PATTERNED LINEAR SYSTEMS AND CONTROL

by

Connor T. Holmes

A thesis submitted in conformity with the requirements for the degree of Masters of Applied Science
Graduate Department of Electrical and Computer Engineering
University of Toronto

© Copyright 2016 by Connor T. Holmes
Abstract

Patterned Linear Systems and Control

Connor T. Holmes
Masters of Applied Science
Graduate Department of Electrical and Computer Engineering
University of Toronto
2016

A method to characterize the symmetries inherent within physical systems via automorphism groups has already been established. In this thesis, we elaborate on this method and define a special reduced form to which the system matrices of a linear system can be decomposed based on these inherent patterns. We employ this decomposition and the resulting block diagonal form to adapt basic control theory concepts such as controllability and stabilizability to a pattern-preserving framework. We give a pole placement algorithm that synthesizes a feedback matrix that is constrained by the patterns of the system. Moreover, we apply our patterned framework to traditional control problems such as stabilization by measurement feedback and output stabilization, and give pattern-preserving feedback synthesis procedures for these problems. Finally, we provide a set of computational tools that can be used to find the aforementioned decomposition and generate pattern-preserving feedbacks for a given patterned system.
"...Ye Powers
And Spirits of this nethermost abyss,
Chaos and ancient Night, I come no spy
With purpose to explore or to disturb
The secrets of your realm, but, by constraint
Wandering this darksome desert, as my way
Lies through your spacious empire up to light,
Alone and without guide, half lost, I seek
What readiest path leads where your gloomy bounds
Confine with Heaven...”

John Milton, Paradise Lost, ln. 968-977
Acknowledgements

First, I would like to express my gratitude toward Professor Mireille E. Broucke for allowing me the opportunity to pursue this academic endeavor and for all of the direction and constructive criticism that she has given, which has, no doubt, molded this Thesis from a simple collection of ideas into a cohesive mathematical theory.

I am deeply indebted to my parents, Bruce and Deborah Holmes, who have made possible everything that I am today and everything that I will be in the future. Their continual love and support is without bound.

I thank my colleagues from the Systems Control Group for offering stimulating conversations and key insights. I would also like to thank the members of my defense committee Lacra Pavel, Manfredi Maggiore, and Josh Taylor for taking time out of their busy lives to read and assess my Thesis in an in depth way and providing me with interesting and useful feedback.

Finalmente, eu agrego a Jessica Sousa Silva quem me apoiava durante todo este processo duro e quem ouviu as minhas reclamações com ouvidos abertos. O seu sorrizo é a luz da minha vida tanto quanto a sua risada para mim é a canção dos anjos.
# Contents

1 Introduction .................................................. 1
   1.1 Distributed Control Systems ............................ 1
   1.2 Patterned Systems ...................................... 2
   1.3 Contributions .......................................... 5
   1.4 Thesis Outline ......................................... 6

2 Background .................................................... 7
   2.1 Notation and Basic Definitions .......................... 7
   2.2 Linear Geometric Control Theory Concepts .............. 8
      2.2.1 Linear Spaces and Subspaces ....................... 8
      2.2.2 Linear Maps .................................... 9
      2.2.3 Linear Equations ................................ 10
      2.2.4 Projection Maps ................................ 11
      2.2.5 Quotient Spaces .................................. 12
      2.2.6 Invariant Subspaces ............................... 13
      2.2.7 Minimal Polynomial ............................... 14
      2.2.8 Modal Decompositions ............................ 15
      2.2.9 Adjoints ....................................... 17
   2.3 The Kronecker Product ................................... 19
   2.4 Graph Theory ............................................. 22
   2.5 Group Theory ............................................ 23
      2.5.1 Representation of a Finite Group .................. 24
      2.5.2 The Irreducible Decomposition Transformation .... 30

3 Patterned Systems ......................................... 33
   3.1 The Automorphism Group .................................. 35
   3.2 The Extended Permutation Representation ............... 37
   3.3 Commuting Relationships ............................... 38
<table>
<thead>
<tr>
<th>Chapter</th>
<th>Title</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>Patterned Matrices and Subspaces</td>
<td>46</td>
</tr>
<tr>
<td>4.1</td>
<td>Patterned Matrices</td>
<td>46</td>
</tr>
<tr>
<td>4.2</td>
<td>Patterned Decomposition</td>
<td>47</td>
</tr>
<tr>
<td>4.3</td>
<td>Patterned Subspaces</td>
<td>52</td>
</tr>
<tr>
<td>5</td>
<td>Patterned Controllability and Observability</td>
<td>56</td>
</tr>
<tr>
<td>5.1</td>
<td>Controllability</td>
<td>58</td>
</tr>
<tr>
<td>5.2</td>
<td>Stabilizability</td>
<td>60</td>
</tr>
<tr>
<td>5.2.1</td>
<td>Example</td>
<td>61</td>
</tr>
<tr>
<td>5.3</td>
<td>Observability</td>
<td>64</td>
</tr>
<tr>
<td>5.4</td>
<td>Detectability</td>
<td>65</td>
</tr>
<tr>
<td>6</td>
<td>Patterned Stabilization by Measurement Feedback</td>
<td>66</td>
</tr>
<tr>
<td>6.1</td>
<td>Patterned Stabilization by Measurement Feedback Problem</td>
<td>66</td>
</tr>
<tr>
<td>6.2</td>
<td>Example</td>
<td>67</td>
</tr>
<tr>
<td>7</td>
<td>Patterned Output Stabilization Problem</td>
<td>71</td>
</tr>
<tr>
<td>7.1</td>
<td>Patterned Controlled Invariance</td>
<td>71</td>
</tr>
<tr>
<td>7.1.1</td>
<td>Controlled Invariant Patterned Subspaces</td>
<td>71</td>
</tr>
<tr>
<td>7.1.2</td>
<td>Patterned Supremal Subspace Algorithm</td>
<td>73</td>
</tr>
<tr>
<td>7.2</td>
<td>Output Stabilization</td>
<td>74</td>
</tr>
<tr>
<td>7.3</td>
<td>Example</td>
<td>77</td>
</tr>
<tr>
<td>8</td>
<td>Conclusion</td>
<td>83</td>
</tr>
<tr>
<td>A</td>
<td>Simplification of the Computation of the Irreducible Decomposition Matrix</td>
<td>86</td>
</tr>
<tr>
<td>A.1</td>
<td>Kernal Intersection Simplification</td>
<td>86</td>
</tr>
<tr>
<td>A.2</td>
<td>Generator Simplification</td>
<td>86</td>
</tr>
<tr>
<td>A.3</td>
<td>Equal Dimension Subsystem Simplification</td>
<td>87</td>
</tr>
<tr>
<td>B</td>
<td>Pattern Decomposition Code</td>
<td>89</td>
</tr>
<tr>
<td>B.1</td>
<td>Sample Decomposition</td>
<td>95</td>
</tr>
<tr>
<td>C</td>
<td>Sample Code</td>
<td>99</td>
</tr>
<tr>
<td>C.1</td>
<td>Stabilization Example Code</td>
<td>99</td>
</tr>
<tr>
<td>C.2</td>
<td>SMFP Example Code</td>
<td>105</td>
</tr>
<tr>
<td>C.3</td>
<td>OSP Example Code</td>
<td>106</td>
</tr>
</tbody>
</table>

Bibliography | 108 |
## List of Figures

1.1 Examples of Patterned Systems: (a) chains: (i) leader-follower and (ii) symmetric, (b) rings: (i) symmetric and (ii) asymmetric, (c) tree ................................................. 3

3.1 Example 1: Example of a System Graph .................................................. 40
3.2 Example 1: Permutation of System Graph .................................................. 41
3.3 4 Link Chain Graph ................................................................................. 42
3.4 4 Link Chain Flip Automorphism .............................................................. 42
3.5 Two Level Binary Tree Graph ................................................................. 44
3.6 Two Level Binary Tree Automorphism ...................................................... 44

5.1 Stabilization Example System Graph - 7 Subsystem Two Level Tree .......... 61

6.1 SMFP Example System Graph - Five Subsystem Block Circulant Pattern .... 68

7.1 OSP Example System Graph - Five Subsystem Chain Pattern ................. 79
7.2 OSP Example - Output and Non-Output State Trajectories ....................... 82

8.1 Illustration of the interactions between subsystems allowed by pattern feedback ......................... 84
Chapter 1

Introduction

In this thesis we build a framework that allows a formalized geometric analysis of the symmetries that are inherent within a given linear dynamical system. This framework also provides a methodology to synthesize feedback controllers for the system while ensuring that its symmetries are preserved. We refer to these symmetries as patterns, and they are mathematically characterized by a set of permutation matrices that commute with matrices of the state space model of the linear system. It is an important fact that this set of matrices actually form a representation of a finite group. Indeed, we leverage this fact - along with the wealth of theory regarding finite groups and representations - to develop a decomposition that will allow us to perform pattern-preserving control synthesis procedures. More specifically, the group characterizes patterns based on the interconnection structure of the distinct subsystems of a given patterned system. We employ notions from algebraic graph theory to establish a firm connection between the structure of the system and the characterization of its patterns. These concepts have firm roots in the study of distributed control systems. To place the reader in the proper context to fully understand the contribution of this work, we now provide a history of distributed control and a survey of significant developments in patterned linear control theory.

1.1 Distributed Control Systems

The most basic definition of a distributed system is any dynamical system that can be viewed as collection of distinct, interacting subsystems. When we consider a system in such a manner, it may become possible to separate the implementation of control systems into local controllers that deal with specific subsystems, rather than one centralized controller that manages the entire system. Such a control system has been deemed a decentralized control system (DCS). Commercial research and development of the DCS began in the early 1960's with the advent of minicomputers that could be used to regulate processes in an industrial setting. The motivation for research in this field was simple; given the large scale of a system such as a chemical plant or car factory, it was infeasible for a single controller to stabilize the entire
system, both in terms of physical limitations as well as limitations of the computational capacity of the controller.

Such technological restrictions led to strict constraints on the allowable communication between local subsystem controllers. From these constraints was born the concept of decentralized control; a methodology in which the system was first divided into subsystems and, subsequently, a local controller was designed for each subsystem such that the controller only used local outputs and affected only local inputs. Needless to say, the synthesis procedure of such a controller is considerably more involved than that of a centralized system. The 1970’s saw an abundance of theoretical contributions to the study of decentralized control. One of most important and notable of these contributions was an early paper by Wang and Davison that defined the so-called fixed modes of a distributed system [1]. These fixed modes were an adaptation of the uncontrollable, unobservable modes of standard control theory to the decentralized case and arose due to local constraints on allowable feedback. The fixed modes are those that cannot be affected by a decentralized controller and, as such, must be stable if decentralized control is to be used. A early survey of decentralized control concepts is given in [2] and several textbooks are available regarding the decentralized control of large scale systems.

With the current advances in communications technology, the constraints imposed by decentralized control often seems overly conservative. On the other hand, a number of papers - such as those of Šiljak [3] or Corfmat and Morse [4] - attempt to employ graph theory to break the constraints imposed by decentralized control and develop a more sophisticated, distributed control synthesis. The application of graph theory introduces the notion of a structured system, which lends itself to the study of interactions between distinct subsystems. Unfortunately, it is evident that the majority of such studies do not characterize the structure of a system beyond the presence or absence of interaction between its subsystems. Some consideration has been given to more general structures, but these are often more computationally intensive [5, 6].

We now turn our attention to the development and study of structured distributed systems, in which the interaction between subsystems is governed by a specific pattern.

1.2 Patterned Systems

Much of the early research on patterned systems considers systems that consist of identical subsystems that interact with one another in accordance with some prescribed constraints [7]. The specific interconnection structure of the subsystems is often referred to as the pattern. A number of common patterns are shown in Figure 1.1 below, which was taken from [8]:
Chapter 1. Introduction

Figure 1.1: Examples of Patterned Systems: (a) chains: (i) leader-follower and (ii) symmetric, (b) rings: (i) symmetric and (ii) asymmetric, (c) tree

In the above figure, circles represent identical subsystems while the arrows represent interactions between the subsystems. Systems that exhibit the ring pattern shown in Figure 1.1 (b) have been given considerable attention by control theory researchers in the past and have become known as circulant systems. In [9], Brockett and Willems demonstrated that the matrices of the state space models of such systems could be diagonalized by a common base matrix. A variety of researchers have used this useful property to examine control problems for circulant systems [10, 11].

This class of circulant systems was further studied by Hamilton and Broucke from the approach of linear geometric control [12]. This research resulted in a paper that defined a general notion of patterned linear systems which supported methods of geometric control [13]. The crucial concept of this paper was the idea that a class of matrices can be defined in terms of polynomials of a base matrix as follows:

\[ \mathcal{F}(M) = \{ T | (\exists t_0, \cdots, t_{n-1} \in \mathbb{R}) T = t_0 I + t_1 M + t_2 M^2 + \cdots + t_{n-1} M^{n-1} \}, \quad M \in \mathbb{R}^{n \times n}. \]
Furthermore, if it could be shown that the matrices of a given system could be classified using a given base matrix as above, then most of the geometric problems formulated in [12] could be solved using a feedback that is in the same class of matrices. Consequentially, the closed-loop system was also a member of this class; the patterns of the system encoded by the base matrix were preserved even after feedback. However, this paper only considered patterned system with subsystems of dimension one, making the theory difficult to apply to the vast majority of distributed systems (especially large-scale distributed systems).

The methodology of Hamilton and Broucke was further pursued by Sniderman and Broucke, by extending the concept of system patterning to block circulant systems [14]. In this case, subsystems could be extended to have dimension greater than one. In the course of this research, a new definition of patterning arose: the patterning of a matrix could be encoded through commuting properties with other matrices. Namely, a matrix $A$ is an element of a defined set of commuting matrices $\mathcal{C}(U, V)$ if and only if $UA = AV$. Using this notion of patterning, a full set of geometric results was elucidated for block circulant systems, including a pole placement algorithm capable of generating a stabilizing feedback which preserves aspects of patterning in the original system (those encoded by the commuting property). In one notable example, this algorithm was also shown to generate decentralizing results [15].

Pre-dating this development by 25 years, Hazewinkel and Martin approached the characterization of patterned systems from the perspective of abstract algebra [16]. This paper presented powerful theories regarding the commutative property between the matrices of the state model of a patterned system and an infinite set of matrices representing the specific algebra. In particular, a decomposition of the system matrices into block diagonal matrices was given, along with concepts of controllability, stabilization, and decentralization. However, since this characterization relies on infinite sets of matrices, the decomposition presented is not always computationally viable or even possible.

Such limitations do not exist for finite groups. More recently, in a paper by Consolini and Tosques [17], the commutative notion of patterning was extended beyond a single commutative relationship to a set of commutative operations defined in terms of a finite group. A key insight of this paper is the use of the automorphism group of an equivalent graph of the patterned system. This group is recognized as a standard way of representing symmetries inherent in mathematical structures [18].

In the paper, deep results of group theory and representation theory were employed to demonstrate that there exists a stabilizing feedback which preserves the system property of commutativity with the matrix representations of the finite group. However, no explicit pole placement algorithm was given and complete results in terms of standard geometric problems were lacking. Even more recently, Consolini and Tosques have presented results which prove that it is possible to find a decentralized output feedback stabilizer if and only if there exists a controller that respects the symmetry of the graph automorphism group of the system [19].
1.3 Contributions

This thesis consolidates the group theoretic results of Consolini and Tosques [17] with the geometric results of Sniderman et al. [14] to formulate a more general geometric theory of patterned linear systems. More specifically, we adopt the framework pioneered by Consolini and Tosques and employ this framework to develop a special decomposition of the system matrices of a patterned system. This is the most important contribution of this thesis and is elaborated in Section 4.2. The aforementioned decomposition reduces the system matrices into a series of subsystems on which control synthesis problems can be solved directly. We will show that the resulting feedback matrix and closed loop system are guaranteed to be patterned. The following is a detailed list of the full contributions of this document:

- We provide a method by which a given system can be represented as edge labeled digraph called the system graph. We formalize the notion of patterning by linking it directly to the automorphism group of this system graph and provide an algebraic method to encode this patterning. We prove that these matrices in fact commute with the system matrices of the state space model of the system.

- Given a system that is patterned by a group $G$, we provide a key decomposition of the system matrices into a block diagonal reduced form. Moreover, the reduced form blocks can be regarded as subsystems in which we can perform standard control synthesis. The feedbacks of each subsystem are then lifted to the original space, yielding a feedback which preserves the pattern of the system. We also show that the spectrum of eigenvalues of a patterned system takes on a specific form.

- We define a notion of patterning for vector subspaces and develop a set of basic properties of these subspaces.

- We give specific procedures for pole placement and stabilization in the case that the resulting feedback is required to be patterned. We also give conditions under which a system is controllable, stabilizable, observable and detectable using pattern-preserving feedback and observer matrices.

- Two classical control problems are revisited in our framework: stabilization by measurement feedback and output stabilization. We show that if a solution to these problems exists on the reduced subsystems, then a solution exists which preserves the pattern of the system. Moreover, we provide methods to construct such a solution.

- We provide a computationally efficient Pattern Toolbox software for MATLAB which generates the transformations that are required to perform the aforementioned decomposition on a given patterned system. Additional, we demonstrate how to use other software to analyze and extract important parameters of the system graph and the group that characterizes the pattern of the system.
Our contributions in this document are mainly theoretical; however, we endeavor to provide several practical examples as well as comprehensive solutions found via the software mentioned above. It is important to bear in mind that almost all of the patterns mentioned in the sections above are amenable to the framework given in this document (certainly including ring, tree, and chain patterns).

1.4 Thesis Outline

Immediately following this introduction is a chapter on background material. This chapter covers all of the preliminary concepts regarding linear geometric control theory, graph theory, group theory, representation theory, and Kronecker products that will be necessary to understand the remainder of the document. Additionally, we build some concepts regarding vector spaces and Kronecker products. In Chapter 3 we present a methodology by which we can represent a system as a graph and, from this graph, determine a group that characterizes the patterns of the system. In the next chapter, we present our definition of patterned matrices and define a special decomposition of these matrices which can be considered the main contribution of this thesis. We also introduce the notion of a patterned subspace. In Chapter 5, we formally define the patterned system and adapt basic control theory concepts such as pole placement and stabilizability to our patterned framework. In Chapters 6 and 7 we address the stabilization by measurement feedback problem and the output stabilization problem, respectively. In Chapter 8, we give some closing remarks as well as some ideas for future directions of this research area. Finally, in the Appendix we give the Pattern Toolbox that was devised to perform the decompositions mentioned above. We also include some computational simplifications that were employed, as well as some sample code that was used to generate and solve the examples shown throughout this document.
Chapter 2

Background

In this chapter we will introduce and give a brief overview of the mathematical machinery that drives the major theories of patterned linear control. One of the most fundamental concepts that will be introduced is that of a finite group, an abstract mathematical set together with a binary operation on its elements. In our context, the group is used to capture the patterns that a specific system exhibits. In order to perform computations on a group, we use representation theory. Representation theory allows us to represent the elements of a given group by matrices that are consistent with the properties of the group. We will see the automorphism group of the graph of subsystems of a system will serve as both a natural and intuitive way to capture patterns in a distributed control system. Drawing on one of the most fundamental concepts of representation theory - the irreducible representation - we will arrive at a decomposition that will be crucial for the remainder of this thesis.

2.1 Notation and Basic Definitions

If $V$ is a finite set, $|V|$ denotes its cardinality. The set containing only the zero element is denoted with a bold face zero, $\mathbf{0}$. The spectrum of a given square matrix $A \in \mathbb{R}^{n \times n}$ is defined as the set of eigenvalues of the matrix and is denoted $\sigma(A)$. We denote the $n \times n$ identity matrix by $I_n$. A homomorphism is a map that preserves selected structure between two algebraic structures, with the structure to be preserved being given by the naming of the homomorphism. An isomorphism is a homomorphism in which the defined map is bijective (characterized by an invertible linear map in our case). Two mathematical objects are said to be isomorphic (denoted by congruency symbol $\cong$) if there exists a linear invertible map between them. An automorphism is an isomorphism mapping a mathematical object to itself. The general linear group $\text{GL}_n(V)$ is the group of isomorphisms of a given vector space $V$ onto itself. The symmetric group $\text{Sym}(n)$ is the group of all permutations of a set of cardinality $n$ [20]. Let $A \otimes B$ denote the Kronecker product between two matrices $A$ and $B$ of any dimension. The direct sum of matrices is
Chapter 2. Background

2.2 Linear Geometric Control Theory Concepts

In this section we define all of the basic concepts from linear geometric control that will be useful to us in the remainder of this document. This section was taken from lecture notes written by Mireille E. Broucke [22] and serves as a review for those reader who are already well versed in linear geometric concepts. Some proofs are omitted.

2.2.1 Linear Spaces and Subspaces

A linear space (or vector space) $X$ over the field $\mathbb{R}$ of reals is a set of elements (called vectors) with two operations: addition of vectors and scalar multiplication. We typically denote linear spaces by script symbols such as $X$, $V$, and so forth. For vectors $x_1, \ldots, x_n$ in $X$, span$\{x_1, \ldots, x_n\}$ denotes the linear span of the vectors, i.e., $\{\sum_{i=1}^{n} c_i x_i : c_i \in \mathbb{R}\}$. We say $X$ is finite-dimensional if there exist vectors $x_1, \ldots, x_n$ such that $X = \text{span}\{x_1, \ldots, x_n\}$. The least such $n$ is the dimension of $X$, denoted $\text{dim}(X)$.

A set of vectors $\{x_1, \ldots, x_n\}$ is linearly independent if

$$(\forall c_i) \quad c_1 x_1 + \cdots + c_n x_n = 0 \implies c_1 = \cdots = c_n = 0.$$ 

A set $\{x_1, \ldots, x_n\}$ is a basis for $X$ if

$$X = \text{span} \{x_1, \ldots, x_n\} \text{ and } \{x_1, \ldots, x_n\} \text{ is linearly independent.}$$ 

Then every $x$ in $X$ can be expressed as $x = c_1 x_1 + \cdots + c_n x_n$, where the coefficients are unique.

**Definition 2.1.** A (non-empty) subset $V$ of $X$ is a subspace, and we write $V \subset X$, if $V$ is closed under addition, i.e.,

$$x, y \in V \implies x + y \in V,$$

and closed under scalar multiplication, i.e.,

$$x \in V, \ c \in \mathbb{R} \implies cx \in V.$$ 

The subspace $\{0\} \subset X$ is sometimes denoted $0$ (the zero subspace).

Let $V, W$ be subspaces of $X$. Then $V \cap W$ and $V + W := \{v + w : v \in V, w \in W\}$ are also subspaces of $X$. $V + W$ is the smallest subspace containing both $V$ and $W$. Similarly $V \cap W$ is the largest subspace.
Chapter 2. Background

The family of all subspaces of $\mathcal{X}$ is partially ordered by subspace inclusion $\subseteq$, and under the operations of $+$ and $\cap$ is easily seen to form a lattice. Note that the set union $\mathcal{V} \cup \mathcal{W}$ is not a subspace in general. Let $\mathcal{V}, \mathcal{W}, \mathcal{R} \subseteq \mathcal{X}$, and suppose $\mathcal{V} \subseteq \mathcal{R}$. Then

$$\mathcal{R} \cap (\mathcal{V} + \mathcal{W}) = (\mathcal{R} \cap \mathcal{V}) + (\mathcal{R} \cap \mathcal{W}) = \mathcal{V} + (\mathcal{R} \cap \mathcal{W}). \tag{2.1}$$

This equation is called the modular distributive rule.

Two subspaces $\mathcal{V}, \mathcal{W}$ are independent if $\mathcal{V} \cap \mathcal{W} = 0$. This is not the same as being orthogonal. More generally, subspaces $\mathcal{V}_1, \ldots, \mathcal{V}_k$ are independent if

$$\mathcal{V}_i \cap \left( \sum_{j \neq i} \mathcal{V}_j \right) = 0 \text{ for every } i.$$

The following three conditions are equivalent:

1. $\mathcal{V}_1, \ldots, \mathcal{V}_k$ are independent.
2. $(\forall v \in \mathcal{V}_1 + \cdots + \mathcal{V}_k) \ (\exists \text{ unique } v_i \in \mathcal{V}_i) \ v = v_1 + \cdots + v_k$.
3. $\dim(\mathcal{V}_1 + \cdots + \mathcal{V}_k) = \dim(\mathcal{V}_1) + \cdots + \dim(\mathcal{V}_k)$.

If $\mathcal{V}, \mathcal{W}$ are independent subspaces, their sum is called the direct sum and is denoted $\mathcal{V} \oplus \mathcal{W}$. The direct sum of more than two subspaces is defined likewise.

Let $\mathcal{V}$ be a subspace of $\mathcal{X}$. A basis $\{v_1, \ldots, v_n\}$ of $\mathcal{X}$ for which $\{v_1, \ldots, v_k\}$ is a basis of $\mathcal{V}$ is called a basis of $\mathcal{X}$ adapted to $\mathcal{V}$. Every subspace has an independent complement, i.e.,

$$\mathcal{V} \subseteq \mathcal{X} \implies (\exists \mathcal{W} \subseteq \mathcal{X}) \ \mathcal{X} = \mathcal{V} \oplus \mathcal{W}.$$

A complement can be constructed using any basis $\{v_1, \ldots, v_n\}$ of $\mathcal{X}$ adapted to $\mathcal{V}$. Namely, span$\{v_{k+1}, \ldots, v_n\}$ is a complement of $\mathcal{V}$. Obviously, the complement is not unique. In Section 2.2.5 we discuss the quotient space which uniquely captures the notion of “$\mathcal{X}$ minus $\mathcal{V}$”.

2.2.2 Linear Maps

Let $\mathcal{X}, \mathcal{Y}$ be two vector spaces. A function $\mathbf{A} : \mathcal{X} \to \mathcal{Y}$ is a linear map iff

$$\mathbf{A}(x_1 + x_2) = \mathbf{A}x_1 + \mathbf{A}x_2, \ x_1, x_2 \in \mathcal{X}$$

$$\mathbf{A}(ax) = a\mathbf{A}x, \ a \in \mathbb{R}, \ x \in \mathcal{X}.$$

A linear map is uniquely determined by its action on a basis. That is, if $\mathbf{A} : \mathcal{X} \to \mathcal{Y}$ is a linear map and if $\{x_1, \ldots, x_n\}$ is a basis for $\mathcal{X}$, then if we know the vectors $\mathbf{A}x_i$, we can compute $\mathbf{A}x$ for every $x \in \mathcal{X}$,
by linearity. Every linear map on finite-dimensional vector spaces has a \textit{matrix representation}. Here we
distinguish a linear map from its matrix representation by unbolding it.\footnote{Eventually this convention will be dropped except when a linear map does not have a matrix representation.} Let $A$ be a linear map $\mathcal{X} \to \mathcal{Y}$, with $\{x_1, \ldots, x_n\}$ a basis for $\mathcal{X}$ and $\{y_1, \ldots, y_p\}$ a basis for $\mathcal{Y}$. A matrix representation for $A$ is obtained as follows: (i) Take the $i$th basis vector $x_i$ of $\mathcal{X}$. (ii) Apply the map $A$ to $x_i$ to get $Ax_i$. (iii) Express $Ax_i$ in the basis for $\mathcal{Y}$. The corresponding coefficients form the $i$th column of $A$.

A linear map induces several special subspaces. Let $A : \mathcal{X} \to \mathcal{Y}$ be a linear map. The kernel (or nullspace) of $A$ is the subspace of $\mathcal{X}$

$$\text{Ker } A := \{ x : Ax = 0 \}.$$

If $V \subset \mathcal{Y}$, the \textit{preimage} of $V$ under $A$ is the subspace of $\mathcal{X}$

$$A^{-1}V := \{ x : Ax \in V \}.$$

The \textit{image} (or range space) of $A$ is the subspace of $\mathcal{Y}$

$$\text{Im } (A) := \{ y : (\exists x \in \mathcal{X}) y = Ax \}.$$

More generally, if $V \subset \mathcal{X}$, the \textit{image} of $V$ under $A$ is

$$A V := \{ y : (\exists x \in V) y = Ax \}.$$

A linear map $A$ is \textit{one-to-one} if $v_1 \neq v_2$ implies $Av_1 \neq Av_2$. A linear map $A$ is \textit{onto} if for every $y \in \mathcal{Y}$ there exists an $x \in \mathcal{X}$ such that $Ax = y$. Clearly $A$ is onto if $\text{Im } (A) = \mathcal{Y}$. Similarly, $A$ is one-to-one iff $\text{Ker } (A) = 0$. Note that $A$ is onto if and only if it has full row rank. $A$ is one-to-one if and only if it has full column rank. The dimension of $\text{Im } (A)$ is called the rank of the matrix. If $\dim(\mathcal{X}) = n$ then

$$\dim \text{Im } (A) + \dim \text{Ker } (A) = n.$$

\section{Linear Equations}

Let $A : \mathcal{X} \to \mathcal{Y}$ and let $I$ be the identity map of $\mathcal{Y}$. Consider the linear equation $AX = I$. A solution $X$ is called a \textit{right inverse} of $A$. A matrix or linear map $A$ has a right inverse iff it is onto. If $A$ is onto, it has a unique right inverse if and only if it is one-to-one.

Now consider the linear equation $XA = I$ where $I$ is the identity map of $\mathcal{X}$. A solution $X$ is called a \textit{left inverse} of $A$. $A$ has a left inverse iff it is one-to-one. If $A$ is one-to-one, it has a unique left inverse iff it is onto.
More generally, the linear equation \[ AX = B \]
has a solution \( X \) if and only if \( \text{Im} \ B \subset \text{Im} \ A \). If \( X \) exists, then it is unique if and only if \( A \) is one-to-one.

The linear equation \[XA = B\]
has a solution \( X \) if and only if \( \text{Ker} \ A \subset \text{Ker} \ B \). If \( X \) exists, then it is unique if and only if \( A \) is onto.

### 2.2.4 Projection Maps

We make use of several special linear maps. Let \( V \subset X \) with \( \dim(V) = k \). Since \( V \) can be regarded as a \( k \)-dimensional vector space in its own right, a vector \( x \in V \) can either be viewed as an element of \( V \) or as an element of the ambient space \( X \). We define the insertion map \( S : V \to X \) to be the linear map which maps a vector \( x \in V \) to the corresponding element \( x \in X \). That is, \( Sx := x \). In coordinates, it maps the \( k \times 1 \) coordinate vector of \( x \) in a basis for \( V \) to the corresponding \( n \times 1 \) coordinate vector for \( x \) in a basis for \( X \). The insertion map is represented by any matrix whose column vectors form a basis for \( V \) relative to a given basis for \( X \). For example, if \( \{v_1, \ldots, v_n\}\) is a basis for \( X \) adapted to \( V \), then \( S \) has a matrix representation

\[
S = \begin{bmatrix} I_k \\ 0 \end{bmatrix}.
\]

Alternatively, let \( \{e_1, \ldots, e_n\}\) be a basis of \( X \) (say the standard unit basis) and let \( \{v_1, \ldots, v_n\}\) be another basis of \( X \) adapted to \( V \). Since each \( v_i \in X \), it can be expressed in the basis \( \{e_1, \ldots, e_n\}\) as \( v_i = v_{i1}e_1 + \cdots + v_{in}e_n \) or as the coordinate vector \( v_i = (v_{i1}, \ldots, v_{in}) \). Then the representation of \( S \) with respect to these bases is

\[
S = \begin{bmatrix} v_{11} & \cdots & v_{1k} \\ \vdots & \ddots & \vdots \\ v_{n1} & \cdots & v_{nk} \end{bmatrix} = \begin{bmatrix} | & | \\ v_1 & \cdots & v_k \end{bmatrix}.
\]

Clearly, \( S \) is one-to-one, \( \text{rank} \ S = k \), and \( V = \text{Im} \ S \subset X \).

Let \( X = V \oplus W \). Then each \( x \in X \) has a unique representation \( x = v + w \) with \( v \in V \), \( w \in W \). The function \( Q : X \to V \) defined as \( x \mapsto v \) is called the natural projection on \( V \) along \( W \). If \( \{v_1, \ldots, v_n\} \) is a basis for \( X \) adapted to \( V \), then \( Q \) has a matrix representation

\[
Q = \begin{bmatrix} I_k \\ 0 \end{bmatrix}.
\]

One can verify that

\[
V = \text{Im} \ Q, \quad W = \text{Ker} \ Q.
\]
Moreover, $Q$ is a left inverse of $S$, i.e. $QS = I_V$, where $I_V$ is the identity map of $V$.

