. Wen, T. W. Lehman, P. L. Sabin, J. W. Ballard, and A. M. Chiodi. Seaglider: A long-range autonomous underwater vehicle for oceanographic research, *Journal of Oceanic Engineering*, vol. 26, no. 4, pp. 424-436, 2001.

[205] S. A. Jenkins, D. E. Humphreys, J. Sherman, J. Osse, C. Jones, N. Leonard, J. Graver, R. Bachmayer, T. Clem, P. Carroll, P. Davis, J. Berry, P. Worley, and J. Wasyl. Underwater glider system study, *Technical Report 53, Scripps Institution of Oceanography*, 2003.

[206] L. Techy, K. A. Morgansen, and C. A. Woolsey. Long-Baseline ranging system for acoustic underwater localization of the seaglider underwater glider, *UWAA Technical Report*, 2010.

[207] L. Techy, K. A. Morgansen, and C. A. Woolsey. Long-Baseline ranging for acoustic underwater localization of the seaglider underwater glider, *American Control Conference*, San Francisco, CA, 2011.

[208] J. G. Graver. *Underwater Gliders: Dynamics, Control, and Design*, PhD thesis, Princeton University, 2005.

[209] J. G. Graver, J. Liu, C. A. Woolsey, and N. E. Leonard. Design and analysis of an underwater glider for controlled gliding, *Conference on Information Sciences and Systems*, pp. 801-806, 1998.

[210] N. E. Leonard and J. G. Graver. Model-based feedback control of autonomous underwater gliders, *Journal of Oceanic Engineering*, vol. 26, no. 4, pp. 633-645, 2001.

[211] R. J. Kraus. *Analytical and numerical optimal motion planning for an Underwater Glider*, PhD thesis, Virginia Tech, 2010.

[212] N. Mahmoudian. *Efficient motion planning and control for underwater gliders*, PhD thesis, Virginia Tech, 2009.

[213] P. H. Milne. *Underwater Acoustic Positioning Systems*, Gulf Publishing Company, 1983.

[214] J. Vaganay, J.J. Leonard, and J.G. Bellingham. Outlier rejection for autonomous acoustic navigation, vol. 3, pp. 2174-218, 1996.

[215] J. L. Crassidis and J. L. Junkins. *Optimal Estimation of Dynamic Systems*, Chapman & Hall/CRC, 2004.

[216] M. Nabi-Abdolyousefi, A. Chapman, and Mehran Mesbahi. Controllability and observability of Cartesian product networks, *IEEE Transaction on Automatic Control*, Submitted.

[217] M. Nabi-Abdolyousefi and M. Mesbahi. Network identification via node knock-out, *IEEE Transactions on Automatic Control*, To appear.

[218] M. Nabi-Abdolyousefi and M. Mesbahi. On the Controllability Properties of Circulant Networks, *IEEE Transactions on Automatic Control*, To appear.

[219] M. Nabi-Abdolyousefi and M. Mesbahi. A sieve method for consensus-type network tomography, *IET Control Theory & Applications*, 2012.

[220] A. Chapman, M. Nabi-Abdolyousefi, and M. Mesbahi. Identification and infiltration of consensus-type networks, *1st IFAC Workshop on Estimation and Control of Networked Systems*, pp. 84-89, 2009.

[221] M. Nabi-Abdolyousefi and M. Mesbahi. Network identification via node knock-out, *49th IEEE Conference on Decision*, Atlanta, GA, USA, 2010.

[222] M. Nabi-Abdolyousefi and M. Mesbahi. System Theory over Random Networks: Controllability and Optimality Properties, *50th IEEE Conference on Decision*, Orlando, FL, USA, 2010.

[223] M. Nabi Abdolyousefi and M. Mesbahi. Decentralized estimators over random networks, *American Control Conference*, San Francisco, CA, USA, 2011.

[224] M. Nabi, M. Mesbahi, N. Fathpour, and F. Y. Hadaegh. Local estimators for multiple spacecraft formation flying, *AIAA Guidance and Control*, HI, USA, 2008.

[225] M. Nabi-Abdolyousefi, M. Fazel, and M. Mesbahi. Graph Identification via Transfer Matrices, Similarity Transformations, and Matrix Approximations, *51th IEEE Conference on Decision and Control*, Maui, HI, USA, 2012.

[226] M. Nabi-Abdolyousefi, A. Chapman, and M. Mesbahi. Controllability and Observability of Cartesian Product Networks, *51th IEEE Conference on Decision and Control*, Maui, HI, USA, 2012.