### 2.2.5 Quotient Spaces

Let $V \subset X$ with $\dim(V) = k$. Subspace $V$ has many complements, but there is a vector space that uniquely captures the notion of “$X$ minus $V$”. It is called the quotient space, denoted $X/V$, and is constructed as follows. Let $x \in X$. Define the coset of $x$,

$$\bar{x} := x + V = \{x + v \mid v \in V\}.$$ 

Thus $\bar{x}$ is a subset of $X$. For example, if $X = \mathbb{R}^2$ and $V$ is geometrically a line through 0, then $\bar{x}$ is the line through $x$ parallel to $V$. The set $X/V$ is defined to be the set of all cosets, i.e.,

$$X/V := \{\bar{x} \mid x \in X\}.$$ 

It is made into a linear space by defining operations of vector sum and scalar multiplication

$$\bar{x}_1 + \bar{x}_2 := x_1 + x_2, \quad c\bar{x} := cx.$$ 

Note that the zero element $\bar{0}$ of $X/V$ is $V$.

One can construct a basis of $X/V$ as follows. Let $\{v_1, \ldots, v_n\}$ be a basis of $X$ adapted to $V$. Then $v_1 = \cdots = v_k = \bar{0}$. Hence if $\bar{x}$ is an arbitrary element of $X/V$, then

$$\bar{x} = c_{k+1}\bar{v}_{k+1} + \cdots + c_n\bar{v}_n.$$ 

It follows that every element of $X/V$ can be written as a linear combination of $\{\bar{v}_{k+1}, \ldots, \bar{v}_n\}$. Moreover, one can show that $\{\bar{v}_{k+1}, \ldots, \bar{v}_n\}$ are linearly independent and therefore form a basis of $X/V$. Thus we arrive at the following.

**Lemma 2.1.** $\dim(X/V) = \dim(X) - \dim(V)$.

The canonical projection $P : X \to X/V$ is the function mapping $x$ to $\bar{x}$. If $\{v_1, \ldots, v_n\}$ is a basis of $X$ adapted to $V$ and $\{\bar{v}_{k+1}, \ldots, \bar{v}_n\}$ is a basis of $X/V$, then $P$ has a matrix representation

$$P = \begin{bmatrix} 0 & I_{n-k} \end{bmatrix}.$$ 

One can verify that $P$ is onto and

$$X/V = \text{Im } P, \quad V = \text{Ker } P.$$
Now let \( \{v_1, \ldots, v_n\} \) be a basis of \( \mathcal{X} \) adapted to \( \mathcal{V} \) and \( \{v_{k+1}, \ldots, v_n\} \) a basis of \( \mathcal{X}/\mathcal{V} \). Define a function that performs the mapping \( v_i \mapsto v_i, \quad i = k + 1, \ldots, n \). This uniquely determines a linear map \( R : \mathcal{X}/\mathcal{V} \to \mathcal{X} \). Moreover, \( R \) is a right inverse of \( P \); that is, \( PR = I_{\mathcal{X}/\mathcal{V}} \), where \( I_{\mathcal{X}/\mathcal{V}} \) is the identity map of \( \mathcal{X}/\mathcal{V} \). In the given bases \( R \) has a matrix representation

\[
R = \begin{bmatrix} 0 \\ I_{n-k} \end{bmatrix}.
\]

Alternatively, let \( \{e_1, \ldots, e_n\} \) be a basis of \( \mathcal{X} \) (say the standard unit basis) and let \( \{v_1, \ldots, v_n\} \) be another basis of \( \mathcal{X} \) adapted to \( \mathcal{V} \). Then the representation of \( R \) with respect to the bases \( \mathcal{X} = \text{span}\{e_1, \ldots, e_n\} \) and \( \mathcal{X}/\mathcal{V} = \text{span}\{v_{k+1}, \ldots, v_n\} \) is

\[
R = \begin{bmatrix} | & | & | \\ v_{k+1} & \cdots & v_n \end{bmatrix}.
\]

If \( \mathcal{W} = \text{span}\{v_{k+1}, \ldots, v_n\} \) and \( Q : \mathcal{X} \to \mathcal{V} \) is the natural projection on \( \mathcal{V} \) along \( \mathcal{W} \), then \( QR = 0 \).

### 2.2.6 Invariant Subspaces

If \( A : \mathcal{X} \to \mathcal{X} \) is a linear map, a subspace \( \mathcal{V} \subset \mathcal{X} \) is \textit{A-invariant} if \( A\mathcal{V} \subset \mathcal{V} \). An \( A \)-invariant subspace \( \mathcal{V} \) induces two linear maps of interest. First, let \( S : \mathcal{V} \to \mathcal{X} \) be the insertion map. The map \( A|_\mathcal{V} : \mathcal{V} \to \mathcal{V} \) called the \textit{restriction} of \( A \) to \( \mathcal{V} \) has the action of \( A \) on \( \mathcal{V} \) (but with codomain \( \mathcal{V} \)) and it is not defined off \( \mathcal{V} \). It is the unique solution of

\[
ASA = SA|_\mathcal{V}.
\]

That is, it satisfies the commutative diagram

\[
\begin{array}{ccc}
\mathcal{X} & \overset{A}{\longrightarrow} & \mathcal{X} \\
\downarrow{S} & & \downarrow{S} \\
\mathcal{V} & \overset{A|_\mathcal{V}}{\longrightarrow} & \mathcal{V}
\end{array}
\]

Second, an \( A \)-invariant subspace \( \mathcal{V} \subset \mathcal{X} \) induces a linear map in the quotient space \( \mathcal{X}/\mathcal{V} \). Let \( P : \mathcal{X} \to \mathcal{X}/\mathcal{V} \) be the canonical projection. The map \( A|_{\mathcal{X}/\mathcal{V}} : \mathcal{X}/\mathcal{V} \to \mathcal{X}/\mathcal{V} \), called the \textit{induced map} in \( \mathcal{X}/\mathcal{V} \) of \( A \), performs the action \( \pi \mapsto \overline{Ax} \). It is easily shown this is a linear map and it is well-defined; that is, it is independent of the choice of representative of \( \pi \). It is the unique solution of

\[
PA = A|_{\mathcal{X}/\mathcal{V}}P.
\]
That is, it satisfies the commutative diagram

\[
\begin{array}{ccc}
X & \xrightarrow{A} & X \\
\downarrow P & & \downarrow P \\
X/V & \xrightarrow{A_{X/V}} & X/V
\end{array}
\]

**Theorem 2.1** (Representation Theorem). Let \( A : X \to X \) be a linear map and \( V \subset X \) such that \( AV \subset V \). Then \( A \) has a matrix representation

\[
A = \begin{bmatrix}
A_1 & * \\
0 & A_2
\end{bmatrix},
\]

where \( A_1 = A_V \) and \( A_2 = A_{X/V} \). Moreover

\[
\sigma(A) = \sigma(A_1) \cup \sigma(A_2).
\]

### 2.2.7 Minimal Polynomial

Let \( A : X \to X \) be a linear map and let \( x \in X \). Consider the sequence of vectors

\[
x, Ax, A^2x, \ldots
\]

There exists an integer \( k \) such that the vectors \( \{x, Ax, \ldots, A^{k-1}x\} \) are linearly independent, while \( A^kx \) can be expressed as a linear combination of the previous vectors, i.e.

\[
A^kx = -\alpha_1x - \alpha_2Ax - \cdots - \alpha_{k-1}A^{k-1}x.
\]

Define the monic polynomial

\[
\psi(s) = s^k + \alpha_ks^{k-1} + \cdots + \alpha_2s + \alpha_1.
\]

Then we can write

\[
\psi(A)x = 0.
\]  \( \text{(2.2)} \)

Every polynomial for which (2.2) holds is called an *annihilating polynomial* of \( x \) with respect to linear map \( A \). The annihilating polynomial constructed above is the one of least degree and is called the *minimal polynomial* of \( x \). It is a fact that that the minimal polynomial of \( x \) divides every annihilating polynomial of \( x \) and that every vector in \( X \) has a minimal polynomial.

Now we can choose a basis \( \{v_1, \ldots, v_n\} \) for \( X \) and let \( \psi_i(s) \) be the minimal polynomial of \( v_i \) with
Chapter 2. Background

15

respect to \( A \). Let \( \psi(s) \) be the least common multiple (l.c.m.) of \( \psi_1(s), \ldots, \psi_n(s) \). Then \( \psi(s) \) is an annihilating polynomial for each \( v_i \). Since every \( x \in \mathcal{X} \) is a linear combination of basis vectors, we see that 
\[ \psi(A)x = c_1\psi(A)v_1 + \cdots + c_n\psi(A)v_n = 0. \]
Thus, \( \psi(s) \) is called an annihilating polynomial of \( \mathcal{X} \).

It can be shown that it is the one of least degree and it is independent of the basis vectors chosen for \( \mathcal{X} \). Therefore, it is called the minimal polynomial of \( \mathcal{X} \) with respect to \( A \) (or more simply the minimal polynomial of \( A \)) and we have that 
\[ \mathcal{X} = \ker \psi(A). \]

The characteristic polynomial of \( A \) is defined as \( \pi(s) = \det(sI - A) \), and it is well-known from the Cayley-Hamilton theorem that \( \pi(A) = 0 \). Thus \( \pi(s) \) is an annihilating polynomial of \( \mathcal{X} \).

Characteristic and minimal polynomials and more generally annihilating polynomials are useful tools for decomposing a space into invariant subspaces with respect to a map \( A \). This depends on several facts about polynomials.

We say two polynomials \( \psi_1 \) and \( \psi_2 \) are coprime if they do not have any common factor. An equivalent condition is that there exist polynomials \( p \) and \( q \) such that 
\[ p(s)\psi_1(s) + q(s)\psi_2(s) = 1. \]

**Lemma 2.2.** Let \( A : \mathcal{X} \to \mathcal{X} \). For any polynomial \( g(s) \), \( \ker g(A) \) is an \( A \)-invariant subspace.

**Theorem 2.2.** Let \( A : \mathcal{X} \to \mathcal{X} \) and let \( g(s) \) be any annihilating polynomial of \( \mathcal{X} \), i.e. \( g(A)x = 0 \) for all \( x \in \mathcal{X} \). Suppose that \( g \) can be factored as 
\[ g(s) = g_1(s)g_2(s), \]
where \( g_1 \) and \( g_2 \) are monic, coprime polynomials. Define \( V_1 = \ker g_1(A) \) and \( V_2 = \ker g_2(A) \). Then the following hold:

1. \( V_1 = \text{Im} \ g_2(A), V_2 = \text{Im} \ g_1(A), \)
2. \( \mathcal{X} = V_1 \oplus V_2, \)
3. \( V_1 \) and \( V_2 \) are \( A \)-invariant,
4. If \( g \) is the minimal (characteristic) polynomial of \( A \), then \( g_1 \) and \( g_2 \) are the minimal (characteristic) polynomials of \( A_{V_1} \) and \( A_{V_2} \), respectively.

2.2.8 Modal Decompositions

Let \( A : \mathcal{X} \to \mathcal{X} \) be a linear map and let \( \psi(s) \) be its minimal polynomial. Suppose we factor \( \psi \) as 
\[ \psi(s) = \psi_1(s) \cdots \psi_p(s), \]
where the \( \psi_i \) are pairwise coprime. Define 
\[ \mathcal{X}_i(A) := \ker \psi_i(A). \]
Because the $\psi_i$ are coprime and by repeatedly applying Theorem 2.2 one gets that

$$\mathcal{X} = \mathcal{X}_1(A) \oplus \cdots \oplus \mathcal{X}_p(A)$$

and $AX_i(A) \subset \mathcal{X}_i(A)$. A decomposition of $\mathcal{X}$ of this form which corresponds to a partition of $\sigma(A)$ into disjoint subsets of $\mathbb{C}$ is called a modal decomposition of $\mathcal{X}$ and the $\mathcal{X}_i(A)$ are called modal subspaces.

In the sequel we are particularly interested in the case when the minimal polynomial is factored as $\psi(s) = \psi^+(s)\psi^-(s)$, where $\psi^+(s)$ contains all the factors in $\mathbb{C}^+$, the closed right-half complex plane, and $\psi^-(s)$ contains all the factors in $\mathbb{C}^-$, the open left-half complex plane. Then $\mathcal{X}$ is decomposed as

$$\mathcal{X} = \mathcal{X}^+(A) \oplus \mathcal{X}^-(A)$$

where

$$\mathcal{X}^+(A) := \ker \psi^+(A), \quad \mathcal{X}^-(A) := \ker \psi^-(A)$$

are called the unstable and stable subspaces of $A$, respectively. More generally, we can consider the case when $\mathcal{X}$ is decomposed as

$$\mathcal{X} = \mathcal{X}^m(A) \oplus \mathcal{X}^c(A)$$

where $\mathcal{X}^m(A)$ is a modal subspace and $\mathcal{X}^c(A)$ is its unique complement. We assume that corresponding to this decomposition is a disjoint partition of $\mathbb{C} = \mathbb{C}^m \cup \mathbb{C}^c$ with $\mathbb{C}^m \cap \mathbb{C}^c = \emptyset$. The minimal polynomial of $\mathcal{X}$ with respect to $A$ factors as

$$\psi(s) = \psi^m(s)\psi^c(s)$$

with $\mathcal{X}^m(A) = \ker \psi^m(A)$ and $\mathcal{X}^c(A) = \ker \psi^c(A)$.

Now let $\mathcal{V} \subset \mathcal{X}$ be such that $AX \subset \mathcal{V}$ and let $P : \mathcal{X} \to \mathcal{X}/\mathcal{V}$ be the canonical projection. Let $A_2 = AX/\mathcal{V}$, the induced map. If $\mathcal{X}/\mathcal{V}$ has a minimal polynomial $\psi_2$ with respect to $A_2$, then it can also be factored as $\psi_2(s) = \psi_2^m(s)\psi_2^c(s)$ where $\psi_2^m$ divides $\psi^m$ and $\psi_2^c$ divides $\psi^c$. Since $\psi_2^m$ and $\psi_2^c$ are coprime we have

$$\mathcal{X}/\mathcal{V} = (\mathcal{X}/\mathcal{V})^m(A_2) \oplus (\mathcal{X}/\mathcal{V})^c(A_2)$$

where $(\mathcal{X}/\mathcal{V})^m(A_2) = \ker \psi_2^m(A_2) \subset \mathcal{X}/\mathcal{V}$ and $(\mathcal{X}/\mathcal{V})^c(A_2) = \ker \psi_2^c(A_2) \subset \mathcal{X}/\mathcal{V}$.

**Lemma 2.3.** Let $\mathcal{V} \subset \mathcal{X}$ such that $AX \subset \mathcal{V}$ and let $P : \mathcal{X} \to \mathcal{X}/\mathcal{V}$. Then the following hold:

(i) $P \mathcal{X}^m(A) = (\mathcal{X}/\mathcal{V})^m(A_{\mathcal{X}/\mathcal{V}})$.

(ii) $\mathcal{X}^m(A) \subset \mathcal{V}$ if and only if $\sigma(AX/\mathcal{V}) \subset \mathbb{C}^c$.

(iii) $\mathcal{X}^m(A) \cap \mathcal{V} = 0$ if and only if $\sigma(AX) \subset \mathbb{C}^c$.

Now suppose we have $\mathcal{V}, \mathcal{W} \subset \mathcal{X}$ such that $\mathcal{V} \oplus \mathcal{W} = \mathcal{X}$ and $AX \subset \mathcal{V}$. Let $S_1 : \mathcal{V} \to \mathcal{X}$ and $S_2 : \mathcal{W} \to \mathcal{X}$
be the insertion maps and let \( Q_1 : \mathcal{X} \to \mathcal{V} \) and \( Q_2 : \mathcal{X} \to \mathcal{W} \) be the natural projections. Also define \( A_1 := Q_1 AS_1 \) and \( A_2 := Q_2 AS_2 \). From (??), we know that \( A_1 = A_\mathcal{V} \). Let \( \psi_1 = \psi_1^+ \psi_1^- \) be the minimal polynomial of \( \mathcal{V} \) with respect to \( A_1 \), and let \( \psi_2 = \psi_2^+ \psi_2^- \) be the minimal polynomial of \( \mathcal{W} \) with respect to \( A_2 \). It can be shown that \( \psi_1 \) divides \( \psi_1 \), the minimal polynomial of \( \mathcal{X} \) with respect to \( A_1 \). Define the unstable subspaces

\[
\mathcal{V}^+ (A_1) := \text{Ker} \psi_1^+ (A_1) \quad \mathcal{W}^+ (A_2) := \text{Ker} \psi_2^+ (A_2).
\]

These modal subspaces are related to the unstable subspace of \( A \).

**Lemma 2.4.** Let \( \mathcal{V}, \mathcal{W} \subset \mathcal{X} \) such that \( \mathcal{V} \oplus \mathcal{W} = \mathcal{X} \) and \( A \mathcal{V} \subset \mathcal{V} \). Let \( S_1 : \mathcal{V} \to \mathcal{X} \) and \( S_2 : \mathcal{W} \to \mathcal{X} \) be the insertion maps and let \( Q_1 : \mathcal{X} \to \mathcal{V} \) and \( Q_2 : \mathcal{X} \to \mathcal{W} \) be the natural projections. Also define \( A_1 := Q_1 AS_1 \) and \( A_2 := Q_2 AS_2 \). Then

(i) \( \mathcal{V}^+ (A_1) = \mathcal{X}^+ (A) \cap \mathcal{V} \).

(ii) \( \mathcal{W}^+ (A_2) = Q_2 \mathcal{X}^+ (A) \).

**Stable and Unstable Modal Subspaces**

Let \( \psi(\lambda) \) be the minimal polynomial of \( A \) and factor it as

\[
\psi(\lambda) = \psi^+(\lambda)\psi^-(\lambda),
\]

where the zeros of \( \psi^+ \) and \( \psi^- \) belong to \( \mathbb{C}^+ \) and \( \mathbb{C}^- \), respectively. Since \( \psi^+ \) and \( \psi^- \) are coprime we have

\[
\mathcal{X} = \mathcal{X}^+ (A) \oplus \mathcal{X}^- (A)
\]

where \( \mathcal{X}^+ (A) = \text{Ker}(\psi)^+ (A) \) and \( \mathcal{X}^- (A) = \text{Ker}(\psi)^- (A) \) are the unstable and stable modal subspaces of \( A \), respectively.

**2.2.9 Adjoint**

\( \mathbb{R}^n \) is a vector space with an inner product \( \langle x, y \rangle_{\mathbb{R}^n} := x^T y \), where \( x^T \) is the transpose of \( x \) and \( x, y \in \mathbb{R}^n \).

More generally, a vector space \( \mathcal{X} \) equipped with an inner product is called an inner product space. Let \( \mathcal{X} \) and \( \mathcal{Y} \) be inner product spaces with inner products \( \langle \cdot, \cdot \rangle_{\mathcal{X}} \) and \( \langle \cdot, \cdot \rangle_{\mathcal{Y}} \), respectively and let \( A : \mathcal{X} \to \mathcal{Y} \) be a linear map. The adjoint of \( A \) is defined as the linear map \( A^* : \mathcal{Y} \to \mathcal{X} \) which satisfies:

\[
\langle y, Ax \rangle_{\mathcal{Y}} = \langle A^* y, x \rangle_{\mathcal{X}}, \quad \forall x \in \mathcal{X}, \quad \forall y \in \mathcal{Y}.
\]
One can show that the adjoint of a linear map always exists and is unique. We denote by $\| \cdot \|_X$ and $\| \cdot \|_Y$ the associated norms on $X$ and $Y$, respectively.

Consider the following example. Let $\mathcal{L}^2([0,t_1])$ denote the space of square integrable, vector-valued functions defined on $[0,t_1]$, i.e., the set of functions $u(\cdot) : [0,t_1] \to \mathbb{R}^k$ with $\int_0^{t_1} \|u(t)\|^2 dt < \infty$. The inner product $\langle \cdot , \cdot \rangle_2$ in $\mathcal{L}^2([0,t_1])$ is defined by

$$\langle u_1 , u_2 \rangle_2 := \int_0^{t_1} u_1^T(t)u_2(t) \, dt . \quad (2.3)$$

Let $F(t)$ be an $n \times k$ real matrix defined for $t \in [0,t_1]$ and satisfying $\int_0^{t_1} \|F(t)\|^2 dt < \infty$. Define the linear map $L : \mathcal{L}^2([0,t_1]) \to \mathbb{R}^n$ by:

$$L(u) = \int_0^{t_1} F(t)u(t) \, dt .$$

One can find the adjoint $L^* : \mathbb{R}^n \to \mathcal{L}^2([0,t_1])$ of $L$ as follows. We have

$$\langle Lu , x \rangle_{\mathbb{R}^n} = \left[ \int_0^{t_1} F(t)u(t) \, dt \right]^T x = \int_0^{t_1} u^T(t)F^T(t)x \, dt = \int_0^{t_1} u^T(t)(L^*x)(t) \, dt = \langle u , L^*x \rangle_2 .$$

Hence

$$\left( L^*x \right)(t) = F^T(t)x , \quad x \in \mathbb{R}^n , \quad t \in [0,t_1] . \quad (2.4)$$

Note that $LL^*$ is a map from $\mathbb{R}^n$ to $\mathbb{R}^n$ so it has a matrix representation given by

$$LL^* = \int_0^{t_1} F(t)F^T(t) \, dt .$$

Let $\mathcal{X}$ be an inner product space with inner product $\langle \cdot , \cdot \rangle$. We say that two vectors $x, y \in \mathcal{X}$ are orthogonal, and we use the notation $x \perp y$, if $\langle x , y \rangle = 0$. Let $\mathcal{V} \subset \mathcal{X}$ be a subspace. The orthogonal complement $\mathcal{V}^\perp$ (called “$\mathcal{V}$ perp”) of $\mathcal{V} \subset \mathcal{X}$ is

$$\mathcal{V}^\perp := \left\{ x \in \mathcal{X} \mid \langle x , v \rangle = 0 , \forall v \in \mathcal{V} \right\} .$$

It is clear that $\mathcal{V}^\perp$ is a subspace of $\mathcal{X}$. Moreover, if $\mathcal{V}$ is finite dimensional, then $\mathcal{X} = \mathcal{V} \oplus \mathcal{V}^\perp$ and $\mathcal{V} = (\mathcal{V}^\perp)^\perp$.

**Theorem 2.3.** Let $\mathcal{X}$ and $\mathcal{Y}$ be inner product spaces such that either $\mathcal{X}$ or $\mathcal{Y}$ is finite dimensional. Let $A : \mathcal{X} \to \mathcal{Y}$ be a linear map.

(i) $\ker A^* = (\text{im } A)^\perp$, \quad $\ker A = (\text{im } A^*)^\perp$.

(ii) $\ker AA^* = \ker A^*$, \quad $\ker A^*A = \ker A$. 

(iii) \( Y = \text{Im} \ A \perp \text{Ker} \ A^* \), \quad \mathcal{X} = \text{Im} \ A^* \perp \text{Ker} \ A.

(iv) \( \text{Im} \ (A) = \text{Im} \ (AA^*) \), \quad \text{Im} \ A^* = \text{Im} \ A^* A.

### 2.3 The Kronecker Product

In this section we define the Kronecker product and highlight a number of its properties that will be imperative to the understanding of the rest of this thesis. The Kronecker product is defined in terms of real matrices, but is extendable to other fields.

**Definition 2.2 (Kronecker Product).** Given matrices \( A \in \mathbb{R}^{m \times n} \) and \( B \in \mathbb{R}^{p \times q} \), the **Kronecker product** of the two matrices, denoted as \( A \otimes B \in \mathbb{R}^{mp \times nq} \), is:

\[
A \otimes B = \begin{bmatrix}
a_{11}B & \ldots & a_{1n}B \\
\vdots & \ddots & \vdots \\
am_{m1}B & \ldots & a_{mn}B
\end{bmatrix},
\]

where \( a_{ij} \) is the element of \( A \) at the \( i \)-th row and \( j \)-th column.

**Lemma 2.5.** \([23, \text{p. 243-246}]\) Given real matrices \( A, B, C \) and \( D \) and \( c \in \mathbb{R} \), the following properties hold:

1. \( A \otimes (B + C) = A \otimes B + A \otimes C \)
2. \( (A + B) \otimes C = A \otimes C + B \otimes C \)
3. \( (cA) \otimes B = A \otimes (cB) = c(A \otimes B) \)
4. \( (A \otimes B) \otimes C = A \otimes (B \otimes C) \)
5. \( (A \otimes B)(C \otimes D) = (AC) \otimes (BD) \)
6. \( (A \otimes B)^{-1} = A^{-1} \otimes B^{-1} \)
7. \( (A \otimes B)^T = A^T \otimes B^T \)

The following lemma regards the Jordan normal form of Kronecker products.

**Lemma 2.6.** Given matrices \( A \) and \( B \) with Jordan normal form \( J \) and \( K \) respectively, the Jordan normal form of \( A \otimes B \) is the same as the Jordan Normal form of \( J \otimes K \).

Although not given explicitly as a lemma, this lemma is proven directly in \([23, \text{p. 261}]\). We now give some of the deeper properties of the Kronecker product that will be useful for us.
**Definition 2.3** (Perfect Shuffle Permutation Matrix). The perfect shuffle permutation matrix is defined as follows,

\[ S_{m,n} = \sum_{i=1}^{m} \sum_{j=1}^{n} E_{ij} \otimes E_{ij}^T, \]

where \( E_{ij} \in \mathbb{R}^{m \times n} \) is such that \( E_{ij} \) has an entry 1 in position \( i, j \), and all other entries are zero.

With this definition in hand, we may give an equation which reverses the order of a Kronecker product.

**Lemma 2.7.** [23, p. 260] Given \( A \in \mathbb{R}^{m \times n} \) and \( B \in \mathbb{R}^{p \times q} \), we can reverse the order of their Kronecker product as follows,

\[ B \otimes A = S_{m,p}^T (A \otimes B) S_{n,q}. \]

The following lemma demonstrates how eigenvalues are affected by the Kronecker product.

**Lemma 2.8.** [23, p. 245] Suppose we have two square matrices \( A \in \mathbb{R}^{n \times n} \) and \( B \in \mathbb{R}^{m \times m} \) and suppose that the corresponding spectrum of each matrix is given by \( \sigma(A) = \{ \lambda_1, ..., \lambda_n \} \) and \( \sigma(B) = \{ \mu_1, ..., \mu_m \} \) respectively. Then the eigenvalues of the Kronecker product \( A \otimes B \) are given by \( \sigma(A \otimes B) = \lambda_i \mu_j \) for \( i = 1, ..., n \) and \( j = 1, ..., m \).

The next lemma will be crucial for the remainder of the section regarding Kronecker products and is generally used in the process of solving certain matrix equations.

**Lemma 2.9.** [23, p. 254] Suppose we have matrices \( A \in \mathbb{R}^{m \times n} \), \( B \in \mathbb{R}^{p \times q} \) and \( X \in \mathbb{R}^{n \times p} \). Then,

\[ (B^T \otimes A) \text{vec}(X) = \text{vec}(AXB). \]

The remainder of the results given in this section are not available in the literature and are therefore proven here.

**Lemma 2.10.** Suppose \( X \in \mathbb{R}^{n \times m} \) and \( M \in \mathbb{R}^{m \times p} \). Then \( \text{vec}(X) \in \text{Im}(M \otimes I_n) \) if and only if \( \text{Im}(X^T) \subseteq \text{Im}(M) \).

**Proof.** ( \( \implies \) ) Suppose \( \text{vec}(X) \in \text{Im}(M \otimes I_n) \). Then there exists a \( Y \in \mathbb{R}^{n \times p} \) such that \( \text{vec}(X) = (M \otimes I_n)\text{vec}(Y) \). Then by Lemma 2.9, \( \text{vec}(X) = \text{vec}(I_nYM^T) = \text{vec}(YM^T) \), Thus \( X = YM^T \) or, equivalently, \( X^T = MY^T \), which implies that \( \text{Im}(X^T) \subseteq \text{Im}(M) \).

( \( \impliedby \) ) Suppose \( \text{Im}(X^T) \subseteq \text{Im}(M) \). Then there exists \( Y \in \mathbb{R}^{n \times p} \) such that \( X^T = MY^T \), or equivalently, \( X = YM^T \). Then \( \text{vec}(X) = \text{vec}(YM^T) = \text{vec}(I_nYM^T) \) and by Lemma 2.9, \( \text{vec}(X) = (M \otimes I_n)\text{vec}(Y) \). Therefore, \( \text{vec}(X) \in \text{Im}(M \otimes I_n) \).

The following lemma is similar, but is expressed for kernel spaces.
Lemma 2.11. Suppose $X \in \mathbb{R}^{n \times m}$ and $M \in \mathbb{R}^{p \times m}$. Then $\text{vec}(X) \in \text{Ker}(M \otimes I_n)$ if and only if $\text{Im}(X^T) \subset \text{Ker}(M)$.

Proof. ($\implies$) Suppose $\text{vec}(X) \in \text{Ker}(M \otimes I_n)$. Then $(M \otimes I_n)\text{vec}(X) = 0$. By Lemma 2.9, we have $\text{vec}(I_nXM^T) = 0$ which implies that $MX^T = 0$. Thus $\text{Im}(X^T) \subset \text{Ker}(M)$.

($\impliedby$) Conversely, suppose $\text{Im}(X^T) \subset \text{Ker}(M)$. Then $MX^T = 0$, so $\text{vec}(I_nXM^T) = 0$. By Lemma 2.9, $(M \otimes I_n)\text{vec}(X) = 0$ and therefore $\text{vec}(X) \in \text{Ker}(M \otimes I_n)$. \hfill $\square$

We now state a theorem regarding a collection of rules about image and kernel spaces of matrices that have been augmented via the Kronecker product with an identity matrix. This theorem will be crucial for the proofs of the succeeding sections.

Theorem 2.4. Suppose $\hat{A} \in \mathbb{R}^{m \times p}$ and $\hat{B} \in \mathbb{R}^{q \times r}$. Let $A = \hat{A} \otimes I_n$ and $B = \hat{B} \otimes I_n$. Then the following statements hold.

1. $\text{Ker}(A) \subset \text{Ker}(B) \iff \text{Ker}(\hat{A}) \subset \text{Ker}(\hat{B})$.

2. $\text{Im}(A) \subset \text{Im}(B) \iff \text{Im}(\hat{A}) \subset \text{Im}(\hat{B})$.

3. $\text{Im}(A) \subset \text{Ker}(B) \iff \text{Im}(\hat{A}) \subset \text{Ker}(\hat{B})$.

4. $\text{Ker}(A) \subset \text{Im}(B) \iff \text{Ker}(\hat{A}) \subset \text{Im}(\hat{B})$.

Proof. We divide the proof as in the theorem; however, the proof for each part is essentially the same.

1. ($\implies$) Suppose $\text{Ker}(A) \subset \text{Ker}(B)$. Let $x \in \text{Ker}(\hat{A})$ and form the matrix $X = \begin{bmatrix} x & \cdots & x \end{bmatrix} \in \mathbb{R}^{p \times n}$ such that $x$ repeats $n$ times. Then $\text{Im}(X) \subset \text{Ker}(\hat{A})$ and by Lemma 2.11, this implies that $\text{vec}(X^T) \in \text{Ker}(A)$. By assumption $\text{vec}(X^T) \in \text{Ker}(B) = \text{Ker}(\hat{B} \otimes I_n)$ and by Lemma 2.11 we have that $\text{Im}(X) \subset \text{Ker}(\hat{B})$. This implies that $x \in \text{Ker}(\hat{B})$ as required.

( $\impliedby$) Suppose $\text{Ker}(\hat{A}) \subset \text{Ker}(\hat{B})$. Let $x \in \text{Ker}(A) = \text{Ker}(\hat{A} \otimes I_n)$. Let $X \in \mathbb{R}^{n \times p}$ such that $x = \text{vec}(X)$. Equivalently, $x = \text{vec}(X) \in \text{Ker}(\hat{A} \otimes I_n)$ so, by Lemma 2.11, $\text{Im}(X^T) \subset \text{Ker}(\hat{A})$. By assumption, $\text{Im}(X^T) \subset \text{Ker}(\hat{B})$. By Lemma 2.11 again, $x = \text{vec}(X) \in \text{Ker}(B)$ as required.

2. ($\implies$) Suppose $\text{Im}(A) \subset \text{Im}(B)$. Let $x \in \text{Im}(\hat{A})$ and form the matrix $X = \begin{bmatrix} x & \cdots & x \end{bmatrix} \in \mathbb{R}^{p \times n}$ such that $x$ repeats $n$ times. Then $\text{Im}(X) \subset \text{Im}(\hat{A})$ and by Lemma 2.10, this implies that $\text{vec}(X^T) \in \text{Im}(A)$. Then, by assumption, $\text{vec}(X^T) \in \text{Im}(B) = \text{Im}(\hat{B} \otimes I_n)$, and by Lemma 2.10, we have that $\text{Im}(X) \subset \text{Im}(\hat{B})$. Thus $x \in \text{Im}(\hat{B})$ as required.

( $\impliedby$) Suppose $\text{Im}(\hat{A}) \subset \text{Im}(\hat{B})$. Let $x \in \text{Im}(A) = \text{Im}(\hat{A} \otimes I)$. Then there exists matrix $X \in \mathbb{R}^{m \times n}$ such that $x = \text{vec}(X)$. Equivalently, $x = \text{vec}(X) \in \text{Im}(\hat{A} \otimes I_n)$ and, by Lemma 2.10, we have that $\text{Im}(X^T) \subset \text{Im}(\hat{A})$. By assumption, $\text{Im}(X^T) \subset \text{Im}(\hat{B})$. By Lemma 2.10, $x = \text{vec}(X) \in \text{Im}(\hat{B} \otimes I_n) = \text{Im}(B)$ as required.
3. ( \( \implies \) ) Suppose \( \text{Im}(A) \subseteq \text{Ker}(B) \). Let \( x \in \text{Im}(\hat{A}) \) and we form the matrix \( X = \begin{bmatrix} 1 & \cdots & 1 \end{bmatrix} \in \mathbb{R}^{p \times n} \) such that \( x \) repeats \( n \) times. Then \( \text{Im}(X) \subseteq \text{Im}(\hat{A}) \) and by Lemma 2.10, this implies that \( \text{vec}(X^T) \in \text{Im}(\hat{A} \otimes I_n) = \text{Im}(A) \). Then, by assumption, \( \text{vec}(X^T) \in \text{Ker}(B) = \text{Ker}(\hat{B} \otimes I_n) \), and by Lemma 2.11, we have that \( \text{Im}(X) \subseteq \text{Ker}(\hat{B}) \). Therefore \( x \in \text{Ker}(\hat{B}) \) as required.

( \( \Leftarrow \) ) Suppose \( \text{Im}(\hat{A}) \subseteq \text{Ker}(\hat{B}) \). Let \( x \in \text{Im}(A) = \text{Im}(\hat{A} \otimes I_n) \) and let \( X \in \mathbb{R}^{m \times n} \) such that \( x = \text{vec}(X) \). So \( x = \text{vec}(X) \in \text{Im}(\hat{A} \otimes I_n) \) and, by Lemma 2.10, we have that \( \text{Im}(X^T) \subseteq \text{Im}(\hat{A}) \). By assumption, \( \text{Im}(X^T) \subseteq \text{Ker}(\hat{B}) \). Now, by Lemma 2.11, \( x = \text{vec}(X) \in \text{Ker}(B) \) as required.

4. ( \( \implies \) ) Suppose \( \text{Ker}(A) \subseteq \text{Im}(B) \). Let \( x \in \text{Ker}(\hat{A}) \) and form the matrix \( X = \begin{bmatrix} 1 & \cdots & 1 \end{bmatrix} \in \mathbb{R}^{p \times n} \) such that \( x \) repeats \( n \) times. Then \( \text{Im}(X) \subseteq \text{Ker}(\hat{A}) \) and by Lemma 2.11, this implies that \( \text{vec}(X^T) \in \text{Ker}(A) \). Then, by assumption, \( \text{vec}(X^T) \in \text{Im}(B) \), and by Lemma 2.10, we have that \( \text{Im}(X) \subseteq \text{Ker}(\hat{B}) \). Therefore \( x \in \text{Ker}(\hat{B}) \) as required.

( \( \Leftarrow \) ) Suppose \( \text{Ker}(\hat{A}) \subseteq \text{Im}(\hat{B}) \). Let \( x \in \text{Ker}(A) = \text{Ker}(\hat{A} \otimes I_n) \). Then there exists matrix \( X \in \mathbb{R}^{n \times p} \) such that \( x = \text{vec}(X) \). Then \( x = \text{vec}(X) \in \text{Ker}(\hat{A} \otimes I_n) \) and, by Lemma 2.11, we have that \( \text{Im}(X^T) \subseteq \text{Ker}(\hat{A}) \). By assumption, \( \text{Im}(X^T) \subseteq \text{Im}(\hat{B}) \). Now, by Lemma 2.10, \( x = \text{vec}(X) \in \text{Im}(B) \) as required.

\( \square \)

### 2.4 Graph Theory

In this section we introduce some elementary notions of graph theory that will be useful to us later when trying to identify the patterns of a control system. We restrict our interest to graphs with labeled edges that can have two directions, i.e., edge labeled digraphs.

**Definition 2.4 (Edge Labeled Digraph).** An edge labeled digraph is defined as the 3-tuple \( G = (V, E, L) \). The tuple consists of three ordered sets defined as follows.

- \( V \) is the set of vertices of the graph, defined as \( V = \{v_1, \ldots, v_h\} \).
- \( E \) is the set of edges that link the vertices, defined as \( E = \{e_{ij} = (v_j, v_i) : e_{ij} \in (V \times V)\} \). Note that if edge \( e_{ij} = (v_j, v_i) \) is defined, there exists an edge directed from vertex \( v_j \) to vertex \( v_i \).
- \( L \) is the labeling function mapping each edge to an index. Let \( \mathcal{I}_L \) denote a set of indices (to be defined later). Then \( L : E \rightarrow \mathcal{I}_L \) such that \( L(e_{ij}) = l_{ij} \in \mathcal{I}_L \).

We define an isomorphism of a graph as follows:

**Definition 2.5 (Graph Isomorphism).** [24, pg. 3] Let \( G = (V, E) \) and \( G' = (V', E') \) be two graphs. We call \( G \) and \( G' \) isomorphic if there exists a bijection \( \phi : V \rightarrow V' \) with \( (v_j, v_i) \in E \) if and only if \( (\phi(v_j), \phi(v_i)) \in E' \) for all \( v_i, v_j \in V \). Such a map is called an isomorphism of the graph \( G \).
Remark 2.1. The definition given above is for non-labeled graphs, which are more common in the literature. We also assume that the labeling function of two isomorphic edge labeled graphs is the same. In our context, this implies that if \( e_{ij} = (v_j, v_i) \in E \) implies that \( e'_{ij} = (\phi(v_j), \phi(v_i)) \in E' \), then \( L(e_{ij}) = L'(e'_{ij}) \).

We define the standard adjacency matrix of a graph.

**Definition 2.6** (Adjacency Matrix). Given a graph \( G \), we define the adjacency matrix to be the matrix \( M \in \mathbb{R}^{k \times k} \), where \( k = |V| \) such that:

\[
(M)_{ij} = \begin{cases} 
1 & \text{if } (v_j, v_i) \in E \\
0 & \text{otherwise}
\end{cases}
\]

We can extend this standard definition of the adjacency matrix to include an encoding of the edge labels of a given graph.

**Definition 2.7** (Labeled Adjacency Matrix). Given a graph \( G \), we define the labeled adjacency matrix to be the matrix of indices \( M_L \in \mathbb{I}^{k \times k} \), where \( k = |V| \), such that:

\[
(M_L)_{ij} = \begin{cases} 
L(e_{ij}) & \text{if } (v_j, v_i) \in E \\
0 & \text{otherwise}
\end{cases}
\]

We have the following lemma regarding the adjacency matrices of isomorphic graphs.

**Lemma 2.12.** [20, Lemma 8.1.1, pg. 164] Let \( G = (V, E) \) and \( G' = (V', E') \) be two directed graphs with the same number of vertices. Then they are isomorphic if and only if there is a permutation matrix \( P \) such that \( P^{-1}MP = M' \) where \( M \) and \( M' \) are the adjacency matrices of \( G = (V, E) \) and \( G' = (V', E') \), respectively.

Remark 2.2. This property also holds for the labeled adjacency matrices \( M_L \) and \( M'_L \) of two isomorphic edge labeled graphs \( G = (V, E, L) \) and \( G' = (V', E', L') \), respectively, as long as they have the same labeling function. This is affirmed in [25], in which it is stated that two labeled graphs are isomorphic if and only if there exists a bijective function that maps between its vertex sets, such that the adjacent edges and labels are preserved. Thus, we have \( P^{-1}M_LP = M'_L \) in the case where the graphs are isomorphic, with \( P \) being the bijective map.

### 2.5 Group Theory

A group is an algebraic structure consisting of a set of elements \( G \) together with an operation "." that combines any two elements of the set. A group satisfies the following axioms:

- Closure: \( a, b \in G \implies a \cdot b = c \in G \)
Chapter 2. Background

• Associativity: \( a, b, c \in G \implies a \cdot (b \cdot c) = (a \cdot b) \cdot c \)

• Identity: \( \exists 1 \in G, \ s.t. \ a \in G \implies a \cdot 1 = 1 \cdot a = a \)

• Invertibility: \( \forall a \in G, \exists a^{-1} \in G \ s.t. \ a \cdot a^{-1} = a^{-1} \cdot a = 1 \)

**Definition 2.8.** The number of elements in a group is referred to as its order, denoted \( |G| \). A group with finite order is called a finite group.

Groups are often used to define a set of invariant operations which identify the symmetries inherent in a given physical object. For example, group theory is used extensively in crystallography and chemistry to identify certain symmetries in molecular structure [26]. We will employ groups in a similar manner to identify the set of permutations to which a given system is invariant.

### 2.5.1 Representation of a Finite Group

**Definition 2.9.** Let \( G \) be a group of finite order. A linear representation \( \rho \) of \( G \) on a representation space \( V \subseteq \mathbb{R}^n \) is a homomorphism,

\[
\rho : G \longrightarrow GL(V) \tag{2.5}
\]

\[
\rho(g) : V \longrightarrow V, \ g \in G. \tag{2.6}
\]

where \( GL(V) \) is the general linear group of isomorphisms on \( V \). Since it is a homomorphism, given elements \( a, b, c \in G \) such that \( a \cdot b = c \), the representation is such that \( \rho(a) \cdot \rho(b) = \rho(c) \).

In the representation literature the representation space is sometimes referred to as the representation itself; however, in the interest of clarity, we will abandon this convention. We can define a representation on any space with dimension greater than zero, as long as the representation on that space upholds the properties of the underlying group.

**Example 2.1.** Consider the group of two elements \( G = \{1, s\} \) such that \( s^2 = 1 \). We can represent this group in a multitude of ways. First, we consider what is referred to as the trivial representation of \( G \), \( \rho^1 : G \longrightarrow GL(\mathbb{R}) \).

\[
\rho^1(g) = 1, \ \forall g \in G.
\]

Note that all of the relationships of \( G \) are preserved in the representation even though almost all of the information in \( G \) is lost. We consider a different, two dimensional representation of \( G \), \( \rho^2 : G \longrightarrow GL(\mathbb{R}^2) \):

\[
\rho_2(1) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}
\]

\[
\rho_2(s) = \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix}.
\]
Note that this representation can be split into two one-dimensional representations.

**Example 2.2.** Consider the symmetric group $S_3$ which gives the permutation actions on the set $\{1, 2, 3\}$. There are six elements in the group, $G = \{123, 132, 213, 231, 312, 321\}$. The elements are denoted by the permuted numbers after the action of the given group element (e.g., $132$ means $1 \to 1, 2 \to 3, 3 \to 2$). The multiplication table of elements is given here with the column group element acting first and the row group element acting subsequently (i.e., column right multiplies row):

<table>
<thead>
<tr>
<th>$S_3$</th>
<th>123</th>
<th>213</th>
<th>132</th>
<th>321</th>
<th>231</th>
<th>312</th>
</tr>
</thead>
<tbody>
<tr>
<td>123</td>
<td>123</td>
<td>132</td>
<td>321</td>
<td>231</td>
<td>312</td>
<td>213</td>
</tr>
<tr>
<td>213</td>
<td>213</td>
<td>231</td>
<td>132</td>
<td>321</td>
<td>123</td>
<td>132</td>
</tr>
<tr>
<td>132</td>
<td>132</td>
<td>123</td>
<td>231</td>
<td>312</td>
<td>213</td>
<td>321</td>
</tr>
<tr>
<td>321</td>
<td>321</td>
<td>312</td>
<td>213</td>
<td>123</td>
<td>321</td>
<td>132</td>
</tr>
<tr>
<td>231</td>
<td>231</td>
<td>213</td>
<td>312</td>
<td>123</td>
<td>132</td>
<td>213</td>
</tr>
<tr>
<td>312</td>
<td>312</td>
<td>132</td>
<td>231</td>
<td>321</td>
<td>321</td>
<td>231</td>
</tr>
</tbody>
</table>

Table 2.1: Multiplication Table for $S_3$

For example, suppose we want to find the group element resulting from applying the $(321)$ permutation followed by the $(231)$ permutation. We have,

\[
\begin{pmatrix}
(231) \\
\text{row}
\end{pmatrix} \cdot \begin{pmatrix}
(321) \\
\text{column}
\end{pmatrix} = \begin{pmatrix}
(132) \\
\text{table result}
\end{pmatrix}.
\]

We can represent the elements of the group with the representation $\rho^3 : G \to GL(\mathbb{R}^5)$, given by:

\[
\rho^3(123) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} ; \quad \rho^3(213) = \begin{bmatrix} -1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} ; \quad \rho^3(132) = \begin{bmatrix} 0 & 0 & -1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix} ; \quad \rho^3(321) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 \\ 1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} ; \quad \rho^3(231) = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}.
\]

We define an important representation that will appear ubiquitously in subsequent chapters.
\textbf{Definition 2.10 (Permutation Representation).} A representation of $G$ is a \textit{permutation representation} if all of its elements are permutation matrices.

\textbf{Definition 2.11 (Isomorphic Representations).} We call two different representations $\rho_1 : G \rightarrow GL(V_1)$ and $\rho_2 : G \rightarrow GL(V_2)$ \textit{isomorphic} (denoted by $\rho_1 \cong \rho_2$) if there exists a linear invertible map $\tau : V_1 \rightarrow V_2$ such that $\tau \rho_1(g) = \rho_2(g)\tau$ for all $g \in G$.

\textbf{Definition 2.12.} Let $G$ be a finite group, $V \subset \mathbb{R}^n$ a representation space and $\rho : G \rightarrow GL(V)$ a representation of $G$ over $V$. The representation $\rho$ is said to be \textit{irreducible} if there does not exist a non-trivial subspace $V' \subset V$ that is invariant to the action of $G$, i.e. $\rho(g)V' \subset V'$, for all $g \in G$. We henceforth denote any irreducible representation with the tilde, i.e. $\tilde{\rho}$.

\textbf{Lemma 2.13.} A representation $\rho : G \rightarrow GL(V)$ is irreducible if and only if there does not exist an invertible map $P \in \mathbb{R}^{n \times n}$, where $n = \dim(V)$, such that for all $g \in G$,

$$P^{-1}\rho(g)P = \begin{bmatrix} \rho_1(g) & * \\ 0 & \rho_2(g) \end{bmatrix},$$

where $\rho_1 : V_1 \rightarrow V_1$ and $\rho_2 : V_2 \rightarrow V_2$ are themselves representations of $G$ over subspaces $V_1$ and $V_2$, respectively.

We recall Schur’s Lemma, which is one of the most fundamental concepts of representation theory.

\textbf{Lemma 2.14 (Schur’s Lemma).} Let $\tilde{\rho}^1 : G \rightarrow GL(V_1)$ and $\tilde{\rho}^2 : G \rightarrow GL(V_2)$ be two irreducible representations of $G$, and let $A : V_1 \rightarrow V_2$ be a linear map such that $\tilde{\rho}^2(g) \cdot A = A \cdot \tilde{\rho}^1(g)$ for all $g \in G$. Then:

1. If $\tilde{\rho}^1$ is not isomorphic to $\tilde{\rho}^2$, then $A = 0$.

2. If $V_1 = V_2$ and $\tilde{\rho}^1 = \tilde{\rho}^2$, then $A$ is a scalar multiple of identity.

\textit{Proof.} (1) Suppose $\tilde{\rho}^1$ is not isomorphic to $\tilde{\rho}^2$ and $A \neq 0$. Let $x \in \ker(A)$. Then, for all $g \in G$, $A\tilde{\rho}^1(g)x = \tilde{\rho}^2(g)Ax = 0$. Thus $\tilde{\rho}^1(g)x \in \ker(A)$, which implies $\tilde{\rho}^1(g)\ker(A) \subset \ker(A)$, for all $g \in G$. But $\tilde{\rho}^1$ is irreducible, so either $\ker(A) = 0$ or $\ker(A) = V_1$. The latter is impossible since $A \neq 0$. Thus, $\ker(A) = 0$.

Next let $x \in \text{Im}(A)$. Then $\tilde{\rho}^2(g)Ax = A\tilde{\rho}^1(g)x$, which implies that $\tilde{\rho}^2(g)\text{Im}(A) \subset \text{Im}(A)$. Since $\tilde{\rho}^2$ is irreducible, either $\text{Im}(A) = 0$ or $\text{Im}(A) = V_2$. Since the former implies $A = 0$, we get $\text{Im}(A) = V_2$. Combining the facts that $\text{Im}(A) = V_2$ and $\ker(A) = 0$, we obtain that $A$ is an isomorphism between $\tilde{\rho}^1$ and $\tilde{\rho}^2$ a contradiction.

(2) Suppose $V_1 = V_2$ and $\tilde{\rho}^1 = \tilde{\rho}^2$. Let $\lambda$ be an eigenvalue of $A$, and let $A' = A - \lambda I$. Clearly $\ker(A') \neq 0$ and $\tilde{\rho}^1(g)A' = A'\tilde{\rho}^1(g)$. Let $x \in \ker(A')$. Then $A'\tilde{\rho}^1(g)x = \tilde{\rho}^2(g)A'x = 0$. Thus $\tilde{\rho}^1(g)x \in \ker(A')$, which implies $\tilde{\rho}^1(g)\ker(A') \subset \ker(A')$, for all $g \in G$. Since $\tilde{\rho}^1$ is irreducible and $\ker(A') \neq 0$, $\ker(A') = V_1$. Therefore, $A' = A - \lambda I = 0$ which implies that $A = \lambda I$ as required. \qed
There are a finite number \( h \) of irreducible representations \( \tilde{\rho}_i : G \rightarrow GL(W_i), i = 1,...,h \), for a given group \( G \) that are unique up to isomorphism \([27]\). Thus any irreducible representation is isomorphic to exactly one of the \( \tilde{\rho}_i \). Henceforth, the irreducible representations will be fixed to be \( \{\tilde{\rho}_1,...,\tilde{\rho}_h\} \), with representation spaces \( \{W_1,...,W_h\} \), respectively. The following lemma regards the decomposition of a representation into irreducible constituents.

**Lemma 2.15.** \([27, \text{Proposition 1.8}]\) Let \( G \) be a finite group and \( \rho : G \rightarrow GL(V) \) be a representation of \( G \) on \( V \subset \mathbb{R}^n \). There exists a decomposition

\[
V = W_1^{\oplus \eta_1} \oplus \cdots \oplus W_h^{\oplus \eta_h}
\]

where the \( W_1,...,W_h \) are the distinct irreducible representation spaces of \( G \) such that \( \rho(g)W_i \subset W_i \). Each \( \eta_i \) is the multiplicity of \( W_i \) in \( V \). Note that it is possible that \( \eta_i = 0 \).

**Remark 2.3.** We note that the irreducible representations spaces \( W_i \) are necessarily real if the vector space \( V \) is real. This is because the \( W_i \) form a direct sum to give \( V \) and thus if even one of the \( W_i \) were non-real it would appear in vector space \( V \), making it non-real.

**Remark 2.4.** It is important to note that any representation space of a given finite group \( G \) will decompose into a direct sum of the \( W_i \), where \( i = 1,...,h \). However, the multiplicities \( \eta_i \) of the decomposition will vary depending on the representation space. For this reason, the multiplicities of the decomposition of a given representation space \( V \) are denoted as \( \eta_i^V \) (i.e. with the superscript \( V \)) to indicate that they pertain specifically to the space \( V \).

In the following lemma we show how this decomposition is extended to the representation itself.

**Lemma 2.16.** Given a real vector space \( V \sim \mathbb{R}^n \) and a representation of \( G \) on \( V \) given by \( \rho : G \rightarrow GL(V) \), there exists an invertible matrix \( T \in \mathbb{R}^{n \times n} \) such that for all \( g \in G \),

\[
\tilde{\rho}(g) = T^{-1}\rho(g)T = \bigoplus_{i=1}^{h} \bigoplus_{j=1}^{\eta_i} \tilde{\rho}_i(g)
\]

where the \( \tilde{\rho}_i(g) \in \mathbb{R}^{n_i \times n_i} \) are the \( h \) irreducible representations each repeated with multiplicity \( \eta_i \) and \( n_i \) is the dimension of the corresponding representation space \( W_i \).
Proof. By Lemma 2.15, representation space $V$ decomposes into the irreducible representation spaces $W_i$, each with multiplicity $\eta_i$. Thus we define the invertible transformation $T : W_1^{\oplus \eta_1} \oplus \cdots \oplus W_h^{\oplus \eta_h} \to V$. Note that, by Remark 2.3, $T$ is a real matrix. For each $g \in G$, $\tilde{\rho}(g) = T^{-1}\rho(g)T$ is a map $\tilde{\rho}(g) : W_i^{\oplus \eta_i} \to W_i^{\oplus \eta_i}$, which represents the action of $G$ on the direct sum of irreducible representation spaces $W_1^{\oplus \eta_1} \oplus \cdots \oplus W_h^{\oplus \eta_h}$. Recall that each irreducible representation space is invariant to the action of $G$, i.e. $\rho(g)W_i \subset W_i$ for all $g \in G$, $i = 1, \ldots, h$. The invariance of each $W_i$ implies that $\tilde{\rho}(g)$ has a block diagonal form with each block corresponding to each irreducible space $W_i$. Furthermore, each block is a representation of $G$ on an irreducible space and hence is an irreducible representation. It remains to prove that blocks corresponding to the same irreducible space $W_i$ are equal. Recall that an irreducible representation is isomorphic to exactly one of the $\tilde{\rho}_i$. Therefore we can select $T$ in such a way that all of the irreducible representations corresponding to a given irreducible space $W_i$ are equal to $\tilde{\rho}_i$. □

For convenience, we define the blocks 

$$ R_i = \frac{\eta_i}{\sum_{k=1}^{h} \eta_k} \tilde{\rho}_i(g) \quad (2.8) $$

such that $\tilde{\rho}(g) = \bigoplus_{i=1}^{h} R_i$. These blocks will be used in subsequent chapters.

Remark 2.5. We call the matrix $T$ the irreducible decomposition transformation. We note here that the transformation $T : W_1^{\oplus \eta_1} \oplus \cdots \oplus W_h^{\oplus \eta_h} \to V$ is defined specifically on the vector space $V$. In subsequent chapters, we will introduced an irreducible decomposition transformation for each new vector space in which we work. Therefore, we denote the transformation as $T_V$ to indicate that it pertains specifically to the space $V$.

The following example demonstrates the decomposition described in Lemma 2.15 above.

**Example 2.3.** We continue with Example 2.2 by considering three of the irreducible representations of the $S_3$ group, defined as follows:

$\tilde{\rho}_1(g)$ is called the "trivial" representation:

$$ \tilde{\rho}_1(123) = 1, \quad \tilde{\rho}_1(213) = 1, \quad \tilde{\rho}_1(132) = 1, $$

$$ \tilde{\rho}_1(321) = 1, \quad \tilde{\rho}_1(231) = 1, \quad \tilde{\rho}_1(312) = 1. $$

$\tilde{\rho}_2(g)$ is called the "sign" representation:

$$ \tilde{\rho}_2(123) = 1, \quad \tilde{\rho}_2(213) = 1, \quad \tilde{\rho}_2(132) = 1, $$

$$ \tilde{\rho}_2(321) = -1, \quad \tilde{\rho}_2(231) = -1, \quad \tilde{\rho}_2(312) = -1. $$
\(\tilde{\rho}_3(g)\) is called the "standard" representation:

\[\tilde{\rho}_3(123) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad \tilde{\rho}_3(213) = \begin{bmatrix} 0 & -1 \\ 1 & -1 \end{bmatrix}, \quad \tilde{\rho}_3(132) = \begin{bmatrix} -1 & -1 \\ 1 & 0 \end{bmatrix}, \quad \tilde{\rho}_3(321) = \begin{bmatrix} 0 & 1 \\ -1 & -1 \end{bmatrix}, \quad \tilde{\rho}_3(231) = \begin{bmatrix} 0 & 0 \\ -1 & 0 \end{bmatrix}, \quad \tilde{\rho}_3(312) = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}.\]

Now, suppose that we know that the irreducible decomposition transformation of our representation of \(S_3\) from Example 2.2 is as follows:

\[T = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix}\]

We apply this transformation to our representation and define \(\tilde{\rho}(g) = T^{-1}\tilde{\rho}_3(g)T\) for all \(g \in G\). Then \(\tilde{\rho}(g)\) is as follows:

\[\tilde{\rho}(123) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}; \quad \tilde{\rho}(213) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}; \quad \tilde{\rho}(132) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}; \quad \tilde{\rho}(321) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}; \quad \tilde{\rho}(231) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}; \quad \tilde{\rho}(312) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}.

By comparing \(\tilde{\rho}(g)\) with our irreducible representations above, it becomes clear that \(\tilde{\rho}(g)\) is a sum of 4 irreducible representations: \(\tilde{\rho}_1(g)\), \(\tilde{\rho}_2(g)\), \(\tilde{\rho}_3(g)\), and \(\tilde{\rho}_2(g)\). That is to say, for all \(g \in G\), \(\tilde{\rho}(g)\) can be expressed as follows:

\[
\tilde{\rho}(g) = \begin{bmatrix} \tilde{\rho}_1(g) & 0 & 0 & 0 \\ 0 & \tilde{\rho}_2(g) & 0 & 0 \\ 0 & 0 & \tilde{\rho}_3(g) & 0 \\ 0 & 0 & 0 & \tilde{\rho}_2(g) \end{bmatrix}.
\]
2.5.2 The Irreducible Decomposition Transformation

This section marks the beginning of the original material presented by this thesis. In particular, we will provide an explicit method by which the matrix $T$ from Lemma 2.16 can be found. However, we must first discover more information about the structure of the representation that we are investigating. Suppose we have a representation $\rho : G \rightarrow GL(V)$ of finite group $G$ on $V \sim \mathbb{R}^n$. We use the GAP [28] software package to easily discover the constituent irreducible representation matrices of the group $G$. However, the multiplicities $\eta_i$ of each irreducible representation space $W_i$ remain unknown. These multiplicities are crucial pieces of information for future theorems and decompositions. The so-called canonical decomposition given below can provide us with this information. See Serre [29, pg. 10] for a definition of the character of a representation.

**Theorem 2.5.** [29, pg. 21] [Canonical Decomposition] Suppose we have a representation $\rho : G \rightarrow GL(V)$ of finite group $G$ in $V \sim \mathbb{R}^n$. There exist subspaces $V_i = W_i^{\oplus \eta_i}$, such that the projection map $p_i : V \rightarrow V_i$ is given by:

$$p_i = \sum_{g \in G} \chi_i(g) \ast \rho(g),$$

where $i = 1, ..., h$, $\chi_i(g)$ is the character of the irreducible representation, and $\rho(g)$ is the representation.

The dimension of each $V_i$ is equal to the rank of each $p_i$, respectively. Therefore,

$$\eta_i = \frac{\dim(V_i^{\oplus n_i})}{n_i} = \frac{\dim(V_i)}{n_i} = \frac{\text{Rank}(p_i)}{n_i},$$

(2.10)

where the projection $p_i$ is as shown above and $n_i$ is the dimension of irreducible representation subspace $W_i$ given in Lemma 2.16. We now have all of the information that we need to find the irreducible decomposition transformation $T$ from Lemma 2.16. Note that with this information (i.e. the $n_i$ and $\eta_i$) and with the help of GAP to determine the irreducible representations, we can determine the exact form of $\tilde{\rho}(g)$ given in Lemma 2.16.

We now search for an invertible $T$ which satisfies the following commutative equation for all $g \in G$,

$$\rho(g)T = T\tilde{\rho}(g)$$

$$\rho(g)T - T\tilde{\rho}(g) = 0.$$ 

Note that the above is an over-constrained set of Sylvester equations. The following (new) method can be used to solve any set of multiple Sylvester equations (given that a solution is known to exist).

From [23, pg. 255], we rewrite each equation in vectorized form such that for all $g \in G$,

$$(I_n \otimes \rho(g) - \tilde{\rho}(g)^T \otimes I_n) \cdot \text{vec}(T) = 0.$$
The equations above form a set of $|G|$ constraint matrices, with each constraint matrix defined as $M(g_i) = (I_n \otimes \rho(g) - \tilde{\rho}(g)^T \otimes I_n)$ for all $g \in G$. We can explicitly write all of our constraints for $T$ by stacking the $M(g_i)$:

$$
M = \begin{pmatrix}
M(g_1) \\
M(g_2) \\
\vdots \\
M(g_{|G|})
\end{pmatrix} \cdot \text{vec}(T) = 0,
$$

(2.11)

where $g_i \in G$. We collect the constraints into a single matrix:

$$
M = \left( I_n \otimes \begin{pmatrix}
\rho(g_1) \\
\rho(g_2) \\
\vdots \\
\rho(g_{|G|})
\end{pmatrix} - \begin{pmatrix}
\tilde{\rho}(g_1)^T \\
\tilde{\rho}(g_2)^T \\
\vdots \\
\tilde{\rho}(g_{|G|})^T
\end{pmatrix} \otimes I_n \right) = \begin{pmatrix}
M(g_1) \\
M(g_2) \\
\vdots \\
M(g_{|G|})
\end{pmatrix}.
$$

Now, we rewrite the constraint in terms of the kernel space of $M$:

$$
M \cdot \text{vec}(T) = 0 \implies \text{vec}(T) \in \text{Ker}(M)
$$

The basis for $\text{Ker}(M)$ is given as a span of vectorized matrices, $\text{Ker}(M) = \text{span}\{\text{vec}(T_1), \ldots, \text{vec}(T_m)\}$. Therefore, $\text{vec}(T) \in \text{Ker}(M)$. Now, we can unvectorize the equations and define a vector space of matrices $\mathcal{M}$ as follows,

$$
T \in \mathcal{M} = \text{span}\{T_1, \ldots, T_m\} \implies T = \sum_{j=1}^{m} \lambda_j T_j
$$

From Lemma 2.16, we know that there exists at least one full rank $T$ that satisfies the above equations. The following lemma uses measure theory to allow us to computationally procure an irreducible decomposition transformation $T$.

**Lemma 2.17.** Suppose we have a set of basis matrices $\{T_1, \ldots, T_m\}$ such that each $T_i \in \mathbb{R}^{n \times n}$ and the span of the basis forms a vector space $\mathcal{M}$. Suppose also that there exists an invertible matrix $T$ in vector space $\mathcal{M}$. Then, a linear combination of the basis matrices with random coefficients yields an invertible matrix with probability 1.

**Proof.** We rely on an well-known result from measure theory stated in [30]. This result states that a polynomial function $\phi : \mathbb{R}^m \rightarrow \mathbb{R}$ is either identically 0, or non-zero almost everywhere. Recall that a matrix is invertible if and only if its determinant is not equal to zero. Let us define the polynomial function $\phi : \mathbb{R}^m \rightarrow \mathbb{R}$ such that,

$$
\phi(x_1, \ldots, x_m) = \det \left( \sum_{i=1}^{m} \lambda_j T_j \right),
$$
where each $T_j \in \mathbb{R}^{n \times n}$ is a basis matrix of vector space $\mathcal{M}$, $\lambda_j \in \mathbb{R}$ are coefficients, and $\text{det}()$ is the determinant function. By assumption, we know that there exists at least one matrix in $\mathcal{M}$ that is invertible and thus has non-zero determinant. Then the polynomial function $\phi(x_1, ..., x_m)$ is not identically zero and, by [30], the function is non-zero almost everywhere. Thus selecting random coefficients in a linear combination of the $T_j$ will almost surely yield an invertible matrix, i.e. with probability 1. 

By the above lemma, we can select the $\lambda_j$ to be random values to find an invertible matrix that satisfies the above equations with probability 1. The result is an irreducible decomposition transformation $T$ which satisfies all of our requirements. Finally, we note that some computation simplifications can be made to the above algorithm to improve performance. These simplifications are detailed in the Appendix A.
Chapter 3

Patterned Systems

We consider the linear time-invariant system,

\[
\dot{x} = Ax + Bu \tag{3.1}
\]
\[
y = Cx + Du \tag{3.2}
\]

where \(x \in \mathcal{X}, u \in \mathcal{U}, \) and \(y \in \mathcal{Y}\) with \(\mathcal{X} \sim \mathbb{R}^n, \mathcal{U} \sim \mathbb{R}^m,\) and \(\mathcal{Y} \sim \mathbb{R}^s,\) and matrices \(A \in \mathbb{R}^{n \times n}, B \in \mathbb{R}^{n \times m},\)
\(C \in \mathbb{R}^{s \times n}\) and \(D \in \mathbb{R}^{s \times m}.\) Here, \(\mathcal{X}\) is the state space, \(\mathcal{U}\) is the input space, and \(\mathcal{Y}\) is the output space. The goal of this chapter is to elaborate a method to mathematically capture the patterns that are inherent within a linear control system. In a number of canonical examples, patterns have been characterized via a commutative property between system matrices and a set of base matrices representing a symmetry operation \([14, 31, 17].\) It is important to establish a method by which these matrices can be determined for a given system. To do so, we must first examine the underlying interconnection structure of the system.

Particularly in large-scale systems, it is often helpful to divide the state, input and output spaces into subspaces and to analyze the structure of the maps between these spaces. As we will see, we can then represent the underlying structure of a system with a digraph. Furthermore, any symmetries or patterns of a given system will be reflected in the digraph of its interconnections. This method has been employed by Siljak to investigate the decentralization of large-scale systems \([3, Chapter 3.2].\) It has also been used by Consolini et. al. to develop a finite group of symmetries inherent to the interconnection structure of a given system \([17].\) We will now demonstrate this methodology and formalize it using concepts introduced in Section 2.4 of Chapter 2. First, we partition the state, input and output spaces as:

\[
\mathcal{X} = \mathcal{X}_1 \oplus \mathcal{X}_2 \oplus \ldots \oplus \mathcal{X}_p,
\]
\[
\mathcal{U} = \mathcal{U}_1 \oplus \mathcal{U}_2 \oplus \ldots \oplus \mathcal{U}_q,
\]
The subspaces are intended to represent interconnected subsystems in a sense to be further clarified below. Henceforth, it is assumed that the designer has some knowledge of how the system can be divided into meaningful subsystems. The system matrices can be correspondingly partitioned as follows.

\[
A = \begin{bmatrix}
A_{11} & \cdots & A_{1p} \\
\vdots & \ddots & \vdots \\
A_{p1} & \cdots & A_{pp}
\end{bmatrix},
\quad
B = \begin{bmatrix}
B_{11} & \cdots & B_{1q} \\
\vdots & \ddots & \vdots \\
B_{p1} & \cdots & B_{pq}
\end{bmatrix},
\quad
C = \begin{bmatrix}
C_{11} & \cdots & C_{1p} \\
\vdots & \ddots & \vdots \\
C_{r1} & \cdots & C_{rp}
\end{bmatrix},
\quad
D = \begin{bmatrix}
D_{11} & \cdots & D_{1q} \\
\vdots & \ddots & \vdots \\
D_{r1} & \cdots & D_{rq}
\end{bmatrix}.
\]

The partitioned state equations are as follows.

\[
\dot{x}_i = A_{ii} x_i + \sum_{j \neq i} A_{ij} x_j + \sum_{k=1}^{q} B_k u_k, \quad x_i \in \mathcal{X}_i, \quad u_k \in \mathcal{U}_k.
\]

We can also write the partitioned output equations.

\[
y_l = \sum_{i=1}^{p} C_{li} x_i + \sum_{k=1}^{q} B_{lk} u_k, \quad y_l \in \mathcal{Y}_l
\]

Since the system is assumed to have some form of patterning, we make the assumption that the partition has been formed in such a way that there is some repetition of blocks in the matrices given above. Therefore, we collect all distinct block matrices appearing in \(A, B, C\) and \(D\), with the restriction that the blocks in the different system matrices are always regarded as distinct. The list of distinct, non-zero blocks is defined as follows.

\[
\mathcal{L} = \{A_1, \ldots, A_{n_A}, B_1, \ldots, B_{n_B}, C_1, \ldots, C_{n_C}, D_1, \ldots, D_{n_D}\}.
\]

Additionally we define the index set \(\mathcal{I}_{\mathcal{L}} = \{1, 2, \ldots, n_A + n_B + n_C + n_D\}\) such that each element of the set \(\mathcal{L}\) has a unique index in the set \(\mathcal{I}_{\mathcal{L}}\). Furthermore, let \((\mathcal{L})_i\) be the element of the set \(\mathcal{L}\) at index \(i \in \mathcal{I}_{\mathcal{L}}\).

Finally, for notational convenience, we define the system matrix \(S : \mathcal{X} \oplus \mathcal{U} \oplus \mathcal{Y} \rightarrow \mathcal{X} \oplus \mathcal{U} \oplus \mathcal{Y}\),

\[
S = \begin{bmatrix}
A & B & 0 \\
0 & 0 & 0 \\
C & D & 0
\end{bmatrix} \in \mathbb{R}^{(n+m+s) \times (n+m+s)}.
\]

The inclusion of zero blocks in \(S\) will be made clear in the following section.
### 3.1 The Automorphism Group

Now we are ready to define a digraph that captures the system’s interconnection structure.

**Definition 3.1 (System Graph).** Consider the system \((3.1)\). We define the \textit{system graph} \(G = (V, E, L)\) to be the edge labeled digraph given by

- \(V = \{v_1, \ldots, v_{p+q+r}\} = \{v_{x_1}, \ldots, v_{x_p}, v_{u_1}, \ldots, v_{u_q}, v_{y_1}, \ldots, v_{y_r}\}\).
- \(E = \{e_{ij} = (v_j, v_i) \in V \times V : S_{ij} \neq 0\}\).
- \(L : E \rightarrow \mathcal{I} \) such that \(L(e_{ij}) = l_{ij} \in \mathcal{I}\) and \(S_{ij} = (\mathcal{L})_{l_{ij}}\).

**Remark 3.1.** We note that the function \(L\) provides us with a method for keeping track of the interconnections between subsystems, inputs, and outputs. This "labeling function" provides the relationship between each edge \(e_{ij}\) of the graph and the non-zero matrix block \(S_{ij} \in \mathcal{L}\) of the system matrix \(S\).

We now state some restrictions on the edges of the system graph that are due to the nature of the system matrices.

**Lemma 3.1.** For each \(v_j \in \{v_{u_1}, \ldots, v_{u_q}\}\), there exists \(v_i \in \{v_{x_1}, \ldots, v_{x_p}, v_{y_1}, \ldots, v_{y_r}\}\) such that \(e_{ij} \in E\) and \(S_{ij} \in \{B_1, \ldots, B_{n_B}, D_1, \ldots, D_{n_D}\}\). Furthermore, for each \(v_i \in \{v_{y_1}, \ldots, v_{y_r}\}\), there exists \(v_j \in \{v_{x_1}, \ldots, v_{x_p}, v_{u_1}, \ldots, v_{u_q}\}\) such that \(e_{ij} \in E\) and \(S_{ij} \in \{C_1, \ldots, C_{n_C}, D_1, \ldots, D_{n_D}\}\).

**Proof.** By definition, an input must affect the states or output of the system in which it is defined. Additionally, inputs cannot be affected by states, outputs, or other inputs. Therefore, the blocks of \(S\) representing the effect of the input vertices are represented by edges from the input vertices to either the state vertices or the output vertices. By definition these blocks will be the non-zero blocks of the \(B\) and \(D\) system matrices. By definition, an output can only be affected by the states or inputs of the system. Additionally, outputs cannot affect states, inputs, or other outputs. Therefore, the blocks of \(S\) representing the effect on the output vertices are represented by edges from either the state vertices or the input vertices to the output vertices. By definition these blocks will be the non-zero blocks of the \(C\) and \(D\) system matrices.

**Remark 3.2.** It is implicit in the preceding lemma that the input vertices \(\{v_{u_1}, \ldots, v_{u_q}\}\) cannot have edges that lead towards them. That is to say, for any edge \(e_{ij} \in E\), the corresponding \(v_i\) cannot be in the set \(\{v_{u_1}, \ldots, v_{u_q}\}\). Similarly, for any edge \(e_{ij} \in E\), the corresponding \(v_j\) cannot be in the set \(\{v_{y_1}, \ldots, v_{y_r}\}\).

We now define the underlying finite group that encodes the patterning of a given system. It is referred to as the \textit{automorphism group} of a graph and is well known in graph theory and group theory literature [20, 24].
Definition 3.2 (Automorphism Group). Given a graph \( G \), we define the automorphism group of \( G \) as the finite group of permutations on the ordered set of vertices \( V \) such that the ordered sets of edges \( E \) and edge label function \( L \) remain unchanged. Equivalently, it is the group of isomorphisms (see Definition 2.5 and Remark 2.1) from \( G \) to itself, i.e. the group of automorphisms of \( G \). We denote the automorphism group as \( \text{AUT}(G) \).

There are various methods used to determine the automorphism group of a given graph, as detailed by [25]. We have chosen to use the software NAUTY [32]. Although it does not allow edge labeled graphs or directed graphs, it is a very efficient tool and fairly easy to use. More information regarding the use of this software is given in the Appendix at the end of the document. Additionally, it is important to note that the automorphism group defined above is a subgroup of the automorphism group that does not preserve edge labels.

Since the automorphism group is a group of permutations on the ordered set of vertices \( V \) of \( G \), we can find a representation \( \rho : G \rightarrow \text{GL}(\mathbb{R}^k) \), where \( k = |V| \). We call such a representation a vertex permutation representation of \( \text{AUT}(G) \). The following important lemma serves as a starting point to establish a firmer connection between our linear system and the automorphism group that encodes the patterns inherent in the system graph.

Lemma 3.2. Consider the system graph \( G \) with labeled adjacency matrix \( M_L \). Let \( \rho : \text{AUT}(G) \rightarrow \text{GL}(\mathbb{R}^n) \) be a vertex permutation representation. Then, for all \( g \in \text{AUT}(G) \), \( \rho(g)M_L = M_L\rho(g) \).

Proof. Let \( G' = (V', E', L') \) be the new graph obtained after applying the permutation \( g \in \text{AUT}(G) \) to the vertices of \( G \). By definition of \( \text{AUT}(G) \), \( G' \) is equal to \( G \) and therefore \( E = E' \) and \( L(e_{ij}) = L'(e_{ij}) \).

By Lemma 2.12 and Remark 2.2, the labeled adjacency matrix \( M'_L \) of \( G' \) is exactly \( M'_L = \rho(g)^{-1}M_L\rho(g) \).

Since \( L(e_{ij}) = L'(e_{ij}) \) and \( E = E' \), we have \( M_L = M'_L \). Thus \( \rho(g)M_L = M_L\rho(g) \) for all \( g \in \text{AUT}(G) \), as required.

We now concern ourselves with the specific structure of an adjacency matrix of the system graph of a given linear system.

Lemma 3.3. Consider the system graph \( G = (V, E, L) \) of (3.1). The labeled adjacency matrix of \( G \) can be partitioned as follows:

\[
M_L = \begin{bmatrix}
M_A & M_B & 0 \\
0 & 0 & 0 \\
M_C & M_D & 0
\end{bmatrix}.
\]

where \( M_A \) represents the interconnection between the state space vertices, \( M_B \) represents the interconnection from the input vertices to the state space vertices, \( M_C \) represents the interconnection from the state space vertices to the output and \( M_D \) represents the interconnection from the input vertices to the output vertices.
Proof. First, we recall that the vertices are ordered as shown in Definition 3.1, i.e. state vertices, input vertices, then output vertices. We note that by Lemma 3.1 and Remark 3.2, the vertices representing the input subspace partitions $\mathcal{U}_i$ ($i = 1, ..., q$) cannot have edges directed towards them. Similarly, the vertices representing the output subspace partitions $\mathcal{Y}_i$ ($i = 1, ..., r$) cannot have edges directed away from them. As a result we have a zero column and zero row in the labeled adjacency matrix corresponding to the outgoing edges of the output vertices and the incoming edges of the input vertices.

We now consider the structure of a specific permutation representation of $\text{AUT}(\mathcal{G})$.

Lemma 3.4. Consider the system graph $\mathcal{G} = (V, E, L)$ of (3.1) and vertex permutation representation $\rho : \text{AUT}(\mathcal{G}) \rightarrow \text{GL}(\mathbb{R}^k)$, where $k = |V|$. Then for all $g \in \text{AUT}(\mathcal{G})$, the permutation representation has the following form:

$$
\rho(g) = \begin{bmatrix}
\rho_X(g) \\
\rho_U(g) \\
\rho_Y(g)
\end{bmatrix}
$$

(3.3)

where $\rho_X(g) \in \mathbb{R}^{p \times p}$, $\rho_U(g) \in \mathbb{R}^{q \times q}$ and $\rho_Y(g) \in \mathbb{R}^{r \times r}$.

Proof. We know that $\rho(g)$ is a vertex permutation matrix. Suppose $\rho(g)_{ij} = 1$, implying that $\rho(g)$ permutes $v_i$ and $v_j$. Now, suppose without loss of generality that $v_i \in \{v_{X_1}, ..., v_{X_p}\}$ and suppose that $v_j \in \{v_{U_1}, ..., v_{U_q}\}$. Since $\rho$ is a representation of $\text{AUT}(\mathcal{G})$, the graph $\mathcal{G}' = (V', E', L')$ resulting from the permutation will be such that $L(e_{ki}) = L'(e_{ki})$. Therefore the edges and edge labels of the vertices $v_i$ and $v_j$ must be the same. Now, recall that the index $l_{kj} = L'(e_{kj})$ of an outgoing edge of $v_j$ indexes the matrix block $S_{ki} \in \{B_1, ..., B_{n_B}\}$ and consider the index $l_{ki} = L(e_{ki})$ of an outgoing edge of vertex $v_i$ which indexes the matrix block $S_{ki}$. It is impossible for $S_{ki} \in \{B_1, ..., B_{n_B}\}$ since a state variable cannot be in the input space $\mathcal{U}$. This implies that $v_i$ and $v_j$ do not have the same edges and edge labels, a contradiction. Therefore $v_i$ and $v_j$ cannot be interchanged via a permutation in the automorphism group. In a similar way, it can be proven that vertices $v_i \in \{v_{X_1}, ..., v_{X_p}\}$ and $v_j \in \{v_{Y_1}, ..., v_{Y_r}\}$ as well as vertices $v_i \in \{v_{U_1}, ..., v_{U_q}\}$ and $v_j \in \{v_{U_1}, ..., v_{U_q}\}$ cannot be permuted by a permutation representation of the automorphism group. Thus we have that $\rho(g)_{ij} = 1$ is only possible if $v_i$ and $v_j$ both belong to same set among $\{v_{X_1}, ..., v_{X_p}\}$, $\{v_{U_1}, ..., v_{U_q}\}$, or $\{v_{Y_1}, ..., v_{Y_r}\}$. The block diagonal structure follows directly.

\[\square\]

3.2 The Extended Permutation Representation

Now we define a new representation of the automorphism group of a system graph which is an extension of the vertex permutation representation.

Definition 3.3. Consider the system (3.1) with system graph $\mathcal{G} = (V, E, L)$, automorphism group $\text{AUT}(\mathcal{G})$, and vertex permutation representation $\rho : \text{AUT}(\mathcal{G}) \rightarrow \text{GL}(\mathbb{R}^k)$, where $k = |V|$. We define
the extended permutation representation \( \rho^{ext} : AUT(\mathcal{G}) \rightarrow GL(\mathcal{X} \oplus \mathcal{U} \oplus \mathcal{Y}) \), given by:

\[
(\rho^{ext}(g))_{ij} = \begin{cases} 
I_{n_j} = I_{n_i} & \text{if } \rho(g)_{ij} = 1 \\
0_{n_i \times n_j} & \text{if } \rho(g)_{ij} = 0
\end{cases},
\]

where \( n_i \) and \( n_j \) are determined by the block \( S_{ij} \in \mathbb{R}^{n_i \times n_j} \) and \( 0_{n_i \times m} \) is an \( n_i \times n_j \) block of zeros.

We note that the extended permutation representation still holds the block diagonal structure shown above for the permutation representation, since we have only extended the dimension of its elements. In fact, we can express \( \rho^{ext}(g) \) as follows.

\[
\rho^{ext}(g) = \begin{bmatrix} 
\rho^{ext}_X(g) & 0 & 0 \\
0 & \rho^{ext}_U(g) & 0 \\
0 & 0 & \rho^{ext}_Y(g)
\end{bmatrix}.
\]

### 3.3 Commuting Relationships

We arrive at the following lemma, which is the final connection between the system matrix, the system graph, and the automorphism group.

**Lemma 3.5.** Consider the system (3.1) with system graph \( \mathcal{G} = (V, E, L) \) and extended permutation representation \( \rho^{ext} : AUT(\mathcal{G}) \rightarrow GL(\mathcal{X} \oplus \mathcal{U} \oplus \mathcal{Y}) \). Suppose \( S \) is the system matrix of this system as defined above. Then for all \( g \in G \), \( \rho^{ext}(g)S = S\rho^{ext}(g) \).

**Proof.** By Lemma 3.2, we have \( \rho(g)M_L = M_L\rho(g) \) for all \( g \in AUT(\mathcal{G}) \). Element-wise, we have \( (\rho(g)M_L)_{ij} = (M_L\rho(g))_{ij} \) for all \( i, j \). Expanding the left-hand side, we have

\[
(\rho(g)M_L)_{ij} = \sum_k \rho(g)_{ik}(M_L)_{kj}.
\]

Since \( \rho(g) \) is a permutation matrix, we can assume that \( \rho(g)_{ik'} = 1 \) and \( \rho(g)_{ij} = 0 \) for all \( j \neq k' \). Then,

\[
\sum_k \rho(g)_{ik}(M_L)_{kj} = (M_L)_{k'j} = l_{k'j}.
\]

We can similarly expand the right-hand side of the equation to get,

\[
(M_L\rho(g))_{ij} = \sum_h (M_L)_{ih}\rho(g)_{hj} = (M_L)_{ih'} = l_{ih'}.
\]

Putting together both sides of the equation we get \( l_{k'j} = l_{ih'} \) and thus \( (L)_{k'j} = (L)_{ih'} \). Since, by definition, \( S_{k'j} = (L)_{k'j} \) and \( S_{ih'} = (L)_{ih'} \), we have \( S_{k'j} = S_{ih'} \).

By definition of the extended permutation representation, \( \rho(g)_{ik'} = 1 \) implies that \( (\rho^{ext}(g))_{ik'} = I_{n_{k'}} \).
Chapter 3. Patterned Systems

Then we have,
\[
S_{k'j} = I_n, \ S_{k'j} = (\rho^{ext}(g))_{ik} S_{k'j} = \sum_k (\rho^{ext}(g))_{ik} S_{k'j} = (\rho^{ext}(g)S)_{ij}.
\]

Similarly,
\[
S_{ih'j} = S_{ih'} \ I_n = S_{ih'}(\rho^{ext}(g))_{h'j} = \sum_h S_{ih}(\rho^{ext}(g))_{h'j} = (S\rho^{ext}(g))_{ij}.
\]

Putting everything together, we obtain,
\[
(\rho^{ext}(g)S)_{ij} = (S\rho^{ext}(g))_{ij},
\]
which implies that \(\rho^{ext}(g)S = S\rho^{ext}(g)\) as desired.

Remark 3.3. We have that \(\rho^{ext}(g)S = S\rho^{ext}(g)\). Recall that we can partition the extended permutation representation into blocks. The resulting multiplication yields the following.
\[
\begin{bmatrix}
\rho_{X}^{ext}(g) \\
\rho_{\mu}^{ext}(g) \\
\rho_{Y}^{ext}(g)
\end{bmatrix}
\begin{bmatrix}
A & B & 0 \\
0 & 0 & 0 \\
C & D & 0
\end{bmatrix}
= 
\begin{bmatrix}
\rho_{X}^{ext}(g) \\
\rho_{\mu}^{ext}(g) \\
\rho_{Y}^{ext}(g)
\end{bmatrix}
\begin{bmatrix}
A & B & 0 \\
0 & 0 & 0 \\
C & D & 0
\end{bmatrix}.
\]

Since these matrices are partitioned in the same way, we can rewrite this equation as a series of commuting relationships.

\[
\rho_{X}^{ext}(g)A = A\rho_{X}^{ext}(g),
\rho_{X}^{ext}(g)B = B\rho_{\mu}^{ext}(g),
\rho_{Y}^{ext}(g)C = C\rho_{X}^{ext}(g),
\rho_{Y}^{ext}(g)D = D\rho_{\mu}^{ext}(g).
\]

These commuting relationships are exactly what we require for our future work. They show that the group that characterizes the patterns of a given system is exactly the automorphism group of the system graph of the linear system.

Example 3.1. Suppose we have a system of the form given in (3.1) and the following system matrices.

\[
A = \begin{bmatrix}
A_1 & A_2 & A_3 & 0 \\
A_2 & A_1 & 0 & A_3 \\
A_3 & 0 & A_1 & A_2 \\
0 & A_3 & A_2 & A_1
\end{bmatrix}; \quad B = \begin{bmatrix}
B_1 \\
B_1 \\
0 \\
0
\end{bmatrix}; \quad C = \begin{bmatrix}
0 & 0 & C_1 & C_1
\end{bmatrix}; \quad D = 0
\]
We divide the state space into 4 subsystem spaces, $X = X_1 \oplus X_2 \oplus X_3 \oplus X_4$. We do not partition the input and output spaces. We note that the system matrices have already been appropriately partitioned.

We also note that there are set of labels is $L = \{A_1, A_2, A_3, B_1, C_1\}$. We construct the graph as shown in Figure 3.1 below.

![Figure 3.1: Example 1: Example of a System Graph](image)

The adjacency matrix of the graph given above is as follows.

$$M = \begin{bmatrix}
1 & 1 & 1 & 0 & 1 & 0 \\
1 & 1 & 0 & 1 & 1 & 0 \\
1 & 0 & 1 & 1 & 0 & 0 \\
0 & 1 & 1 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 1 & 0 & 0 \\
\end{bmatrix}$$

Furthermore, we have the labeled adjacency matrix.

$$M_L = \begin{bmatrix}
A_1 & A_2 & A_3 & 0 & B_1 & 0 \\
A_2 & A_1 & 0 & A_3 & B_1 & 0 \\
A_3 & 0 & A_1 & A_2 & 0 & 0 \\
0 & A_3 & A_2 & A_1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & C_1 & C_1 & 0 & 0 \\
\end{bmatrix}$$

There are only two permissible permutations in the automorphism group of this graph. The first is the identity permutation, that is, the permutation in which the vertices are permuted to themselves. The second involves flipping the graph along the vertical axis as shown below.
Figure 3.2: Example 1: Permutation of System Graph

Note that the permutation does not affect the edges or the edge labels. The vertex permutation representation of this permutation is given by the following matrix.

\[
\rho(g) = \begin{bmatrix}
0 & 1 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 \\
\end{bmatrix}.
\]

Note that this representation commutes with \(M\) and \(M_L\). If the system had subsystems with multiple variables, the extended permutation representation could be found by replacing the non-zero entries with appropriately sized identity matrices.

We now present some additional examples of patterned systems, their system graphs, and some viable permutations of the system graph vertices. For simplicity, only the state space matrices are given.

**Example 3.2.** Consider the following state matrix:

\[
A = \begin{bmatrix}
A_1 & A_2 & 0 & 0 & 0 \\
A_2 & A_1 & A_2 & 0 & 0 \\
0 & A_2 & A_1 & A_2 & 0 \\
0 & 0 & A_2 & A_1 & A_2 \\
0 & 0 & 0 & A_2 & A_1 \\
\end{bmatrix},
\]

where \(A_1 \in \mathbb{R}^{n \times n}\) and \(A_2 \in \mathbb{R}^{n \times n}\) are arbitrary matrices. The associated system graph is:
We note that the system has a 4-link bidirectional chain structure. The only member of the automorphism group (other than identity) for the 4-link chain is the permutation which flips the entire chain:

\[ \rho(g_1) = \begin{bmatrix}
0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 1 & 0 \\
0 & 0 & 1 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0
\end{bmatrix} \]

Since all of the subsystems have the same dimension, we can define the extended permutation representation as \( \rho^{ext}(g) = \rho(g) \otimes I_n \). We note that the extended permutation representation commutes with \( A \) as follows:

\[ \rho^{ext}(g) \cdot A = \begin{bmatrix}
0 & 0 & 0 & 0 & I_n \\
0 & 0 & 0 & I_n & 0 \\
0 & 0 & I_n & 0 & 0 \\
0 & I_n & 0 & 0 & 0 \\
I_n & 0 & 0 & 0 & 0
\end{bmatrix} \cdot \begin{bmatrix}
A_1 & A_2 & 0 & 0 & 0 \\
A_2 & A_1 & A_2 & 0 & 0 \\
0 & A_2 & A_1 & A_2 & 0 \\
0 & 0 & A_2 & A_1 & A_2 \\
0 & 0 & 0 & A_2 & A_1
\end{bmatrix} = \begin{bmatrix}
0 & 0 & 0 & 0 & I_n \\
0 & 0 & 0 & I_n & 0 \\
0 & 0 & I_n & 0 & 0 \\
0 & I_n & 0 & 0 & 0 \\
I_n & 0 & 0 & 0 & 0
\end{bmatrix} = A \cdot \rho^{ext}(g) \]
Now we substitute values. To keep it simple we will use scalars; $A_1 = 7$ and $A_2 = 3$ and since $n = 1$, $I_n = 1$. The system matrix becomes the following:

$$A = \begin{bmatrix}
7 & 3 & 0 & 0 & 0 \\
3 & 7 & 3 & 0 & 0 \\
0 & 3 & 7 & 3 & 0 \\
0 & 0 & 3 & 7 & 3 \\
0 & 0 & 0 & 3 & 7
\end{bmatrix}$$

The permutation representation (incidentally equal to the extended permutation) is as follows:

$$\rho(1) = \begin{bmatrix}
1 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 1
\end{bmatrix}; \quad \rho(2) = \begin{bmatrix}
0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 1 & 0 \\
0 & 0 & 1 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0
\end{bmatrix}$$

**Example 3.3.** Consider the following state matrix:

$$A = \begin{bmatrix}
A_1 & A_2 & A_2 & 0 & 0 & 0 & 0 \\
A_2 & A_1 & 0 & A_2 & A_2 & 0 & 0 \\
A_2 & 0 & A_1 & 0 & 0 & A_2 & A_2 \\
0 & A_2 & 0 & A_1 & 0 & 0 & 0 \\
0 & A_2 & 0 & 0 & A_1 & 0 & 0 \\
0 & 0 & A_2 & 0 & 0 & A_1 & 0 \\
0 & 0 & A_2 & 0 & 0 & 0 & A_1
\end{bmatrix}$$

where $A_1 \in \mathbb{R}^{n \times n}$ and $A_2 \in \mathbb{R}^{n \times n}$ are arbitrary matrices. The associated system graph is:
From the graph, it is clear that the system has a two layer binary tree structure. The automorphism group for the tree contains 8 elements. We demonstrate the element which flips the entire tree and subsequently flips two of the lower branches in the following figure:

The vertex permutation representation for this permutation is as follows:

$$\rho(7) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$
The following matrices are the vertex permutation representations for the automorphism group:

\[
\begin{align*}
\rho(1) &= \begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix} &
\rho(2) &= \begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix} &
\rho(3) &= \begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix} \\
\rho(4) &= \begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0
\end{bmatrix} &
\rho(5) &= \begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0
\end{bmatrix} &
\rho(6) &= \begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 1 & 0 & 0
\end{bmatrix} \\
\rho(7) &= \begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix} &
\rho(8) &= \begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}
\end{align*}
\]

If \( n = 1 \), it can be shown that all of these matrices permute with the state matrix.
Chapter 4

Patterned Matrices and Subspaces

We know from the previous chapter that, given a system graph, we can determine a group of symmetry operations and develop a commuting property between the representation of this group and the matrices of the corresponding linear system. In this chapter, we further investigate the implications of such a commuting property and develop a decomposition that will serve as the crux of the remainder of this thesis. Furthermore, we investigate the properties of subspaces that exhibit a similar notion of patterning. Note that from this point we generalize the theory to include any finite group whose representation commutes with the matrices of a system (i.e. not just \( AUT(G) \)).

4.1 Patterned Matrices

Armed with the finite group characterization of patterns, we endeavor to further characterize the properties of a matrix that possesses the patterning prescribed by a given finite group \( G \). We begin with a fundamental definition.

**Definition 4.1 (G-Patterned Matrix).** Let \( G \) be a finite group, \( V_1 = \mathbb{R}^n \) and \( V_2 = \mathbb{R}^m \) be vector spaces, and \( \rho^1 : G \rightarrow GL(V_1) \) and \( \rho^2 : G \rightarrow GL(V_2) \) be two representations of \( G \). We say the matrix \( A \in \mathbb{R}^{n \times m} \) is G-patterned if \( \rho^1(g) \cdot A = A \cdot \rho^2(g) \) for all \( g \in G \).

This definition is nearly identical to the definition of \( G \)-equivariance given in by Consolini and Tosques [17] as well as Russo and Slotine [33]. Furthermore, we note the following interesting properties.

**Lemma 4.1 (Properties of Patterned Matrices).** Let \( G \) be a finite group with representations \( \rho^m : G \rightarrow GL(\mathbb{R}^m) \), \( \rho^n : G \rightarrow GL(\mathbb{R}^n) \), and \( \rho^p : G \rightarrow GL(\mathbb{R}^p) \) on \( \mathbb{R}^m \), \( \mathbb{R}^n \), and \( \mathbb{R}^p \), respectively. Furthermore, let \( \rho^{m+p} : G \rightarrow GL(\mathbb{R}^{m+p}) \) be a representation on \( \mathbb{R}^{m+p} \) such that \( \rho^{m+p}(g) = \begin{bmatrix} \rho^m(g) & 0 \\ 0 & \rho^p(g) \end{bmatrix} \) for all \( g \in G \). Then the following properties hold:

1. Let \( A_1 \in \mathbb{R}^{n \times m} \) and \( A_2 \in \mathbb{R}^{m \times p} \) be G-patterned matrices. Then the product \( A_1A_2 \) is G-patterned.
2. Let \( A_1 \in \mathbb{R}^{n \times m} \) and \( A_2 \in \mathbb{R}^{n \times m} \) be \( G \)-patterned matrices. Then the sum \( A_1 + A_2 \) is \( G \)-patterned.

3. Let \( A \in \mathbb{R}^{n \times m} \) be a \( G \)-patterned matrix, and \( \lambda \in \mathbb{R} \) be a scalar. Then the matrix \( \lambda A \) is \( G \)-patterned.

4. Let \( A \in \mathbb{R}^{n \times n} \) be an invertible \( G \)-patterned matrix. Then, \( A^{-1} \) is also \( G \)-patterned by the same representation as \( A \).

5. Let \( A_1 \in \mathbb{R}^{n \times m} \) and \( A_2 \in \mathbb{R}^{n \times p} \) be \( G \)-patterned matrices. Then the concatenation \([A_1 \ A_2]\) is also \( G \)-patterned.

Proof. We prove each part separately by direct computation. For all \( g \in G \):

1. \( \rho^n(g)A_1A_2 = A_1\rho^m(g)A_2 = A_1A_2\rho^n(g) \), as required.

2. \( \rho^n(g)(A_1 + A_2) = \rho^p(g)A_1 + \rho^m(g)A_2 = A_1\rho^m(g) + A_2\rho^m(g) = (A_1 + A_2)\rho^m(g) \), as required.

3. \( \rho^n(g)\lambda A = \lambda\rho^n(g)A = \lambda A\rho^m(g) \), as required.

4. \( \rho^n(g) = \rho^n(g)I = \rho^n(g)AA^{-1} = A\rho^n(g)A^{-1} \). Multiplying by the inverse we obtain the result \( \rho^n(g)A^{-1} = A^{-1}\rho^n(g) \).

5. \( \rho^n(g)[A_1 \ A_2] = [\rho^n(g)A_1 \ \rho^n(g)A_2] = [A_1\rho^m(g) \ A_2\rho^p(g)] = [A_1 \ A_2] \begin{bmatrix} \rho^m(g) & 0 \\ 0 & \rho^p(g) \end{bmatrix} = [A_1 \ A_2] \rho^{m+p}(g) \).

\( \square \)

4.2 Patterned Decomposition

We now begin the main contribution of this section by stating a powerful theorem which equates \( G \)-patterning to a special block diagonal form. Recall from Lemma 2.16 that a given representation space \( \mathcal{V} \) can decomposed into a direct sum of the \( h \) irreducible representation spaces \( \mathcal{W}_i \), each repeated with multiplicity \( \eta_i \) and having dimension \( n_i \).

**Theorem 4.1** (Patterned Matrix Structure). Let \( G \) be a finite group and let \( \rho^1 : G \rightarrow GL(\mathbb{R}^n) \) and \( \rho^2 : G \rightarrow GL(\mathbb{R}^m) \) be representations of \( G \) on \( \mathbb{R}^n \) and \( \mathbb{R}^m \), respectively. Let \( T_1 \in \mathbb{R}^{n \times n} \) and \( T_2 \in \mathbb{R}^{m \times m} \) be the associated irreducible decomposition transformations. A matrix \( A \in \mathbb{R}^{n \times m} \) is \( G \)-patterned if and only if \( \hat{A} = T_1^{-1}AT_2 \) has the following block diagonal form,

\[
\hat{A} = \begin{bmatrix}
\hat{A}_1 & 0 & \ldots & 0 \\
0 & \hat{A}_2 & \ldots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \ldots & \hat{A}_h
\end{bmatrix}, \quad \hat{A}_i = \hat{A}_i \otimes I_{n_i},
\]

(4.1)
where \( \hat{A}_i \in \mathbb{R}^{\eta_i^1 \times \eta_i^2} \), and \( \eta_i^1 \) and \( \eta_i^2 \) are the multiplicities of the irreducible spaces in \( \mathbb{R}^n \) and \( \mathbb{R}^m \). That is,

\[
\mathbb{R}^n = W_1^{\oplus \eta_1^1} \oplus \cdots \oplus W_h^{\oplus \eta_h^1},
\]

\[
\mathbb{R}^m = W_1^{\oplus \eta_1^2} \oplus \cdots \oplus W_h^{\oplus \eta_h^2},
\]

where \( W_i \sim \mathbb{R}^{n_i} \).

**Proof.** We first partition \( \hat{A} \) as follows,

\[
\hat{A} = \begin{bmatrix}
\hat{A}_{11} & \cdots & \hat{A}_{1h} \\
\vdots & \ddots & \vdots \\
\hat{A}_{h1} & \cdots & \hat{A}_{hh}
\end{bmatrix}, \quad \hat{A}_{ij} \in \mathbb{R}^{\eta_i^1 \times \eta_j^1}.
\] (4.2)

We show that \( A \) is \( G \)-patterned if and only if,

\[
\hat{A}_{ij} = \begin{cases} 
\hat{A}_i \otimes I_{n_i} & \text{if } i = j \\
0 & \text{if } i \neq j 
\end{cases}
\]

From Lemma 2.16, we know that for each representation \( \rho^1 \) and \( \rho^2 \), there exist transformation matrices \( T_1 \) and \( T_2 \) such that for \( q = 1, 2 \) and all \( g \in G \),

\[
\rho^j(g) = T_q^{-1} \rho^j(g) T_q = \begin{bmatrix}
R_1^q(g) \\
\vdots \\
R_h^q(g)
\end{bmatrix},
\] (4.3)

where \( R_i^q(g) = \oplus_{k=1}^{\eta_i^q} \hat{\rho}_k(g) \) for all \( i = 1, \ldots, h \) and \( q = 1, 2 \) as in (2.8).

( \( \Rightarrow \) ) Let \( A \) be \( G \)-patterned. By definition, \( \rho^1(g) \cdot A = A \cdot \rho^2(g) \) for all \( g \in G \). Applying \( T_1 \) and \( T_2 \), we have \( \hat{\rho}^1(g) \hat{A} = \hat{\rho}^2(g) \hat{A} \), where \( \hat{A} = T_1^{-1} A T_2 \). Using (4.2) and (4.3),

\[
\begin{bmatrix}
R_1^1(g) \\
\vdots \\
R_h^1(g)
\end{bmatrix} \begin{bmatrix}
\hat{A}_{11} & \cdots & \hat{A}_{1h} \\
\vdots & \ddots & \vdots \\
\hat{A}_{h1} & \cdots & \hat{A}_{hh}
\end{bmatrix} = \begin{bmatrix}
\hat{A}_{11} & \cdots & \hat{A}_{1h} \\
\vdots & \ddots & \vdots \\
\hat{A}_{h1} & \cdots & \hat{A}_{hh}
\end{bmatrix} \begin{bmatrix}
R_1^2(g) \\
\vdots \\
R_h^2(g)
\end{bmatrix}.
\]

That is, for all \( g \in G \) and \( i, j = 1, \ldots, h \),

\[
R_i^1(g) \hat{A}_{ij} = \hat{A}_{ij} R_j^2(g).
\]
Using the definition of $R_i^q(g)$, we get,

\[
\begin{bmatrix}
\tilde{\rho}_i(g) \\
\vdots \\
\tilde{\rho}_i(g)
\end{bmatrix}_{\eta_i^1 \text{ times}} \begin{bmatrix}
\tilde{A}_{ij} = \tilde{A}_{ij}
\end{bmatrix}_{\eta_i^2 \text{ times}} \begin{bmatrix}
\tilde{\rho}_j(g) \\
\vdots \\
\tilde{\rho}_j(g)
\end{bmatrix}
\]

This leads to a further partitioning of the $\tilde{A}_{ij}$,

\[
\tilde{A}_{ij} = \begin{bmatrix}
(\tilde{A}_{ij})_{11} & \cdots & (\tilde{A}_{ij})_{1\eta_i^2} \\
\vdots & \ddots & \vdots \\
(\tilde{A}_{ij})_{\eta_i^1 1} & \cdots & (\tilde{A}_{ij})_{\eta_i^1 \eta_i^2}
\end{bmatrix}, \quad (\tilde{A}_{ij})_{kl} \in \mathbb{R}^{n_i \times n_j}.
\]

Then (4.4) becomes for all $g \in G$,

\[
\tilde{\rho}_i(g) \cdot (\tilde{A}_{ij})_{kl} = (\tilde{A}_{ij})_{kl} \cdot \tilde{\rho}_j(g), \quad \forall i, j, k, l.
\] (4.5)

Now suppose $i \neq j$. Then $\tilde{\rho}_i$ and $\tilde{\rho}_j$ are not isomorphic ($\tilde{\rho}_i \not\sim \tilde{\rho}_j$). By Schur’s lemma, for all $k, l$ and $i \neq j$, $\tilde{A}_{ij} = 0$. If $i = j$ then $\tilde{\rho}_i = \tilde{\rho}_j$ and, again by Schur’s Lemma, $(\tilde{A}_{ij})_{kl} = a_{kl}^1 \cdot I_{n_i}$ where $a_{kl}^1 \in \mathbb{R}$. Then we can define,

\[
\tilde{A}_i = \tilde{A}_{ii} = \tilde{A}_i \otimes I_{n_i} = \begin{bmatrix}
a_{11}^i & \cdots & a_{1\eta_i^2}^i \\
\vdots & \ddots & \vdots \\
a_{\eta_i^1 1}^i & \cdots & a_{\eta_i^1 \eta_i^2}^i
\end{bmatrix} \otimes I_{n_i}.
\] (4.6)

Note that $\tilde{A}_i \in \mathbb{R}^{\eta_i^1 \times \eta_i^2}$.

( $\Leftarrow$ ) Suppose that $\tilde{A} = T_1^{-1}AT_2$ has the block diagonal form given in (4.1). We want to show that $\rho^1(g) \cdot A = A \cdot \rho^2(g)$ for all $g \in G$. It suffices to show that $\tilde{\rho}^3(g)\tilde{A} = \tilde{A}\tilde{\rho}^2(g)$ for all $g \in G$. Using our above derivations, we must equivalently show for all $i, j$ and all $g \in G$,

\[
R_i^1(g)\tilde{A}_{ij} = \tilde{A}_{ij}R_j^2(g)
\] (4.7)

First, if $i \neq j$ then, by assumption, $\tilde{A}_{ij} = 0$ and (4.7) holds automatically. Second, suppose $i = j$. Using (4.4), (4.6), and our assumptions, (4.7) becomes,

\[
R_i^1\tilde{A}_{ii} = R_i^1(\tilde{A}_i \otimes I_{n_i}) = \begin{bmatrix}
\tilde{\rho}_i(g) \\
\vdots \\
\tilde{\rho}_i(g)
\end{bmatrix}_{\eta_i^2 \text{ times}} \begin{bmatrix}
\begin{bmatrix}
a_{11}^i I_{n_i} & \cdots & a_{1\eta_i^2}^i I_{n_i} \\
\vdots & \ddots & \vdots \\
a_{\eta_i^1 1}^i I_{n_i} & \cdots & a_{\eta_i^1 \eta_i^2}^i I_{n_i}
\end{bmatrix}
\end{bmatrix}
\]
Thus the commuting property holds in both cases and $A$ is $G$-patterned.

Remark 4.1. We recall from Remark 2.5 that, in the above theorem, $T_1^{-1}$ is a map $T_1^{-1} : V_1 \rightarrow W_1^{\oplus n_1^i} \oplus \ldots \oplus W_h^{\oplus n_h^i}$ and $T_2$ is a map $T_2 : W_1^{\oplus n_1^i} \oplus \ldots \oplus W_h^{\oplus n_h^i} \rightarrow V_2$. This implies that $\hat{A} = T_1^{-1}A T_2$ is in fact a map $\hat{A} : W_1^{\oplus n_1^i} \oplus \ldots \oplus W_h^{\oplus n_h^i} \rightarrow W_1^{\oplus n_1^i} \oplus \ldots \oplus W_h^{\oplus n_h^i}$. By investigating the structure of $\hat{A}$ and recalling that its diagonal blocks are only nonzero when mapping between isomorphic irreducible representation spaces, we find that for all $i = 1, \ldots, h$ each $\hat{A}_i$ is itself a map $\hat{A}_i : W_i^{\oplus n_i^i} \rightarrow W_i^{\oplus n_i^i}$.

Remark 4.2. In the literature, a faithful representation as a representation in which each element of a group $G$ is represented by a distinct mapping $\rho(g)$. In our context, faithfulness indicates that the patterns encoded by the group are represented algebraically in the most complete way possible. Moreover, a more faithful set of representations will yield a more intricate decomposition, since more of the systems structure is being preserved by the commuting relationship. For example, the representation that maps all of the group elements to the identity matrix has no faithfulness and thus the decomposed matrix consists of a single block with no structure.

We now define a reduced form, which will be used for conciseness in future proofs. This form reduces the information contained in a $G$-patterned matrix to only that part which is essential to perform standard control synthesis.

Definition 4.2 (Pattern Reduced Form). Let $A \in \mathbb{R}^{n \times m}$ be a $G$-patterned matrix and let $\hat{A}$ be the pattern decomposition form of $A$ as shown in Theorem 4.1. We define the Pattern Reduced Form of matrix $A$ as the matrix $\hat{A} \in \mathbb{R}^{n \times \gamma_2}$ (with $\gamma_1 = \sum_{i=1}^{h} n_i^1$ and $\gamma_2 = \sum_{i=1}^{h} n_i^2$) such that $\hat{A}$ is the direct sum of the $\hat{A}_i$ from the pattern decomposition of $A$:

$$\hat{A} = \hat{A}_1 \oplus \hat{A}_2 \oplus \ldots \oplus \hat{A}_h = \begin{bmatrix} \hat{A}_1 & 0 & \ldots & 0 \\ 0 & \hat{A}_2 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & \hat{A}_h \end{bmatrix}$$

We call $\{\hat{A}_1, \ldots, \hat{A}_h\}$ the reduced form blocks of matrix $A$. 
Lemma 4.3. Let $T_m$, $T_n$, and $T_p$ be the irreducible decomposition transformation on $\mathbb{R}^m$, $\mathbb{R}^n$, and $\mathbb{R}^p$, respectively. Then the following properties about the reduced form hold:

1. Let $A_1 \in \mathbb{R}^{n \times m}$ and $A_2 \in \mathbb{R}^{m \times p}$ be $G$-patterned matrices. Then the reduced form of the product is $\tilde{A}_1 \tilde{A}_2 = \hat{A}_1 \hat{A}_2$.

2. Let $A_1 \in \mathbb{R}^{n \times m}$ and $A_2 \in \mathbb{R}^{n \times m}$ be $G$-patterned matrices. Then the reduced form of the sum is $A_1 + A_2 = \hat{A}_1 + \hat{A}_2$.

3. Let $A_1 \in \mathbb{R}^{n \times m}$ and $A_2 \in \mathbb{R}^{n \times p}$ be $G$-patterned matrices. Then the reduced form of the concatenation is $[A_1 \ A_2] = [\hat{A}_1 \ \hat{A}_2]$.

Proof. Note that by Lemma 4.1, the product, sum and concatenation of $A_1$ and $A_2$ are $G$-patterned matrices. Thus, in each case we simply apply the appropriate irreducible decomposition transformation matrix and proceed with direct computation:

1. We have $\tilde{A}_1 \tilde{A}_2 = \bigoplus_{i=1}^h A_1 A_2 \otimes I_n$. We also have that $\tilde{A}_1 \tilde{A}_2 = T_n^{-1} A_1 A_2 T_p = T_n^{-1} A_1 T_m T_n^{-1} A_2 T_p = \hat{A}_1 \hat{A}_2 = \bigoplus_{i=1}^h (\hat{A}_1 \otimes I_n)(\hat{A}_2 \otimes I_n) = \bigoplus_{i=1}^h (\tilde{A}_1 \tilde{A}_2) \otimes I_n$. Applying the reduction procedure to the left and right sides of the previous equation we see that $\tilde{A}_1 \tilde{A}_2 = \hat{A}_1 \hat{A}_2$.

2. We have $\tilde{A}_1 + \tilde{A}_2 = T_n^{-1} (A_1 + A_2) T_m = T_n^{-1} A_1 T_m + T_n^{-1} A_2 T_m = \hat{A}_1 + \hat{A}_2$. Applying the reduction procedure to the left and right sides of the previous equation we see that $\tilde{A}_1 + \tilde{A}_2 = \hat{A}_1 + \hat{A}_2$.

3. We have $[\tilde{A}_1 \ A_2] = T_n^{-1} [A_1 \ A_2] \begin{bmatrix} T_m & 0 \\ 0 & T_p \end{bmatrix} = [T_n^{-1} A_1 T_m \ T_n^{-1} A_2 T_p] = [\hat{A}_1 \ \hat{A}_2]$. Applying the reduction procedure to the left and right sides of the previous equation we see that $[\tilde{A}_1 \ A_2] = [\hat{A}_1 \ A_2]$.

We now give some results regarding patterned matrices and polynomials of these matrices, which will become important in the final chapter of this document.

Lemma 4.3. Let $G$ be a finite group and $\rho : G \rightarrow GL(X)$ be a representation of $G$ on $X$ with the irreducible decomposition matrix $T_X$. Let $A \in \mathbb{R}^{n \times n}$ be a square, $G$-patterned matrix. Then,

$$A^k = T_X^{-1} \tilde{A}^k T_X = T_X^{-1} \begin{bmatrix} \hat{A}_1^k \\ \vdots \\ \hat{A}_h^k \end{bmatrix} T_X.$$
Chapter 4. Patterned Matrices and Subspaces

The proof of the previous lemma follows clearly from direct multiplication, due to the block diagonal nature of \( \hat{A} \).

**Lemma 4.4.** Let \( G \) be a finite group and \( \rho: G \to GL(\mathcal{X}) \) be a representation of \( G \) on \( \mathcal{X} \) with the irreducible decomposition matrix \( T_\mathcal{X} \). Let \( A \in \mathbb{R}^{n \times n} \) be a square, G-patterned matrix. Let \( \psi(s) = \sum_{k=1}^{n-1} \lambda_k s^k \) be a polynomial. Then \( \psi(A) \) is G-patterned and the reduced form of \( \psi(A) \) is \( \psi(\hat{A}) \).

**Proof.** From Lemma 4.1, we have that \( \psi(A) \) is G-patterned such that \( \rho^X(g)\psi(A) = \psi(A)\rho^X(g) \). Applying the transformation \( T_\mathcal{X} \) and by Lemma 4.3, \( T_\mathcal{X}^{-1}\psi(A)T_\mathcal{X} = \psi(T_\mathcal{X}^{-1}AT_\mathcal{X}) = \psi(\hat{A}) = \bigoplus_i \psi(\hat{A}_i) \). Furthermore, for all \( i = 1, \ldots, h \), \( \psi(\hat{A}_i) = \psi(\hat{A}_i \otimes I_{n_i}) = \sum_{k=1}^{n_i-1} \lambda_k (\hat{A}_i \otimes I_{n_i})^k = (\sum_{k=1}^{n_i-1} \lambda_k \hat{A}_i^k) \otimes I_{n_i} = \psi(\hat{A}_i) \otimes I_{n_i} \). Altogether, \( \psi(\hat{A}) = \bigoplus_i \psi(\hat{A}_i) = \bigoplus_i \psi(\hat{A}_i) \otimes I_{n_i} \) and thus the reduced form of \( \psi(A) \) is \( \psi(\hat{A}) \).

\[ \square \]

### 4.3 Patterned Subspaces

In this section we define the so-called patterned subspace along with properties that will be useful for us in the remainder of the document. Let \( G \) be a finite group and let \( \rho^X: G \to GL(\mathcal{X}) \) be a representation of \( G \) on \( \mathcal{X} \sim \mathbb{R}^n \).

**Definition 4.3** (G-patterned subspace). We say that \( \mathcal{V} \subset \mathcal{X} \) is a G-patterned subspace if it is invariant to the action of \( G \) i.e. \( \rho^X(g)\mathcal{V} \subset \mathcal{V} \) for all \( g \in G \). If \( \mathcal{V} \) is G-patterned then we can define \( \rho^\mathcal{V}: G \to GL(\mathcal{V}) \), the restriction of \( \rho^X(g) \) to the subspace \( \mathcal{V} \).

We now introduce some basic properties of G-patterned subspaces.

**Lemma 4.5.** Let \( \mathcal{V} \subset \mathcal{X} \sim \mathbb{R}^n \) be a subspace with insertion map \( S \in \mathbb{R}^{n \times m} \) such that \( \mathcal{V} = Im(S) \subset \mathcal{X} \). Then \( \mathcal{V} \) is G-patterned if and only if \( S \) is a G-patterned matrix, i.e. \( \rho^X(g)S = S\rho^\mathcal{V}(g) \) for all \( g \in G \).

**Proof.** ( \( \Rightarrow \) ) Suppose \( \mathcal{V} \) is G-patterned. Then, as per Section 2.2.6, we have that \( \rho^X(g)S = S\rho^\mathcal{V}(g) \) for all \( g \in G \), where each \( \rho^\mathcal{V}(g) \) is the restriction of \( \rho^X(g) \) to the space \( \mathcal{V} \). We see that the restricted representations themselves form a representation of \( G \) on \( \mathcal{V} \), i.e. \( \rho^\mathcal{V}: G \to GL(\mathcal{V}) \).

( \( \Leftarrow \) ) Suppose \( \rho^X(g)S = S\rho^\mathcal{V}(g) \) and let \( x \in \mathcal{V} \) such that \( x = Sv \). Then, for all \( g \in G \), \( \rho^X(g)x = \rho^X(g)Sv = S\rho^\mathcal{V}(g)v \in \mathcal{V} \). Therefore \( \rho^X(g)\mathcal{V} \subset \mathcal{V} \) for all \( g \in G \), as required.

\[ \square \]

**Lemma 4.6.** A subspace \( \mathcal{V} \subset \mathcal{X} \) is G-patterned if and only if any complement of \( \mathcal{V} \), \( \mathcal{V}^c \) such that \( \mathcal{X} = \mathcal{V} \oplus \mathcal{V}^c \), is also G-patterned.

**Proof.** The proof of this lemma is based on a lemma proposed in [29, p. 6] which states that, given a subspace that is invariant to the action of \( G \), its complement is also invariant to the action of \( G \). Therefore, since \( \mathcal{V} \) is invariant to the action of \( G \) by definition, so is its complement. This implies that the complement is G-patterned by definition.

\[ \square \]
Lemma 4.7. Let $V \subset X$ be a subspace and let $S \in \mathbb{R}^{m \times n}$ such that $V = \text{Ker}(S) \subset X$. Then $V$ is $G$-patterned if and only if $\rho^X(g)S^T = S^T\rho^V(g)$ for all $g \in G$, where $V^\perp = \text{Im}(S^T)$ is the orthogonal complement of $V$.

Proof. By Lemma 4.6 above, we know that $V$ is $G$-patterned if and only if it has a $G$-patterned complement. Therefore we examine the orthogonal complement $V^\perp = (\text{Ker}(S))^\perp = \text{Im}(S^T)$ and find that, by definition, it is $G$-patterned if and only if $\rho^X(g)S^T = S^T\rho^{V^\perp}(g)$.

Lemma 4.8. If $V_1 \subset X$ and $V_2 \subset X$ are $G$-patterned subspaces, then $V = V_1 + V_2$, is also $G$-patterned.

Proof. Given that both $V_1$ and $V_2$ are $G$-patterned, we have that, for all $g \in G$, $\rho^X(g)V_1 \subset V_1$ and $\rho^X(g)V_2 \subset V_2$. Now, we observe that $\rho^X(g)(V) = \rho^X(g)(V_1 + V_2) = \rho^X(g)V_1 + \rho^X(g)V_2 \subset V_1 + V_2 = V$, which implies that $V$ is $G$-patterned by definition.

Lemma 4.9. If $V_1 \subset X$ and $V_2 \subset X$ are $G$-patterned subspaces, then $V = V_1 \cap V_2$, is also $G$-patterned.

Proof. Suppose $v \in V = V_1 \cap V_2$ so that $v \in V_1$ and $v \in V_2$. Since the subspaces are $G$-patterned, we have that, for all $g \in G$, $\rho^X(g)v \in V_1$ and $\rho^X(g)v \in V_2$, which implies that $\rho^X(g)v \in V$. Thus $\rho^X(g)V \subset V$ for all $g \in G$ and $V$ is $G$-patterned.

Lemma 4.10. Let $A \in \mathbb{R}^{n \times n}$ is a $G$-patterned matrix such that $\rho^X(g)A = A\rho^X(g)$ for all $g \in G$ and let $V \subset X$ is a $G$-patterned subspace. Then the preimage $W = A^{-1}V$ is also a $G$-patterned subspace.

Proof. Let $x \in A^{-1}V$. Since $Ax \in V$ and $\rho(g)V \subset V$ for all $g \in G$, we have $A\rho(g)x = \rho(g)Ax \in V$ for all $g \in G$. Thus, $\rho(g)x \in A^{-1}V$, as required.

The following three lemmas will be used in the last chapters of this thesis.

Lemma 4.11. Let $X \sim \mathbb{R}^n$ be a vector space and let $V \sim \mathbb{R}^m$ and $V' \sim \mathbb{R}^p$ be $G$-patterned subspaces of $X$. Let $S \in \mathbb{R}^{m \times n}$ and $S' \in \mathbb{R}^{n \times p}$ be the insertion maps of $V$ and $V'$, respectively, such that $V = \text{Im}(S)$ and $V' = \text{Im}(S')$. Moreover, let $\tilde{S}$ and $\tilde{S}'$ be the reduced forms of $S$ and $S'$ respectively. Then $V \subset V'$ if and only if $\text{Im}(\tilde{S}) \subset \text{Im}(\tilde{S}')$.

Proof. We have that $\text{Im}(S) \subset \text{Im}(S')$ is equivalent to $\text{Im}(\tilde{S}) \subset \text{Im}(\tilde{S}')$. Due to the block diagonal structure of $\tilde{S}$ and $\tilde{S}'$, the previous statement is true if and only if $\text{Im}(\tilde{S}_i) \subset \text{Im}(\tilde{S}'_i)$ for all $i = 1, \ldots, h$. Applying Theorem 2.4, we have that $\text{Im}(\tilde{S}_i) \subset \text{Im}(\tilde{S}'_i)$ is true if and only if $\text{Im}(\tilde{S}_i) \subset \text{Im}(\tilde{S}'_i)$ for all $i = 1, \ldots, h$. Collecting the equations into a block diagonal form, we have $\text{Im}(\tilde{S}) \subset \text{Im}(\tilde{S}')$.

Lemma 4.12. Let $S \in \mathbb{R}^{m \times n}$ and $S' \in \mathbb{R}^{p \times n}$ be $G$-patterned matrices. Then $\text{Ker}(S) \subset \text{Ker}(S')$ if and only if $\text{Ker}(\tilde{S}) \subset \text{Ker}(\tilde{S}')$.

Proof. We have that $\text{Ker}(S) \subset \text{Ker}(S')$ if and only if $\text{Ker}(\tilde{S}) \subset \text{Ker}(\tilde{S}')$. Due to the block diagonal structure of $\tilde{S}$ and $\tilde{S}'$, the previous statement is true if and only if $\text{Ker}(\tilde{S}_i) \subset \text{Ker}(\tilde{S}'_i)$ for all $i = 1, \ldots, h$. Due to the
Applying Theorem 2.4, we have that Ker($\tilde{S}_i$) $\subset$ Ker($\tilde{S}_i'$) is true if and only if Ker($\tilde{S}_i$) $\subset$ Ker($\tilde{S}_i'$) for all $i = 1, \ldots, h$. Collecting the equations into a block diagonal form, we have Ker($\tilde{S}$) $\subset$ Ker($\tilde{S}'$).

**Lemma 4.13.** Let $S \in \mathbb{R}^{n \times n}$ and $S' \in \mathbb{R}^{n \times p}$ be $G$-patterned matrices. Then Ker($S$) $\subset$ Im($S'$) if and only if Ker($\tilde{S}$) $\subset$ Im($\tilde{S}'$).

**Proof.** We have that Ker($S$) $\subset$ Im($S'$) if and only if Ker($\tilde{S}$) $\subset$ Im($\tilde{S}'$). Due to the block diagonal structure of $\tilde{S}$ and $\tilde{S}'$, the previous statement is true if and only if Ker($\tilde{S}_i$) $\subset$ Im($\tilde{S}_i'$) for all $i = 1, \ldots, h$. Applying Theorem 2.4, we have that Ker($\tilde{S}_i$) $\subset$ Ker($\tilde{S}_i'$) is true if and only if Ker($\tilde{S}_i$) $\subset$ Ker($\tilde{S}_i'$) for all $i = 1, \ldots, h$. Collecting the equations into a block diagonal form, we have Ker($\tilde{S}$) $\subset$ Im($\tilde{S}'$).

Some extra details regarding the subspaces of block diagonal matrices has been removed from the above proofs in the interest of conciseness. For the sake of clarity, the following remark demonstrates these details in the case of Lemma 4.12.

**Remark 4.4.** Let $G$ be a finite group with representations $\rho^X : G \to GL(X)$, $\rho^U : G \to GL(U)$, and $\rho^Y : G \to GL(Y)$ on the vector spaces $X \sim \mathbb{R}^n$, $U \sim \mathbb{R}^p$, and $Y \sim \mathbb{R}^m$, respectively. Let matrices $S : X \to Y$ and $S' : X \to U$ be $G$-patterned. We wish to show that Ker($S$) $\subset$ Ker($S'$) $\iff$ Ker($\tilde{S}_i$) $\subset$ Ker($\tilde{S}'_i$) for all $i = 1, \ldots, h$.

First, note that Ker($S$) $\subset$ Ker($S'$) $\iff$ Ker($\tilde{S}$) $\subset$ Ker($\tilde{S}'$). Recall from Lemma 2.15 that, given a finite group $G$, the state space $X \sim \mathbb{R}^n$ can be expressed as a direct sum of the irreducible subspaces $X = W^\oplus_{i=1}W_i^\oplus \oplus \cdots \oplus W^\oplus_{h} W_h^\oplus$. Let $X_i = W^\oplus_i$ such that $X = X_1 \oplus \cdots \oplus X_h$, and let $T_i : X_i \to X$ be the insertion map of each subspace. For completeness, let $U = U_1 \oplus \cdots \oplus U_h$ and $Y = Y_1 \oplus \cdots \oplus Y_h$, where $U_i = W^\oplus_i U$ and $Y_i = W^\oplus_i Y$. Then from Remark 4.1, we have that $\tilde{S}_i : X_i \to Y_i$ and $\tilde{S}'_i : X_i \to U_i$ for all $i = 1, \ldots, h$. Let $x_i \in X_i$ such that $x_i \in$ Ker($\tilde{S}_i$). Consider the following equation.

\[
\tilde{S}_i x_i = \begin{bmatrix}
\tilde{S}_1 \\
\vdots \\
\tilde{S}_i \\
\vdots \\
\tilde{S}_h
\end{bmatrix} = \begin{bmatrix}
0 \\
0 \\
\vdots \\
0
\end{bmatrix}.
\]

Thus, $T_i x_i \in$ Ker($\tilde{S}$). So, by assumption, $T_i x_i \in$ Ker($\tilde{S}'$). We have,

\[
0 = \tilde{S}'_i T_i x_i = \begin{bmatrix}
0 \\
0 \\
\vdots \\
0
\end{bmatrix} = \begin{bmatrix}
\tilde{S}'_i \\
\tilde{S}'_i \\
\vdots \\
\tilde{S}'_i
\end{bmatrix} x_i = \begin{bmatrix}
0 \\
0 \\
\vdots \\
0
\end{bmatrix}.
\]
This implies that $x_i \in \text{Ker}(\tilde{S}'_i)$, as required.

In the reverse direction, let $x \in \text{Ker}(\tilde{S})$ and partition $x$ according to the blocks of $\tilde{S}$ such that $x = \begin{bmatrix} x_1 & \cdots & x_h \end{bmatrix}^T$. Thus, $\tilde{S}_i x_i = 0$ for all $i = 1, \ldots, h$. Then, by assumption, $\tilde{S}'_i x_i = 0$ for all $i = 1, \ldots, h$, which implies that $\tilde{S}' x = 0$. Therefore, $x \in \text{Ker}(\tilde{S}')$, as required.
Chapter 5

Patterned Controllability and Observability

Given the characterization of patterned matrices from the previous chapter, we can now define the notion of a patterned system. Furthermore, in this chapter we will introduce the basic notions on the existence of controllers that can stabilize a patterned system while also preserving the patterns that characterize the system. Additionally, pattern-preserving observability and detectability are developed as dual concepts to controllability and stabilizability.

5.0.1 Patterned Dynamical System

We begin by recalling the standard linear time-invariant (LTI) dynamic system with output $y$. Let $\mathcal{X} \sim \mathbb{R}^n$ be the state space, $\mathcal{U} \sim \mathbb{R}^m$ be the input space, and $\mathcal{Y} \sim \mathbb{R}^p$ be the output space. The system is:

$$\dot{x} = Ax + Bu \quad (5.1)$$
$$y = Cx \quad (5.2)$$

where $x \in \mathcal{X}$, $u \in \mathcal{U}$, $y \in \mathcal{Y}$, $A \in \mathbb{R}^{n \times n}$ such that $A : \mathcal{X} \to \mathcal{X}$, $B \in \mathbb{R}^{n \times m}$ such that $B : \mathcal{U} \to \mathcal{X}$, $C \in \mathbb{R}^{p \times n}$ such that $C : \mathcal{X} \to \mathcal{Y}$. We also denote the feedback matrix $K \in \mathbb{R}^{m \times n}$ such that $K : \mathcal{X} \to \mathcal{U}$. Furthermore we say that feedback matrix $K$ is $G$-patterned if $\rho^u(g)K = K\rho^x(g)$. We can easily extend the concept of a $G$-patterned matrix to a LTI dynamical system.

**Definition 5.1** (G-Patterned System). Let $G$ be a finite group, let $A$, $B$ and $C$ in (5.1) above and let each matrix be $G$-patterned. That is to say, there exist $\rho^x(G) : G \to \text{GL}(\mathcal{X})$, $\rho^u(G) : G \to \text{GL}(\mathcal{U})$ and $\rho^y(G) : G \to \text{GL}(\mathcal{Y})$ such that $\rho^x(g)A = A\rho^x(g)$, $\rho^x(g)B = B\rho^u(g)$ and $\rho^y(g)C = C\rho^x(g)$. Then we say the system $(C, A, B)$ is a $G$-patterned system.
Once again, this definition is nearly identical to the definition of a $G$-equivariant system given by Consolini and Tosques [17].

By Theorem 4.1, we note that there exist invertible irreducible decomposition transformations, $T_X$, $T_u$ and $T_y$, which render the matrices $A, B, C, K$ into their respective decomposed forms $\hat{A} = T_X^{-1}AT_X$, $\hat{B} = T_X^{-1}BT_U$, $\hat{C} = T_Y^{-1}CT_X$ and $\hat{K} = T_U^{-1}KT_X$. We say $(\hat{C}, \hat{A}, \hat{B})$ is the reduced form of the system $(C, A, B)$. Note that the dimensions still agree in the reduced form. Additionally, we call each $(\hat{C}_i, \hat{A}_i, \hat{B}_i)$ a reduced subsystem of the $G$-patterned system $(C, A, B)$.

We now state some of the properties of the eigenvalue spectrum of a square $G$-patterned matrix.

**Lemma 5.1.** Let $A \in \mathbb{R}^{n \times n}$ be a $G$-patterned matrix. The spectrum $\sigma(A)$ is equivalent to the disjoint union of the spectra of the $\hat{A}_i$ each repeated $n_i$ times, where $n_i$ is the dimension of $W_i$ as in Lemma 2.16. That is,

$$\sigma(A) = \bigoplus_{i=1}^{h} \bigoplus_{j=1}^{n_i} \sigma(\hat{A}_i).$$

**Proof.** First, let $M = \bigoplus_{i=1}^{h} M_i$ be an arbitrary block diagonal matrix. We note that $\sigma(M) = \sigma(\bigoplus_{i=1}^{h} M_i) = \bigoplus_{i=1}^{h} \sigma(M_i)$ by a property of block diagonal matrices. From this fact and Theorem 4.1, we have $\sigma(A) = \sigma(\hat{A}) = \sigma(\bigoplus_{i=1}^{h} \hat{A}_i) = \sigma(\bigoplus_{i=1}^{h} \hat{A}_i \otimes I_{n_i}) = \bigoplus_{i=1}^{h} \sigma(\hat{A}_i \otimes I_{n_i}) = \bigoplus_{i=1}^{h} \bigoplus_{j=1}^{n_i} \sigma(\hat{A}_i).$}

We can show a similar result regarding the Jordan normal form a $G$-patterned matrix.

**Lemma 5.2.** Let $A \in \mathbb{R}^{n \times n}$ be a $G$-patterned matrix and let $\hat{J}_i$ be the Jordan normal form of each of the the $\hat{A}_i$. Then the Jordan normal form of $A$ is given by,

$$J = \bigoplus_{i=1}^{h} \bigoplus_{j=1}^{n_i} \hat{J}_i.$$

**Proof.** Since $A$ and $\hat{A}$ are similar matrices, their Jordan normal form is the same. Consider $\tilde{A}_i = \hat{A}_i \otimes I_{n_i}$. By Lemma 2.6, the Jordan normal form of $\tilde{A}_i$ is the same as that of $\hat{J}_i \otimes I_{n_i}$. Moreover, by Lemma 2.7, there exists a permutation matrix $S_i$ such that $S_i(\hat{J}_i \otimes I_{n_i})S_i^{-1} = I_{n_i} \otimes \hat{J}_i = \bigoplus_{j=1}^{n_i} \hat{J}_i$, which is already in Jordan normal form. Thus, for all $i = 1,...,h$, there exists an invertible matrix $P_i$ such that $\tilde{A}_i = P_i(I_{n_i} \otimes \hat{J}_i)P_i^{-1}$. Now, we have,

$$\hat{A} = \bigoplus_{i=1}^{h} \hat{A}_i = \bigoplus_{i=1}^{h} P_i(I_{n_i} \otimes \hat{J}_i)P_i^{-1} = P(\bigoplus_{i=1}^{h} \bigoplus_{j=1}^{n_i} \hat{J}_i)P^{-1},$$

where $P = \bigoplus_{i=1}^{h} P_i$ (note that the direct sum here implies block diagonal matrix concatenation). Since the $\hat{J}_i$ are already in Jordan normal form, we have that the Jordan normal form of $\hat{A}$ (and thus also of $A$) is $J = \bigoplus_{i=1}^{h} \bigoplus_{j=1}^{n_i} \hat{J}_i$, as required. 

\[\square\]
As a direct result of this property, we give the following result regarding minimal polynomials.

**Lemma 5.3.** Let $A \in \mathbb{R}^{n \times n}$ be a $G$-patterned matrix. Let $\psi(s)$ be the minimal polynomial of $A$. Then it is also the minimal polynomial of the reduced form of $A$, $\hat{A}$.

*Proof.* by Theorem 3.3.6 of [34, p. 345] that the minimal polynomial of a matrix $A$ is entirely determined by the Jordan blocks of $A$. By Lemma 5.2, we have that the distinct Jordan blocks of $A$ and $\hat{A} = \bigoplus_{i=1}^{h} \hat{A}_i$ are the same. Thus their minimal polynomials are the same, as required.

Equation (5.3) shows that there is redundancy in the spectrum of a patterned matrix. More specifically, the eigenvalues of each $\hat{A}_i$ repeat $n_i$ times. The same repetition occurs for the blocks of the Jordan normal form. This redundancy is, in fact, a direct result of the types of symmetries that are present within the system. We formalize this repetition in the spectrum of a $G$-patterned matrix as follows.

**Definition 5.2** (Patterned Spectrum). Let $G$ be a finite group with representation $\rho : G \rightarrow GL(\mathcal{X})$ on vector space $\mathcal{X} \sim \mathbb{R}^n$. Then we say that the spectrum $\mathcal{L}$ is $G$-patterned if it can be expressed as $\mathcal{L} = \bigoplus_{i=1}^{h} \bigoplus_{j=1}^{n_i} \mathcal{L}_i$, where the $\mathcal{L}_i$ are any arbitrary symmetric spectra with cardinalities given by $|\mathcal{L}_i| = \eta_i$, where $\eta_i$ and $n_i$ are as in Lemma 5.1 above.

As we will see in the next section, the patterning of the spectrum restricts our ability to assign the eigenvalues of a system via feedback, subject to the restriction that the patterns of the system be preserved.

## 5.1 Controllability

A primary goal of this work is to elucidate the possible spectra that can be assigned to a given patterned system via patterned feedback. The multiplicity or repetition of certain eigenvalues within the system is an inherent property of the group symmetries in the system. Since we seek to preserve the symmetries in the system, the repetition of the eigenvalues must also be preserved. In terms of controllability, we must only assign the spectrum of the reduced system if we want to preserve symmetry. Therefore, we define a notion of controllability based on controllability of the reduced subsystems.

**Definition 5.3** (Pattern Controllability). Suppose we have a $G$-patterned system $(A, B)$. Then we say that the pair $(A, B)$ is pattern controllable if the reduced form $(\hat{A}, \hat{B})$ is controllable in the standard linear control theory sense.

*Remark* 5.1. We note that the definition of pattern controllability above is equivalent to controllability of each of the reduced subsystems $(\hat{A}_i, \hat{B}_i)$ for all $i = 1, \ldots, h$ in the standard linear control theory sense. In fact, in future proofs we will use exactly this condition to show pattern controllability. ▶

We recall that we define the *controllable subspace* of a system is defined as follows:
where $\mathcal{B} = \text{Im}(B)$. We note the following lemma about the controllable subspace of a $G$-patterned system.

**Lemma 5.4.** Suppose a system $(A, B)$ is $G$-patterned. Then the controllable subspace $\mathcal{C} = \langle A|B \rangle$ is a $G$-patterned subspace.

**Proof.** By Lemma 4.1, $A^{i-1}B$ is $G$-patterned and by Lemma 4.5, $\text{Im}(A^{i-1}B) = A^{i-1}\text{Im}(B)$ is a $G$-patterned subspace. Now, by Lemma 4.8, $\langle A|B \rangle$ is $G$-patterned. \hfill $\square$

This, in turn, also implies that the reachable space $\text{Im}(L_e)$ of a $G$-patterned system is always patterned since $\text{Im}(L_e) = \mathcal{C}$. We recall the definition of the controllability matrix:

**Definition 5.4** (Controllability Matrix). Let $(A, B)$ be as in (3.1). Then the controllability matrix is defined as $M_e = \begin{bmatrix} B & AB & A^2B & \cdots & A^{n-1}B \end{bmatrix}$. Furthermore, $\mathcal{C} = \text{Im}(M_e)$.

By Lemma 4.5, $M_e$ is a $G$-patterned matrix. In pursuit of our pole-placement goal, we first consider the following two lemmas, which will bridge the gap between pattern controllability and patterned pole placement.

**Lemma 5.5.** Let $A$, $B$, and $K$ be as in (5.1) and let each matrix be $G$-patterned. Then the matrix $A + BK$ is also $G$-patterned. Furthermore, $A + BK = \hat{A} + \hat{B}K$ and $A + BK = \hat{A} + \hat{B}K$.

**Proof.** This lemma follows directly from Lemma 4.1 and Lemma 4.2. \hfill $\square$

We now state our main pole placement result and give a proof that constructively demonstrates how to determine a patterned feedback $K$.

**Theorem 5.1** (Patterned Pole Placement). Let $(A, B)$ be $G$-patterned system and let $\mathcal{L}$ be an arbitrary a $G$-patterned spectrum. Then there exists a $G$-patterned matrix $K \in \mathbb{R}^{m \times n}$ such that $\sigma(A + BK) = \mathcal{L}$ if and only if $(A, B)$ is pattern controllable.

**Proof.** $(\implies)$ Suppose we have a $G$-patterned matrix $K$ such that $\sigma(A + BK) = \mathcal{L}$. Since $A$, $B$, and $K$ are patterned we know that $A + BK$ is patterned by Lemma 5.5. By Lemma 5.1 and Lemma 5.5, we have $\sigma(A + BK) = \bigcup_{i=1}^{h} \bigcup_{j=1}^{n_i} \sigma(\hat{A}_i + \hat{B}_i\hat{K}_i)$. By Definition 5.2, $\mathcal{L} = \bigcup_{i=1}^{h} \bigcup_{j=1}^{n_i} \mathcal{L}_i$. Thus $\bigcup_{i=1}^{h} \bigcup_{j=1}^{n_i} \sigma(\hat{A}_i + \hat{B}_i\hat{K}_i) = \bigcup_{i=1}^{h} \bigcup_{j=1}^{n_i} \mathcal{L}_i$, which implies that $\bigcup_{j=1}^{n_i} \sigma(\hat{A}_i + \hat{B}_i\hat{K}_i) = \bigcup_{j=1}^{n_i} \mathcal{L}_i$ must be true for $i = 1, \ldots, h$. This implies that $\sigma(\hat{A}_i + \hat{B}_i\hat{K}_i) = \mathcal{L}_i$. Since $\mathcal{L}_i$ is an arbitrary spectrum, this equation is equivalent to controllability of the reduced subsystem $(\hat{A}_i, \hat{B}_i)$ for all $i = 1, \ldots, h$. By Remark 5.1, this is exactly the condition for pattern controllability of $(A, B)$.

$(\impliedby)$ Now assume $(A, B)$ is pattern controllable. By Remark 5.1, this implies that the $(\hat{A}_i, \hat{B}_i)$ are controllable for all $i = 1, \ldots, h$. Thus, for each $(\hat{A}_i, \hat{B}_i)$, there exists a $\hat{K}_i$ such that $\sigma(\hat{A}_i + \hat{B}_i\hat{K}_i) = \mathcal{L}_i$,
where $L_i$ is an arbitrary spectrum. Then $\bigcup_{i=1}^{n_i} \sigma(\hat{A}_i + \hat{B}_i \hat{K}_i) = \bigcup_{j=1}^{n_j} L_j$ must be true for $i = 1, ..., h$. Forming the union of these spectra gives $\bigcup_{i=1}^{n_i} \bigcup_{j=1}^{n_j} \sigma(\hat{A}_i + \hat{B}_i \hat{K}_i) = \bigcup_{i=1}^{n_i} \bigcup_{j=1}^{n_j} L_i$ which implies that $\sigma(A + BK) = \mathcal{L}$. Furthermore, we note that for each $\hat{K}_i$ we can define a matrix $\tilde{K}_i = \hat{K}_i \otimes I_{n_i}$ and form the direct sum $\tilde{K} = \bigoplus_{i=1}^{h} \tilde{K}_i$. Applying Theorem 4.1, we see that $K = T_u^{-1} \tilde{K} T_X$ such that $K : \mathcal{X} \rightarrow \mathcal{U}$ and that $K$ is $G$-patterned by construction.

Thus, if the system is pattern controllable, we will be able to use a patterned feedback to place all of the poles of the patterned spectrum. By Lemma 5.1, this implies that all of the poles of the system can be placed arbitrarily, albeit with some necessary multiplicity resulting from the pattern preservation. Note that the second part of the proof also provides a method by which a patterned feedback $K$ can be constructed.

### 5.2 Stabilizability

We can define a notion of stabilizability as follows,

**Definition 5.5 (Pattern Stabilizability).** The $G$-patterned system $(A, B)$ is **pattern stabilizable** if there exists a $G$-patterned matrix $K \in \mathbb{R}^{m \times n}$ such that $\sigma(A + BK) \subset \mathbb{C}^{-}$.

**Theorem 5.2.** The $G$-patterned system $(A, B)$ is pattern stabilizable if and only if the reduced form $(\hat{A}, \hat{B})$ is stabilizable in the standard linear control sense.

**Proof.** ($\Rightarrow$) Suppose that we have a $G$-patterned feedback $K$ such that $\sigma(A + BK) \subset \mathbb{C}^{-}$. By Lemma 5.1 and Lemma 5.5 we have that $\sigma(A + BK) = \bigcup_{i=1}^{n_i} \bigcup_{j=1}^{n_j} \sigma(\hat{A}_i + \hat{B}_i \hat{K}_i) \subset \mathbb{C}^{-}$ for all $i = 1, ..., h$. This of course implies that the reduced system $(\hat{A}_i, \hat{B}_i)$ is stabilizable in the standard linear control sense. By the block diagonal structure of $\hat{A}$ and $\hat{B}$, stabilizability of $(\hat{A}_i, \hat{B}_i)$ for all $i = 1, ..., h$ is equivalent to stabilizability of $(\hat{A}, \hat{B})$.

($\Leftarrow$) As above, stabilizability of $(\hat{A}, \hat{B})$ is equivalent to stabilizability of $(\hat{A}_i, \hat{B}_i)$ for all $i = 1, ..., h$. Thus, for each $i = 1, ..., h$, there exists a feedback matrix $\hat{K}_i \in \mathbb{R}^{n_i \times n_X}$ such that $\sigma(\hat{A}_i + \hat{B}_i \hat{K}_i) \subset \mathbb{C}^{-}$. Then, we note that for each $\hat{K}_i$ we can define a matrix $\tilde{K}_i = \hat{K}_i \otimes I_{n_i}$ and form the direct sum $\tilde{K} = \bigoplus_{i=1}^{h} \tilde{K}_i$. Applying Theorem 4.1, we see that $K = T_u^{-1} \tilde{K} T_X$ such that $K : \mathcal{X} \rightarrow \mathcal{U}$ and that $K$ is $G$-patterned by construction. As above, $\sigma(A + BK) = \bigcup_{i=1}^{n_i} \bigcup_{j=1}^{n_j} \sigma(\hat{A}_i + \hat{B}_i \hat{K}_i) \subset \mathbb{C}^{-}$, as required. □
5.2.1 Example

We present a system that has the general structure of a two level binary tree. We define the system matrices as follows:

\[
A = \begin{bmatrix}
A_1 & A_4 & A_4 & 0 & 0 & 0 & 0 \\
A_4 & A_2 & 0 & A_4 & A_4 & 0 & 0 \\
A_4 & 0 & A_2 & 0 & 0 & A_4 & A_4 \\
0 & A_4 & 0 & A_3 & 0 & 0 & 0 \\
0 & A_4 & 0 & 0 & A_3 & 0 & 0 \\
0 & 0 & A_4 & 0 & 0 & 0 & A_3 \\
0 & 0 & A_4 & 0 & 0 & 0 & A_5 \\
\end{bmatrix}
\]

\[
A_1 = \begin{bmatrix} 2 & 2 \\ 4 & 6 \end{bmatrix}; \quad A_2 = \begin{bmatrix} -1 & 2 \\ 3 & -4 \end{bmatrix}; \\
A_3 = \begin{bmatrix} -7 & 8 \\ 3 & 2 \end{bmatrix}; \quad A_4 = \begin{bmatrix} -5 & 0 \\ 0 & 3 \end{bmatrix};
\]

\[
B = \begin{bmatrix}
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
\end{bmatrix}^T
\]

As explained in Section 2.4, the system graph is:

![Figure 5.1: Stabilization Example System Graph - 7 Subsystem Two Level Tree](image-url)
Chapter 5. Patterned Controllability and Observability

In the graph, the $X_i$ represent the subsystems with the directed arrows representing the interactions between them via the $A$ matrix. The inputs $u_j$ represent the inputs from the matrix $B$. The symmetry or pattern of the system is immediately apparent upon viewing the graph representation. However, for this and more complicated systems there exists software to aid in determining the full group of automorphic permutation operations ($G$) along with their matrix representations $\rho^x(g)$ and $\rho^u(g)$. To find the permissible permutations we use NAUTY [32] and for the remainder of the group theoretic information, we rely on the well known Groups, Algorithms, Programming (GAP) System for Computational Discrete Algebra [28]. The symmetry inherent within this system was found to correspond to the dihedral $D_8$ group. This group is commonly denoted as follows: $G = \{e, a, a^2, a^3 = a^{-1}, x, ax, a^2x, a^3x\}$ where $x$ and $a$ are such that $a^4 = x^2 = e$ and $xax^{-1} = a^{-1}$ [35]. The system is, by definition, $G$-patterned with the $D_8$ group. For the purpose of exposition we demonstrate the state space and input space representations of a couple of the group elements (note that these are the permutation matrices corresponding to the permutations found via the procedure shown in Section 2.4).

$$
\rho^x(ax) = \begin{bmatrix}
I_2 & 0 & 0 & 0 & 0 \\
0 & I_2 & 0 & 0 & 0 \\
0 & 0 & I_2 & 0 & 0 \\
0 & 0 & 0 & I_2 & 0 \\
0 & 0 & 0 & 0 & I_2 \\
\end{bmatrix};
\rho^x(a^2x) = \begin{bmatrix}
I_2 & 0 & 0 & 0 & 0 \\
0 & I_2 & 0 & 0 & 0 \\
0 & 0 & I_2 & 0 & 0 \\
0 & 0 & 0 & I_2 & 0 \\
0 & 0 & 0 & 0 & I_2 \\
\end{bmatrix};
\rho^x(ax) = \begin{bmatrix}
1 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 1 \\
\end{bmatrix};
\rho^x(a^2x) = \begin{bmatrix}
1 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 1 \\
0 & 0 & 1 & 0 & 0 \\
\end{bmatrix}.
$$

It can be verified that the system matrices commute with these representations as prescribed by the definition of a $G$-patterned system. From the GAP software we know that there are 5 irreducible representations that pertain to the $D_8$ group, denoted here as $\tilde{\rho}_1(g), \tilde{\rho}_2(g), \tilde{\rho}_3(g), \tilde{\rho}_4(g), \tilde{\rho}_5(g)$. The first four representations are one dimensional ($n_1 = n_2 = n_3 = n_4 = 1$), while the fifth has 2 dimensions ($n_5 = 2$). Additionally, we determine the irreducible representations that compose the two representations above, along with the multiplicities of each irreducible representation. The multiplicities of the irreducible representations composing $\rho^x(g)$ are as follows: $\eta_1 = 6, \eta_2 = 0, \eta_3 = 0, \eta_4 = 4, \eta_5 = 2$. The multiplicities of the irreducible representations composing $\rho^u(g)$ are as follows: $\eta_1 = 2, \eta_2 = 0, \eta_3 = 0, \eta_4 = 1, \eta_5 = 1$.

Knowing this, we can determine the irreducible decomposition transformations $T_x$ and $T_u$ such that,

$$
\tilde{\rho}^x(g) = T_x^{-1} \rho^x(g) T_x = \text{diag} \left( \tilde{\rho}_1(g), \ldots, \tilde{\rho}_1(g), \tilde{\rho}_2(g), \ldots, \tilde{\rho}_4(g), \tilde{\rho}_5(g), \tilde{\rho}_5(g) \right) \quad (\text{6 times})
$$

$$
\tilde{\rho}^u(g) = T_u^{-1} \rho^u(g) T_u = \text{diag} (\tilde{\rho}_1(g), \tilde{\rho}_1(g), \tilde{\rho}_1(g), \tilde{\rho}_5(g)).
$$
Now we can apply the decomposition matrices to the system matrices to give the following pattern decomposed forms of $A$ and $B$,

\[
\tilde{A} = T_X^{-1}AT_X = \text{diag} \left( \tilde{A}_1, \tilde{A}_4, \tilde{A}_5 \right)
\]

\[
\tilde{A}_1 = \begin{bmatrix}
3 & 2 & 7/2 & 0 & 0 \\
0 & 7/2 & -1 & 2 & -7/2 \\
0 & 0 & 3 & -4 & 0 \\
0 & 0 & 0 & 7/2 & 2 \\
0 & 0 & 0 & 0 & 4/24
\end{bmatrix}; \quad \tilde{A}_4 = \begin{bmatrix}
-1 & 2 & -7/2 & 0 & 4/24 \\
0 & 9/2 & 8 & 0 & 2 \\
0 & 0 & 9/2 & 0 & 3 \\
0 & 0 & 0 & 9/2 & 2 \\
0 & 0 & 0 & 0 & 4/24
\end{bmatrix}; \quad \tilde{A}_5 = \begin{bmatrix}
2 & 0 & 3 & 0 & 0 \\
8/3 & 0 & 3 & 0 & 0 \\
0 & 0 & 0 & 0 & -7
\end{bmatrix}
\]

\[
\tilde{B} = T_X^{-1}BT_X = \text{diag} \left( \tilde{B}_1, \tilde{B}_4, \tilde{B}_5 \right)
\]

\[
\tilde{B}_1 = \begin{bmatrix}
0 \\
0 \\
0 \\
0 \\
0
\end{bmatrix}; \quad \tilde{B}_4 = \begin{bmatrix}
0 \\
0 \\
0 \\
0 \\
1
\end{bmatrix}; \quad \tilde{B}_5 = \begin{bmatrix}
-1 & 0 & 0 & 1 & 0 \\
0 & 0 & -1 & 0 & 1 \\
0 & 0 & 0 & 0 & 0
\end{bmatrix}
\]

Employing Theorem 4.1, we note that the reduced forms of $\tilde{A}$ and $\tilde{B}$ are as follows,

\[
\tilde{A}_1 = \hat{A}_1; \quad \tilde{A}_4 = \hat{A}_4; \quad \tilde{A}_5 = \begin{bmatrix} 2 & 3 \\ 3 & 7 \end{bmatrix}
\]

\[
\tilde{B}_1 = \hat{B}_1; \quad \tilde{B}_4 = \hat{B}_4; \quad \tilde{B}_5 = \begin{bmatrix} -1 \\ 0 \end{bmatrix}
\]

Each patterned-reduced subsystem $(\hat{A}, \hat{B})$ is controllable which implies that the entire system is pattern controllable. As a result we can place all of the eigenvalues of the patterned spectrum. We assign the spectra as follows: $L_1 = \{-1, -2, -3, -4, -5, -6\}, L_4 = \{-7, -8, -9, -10\}, L_5 = \{-11, -12\}$. We use the MATLAB \textit{place} function with the reduced matrices to find reduced feedback matrices,

\[
\hat{K}_1 = \begin{bmatrix}
20.8369 & 1.7377 & 10.5936 & 0.0816 & -19.3971 & -3.1155
\end{bmatrix}
\]

\[
\hat{K}_4 = \begin{bmatrix}
46.475 & 7.7522 & -32.3612 & -24
\end{bmatrix}
\]

\[
\hat{K}_5 = \begin{bmatrix}
18 & 5.5
\end{bmatrix}
\]

Finally we put the feedback matrices in their unreduced form, collect them in a single matrix as $\hat{K} = \text{diag} \left( \hat{K}_1, \hat{K}_4, \hat{K}_5 \otimes I_2 \right)$. We now reverse the pattern decomposition, which gives $K = T_X \hat{K} T_X^{-1}$. The resulting $K$ places the patterned spectrum and is, of course, $G$-patterned by Theorem 4.1. The resulting spectrum of the closed loop system is as follows:

\[
\sigma(A + BK) = \{-1, -2, -3, -4, -5, -6, -7, -8, -9, -10, -11, -12, -12\}
\]

As expected, we see repetition of the eigenvalues of $L_5$. 

\textit{Chapter 5. Patterned Controllability and Observability} 63
5.3 Observability

In this section, we adapt the main theorems of observability to our patterned system framework. It should come as no surprise that the results developed for controllability and stabilization of patterned systems can be extended to notions of observability. We employ the same duality strategies used to prove observability theorems in the standard case. For the sake of brevity, the majority of proofs are simply referred to the equivalent proofs in the previous section. It is assumed that the reader is familiar with standard linear observer theory.

Consider the system defined in (5.1). We also denote the observer matrix $G$ with standard linear observer theory. This condition is sufficient to demonstrate pattern observability.

**Lemma 5.6.** $(C, A)$ is observable if and only if $(A^T, C^T)$ is controllable.

We now provide a new definition of observability for G-patterned systems.

**Definition 5.6.** Suppose we have a $G$-patterned system $(C, A)$. Then the pair $(C, A)$ is pattern observable if the reduced form $(\hat{C}, \hat{A})$ is observable in the standard linear control theory sense.

**Remark 5.2.** As before, we note that the definition of pattern observability above is equivalent to observability of each of the reduced subsystems $(\hat{C}_i, \hat{A}_i)$ for all $i = 1, \ldots, h$ in the standard linear control theory sense. This condition is sufficient to demonstrate pattern observability.

**Remark 5.3.** By Lemma 5.6 above, a $G$-patterned system is pattern observable if and only if each reduced subsystem $(\hat{A}_i^T, \hat{C}_i^T)$ is controllable in the standard linear control sense for all $i = 1, \ldots, h$.

The following is the main theorem of observability adapted to our $G$-patterned framework.

**Theorem 5.3 (Patterned Observability).** Let $(C, A)$ be $G$-patterned system and let $\mathcal{L}$ be an arbitrary $G$-patterned spectrum. Then, there exists a $G$-patterned matrix $L \in \mathbb{R}^{n \times p}$ such that $\sigma(A + LC) = \mathcal{L}$ if and only if $(C, A)$ is pattern observable.

**Proof.** ($\implies$) Suppose we have a $G$-patterned matrix $L$ such that $\sigma(A + LC) = \mathcal{L}$. Since $A$, $B$, and $L$ are patterned, we know that $A + LC$ is patterned by the Lemmas 4.1 and 4.1. By Lemma 5.1 and Lemma 5.5 (replacing the appropriate matrices), we know that the patterned spectrum can be written as a union of the reduced subsystem spectra, $\sigma(A + LC) = \bigcup_{i=1}^{h} \bigcup_{j=1}^{n_i} \sigma(\hat{A}_i + \hat{L}_i \hat{C}_i)$ and by Definition 5.2, $\mathcal{L} = \bigcup_{i=1}^{h} \bigcup_{j=1}^{n_i} \mathcal{L}_i$. Thus $\bigcup_{i=1}^{h} \bigcup_{j=1}^{n_i} \sigma(\hat{A}_i + \hat{L}_i \hat{C}_i) = \bigcup_{i=1}^{h} \bigcup_{j=1}^{n_i} \mathcal{L}_i$, which implies that $\bigcup_{i=1}^{h} \bigcup_{j=1}^{n_i} \sigma(\hat{A}_i + \hat{L}_i \hat{C}_i) = \bigcup_{i=1}^{h} \bigcup_{j=1}^{n_i} \mathcal{L}_i$, must be true for $i = 1, \ldots, h$. This implies that $\sigma(\hat{A}_i + \hat{L}_i \hat{C}_i) = \mathcal{L}_i$. Since the spectrum of a matrix is invariant to the transpose operation, we have $\sigma(\hat{A}_i^T + \hat{C}_i^T \hat{L}_i^T) = \mathcal{L}_i$. Since $\mathcal{L}_i$ is an arbitrary spectrum, this equation is equivalent to controllability of the reduced subsystem $(\hat{A}_i^T, \hat{C}_i^T)$ for all $i = 1, \ldots, h$. By Remark 5.3, this is exactly the condition for pattern observability of $(C, A)$. 


(⇐) Now assume \((C, A)\) is pattern observable, which implies that the \((\hat{A}_i^T, \hat{C}_i^T)\) are controllable for all \(i = 1, ..., h\). Thus, for each \((\hat{A}_i^T, \hat{C}_i^T)\), there exists a \(\hat{L}_i\) such that \(\sigma(\hat{A}_i^T + \hat{C}_i^T \hat{L}_i) = \mathcal{L}_i\), where \(\mathcal{L}_i\) is an arbitrary spectrum. By transposing, we get \(\sigma(\hat{A}_i + \hat{L}_i \hat{C}_i) = \mathcal{L}_i\). Then \(\bigcup_{i=1}^h \bigcup_{j=1}^{n_i} \sigma(\hat{A}_i + \hat{L}_i \hat{C}_i) = \bigcup_{i=1}^h \bigcup_{j=1}^{n_i} \mathcal{L}_i\) must be true for \(i = 1, ..., h\). Forming the union of the aforementioned spectra, \(\mathcal{L} = \bigcup_{i=1}^h \bigcup_{j=1}^{n_i} \mathcal{L}_i\) and form the direct sum \(\hat{L} = \bigoplus_{i=1}^h \hat{L}_i\). Applying Theorem 4.1, we see that \(L = T_{\mathcal{X}}^{-1} \hat{L} T_{\mathcal{Y}}\) such that \(L : \mathcal{Y} \rightarrow \mathcal{X}\) and that \(L\) is \(G\)-patterned by construction.

This theorem allows to place the poles of a given observer matrix \(L\) with the guarantee that this matrix is \(G\)-patterned, as is the closed loop system that governs the observer dynamics. Note that, as in the standard pole placement theorem, we are required to place the poles of the observer systems with some multiplicity of given poles. As in the controllability case, the second part of the proof above provides a method by which we may construct a \(G\)-patterned observer matrix \(L\).

### 5.4 Detectability

We define detectability in the pattern framework as follows:

**Definition 5.7** (Pattern Detectability). The \(G\)-patterned system \((C, A)\) is pattern detectable if there exists a \(G\)-patterned matrix \(L \in \mathbb{R}^{n \times p}\) such that \(\sigma(A + LC) \subset \mathbb{C}^-\).

We now present conditions under which a \(G\)-patterned system is detectable.

**Theorem 5.4.** The \(G\)-patterned system \((C, A)\) is pattern detectable if and only if the reduced form \((\hat{C}, \hat{A})\) is detectable in the standard linear control sense.

**Proof.** We first note that, by the block diagonal structure of \(\hat{C}\) and \(\hat{A}\), detectability of \((\hat{C}, \hat{A})\) is equivalent to detectability of \((\hat{C}_i, \hat{A}_i)\) for all \(i = 1, ..., h\). By Lemma 5.1 and Lemma 5.5 (replacing the appropriate matrices) we have that \(\sigma(A + LC) = \bigcup_{i=1}^h \bigcup_{j=1}^{n_i} \sigma(\hat{A}_i + \hat{L}_i \hat{C}_i)\) for all \(i = 1, ..., h\). Therefore, \(\sigma(A + LC) \subset \mathbb{C}^-\) if and only if \(\bigcup_{i=1}^h \bigcup_{j=1}^{n_i} \sigma(\hat{A}_i + \hat{L}_i \hat{C}_i) \subset \mathbb{C}^-\), which is clearly true if and only if, for all \(i = 1, ..., h\), \(\sigma(\hat{A}_i + \hat{L}_i \hat{C}_i) \subset \mathbb{C}^-\), where \(\hat{L}_i \in \mathbb{R}^n_{\mathcal{X} \times \mathcal{Y}}\). This is, of course, the definition of stabilizability for \((\hat{C}_i, \hat{A}_i)\) in the standard linear control sense. \(\square\)
Chapter 6

Patterned Stabilization by Measurement Feedback

We now approach some more complicated linear control problems from the perspective of the patterned linear control framework. In this chapter, we will investigate the stabilization of a patterned system using information only available from a specific measurement. As we will see, the proof is fairly straightforward and leverages a previously developed lemma regarding the kernel spaces of patterned matrices.

6.1 Patterned Stabilization by Measurement Feedback Problem

Consider the system (5.1). We define the following problem:

**Definition 6.1** (Stabilization By Measurement Feedback Problem (SMFP)). Find a feedback $u = Kx$ such that

$$\text{Ker}(C) \subset \text{Ker}(K)$$  \hspace{1cm} (6.1)

$$\sigma(A + BK) \subset \mathbb{C}^-.$$  \hspace{1cm} (6.2)

**Theorem 6.1** (Patterned Stabilization by Measurement Feedback Problem). Given a $G$-patterned system (5.1), the SMFP is solved with a patterned feedback $K$ if and only if the SMFP is solved for the reduced system $(\hat{C}, \hat{A}, \hat{B})$.

**Proof.** ($\Rightarrow$) Suppose there exists a patterned $K$ that solves the SMFP. That is, $\text{Ker}(C) \subset \text{Ker}(K)$, $\sigma(A + BK) \subset \mathbb{C}^-$, and $\rho^H(g)K = K\rho^V(g)$, for all $g \in G$. By Lemma 4.12, the first condition is true if and only if $\text{Ker}(\hat{C}) \subset \text{Ker}(\hat{K})$. Furthermore, by Lemmas 5.1 and 5.5, $\sigma(\hat{A} + \hat{B}\hat{K}) \subset \mathbb{C}^-$ (as in the proof
of Theorem 5.2). Thus, the SMFP for the reduced system \((\hat{C}, \hat{A}, \hat{B})\) is solved by \(\hat{K}\).

(\iff) Suppose there exists a feedback \(\hat{K}\) such that \(\sigma(\hat{A} + \hat{B}\hat{K}) \subset \mathbb{C}^{-}\) and \(\text{Ker}(\hat{C}) \subset \text{Ker}(\hat{K})\). Since \((\hat{C}, \hat{A}, \hat{B})\) are all block diagonal, without loss of generality we can take \(\hat{K} = \bigoplus_{i=1}^{h} \hat{K}_i\). Then, we note that for each \(\hat{K}_i\) we can define a matrix \(\tilde{K}_i = \hat{K}_i \otimes I_{n_i}\) and form the direct sum \(\tilde{K} = \bigoplus_{i=1}^{h} \tilde{K}_i\). Applying Theorem 4.1, we see that \(K = T_{u}^{-1} \tilde{K} T_{x}\) such that \(K : X \rightarrow U\) and that \(K\) is \(G\)-patterned by construction; that is, \(\rho^{U}(g)K = K \rho^{X}(g)\). By Lemma 4.12, \(\text{Ker}(\hat{C}) \subset \text{Ker}(\hat{K})\) implies \(\text{Ker}(C) \subset \text{Ker}(K)\). Furthermore, by Lemmas 5.1 and 5.5, \(\sigma(\hat{A} + \hat{B}\hat{K}) \subset \mathbb{C}^{-}\) implies \(\sigma(A + BK) \subset \mathbb{C}^{-}\). Therefore the SMFP of \((C, A, B)\) is solved by the \(G\)-patterned feedback \(K\).

\[\text{Example 6.1.}\] We consider the system given by the following \(A, B,\) and \(C\) matrices.

\[
A = \begin{bmatrix}
-2.2 & 5.0 & -0.37 & 0 & 0 & 0 & 0 & 0 & -0.37 & 0 \\
0 & 5.4 & 0 & -0.49 & 0 & 0 & 0 & 0 & 0 & -0.49 \\
-0.37 & 0 & -2.2 & 5.0 & -0.37 & 0 & 0 & 0 & 0 & 0 \\
0 & -0.49 & 0 & 5.4 & 0 & -0.49 & 0 & 0 & 0 & 0 \\
0 & 0 & -0.37 & 0 & -2.2 & 5.0 & -0.37 & 0 & 0 & 0 \\
0 & 0 & 0 & -0.49 & 0 & 5.4 & 0 & -0.49 & 0 & 0 \\
0 & 0 & 0 & 0 & -0.37 & 0 & -2.2 & 5.0 & -0.37 & 0 \\
0 & 0 & 0 & 0 & 0 & -0.49 & 0 & 5.4 & 0 & -0.49 \\
-0.37 & 0 & 0 & 0 & 0 & 0 & -0.37 & 0 & -2.2 & 5.0 \\
0 & -0.49 & 0 & 0 & 0 & 0 & 0 & -0.49 & 0 & 5.4
\end{bmatrix}
\]

\[B = \begin{bmatrix}
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}
\]

\[C = \begin{bmatrix}
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0
\end{bmatrix}
\]

We divide the state space into subsystems of dimension 2 and observe that this system exhibits a block circulant pattern [14]. This patterning becomes even more apparent when we consider the system graph:
Note that, in the above system matrix, $A_1 = \begin{bmatrix} -2 & 5.0 \\ 0 & 5.4 \end{bmatrix}$, $A_2 = \begin{bmatrix} -0.37 & 0 \\ 0 & -0.49 \end{bmatrix}$, $B_1 = [0]$, and $C_1 = [0 \ 1]$. Using the NAUTY and GAP software we determine the automorphism group of the system graph and the relevant group theory information. We give two examples of the representations (permutation matrices) that commute with the $A$ matrix of this system. In fact, these elements the generators of the automorphism group (see Definition A.1 in Appendix A):

$$
\rho^X(g_1) = \begin{bmatrix}
0 & 0 & 0 & I_2 & 0 \\
0 & 0 & I_2 & 0 & 0 \\
0 & I_2 & 0 & 0 & 0 \\
I_2 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & I_2
\end{bmatrix}, \quad \rho^X(g_2) = \begin{bmatrix}
0 & I_2 & 0 & 0 & 0 \\
I_2 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & I_2 \\
0 & 0 & I_2 & 0 & 0 \\
0 & 0 & I_2 & 0 & 0
\end{bmatrix},
$$

where $I_2$ is the two dimensional identity matrix. For this group, there are four irreducible representations with dimensions $n_1 = n_2 = 1$ and $n_3 = n_4 = 2$. Using MATLAB code detailed in Appendix B, we determine that the multiplicities of the irreducibles on each vector space are as follows: $\eta_1^X = 2$, $\eta_2^X = 0$, $\eta_3^X = 2$, $\eta_4^X = 2$, $\eta_1^\ell = 2$, $\eta_2^\ell = 0$, $\eta_3^\ell = 1$, $\eta_4^\ell = 1$, $\eta_1^Y = 1$, $\eta_2^Y = 0$, $\eta_3^Y = 1$, and $\eta_4^Y = 1$. The irreducible transformation matrices were also calculated by the MATLAB code. Recall that these matrices are randomly generated subject to given constraints. We show the transformations for proof of concept:

$$
T_X = \begin{bmatrix}
-0.00651 & 0 & -0.116 & 0.357 & 0 & -0.254 & 0.185 & 0 & 0 \\
0 & -0.00651 & 0 & 0 & -0.116 & 0 & 0 & -0.254 & 0.185 \\
0 & 0 & 0.304 & 0.221 & 0 & 0 & 0.0971 & -0.299 & 0 \\
0 & 0 & 0 & 0.304 & 0.221 & 0 & 0 & 0.0971 & -0.299 \\
-0.00651 & 0 & 0 & 0 & 0.304 & -0.221 & 0 & 0 & 0.0971 \\
0 & -0.00651 & 0 & 0 & 0 & 0 & 0.0971 & -0.299 & 0 \\
-0.00651 & 0 & -0.116 & 0.357 & 0 & -0.116 & -0.357 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.314 & 3.52 \times 10^{-17} \\
-0.00651 & 0 & -0.375 & 3.09 \times 10^{-16} & 0 & 0 & 0 & 0 & 0.314 & 3.52 \times 10^{-17}
\end{bmatrix}
$$
we note that the closed loop matrix commutes with the permutations of the automorphism group given
This matrix is patterned by the automorphism group of the system graph shown above. As expected,
Applying these transformations we obtain the following reduced subsystems:

\[
\begin{align*}
\hat{A}_1 &= \begin{bmatrix} -2.94 & 5.0 \\ 0 & 4.41 \end{bmatrix}, & \hat{B}_1 &= \begin{bmatrix} 0 \\ 14.87 \end{bmatrix}, & \hat{C}_1 &= \begin{bmatrix} 0 & 0.039 \end{bmatrix}; \\
\hat{A}_3 &= \begin{bmatrix} -2.43 & 5.0 \\ 0 & 5.09 \end{bmatrix}, & \hat{B}_3 &= \begin{bmatrix} 0 \\ 0.8 \end{bmatrix}, & \hat{C}_3 &= \begin{bmatrix} 0 & 1.56 \end{bmatrix}; \\
\hat{A}_4 &= \begin{bmatrix} -1.6 & 5 \\ 0 & 6.2 \end{bmatrix}, & \hat{B}_4 &= \begin{bmatrix} 0 \\ 0.47 \end{bmatrix}, & \hat{C}_4 &= \begin{bmatrix} 0 & 0.75 \end{bmatrix}.
\end{align*}
\]
Furthermore, we see that the following reduced form feedback matrices solve the SMFP on each system
are as follows:

\[
\hat{K}_1 = \begin{bmatrix} 0 & -7 \end{bmatrix}; \quad \hat{K}_3 = \begin{bmatrix} 0 & -10 \end{bmatrix}; \quad \hat{K}_4 = \begin{bmatrix} 0 & -12 \end{bmatrix}.
\]

Clearly, we have \(\text{Ker}(\hat{C}_i) \subset \text{Ker}(\hat{K}_i)\) for \(i = 1, 3, 4\) (\(i = 2\) is not considered since the multiplicity of this irreducible is zero for this system). We reverse the decomposition procedure on to yield the following feedback matrix:

\[
K = T_{t_i}(\hat{K}_1 \oplus \hat{K}_3 \oplus I_2 \oplus \hat{K}_4 \oplus I_2) = \\
\begin{bmatrix}
0 & -26.3 & 0 & -20.0 & 0 & -18.9 & 0 & -18.9 & 0 & -20.0 \\
0 & -20.0 & 0 & -26.3 & 0 & -20.0 & 0 & -18.9 & 0 & -18.9 \\
0 & -18.9 & 0 & -20.0 & 0 & -26.3 & 0 & -20.0 & 0 & -18.9 \\
0 & -18.9 & 0 & -18.9 & 0 & -20.0 & 0 & -26.3 & 0 & -20.0 \\
0 & -20.0 & 0 & -18.9 & 0 & -18.9 & 0 & -20.0 & 0 & -26.3
\end{bmatrix}.
\]

We note that \(\text{Ker}(C) \subset \text{Ker}(K)\) and is stabilizing as desired. Furthermore, the closed loop system is
given as follows:

\[
A + BK = \\
\begin{bmatrix}
-2.2 & 5.0 & -0.371 & 0 & 0 & 0 & 0 & 0 & -0.371 & 0 \\
-0.371 & 0 & -2.2 & 5.0 & -0.371 & 0 & 0 & 0 & 0 & 0 \\
0 & -20.5 & 0 & -20.5 & 0 & -18.9 & 0 & -18.9 & 0 & -20.5 \\
0 & -18.9 & 0 & -20.5 & 0 & -20.9 & 0 & -18.9 & 0 & -20.5 \\
0 & 0 & 0 & 0 & -0.371 & 0 & -2.2 & 5.0 & -0.371 & 0 \\
0 & -18.9 & 0 & -20.5 & 0 & -20.9 & 0 & -20.9 & 0 & -20.5 \\
-0.371 & 0 & 0 & 0 & 0 & -0.371 & 0 & -2.2 & 5.0 & 0 \\
0 & -20.5 & 0 & -18.9 & 0 & -18.9 & 0 & -20.5 & 0 & -20.9
\end{bmatrix}.
\]
This matrix is patterned by the automorphism group of the system graph shown above. As expected,
we note that the closed loop matrix commutes with the permutations of the automorphism group given
above. The MATLAB code used to solve this problem is given in Appendix C.
Chapter 7

Patterned Output Stabilization

Problem

In this chapter we will address the output stabilization problem and adapt it to our patterned control framework. Since the concept of controlled invariance is essential to the solution of the output stabilization problem, we first develop a notion of controlled invariance for a patterned system under a patterned feedback. We propose an algorithm that can be used to identify an appropriate supremal controlled invariant space. With these tools in hand, we proceed to provide a solution of the patterned output stabilization problem.

7.1 Patterned Controlled Invariance

In this section we show that a patterned subspace that is also controlled invariant can be made invariant using a feedback matrix $K$ that is patterned in the same way. Furthermore, as in previous results shown in this document, we can synthesize the patterned feedback matrix $K$ through the patterned decomposition of the system, followed by the standard synthesis procedure for controlled invariance.

7.1.1 Controlled Invariant Patterned Subspaces

A subspace $V \subset X$ is called a controlled invariant subspace for a system (5.1) if there exists a map $K : X \rightarrow U$ such that $V$ is $(A + BK)$-invariant [12]. We refer to $K$ as a "friend of $V$". We define the class of controlled invariant subspaces of $X$ for a given linear system and denote this class by $\mathcal{J}(X)$. More generally, given a subspace $K \subset X$, we define the following.

$$\mathcal{J}(K) := \{ V \in \mathcal{J}(X) : V \subset K \}.$$
We now define a new class of controlled invariant *patterned subspaces*, denoted,

\[ \mathcal{J}_G(K) := \{ V \in \mathcal{J}(K) : \rho^X(g)V \subset V, \ \forall g \in G \}. \]

Note that the only modification here is that we require that \( V \) be a \( G \)-patterned subspace.

It is well-known (See Lemma 4.2 of Wonham [12]) that controlled invariant subspaces can be characterized without explicit reference to a given feedback matrix \( K \). The following lemma extends this fact to *patterned controlled invariant subspaces*.

**Lemma 7.1.** Suppose \((A,B)\) is a \( G \)-patterned system. Let \( V \subset K \subset X \) be a \( G \)-patterned subspace such that \( V = \text{Im}(S) \), where \( S \) is a \( G \)-patterned insertion map. Then \( V \in \mathcal{J}_G(K) \) if and only if,

\[ \hat{A}V \subset \hat{V} + \hat{B}, \]

where \( \hat{V} = \text{Im}(\hat{S}) \) and \( \hat{B} = \text{Im}(\hat{B}) \).

**Proof.** Suppose \( V \in \mathcal{J}_G(K) \). By Lemma 4.2 of [12], we have \( AV \subset V + B \), where \( B = \text{Im}(B) \). Equivalently,

\[ \text{Im}(AS) \subset \text{Im}(W), \]

where \( W = [S \ B] \) and both \( W \) and \( AS \) are \( G \)-patterned by Lemma 4.1. By Lemma 4.11, we have,

\[ \text{Im}(\hat{A}S) \subset \text{Im}(\hat{W}). \]

Furthermore, by Lemma 4.2, we have,

\[ \hat{A} \text{Im}(\hat{S}) = \text{Im}(\hat{A}\hat{S}) \subset \text{Im} \left[ \hat{S} \ \hat{B} \right] = \text{Im}(\hat{S}) + \text{Im}(\hat{B}). \]

That is,

\[ \hat{A}\hat{V} \subset \hat{V} + \hat{B}, \]

as required. The converse proof is identical, but in reverse.

**Lemma 7.2.** If \( V \in \mathcal{J}_G(K) \), then there exists a 'friend' \( K \in \mathbb{R}^{n \times m} \) which is a \( G \)-patterned matrix.

**Proof.** If \( V \in \mathcal{J}_G(K) \), then we have that \( \hat{A}_i \cdot \text{Im}(\hat{S}_i) \subset \text{Im}(\hat{S}_i) + \text{Im}(\hat{B}_i) \) for all \( i = 1...h \) by the preceding Lemma. By definition, this implies that there are matrices \( \hat{K}_i \) such that \( (\hat{A}_i + \hat{B}_i\hat{K}_i)\text{Im}(\hat{S}_i) \subset \text{Im}(\hat{S}_i) \) for all \( i = 1...h \). We can rewrite this equation as \( \text{Im}((\hat{A}_i + \hat{B}_i\hat{K}_i)\hat{S}_i) \subset \text{Im}(\hat{S}_i) \), and apply Theorem 2.4 Part 2 to obtain \( \text{Im}((\hat{A}_i + \hat{B}_i\hat{K}_i)\hat{S}_i) \subset \text{Im}(\hat{S}_i) \), which implies that \( (\hat{A}_i + \hat{B}_i\hat{K}_i)\text{Im}(\hat{S}_i) \subset \text{Im}(\hat{S}_i) \) for all \( i = 1...h \). We can collect these equations and express them as \( (\hat{A} + \hat{B}\hat{K})\text{Im}(\hat{S}) \subset \text{Im}(\hat{S}) \) and, multiplying
by the appropriate irreducible decomposition transformation matrix $T_X$ we find that $(A + BK)V \subset V$. The feedback $K$ that renders the subspace $V$ invariant is therefore $G$-patterned by construction.

\subsection{Patterned Supremal Subspace Algorithm}

We define the supremal subspace of a class of subspaces as follows.

\begin{definition}
Let $\mathcal{J}(K)$ be a class of subspaces. Then the supremum of $\mathcal{J}(K)$, denoted $V^\ast = \sup \mathcal{J}(K)$, is the largest element of $\mathcal{J}(K)$. That is to say, for all $V \in \mathcal{J}(K)$, $V \subseteq V^\ast$.
\end{definition}

In order to show that $\mathcal{J}_G(K)$ has a supremum we must give the following lemma.

\begin{lemma}
The class of subspaces $\mathcal{J}_G(K)$ is closed under subspace addition.
\end{lemma}

\begin{proof}
Let $V_1, V_2 \in \mathcal{J}_G(K)$. Thus $V_1$ and $V_2$ are $G$-patterned and controlled invariant. By Lemma 4.8, $V_1 + V_2$ is $G$-patterned. Furthermore we have, $A(V_1 + V_2) = AV_1 + AV_2 \subset V_1 + V_2 + B$.
\end{proof}

The above implies that $\mathcal{J}_G(K)$ is a semi-lattice. We now show that $\mathcal{J}_G(K)$ has a supremum.

\begin{lemma}
$\mathcal{J}_G(K)$ has a supremal element $V^\ast$.
\end{lemma}

\begin{proof}
Since, by Lemma 7.3, $\mathcal{J}_G(K)$ is closed under addition, Lemma 4.4 of [12] states that $\mathcal{J}_G(K)$ has a supremal element $V^\ast$ and it is unique.
\end{proof}

From [12], we also know that the supremal controlled invariant subspace in a given class $\mathcal{J}(K)$ can be found via the following algorithm.

\begin{algorithm} [Supremal Controlled-Invariant Subspace]
The algorithm is as follows:
\begin{itemize}
  \item $V^0 = K$
  \item $V^k = K \cap (A^{-1}(V^{k-1} + B))$,
\end{itemize}
where $V^k \subseteq V^{k-1}$ and for some $k^* \leq \dim(K)$, $V^j = \sup \mathcal{J}(K)$ for all $j \geq k^*$.
\end{algorithm}

We now state the following lemma regarding the use of the aforementioned algorithm to find the \textit{patterned} supremal controlled invariant subspace.

\begin{lemma}
Let $(A, B)$ be a $G$-patterned system. If the subspace $K$ in Algorithm 7.1 is $G$-patterned, then the algorithm also yields the supremum of the patterned class, $\sup \mathcal{J}_G(K)$.
\end{lemma}

\begin{proof}
By [12], we know that the supremum resulting from Algorithm 7.1 is controlled invariant. It remains to be proven that the resulting subspace is $G$-patterned. We begin with a $G$-patterned subspace $K$ and iteratively perform operations of subspace addition, preimage and intersection. We know that both $A$ and $B$ are patterned. $B$ is patterned by definition, therefore Lemma 4.8 guarantees that, at each iteration, $V^{k-1} + B$ is patterned. Furthermore by Lemmas 4.10 and 4.9, the next subspace in the iteration
\[ V^k = \mathcal{K} \cap A^{-1}(V^{k-1} + \mathcal{B}) \] is also \( G \)-patterned. Therefore, by induction, each stage of the iteration yields a subspace that is \( G \)-patterned. The resulting supremum will also be \( G \)-patterned.

Alternatively, since \( \mathcal{K} \) is \( G \)-patterned, it is computationally and intuitively advantageous to first divide the system into the reduced subsystems and then find the supremum on the subsystems. The following lemma guarantees that this is possible.

**Lemma 7.6.** Let \((A, B)\) be a \( G \)-patterned system and let \( \mathcal{K} \sim \mathbb{R}^m \) and \( V^* \sim \mathbb{R}^p \) be \( G \)-patterned subspaces such that \( V^* \subset \mathcal{K} \subset X \sim \mathbb{R}^n \). Moreover, let the insertion map of \( \mathcal{K} \) be \( Q \in \mathbb{R}^{n \times m} \) and the insertion map of \( V^* \) be \( S^* \in \mathbb{R}^{n \times p} \). Then \( \dot{V}^*_i = \text{Im}(\hat{S}^*_i) = \sup \mathcal{J}(\hat{K}_i) \) for all \( i = 1, \ldots, h \) implies that \( V^* = \sup \mathcal{J}_G(\mathcal{K}) \), where \( \dot{V}^*_i = \text{Im}(\hat{S}^*_i), \hat{K}_i = \text{Im}(\hat{Q}^*_i) \) and \( \mathcal{J}(\hat{K}_i) \) is defined on reduced subsystem \((\hat{A}_i, \hat{B}_i)\).

**Proof.** Let \( V \in \mathbb{R}^l \) be a subspace such that \( V \in \sup \mathcal{J}_G(\mathcal{K}) \) and let \( S \in \mathbb{R}^{n \times l} \) be the insertion map of \( V \) such that \( \mathcal{V} = \text{Im}(S) \). By Lemma 7.1, we have that \( \hat{A}_i \dot{V}^*_i \subset \dot{V}^*_i + \hat{B}_i \) for all \( i = 1, \ldots, h \). Note that \( \mathcal{V} \subset \mathcal{K} \) is equivalent to \( \text{Im}(S) \subset \text{Im}(Q) \). By Lemma 4.11, this implies that, in terms of the reduced matrices, \( \text{Im}(\hat{S}_i) \subset \text{Im}(\hat{Q}_i) \) for all \( i = 1, \ldots, h \). Combining these facts, we know that \( \dot{V}^*_i \subset \mathcal{J}(\hat{K}_i) \) for all \( i = 1, \ldots, h \). Then, by the definition of supremum and by assumption, we have that \( \dot{V}^*_i \subset \mathcal{J}(\hat{K}_i) \) which implies \( \text{Im}(\hat{S}^*_i) \subset \text{Im}(\hat{S}^*_i) \). Again, by Lemma 4.11, this implies that \( \text{Im}(S) \subset \text{Im}(S^*) \), which is equivalent to \( \mathcal{V} \subset V^* \), as required. \( \square \)

**Remark 7.1.** As usual, we can collect these results for the reduced subsystem above into one result by arranging them into a block diagonal matrices as follows. We define \( \check{V}^* = \bigoplus_{i=1}^h \dot{V}^*_i = \text{Im}(\bigoplus_{i=1}^h \hat{S}^*_i) \) and \( \check{K} = \bigoplus_{i=1}^h \hat{K}_i = \text{Im}(\bigoplus_{i=1}^h \hat{Q}_i) \). Now, we note that \( \check{V}^* = \sup \mathcal{J}(\check{K}) \), where \( \mathcal{J}(\check{K}) \) is defined for the reduced form \((\hat{A}, \hat{B})\).

With the above lemmas in hand, we can find supremal patterned controlled invariant subspaces and are thus prepared to confront the Output Stabilization Problem.

### 7.2 Output Stabilization

Once again, we consider the linear system (5.1). It is a natural objective of control theory to control the dynamics of a specific set of outputs or measurements of this system. We characterize this set of outputs by the matrix \( D \in \mathbb{R}^{f \times n} \) such that \( z = Dx \), where \( z \in \mathcal{Z} \sim \mathbb{R}^f \) is the output vector and \( x \in \mathcal{X} \sim \mathbb{R}^n \) is the state vector. This problem is commonly referred to as the **output stabilization problem**, and is defined as follows.

**Definition 7.2** (Output Stabilization Problem (OSP)). Given the linear system (5.1), let \( z \in \mathcal{Z} \sim \mathcal{R}^f \) be such that \( z = Dx \), where \( D \in \mathbb{R}^{f \times n} \). Find a state feedback \( u = Kx \) such that,

\[
z(t) = De^{(A+BK)t} 
\longrightarrow 0 \quad \text{as} \quad t \longrightarrow \infty.
\] (7.1)
Chapter 7. Patterned Output Stabilization Problem

We also recall that the following theorem gives solvability conditions for OSP,

**Theorem 7.1.** OSP is solvable if and only if

\[ \mathcal{X}^+(A) \subset (A|B) + V^*, \tag{7.2} \]

where

\[ V^* = \sup \mathcal{J} (\text{Ker}(D)). \tag{7.3} \]

We now express the problem statement of the Patterned Output Stabilization Problem (POSP), that is, OSP in the case that the feedback is required to be patterned by a specific group \( G \). Obviously, we require some additional assumptions about the system.

**Definition 7.3** (Patterned Output Stabilization Problem (POSP)). Given a \( G \)-patterned system \((A, B)\) and a matrix \( D \) such that \( \text{Ker}(D) \) is \( G \)-patterned, find a \( G \)-patterned feedback \( u = Kx \) such that \( z(t) = De^{A+B\psi} \to 0 \) as \( t \to \infty \).

We find that this problem is solvable under a specific condition. Not surprisingly, this necessary and sufficient condition for solvability relies on the solvability of the standard OSP on the reduced form of system (i.e. on the reduced subsystems).

**Theorem 7.2.** Let \( G \) be a finite group and let \((A, B)\) be a \( G \)-patterned system. Also let \( \mathcal{K} = \text{Ker}(D) \) be a \( G \)-patterned subspace. Then the POSP is solvable if and only if the standard OSP is solvable on the reduced system \((\hat{A}, \hat{B})\).

**Proof.** (\( \Rightarrow \)) Suppose we have a \( G \)-patterned feedback that solves the POSP. Then \( K \) also solves the OSP, so by Theorem 7.1 we have \( \mathcal{X}^+(A) \subset (A|B) + V^* \), where \( V^* = \sup \mathcal{J} (\text{Ker}(D)) \). Since \( \text{Ker}(D) \) is \( G \)-patterned, by Lemma 7.5, \( V^* = \sup \mathcal{J}_G (\text{Ker}(D)) \). Now (7.2) can be rewritten as:

\[ \text{Ker}(\psi^+(A)) \subset \text{Im}(M_c) + \text{Im}(S^*) = \text{Im} [M_c S^*], \]

where \( S^*: V^* \to \mathcal{X} \) is the insertion map of \( V^* \). By Lemma 4.4, \( \psi^+(A) \) is \( G \)-patterned. By 5.4, \( M_c \) is \( G \)-patterned. By 4.5, we have that \( S^* \) is \( G \)-patterned. Thus, by Lemma 4.1, \([M_c S^*] \) is \( G \)-patterned. We can apply Lemmas 4.11 and 4.2 to obtain,

\[ \text{Ker}(\psi^+(A)) \subset \text{Im} \left[ \hat{M}_c \ S^* \right] = \text{Im}(\hat{M}_c) + \text{Im}(\hat{S}^*). \]

By the application of Lemma 4.2, we have that,

\[ \text{Im}(\hat{M}_c) = \text{Im} \left[ \hat{B} \ \hat{A}\hat{B} \ \hat{A}^2\hat{B} \ \cdots \ \hat{A}^{n-1}\hat{B} \right] = \langle \hat{A}|\hat{B} \rangle, \]
where $\mathcal{B} = \text{Im}(\hat{B})$. Furthermore, by Lemma 5.3, $\mathcal{X}^+(\hat{A}) = \text{Ker}(\psi^+(\hat{A}))$. By Lemma 7.6 and Remark 7.1 we have,

$$\text{Im}(\hat{S}^*) = \hat{\mathcal{V}}^* = \sup \mathfrak{I}(\hat{K}),$$

where $\hat{S}^*$ is the reduced form of $S^*$, $\hat{K} = \text{Im}(\hat{Q})$, and $\hat{Q}$ is the reduced form of $Q$. Altogether, we obtain,

$$\mathcal{X}^+(\hat{A}) \subset \langle \hat{A}|\mathcal{B} \rangle + \hat{\mathcal{V}}^*,$$

where, $\hat{\mathcal{V}}^* = \sup \mathfrak{I}(\hat{K})$, for $\hat{K} = \text{Im}(\hat{Q})$. Thus the solvability of the OSP for the reduced system $(\hat{A}, \hat{B})$, as required.

$(\Leftarrow)$ The converse proof reverses the statements above to obtain (7.2) and (7.3) above. It remains only to show that $K$ is $G$-patterned. To show this, we consider the feedback $\hat{K}$ that solves the OSP on the reduced subsystem $(\hat{A}, \hat{B})$. The existence of such a $\hat{K}$ is guaranteed by the condition $\mathcal{X}^+(\hat{A}) \subset \langle \hat{A}|\mathcal{B} \rangle + \hat{\mathcal{V}}^*$. We note that since $\hat{A}$, $\hat{B}$, and $\hat{S}^*$ are block diagonal, we can assume (without loss of generality) that $\hat{K} = \oplus_{i=1}^h \hat{K}_i$ such that $\hat{K}_i \in \mathbb{R}^{d_i \times \tilde{n}}$. Then, we note that for each $\hat{K}_i$ we can define a matrix $\hat{K}_i = \hat{K}_i \otimes I_{n_i}$ and form the direct sum $\hat{K} = \oplus_{i=1}^h \hat{K}_i$. Applying Theorem 4.1, we see that $K = T_u^{-1}\hat{K}T_X$ such that $K : \mathcal{X} \rightarrow \mathcal{U}$ and that $K$ is $G$-patterned by construction. As shown above, the resulting $K$ solves the OSP and is $G$-patterned, as required.

In the following remark, we provide an alternative proof of sufficiency for Theorem 7.2 which provides some insight into how the patterned feedback is constructed using geometric control techniques.

**Remark 7.2.** By assumption, we have that $\mathcal{X}^+(\hat{A}) \subset \langle \hat{A}|\mathcal{B} \rangle + \hat{\mathcal{V}}^*$ such that $\hat{\mathcal{V}}^* = \sup \mathfrak{I}(\hat{K})$. By Remark 7.1, we have that $\hat{\mathcal{V}}^* = \text{Im}(\hat{S}^*_i)$ is controlled invariant for the system $(\hat{A}_i, \hat{B}_i)$. By [12], this implies that there exists a feedback transformation $(M_i, \hat{K}^1_i)$ such that,

$$M_i^{-1}(\hat{A}_i + \hat{B}_i\hat{K}^1_i)M_i = \begin{bmatrix} \hat{A}_i^1 & * \\ 0 & \hat{A}_i^2 \end{bmatrix}, \quad M_i^{-1}\hat{B}_i = \begin{bmatrix} \hat{B}_i^1 \\ \hat{B}_i^2 \end{bmatrix}.$$  

Clearly, we would like to influence the dynamics of $\hat{A}_i^2$ to make subspace $\hat{V}_i^*$ attractive. By assumption and the block diagonal structure of the reduced system, we have that $\mathcal{X}^+(\hat{A}_i) \subset \langle \hat{A}_i|\mathcal{B}_i \rangle + \hat{\mathcal{V}}^*_i$. Again by [12], we note that $M_i = [ * \ P_i^T]^T$ such that $P_i$ is the canonical projection of $\hat{\mathcal{V}}^*_i$ (i.e. $\hat{\mathcal{V}}^*_i = \text{Ker}(P_i)$). We apply $P_i$ to the previous expression to give $P_i\mathcal{X}^+(\hat{A}_i) \subset P_i \left( \langle \hat{A}_i|\mathcal{B}_i \rangle + \hat{\mathcal{V}}^*_i \right) = P_i \langle \hat{A}_i|\mathcal{B}_i \rangle$. Thus, by Lemma 2.3, we have $\mathcal{X}^+(\hat{A}_i^2) \subset \langle \hat{A}_i^2|\mathcal{B}_i^2 \rangle$. By [12], this implies that there exists another feedback $\hat{K}_i^2$ such that the above dynamics decay to the subspace $\hat{V}_i$. That is, we can define a sum of the two feedbacks $\hat{K}_i = \hat{K}_i^1 + \hat{K}_i^2 P_i$ such that,

$$e^{(\hat{A}_i + \hat{B}_i\hat{K}_i)t} \rightarrow \hat{V}_i, \quad t \rightarrow \infty.$$
We note that,
\[ e^{(\hat{A}_i + \hat{B}_i \hat{K}_i)t} \otimes I_{n_i} = \sum_{k=0}^{\infty} ((\hat{A}_i + \hat{B}_i \hat{K}_i) \otimes I_{n_i})^k t^k = e^{(\hat{A}_i + \hat{B}_i \hat{K}_i)t}. \]

Moreover, \( \hat{V}_i \otimes I_{n_i} = \text{Im}(\hat{S}_i^*) \otimes I_{n_i} = \text{Im}(\tilde{S}_i^*). \) Therefore, for all \( i = 1...h, \) we have,
\[ e^{(\hat{A}_i + \hat{B}_i \hat{K}_i)t} \rightarrow \text{Im}(\tilde{S}_i^*) \quad t \rightarrow \infty. \]

Recall that \( V^* = \text{Im}(S^*) = \text{Im}(\tilde{S}) = \text{Im}(\oplus_{i=1}^{h} \tilde{S}_i^*). \) We collect all of the equations into one expression via matrix direct sum,
\[ e^{(\hat{A} + \hat{B} \hat{K})t} \rightarrow V^* \quad t \rightarrow \infty. \]

We apply the transformations \( T_x \) and \( T_u \) to the state evolution, to obtain,
\[ x(t) = e^{(A+BK)t} \rightarrow V^* \quad t \rightarrow \infty. \]

We note that \( K = T_u^{-1} \hat{K} T_x \) Finally, since \( V^* \subset \text{Ker}(D), \) we have,
\[ z(t) = Dx(t) = De^{(A+BK)t} \rightarrow 0, \quad t \rightarrow \infty, \]

which demonstrates that, by definition, output stabilization has been solved. Furthermore, the stabilizing feedback \( u = Kx \) is \( G \)-patterned by construction (Theorem 4.1), and is given by the following expression,
\[ K = T_u^{-1} \hat{K} T_x = T_u^{-1} \begin{bmatrix} \hat{K}_1 & \ddots & \cdot \cdot \cdot \hat{K}_h \end{bmatrix} T_x, \]

where, for all \( i = 1,...,h, \hat{K}_i = \hat{K}_i \otimes I_{n_i}. \) Recall that the \( \hat{K}_i = \hat{K}_i^1 + \hat{K}_i^2 P_i \) were the output stabilizing feedbacks of the patterned reduced systems.

\[ 7.3 \quad \text{Example} \]

**Example 7.1.** In this example, we consider a chain system in which is subdivided into five subsystems. The input is only allowed to enter into the system at the outer edges of the chain, while the output is taken from subsystems closer to the center. Note that there is an unstable subsystem at the center of
the chain. The system is given by the following matrices:

\[
A = \begin{bmatrix}
-1 & 10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & -5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 3 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & -4 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 4 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 3 & 1 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & -4 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 10 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -5 \\
\end{bmatrix},
\]

\[
B = \begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
\end{bmatrix}^T,
\]

\[
D = \begin{bmatrix}
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
\end{bmatrix}.
\]

Furthermore, we define \( V = \ker(D) = \im(S) \) such that \( S \) is defined as follows:

\[
S = \begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
\end{bmatrix}^T.
\]

We consider the system graph associated with this system:
In the above figure, $A_1 = \begin{bmatrix} 1 & 4 \\ 0 & 2 \end{bmatrix}$, $A_2 = \begin{bmatrix} 3 & 1 \\ 0 & -4 \end{bmatrix}$, $A_3 = \begin{bmatrix} -1 & 10 \\ 0 & -5 \end{bmatrix}$, $A_4 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, $A_5 = \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix}$, $B_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, and $D_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$. When we consider the system graph, we note that only one possible permutation (apart from the identity) can make up the automorphism group. This permutation involves switching the first and fifth vertices as well as the second and the third. The representation of this permutation on the state space is as follows:

$$
\rho^X(g_1) = \begin{bmatrix}
0 & 0 & 0 & 0 & I_2 \\
0 & 0 & 0 & I_2 & 0 
0 & 0 & I_2 & 0 & 0 
0 & I_2 & 0 & 0 & 0 
I_2 & 0 & 0 & 0 & 0
\end{bmatrix}
$$

There are only two one-dimensional irreducible representation spaces for this group (i.e. $n_1 = n_2 = 1$). We use the MATLAB code to determine the following multiplicities with which the $W_i$ compose $X$, $U$ and $K = \text{Ker}(D)$ respectively: $\eta_1^X = 6$, $\eta_2^X = 4$, $\eta_1^U = 3$, $\eta_2^U = 2$, $\eta_1^K = 3$, $\eta_2^K = 2$. With this information in hand we can decompose the $A$, $B$, and $S$ matrices, to yield the following systems:

$$
\hat{A}_1 = \begin{bmatrix}
6.13 & 4.24 & 7.42 & -5.94 & 7.53 & 6.35 \\
4.74 & 7.91 & 3.52 & -7.55 & 6.51 & 0.734 
3.06 & 12.6 & 3.88 & -10.7 & 8.37 & -2.79 
-4.25 & 5.07 & -1.08 & -0.216 & 1.15 & -1.73 
-19.6 & -24.0 & -12.7 & 23.9 & -21.4 & -6.03 
4.43 & 3.83 & -2.78 & -1.59 & 0.609 & -0.313
\end{bmatrix},
\hat{B}_1 = \begin{bmatrix}
-0.262 & -0.283 
-0.814 & 0.0393 
-1.23 & -1.03 
-0.283 & -0.683 
2.4 & 1.2 
-0.516 & 0.0544
\end{bmatrix}
$$
Chapter 7. Patterned Output Stabilization Problem

\[
\hat{S}_1 = \begin{bmatrix}
0.488 & 0.505 & -0.078 & -0.302 \\
-0.356 & 0.346 & -0.206 & -1.02 \\
-0.00912 & 0.584 & -0.716 & -1.3 \\
0.0322 & -0.683 & -0.698 & 0.332 \\
0.311 & -1.76 & 0.802 & 3.16 \\
-0.229 & 0.392 & -0.0386 & -0.779 \\
\end{bmatrix};
\]

\[
\hat{A}_2 = \begin{bmatrix}
-11.4 & 1.19 & -10.4 & -10.9 \\
-10.5 & -1.03 & -10.5 & -13.6 \\
-3.46 & 2.73 & -2.61 & -2.55 \\
9.8 & -4.04 & 7.4 & 8.04 \\
\end{bmatrix}, \quad \hat{B}_2 = \begin{bmatrix}
1.5 \\
1.45 \\
0.111 \\
-0.835 \\
\end{bmatrix}, 
\]

\[
\hat{S}_2 = \begin{bmatrix}
1.5 \\
1.5 \\
0.227 \\
-0.843 \\
\end{bmatrix}
\]

Now that we have decomposed the system, we proceed to solve the output stabilization problem on each of the subsystems. Note that for the following calculations we use the Geometric Approach MATLAB toolbox developed by Basile and Marro [36]. We first find \( \sup \mathcal{J}(\hat{S}_i) \) for each reduced subsystem. As it turns out, there is a supremum for the first subsystem, but not for the second (algorithm converges on zero). The first \( \mathcal{V}_1^* = \text{Im}(\hat{S}_1^*) \) is given by the following insertion map:

\[
\hat{S}_1^* = \begin{bmatrix}
-0.311 & 7.95 \cdot 10^{-17} \\
0.0455 & 0.455 \\
-0.488 & 0.486 \\
0.239 & 0.0335 \\
0.0734 & -0.668 \\
0.775 & 0.332 \\
\end{bmatrix}
\]

Next, we verify that the OSP is, in fact, solvable by checking the condition \( \mathcal{X}^+(\hat{A}_i) \subset \langle \hat{A}_i | \hat{B}_i \rangle + \mathcal{V}_i^* \). OSP is solvable for both subsystems and hence the POSP is solvable. Considering the first subsystem, we generate a feedback that guarantees invariance of \( \mathcal{V}_1^* \):

\[
\hat{K}_{11} = \begin{bmatrix}
0.782 & 0.687 & 2.08 & -0.543 & -1.36 & -1.37 \\
1.02 & 0.892 & 2.71 & -0.706 & -1.77 & -1.77 \\
\end{bmatrix}
\]

Next, we place the poles of the dynamics off of \( \mathcal{V}_1^* \) to ensure that the output dynamics are stable with
a second feedback:

\[ \hat{K}_{12} = \begin{bmatrix} 13.7 & -65.0 & 55.8 & -10.3 & 18.6 & 45.8 \\ -11.0 & 86.2 & -68.1 & -4.89 & -15.6 & -49.3 \end{bmatrix} \]

The control invariance and stabilization feedbacks are summed to form \( \hat{K}_1 \). Despite the fact that the second reduced subsystem's controlled invariant subspace is trivial, we can still stabilize the second system because it is controllable. We stabilize the reduced subsystem using the following feedback:

\[ \hat{K}_2 = \begin{bmatrix} 73.2 & 168.0 & -74.6 & 135.0 \\ -86.9 & -361.0 & 223.0 & -171.0 \end{bmatrix} \]

Putting the feedbacks together from each subsystem and reversing the decomposition we obtain:

\[
K = T_u (\hat{K}_1 \oplus \hat{K}_2) T^{-1}_X
\]

\[
= \begin{bmatrix} -10.1 & -10.6 & -47.3 & -6.5 & -1.34 \cdot 10^{-13} & -13.9 & 3.1 & -0.0597 & 0.443 & -0.561 \\ -1.57 & -0.876 & -4.11 & -2.93 & -4.86 \cdot 10^{-14} & -11.7 & -34.0 & -3.14 & -4.81 & 2.56 \\ 0.443 & -0.561 & 3.1 & -0.0597 & -1.28 \cdot 10^{-13} & -13.9 & -47.3 & -6.5 & -10.1 & -10.6 \\ -4.81 & 2.56 & -34.0 & -3.14 & -5.44 \cdot 10^{-14} & -11.7 & -4.11 & -2.93 & -1.57 & -0.876 \end{bmatrix}
\]

We obtain the closed loop system:

\[
A + BK =
\begin{bmatrix} -11.1 & -0.646 & -47.3 & -6.5 & -1.34 \cdot 10^{-13} & -13.9 & 3.1 & -0.0597 & 0.443 & -0.561 \\ -1.57 & -5.88 & -4.11 & -2.93 & -4.86 \cdot 10^{-14} & -11.7 & -34.0 & -3.14 & -4.81 & 2.56 \\ 1.0 & 0 & 3.0 & 1.0 & 0 & 1.0 & 0 & 0 & 0 & 0 \\ 0 & 1.0 & 0 & -4.0 & 0 & 1.0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1.0 & 4.0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2.0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1.0 & 3.0 & 1.0 & 1.0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1.0 & 0 & -4.0 & 0 & 1.0 \\ 0.443 & -0.561 & 3.1 & -0.0597 & -1.28 \cdot 10^{-13} & -13.9 & -47.3 & -6.5 & -11.1 & -0.646 \\ -4.81 & 2.56 & -34.0 & -3.14 & -5.44 \cdot 10^{-14} & -11.7 & -4.11 & -2.93 & -1.57 & -5.88 \end{bmatrix}
\]

We note that the closed loop system is patterned and indeed commutes with the permutation mentioned at the beginning of the example. Furthermore, the following graphs of the trajectories of output and non-output states shows that the output has been stabilized, but other states of the system are unstable.
Figure 7.2: OSP Example - Output and Non-Output State Trajectories
Chapter 8

Conclusion

Symmetry is a concept that is so natural to the human mind that even a child with basic knowledge of geometry can understand it. It is a concept that is so ingrained in the human psyche that we have defined our notions of artistic beauty based on its principles. The patterns that reflect symmetry appear in our architecture, our aesthetics, even the molecules that compose us. Few realize that there is a mathematical construct - group theory - that provides a formalized characterization of these patterns. Equally as ubiquitous in modern society, though significantly less understood in general, are distributed systems. Distributed systems comprise our transit, our electrical grids, our factories, and our communication networks. In fact, with the recent advances in communications technology, distributed systems are becoming even more prominent in the systems of mankind. For example, multi-agent systems such as swarms of drones have become a new tool to gather information and data in large industries such as agriculture or mining. As in most systems, we require a specialized, distributed control theory to properly regulate such systems.

It is only natural, then, that a theory should arise to unify these two great concepts that are so prevalent in our society. Such a theory would allow us to harness the mathematically rigourous characterization of symmetry presented by group and representation theory, and employ this characterization to master the control of distributed systems.

Such a unification is the goal to which this thesis is dedicated and, indeed, makes its largest contribution. We have formalized a method by which, through graph theory, we can identify the patterns inherent in any linear dynamical system and formulate a finite group to characterize these patterns in Chapter 3. Using this finite group, we have established a decomposition of the patterned system into a series of reduced, non-patterned subsystems in Chapter 4. With this decomposition in hand, we showed in Chapter 5 that pole placement and stabilization of these subsystems will result in pole placement and stabilization of the entire system with a controller that preserves the patterns characterized by the finite group. Similarly, in Chapters 6 and 7 we demonstrated that classical control problems such as
stabilization by measurement feedback and output stabilization can be solved with a pattern-preserving feedback if they are solved on the reduced subsystems in the standard linear sense. Throughout this document we provide constructive procedures for synthesis of the feedback matrices that we employ and have provided ample examples to further demonstrate the validity of the theory.

Patterned solutions to other classical problems, such as the restricted regulator problem and the disturbance decoupling problem, were not investigated in this document. However, we assume that a similar method of decomposition and control synthesis for the reduced system will yield a patterned solution.

It is important to note that a controller that preserves the patterning of a given system is not necessarily a distributed controller. This is because, although patterning imposes some constraints on the communication between subsystems, these constraints are not as strict as those of a distributed controller. We illustrate this disconnect with the following example. Suppose we have a system with a system graph as shown on the left of Figure 8.1 below:

![Figure 8.1: Illustration of the interactions between subsystems allowed by pattern feedback](image)

Suppose we require that the distributed controller of this system does not allow communication between subsystems 1 and 4 as well as between 2 and 3. We note that the patterns of this system involve flipping diagonally, flipping vertically, and rotating the system graph above. It is clear from the figure that a patterned feedback would preserve these patterns, but would allow the communications that we have prohibited.

Herein lies the gap between the theory of patterned system presented in this document and the ultimate goal of a formalized, algebraic method for the formulation of distributed controllers. There have been some attempts to bridge this gap as shown in [15] and [19], but no formalized method has yet been developed to take advantage of patterning to yield a true distributed controller.

We suspect that, to produce a distributed controller from a patterned one, additional constraints will have to be applied at the stage of feedback synthesis for the reduced subsystems. A prudent research
strategy would be to revisit established results on distributed controllers in the literature in light of our patterned decomposition. One prominent example of this would be the investigation of the implications of the fixed modes of a decentralized controller in the reduced subsystems. Moreover, the implications of any constraint on controller structure should be investigated from the perspective of the reduced system.

Another avenue of future work includes the extension of the concept of patterned systems to nonlinear systems. The obvious nonlinear analog of the commuting property is $f(Px) = Pf(x)$ for some non-linear function $f$ and representation matrix $P$.

The study of Lie groups is also very promising, since it is commonly known that these groups represent the ”best-developed theory of continuous symmetry of mathematical objects and structures” according to a quick Wikipedia search [37]. For example, these groups can be used to characterize the symmetry of manifolds, a concept that is enticing for any non-linear control theorist with some knowledge of differential geometry. A more specific example of a Lie group is the $SO(3)$ rotation group, which finds important applications in many physical systems, including those of aerospace engineering and robotics.

To conclude, although this thesis provides a significant step forward in the characterization of patterns in distributed systems, as ever, research in this field is far from being closed.
Appendix A

Simplification of the Computation of the Irreducible Decomposition Matrix

The computation of irreducible decomposition matrix $T$ has been found to be the most computationally intensive stage of Patterned Linear Control. As such, three computational simplifications are presented in this appendix.

A.1 Kernal Intersection Simplification

In practice, the matrix $M$ can be very large (depending on the size of the group) and is prone to floating point arithmetic or rounding errors in computational software. It is advisable to calculate each $\text{Ker}(M(g))$ individually and find the basis of the intersection of the resulting subspaces iteratively. This achieves the same result and can be done easily in MATLAB using the `null` and `orth` commands, as elaborated in the Appendix B. This method is not as susceptible to the aforementioned errors and generally requires less memory to perform.

A.2 Generator Simplification

Another simplification can be made using the so-called generators of a group, which are defined as follows.

**Definition A.1** (Generating set of a Group). Given a group $G$, there exists a subset $S$ of its elements such that any element of the group can be expressed as a combination of the elements in the subset and
Appendix A. Simplification of the Computation of the Irreducible Decomposition Matrix

their inverses via the group operation. Elements of such a subset are referred to as the \textit{generators of the group} and the subset itself is referred to as the \textit{generating set of the group}.

It is often the case that the generating set is much smaller than the entire set of elements in the group. Some of the calculations that we perform need only use the generators of the group to be completed. The computation of the irreducible decomposition matrix $T$ is one such calculation.

Lemma A.1. Let $G$ be a finite group, $\mathcal{V} \in \mathbb{R}^n$ a vector space, $\rho : G \rightarrow GL(\mathcal{V})$ be a representation of $G$ over $\mathcal{V}$, and $\tilde{\rho} : G \rightarrow GL(\mathcal{V})$ be the irreducible form of $\rho(g)$. Let $S$ be the generating set of $G$. Then, if $\rho(s)T = T\tilde{\rho}(s)$ for all $s \in S$ then $\rho(g)T = T\tilde{\rho}(g)$ for all $g \in G$.

Proof. Let $g \in G$ and $S$ be the generating set of $G$. Then for some $s_1, s_2 \in S$, $g = s_1 \cdot s_2$, where "\cdot" is the group operation. Then we have that $\rho(g) = \rho(s_1) \cdot \rho(s_2)$ and $\tilde{\rho}(g) = \tilde{\rho}(s_1) \cdot \tilde{\rho}(s_2)$. By assumption we have that $\rho(s_1)T = T\tilde{\rho}(s_2)$ and $\rho(s_2)T = T\tilde{\rho}(s_2)$. Thus we have $\rho(g)T = \rho(s_1) \cdot \rho(s_2)T = \rho(s_1) \cdot T\tilde{\rho}(s_2) = T\tilde{\rho}(s_1) \cdot \tilde{\rho}(s_2) = T\tilde{\rho}(g)$.

The significance of this lemma is that the matrix $M$ in (2.11) above needs only include the representations of the elements of the set $S$, as shown below.

$$M = \left( I_n \otimes \begin{bmatrix} \rho(1) \\ \rho(2) \\ \vdots \\ \rho(|S|) \end{bmatrix} - \begin{bmatrix} \tilde{\rho}(1)^T \\ \tilde{\rho}(2)^T \\ \vdots \\ \tilde{\rho}(|S|)^T \end{bmatrix} \otimes I_n \right).$$

Note that only the representations of the elements of $S$ are used above. In some cases, this significantly reduces computation times as the cardinality set $S$ is most often much smaller than the cardinality of $G$.

A.3 Equal Dimension Subsystem Simplification

In the case that all of the subsystems of a given system have the same dimension, we encounter another significant simplification: it turns out that we only need calculate the irreducible decomposition transformation $T$ for a simplified version of the system in which each subsystem has only one state. Needless to say, this greatly simplifies the calculations required, especially when the subsystems are large in dimension.

Lemma A.2. Suppose we have permutation representation $\rho : G \rightarrow GL(\mathcal{V})$ of $G$ in $\mathcal{V} \sim \mathbb{R}^n$ which acts on a system with one-dimensional subsystems and $\rho' : G \rightarrow GL(\mathcal{V}')$ of $G$ in $\mathcal{V}' \sim \mathbb{R}^{n-m}$ which acts on the same system, but with $m$-dimensional subsystems. If $T$ is the irreducible decomposition transformation of the first representation then the irreducible decomposition of the second representation is given by
The structure of the system is not altered, we can form the permutation representation of the altered system.

Now, suppose we wish to extend the system such that each subsystem is one-dimensional. Since the structure of the system is not altered, we can form the permutation representation of the altered system \( \rho' : G \rightarrow GL(V') \) of \( G \) in \( V' \sim \mathbb{R}^n \) via the Kronecker product with identity, \( \rho' = \rho \otimes I_m \). We are interested in the irreducible decomposition transformation \( T' \) of this new system. Suppose that we have already found a \( T \) for the original system, that is, \( \rho^V(g)T = T\tilde{\rho}^V(g) \) for all \( g \in G \). Now, we note that for all \( g \in G \),

\[
(\rho^V(g) \otimes I_m)(T \otimes I_m) = (\rho^V(g)T) \otimes I_m = (T\tilde{\rho}^V(g)) \otimes I_m = (T \otimes I_m)(\tilde{\rho}^V(g) \otimes I_m)
\]

We note that \( \tilde{\rho}^V(g) \otimes I_m \) is certainly not a direct sum of irreducible representations as called for by Lemma 2.16. However, we can distribute the Kronecker product as follows, partitioning \( \tilde{\rho}^V(g) = \bigoplus_{i=1}^{h} R_i \) as in Lemma 2.16. We then find that \( \tilde{\rho}^V(g) \otimes I_m = \bigoplus_{i=1}^{h} (R_i \otimes I_m) = \bigoplus_{i=1}^{h} \bigoplus_{i=1}^{n_i} \tilde{\rho}^V_i(g) \).

With the help of Lemma 2.7, we see that we can reverse each product with perfect shuffle matrix such that for \( i = 1, ..., h \), \( R_i \otimes I_m = S_i(I \otimes R_i)S_i \) where \( S_i = \sum_{j=1}^{m} e_i^T \otimes I_{n_i}\eta_i \otimes e_i \) is the appropriate perfect shuffle matrix. We note that \( I_m \otimes R_i = I_m \otimes \bigoplus_i \tilde{\rho}^V_i(g) \) is, in fact, in irreducible form, since \( \tilde{\rho}^V_i(g) \) is just repeated \( m \) times.

Furthermore we note that \( I_m \otimes R_i = \tilde{\rho}^V_i(g) \). We can collect the \( S_i \) into a block matrix \( S = \bigoplus_{i=1}^{h} S_i \).

Continuing from our previous equation,

\[
(\tilde{\rho}^V(g) \otimes I_m) = \bigoplus_{i=1}^{h} (R_i \otimes I_m) = \bigoplus_{i=1}^{h} S_i^T (I_m \otimes R_i)S_i = S^T \left[ \bigoplus_{i=1}^{h} (I_m \otimes R_i) \right] S = S^T \bigoplus_{i=1}^{h} \tilde{\rho}^V_i(g)S = S^T \tilde{\rho}^V(g)S,
\]

which implies that \( (\tilde{\rho}^V(g) \otimes I_m)S = S\tilde{\rho}^V(g) \). Putting it all together we have the following equation.

\[
(\rho^V(g) \otimes I_m)(T \otimes I_m)S = (T \otimes I_m)(\tilde{\rho}^V(g) \otimes I_m)S = (T \otimes I_m)(I_m \otimes \tilde{\rho}^V(g)) = (T \otimes I_m)S\tilde{\rho}^V(g).
\]

The matrix \( T' = (T \otimes I_m)S \) is therefore the irreducible decomposition transformation of representation \( \tilde{\rho}^V(g) \) by definition.

As a result of this lemma, given a systems with subsystems of equal dimension, we need only perform the algorithm to find the \( T \) matrix for an equivalent system with one dimensional subsystems. The matrix \( S \) is trivially calculated. This greatly simplifies our calculations and computation time and space, and essentially removes the computational limit on subsystem size.
Appendix B

Pattern Decomposition Code

The following lemma is used purely for analysis of the permissible permutations of the automorphism group of the system graph. It is used to account for the fact that the NAUTY software does not support edge labels. In the following lemma, we employ the following equation:

\[
(M_{L_k})_{ij} = \begin{cases} 
    k & \text{if } (v_j, v_i) \in E \text{ and } L(e_{ij}) = k \\
    0 & \text{otherwise}
\end{cases}.
\] (B.1)

We note that \( M_{L_k} = \sum_{k} M_{L_k} \) and that \( k = 1, \ldots, |I_L| \). The matrix \( M_{L_k} \) is the labeled adjacency matrix when we only consider the edges that have label \( k \).

**Lemma B.1.** Let \( G = (V, E, L) \) be a system graph and let the corresponding labeled adjacency matrix be \( M_L \). Now, let \( E_k \subset E \) be such that \( e_{ij} \in E_k \) if and only if \( l_{kj} = k \) and define \( G_k = (V, E_k, L) \). Then \( \text{AUT}(G) = \cap_{k} \text{AUT}(G_k) \).

**Proof.** First note that \( M_{L_k} \) is the labeled adjacency matrix of \( G_k \).

( \( \Rightarrow \) ) Let \( g \in \text{AUT}(G) \) and let \( \rho : G \to GL(\mathbb{R}^m) \) (where \( m = |V| \)) be a vertex permutation representation. From Lemma 3.2, we know that \( \rho(g)M_L = M_L\rho(g) \) which implies that \( \rho(g)(\sum_{k} M_{L_k}) = (\sum_{k} M_{L_k})\rho(g) \). Since the elements of \( M_{L_i} \) cannot be equal to the elements of \( M_{L_j} \) if \( i \neq j \), we have that \( \rho(g)M_{L_k} = M_{L_k}\rho(g) \), for all \( k = 1, \ldots, |I_L| \). Therefore \( g \in \text{AUT}(G_k) \) for all \( k = 1, \ldots, |I_L| \) by Lemma 3.2.

( \( \Leftarrow \) ) Let \( g \in \text{AUT}(G_k) \) for all \( k = 1, \ldots, |I_L| \). By Lemma 3.2 this implies that \( \rho(g)M_{L_k} = M_{L_k}\rho(g) \), for all \( k = 1, \ldots, |I_L| \). Hence the sum also has the commuting property, i.e. \( \rho(g)(\sum_{k} M_{L_k}) = (\sum_{k} M_{L_k})\rho(g) \). Again by Lemma 3.2, we have that \( g \in \text{AUT}(G) \), as required. \( \square \)

This lemma basically allows us to run NAUTY for each graph \( G_k \), since the labels of \( G_k \) are all the same. The intersection of the resulting automorphism groups is therefore the automorphism group of the graph \( G \).
We now give the code of the Pattern Toolbox that was developed to perform all of our pattern-preserving computations. The main section of code is a the Pattern MATLAB class definition defined as follows.

```matlab
classdef Pattern
properties (SetAccess=private, GetAccess=public)
%Numbers
n; %Permutation Representation Dimension
o; %Order of the underlying Group
numGen; %Number of Generators Used
numChar; %Number of Characters/Reducibles/Conjugacy Classes

%Matrices/Vectors
perm; %Permutation Representation
irred; %Permutation Decomposed into Irreducibles
irreducs; %Irreducible Representations
elemMap; %Map of Elements
charTable; %Character Table
conjClass; %Conjugacy Class of Element
etas; %Multiplicities of the Irreducibles in the Permutation Rep.
end
methods
%Constructor for a Pattern object
%NOTE: correspondence between the permGens and the irreducGens is
%VERY important. Make sure that they ordered in the same manner
%as expressed in GAP
%NOTE: This class is used to define a base pattern of a system,
%Therefore the sub-systems are assumed to be one dimensional. The
%IrredTrans method allows input to change the dimension of the
%sub-systems
function obj = Pattern(order, permGens, irreducGens, charTable)
%order - the order of the underlying group
%permGens - array of the permutation representation generator
%matrices (assuming f-Dimensional sub-systems
%irreducGens - cell array containing the arrays of the of the
%irreducible representation generating matrices
%NOTE: expected to be a cell of (n x n x n gens) matrices
%charTable - character table

[obj.n, dummy, obj.numGen] = size(permGens);
[obj.numChar, dummy] = size(charTable);
obj.o = order;

%Define Character Table
obj.charTable = charTable;

%Define permutation matrix representation
%The ordering of elements is also determined here as well as
%which elements produce the others
obj.perm = zeros(obj.n, obj.n, obj.numGen);
obj.elemMap = zeros(2, obj.n);
for i = 1:n
obj.elemMap(1, i) = 0; %the generators do not have source elements
obj.elemMap(2, i) = 0;

%copy generators and kron multiply to account for non-zero
%subsystem dimension
obj.perm(:,:,i) = permGens(:,:,i);
end

%The following algorithm finds all elements of the group by
%Essentially exploring the multiplication table, and recording
%Any new elements
while (n == obj.n)
i = 1;
    while i <= n
        element = obj.perm(:,:,i) * obj.perm(:,:,j); %Define potentially new group element
        j = 1;
            while j <= n
```
Appendix B. Pattern Decomposition Code

% Define irreducible representations using irreducible generators
for i = 1:obj.numChars %for each irreducible representation
obj.irreducibles{i} = zeros(obj.charTable(i,1), obj.charTable(i,1), obj.o);
for j = 1:obj.o %for each element
if (obj.elemMap(1,j)==0) %if element is generator then define from generator
    obj.irreducibles{i}(j) = irredGens{j};
else
    dummy = obj.elemMap(1,j); % to find the elements that together produce this element
    b = obj.elemMap(2,j); % there is no fear of these elements not being defined (due to the way we created
    obj.irreducibles{i}(j) = obj.irreducibles{i}(j, dummy)*obj.irreducibles{i}(j, b);
end
end
end
end

% Define the Conjugacy Class of each element based on character Table
for i = 1:obj.o %for each element
char = trace(obj.irreducibles{i}(:,i));
if (obj.numChars>1)
    for j = 1:obj.numChars %for each irreducible
        char = [char; trace(obj.irreducibles{j}(:,i))];
    end
end
end
end
end

% Define decomposed permutation matrix
for i = 1:obj.numChars %for each irreducible
    S = rank(S)/obj.charTable(i,1);
end
end
end
end
end
for i = 1:obj.o
  block = []; % we will build the representation with this block
  for j = 1:obj.numChars
    if obj.eta(j) > 0
      for k = 1:obj.eta(j) % where we add eta irreducible blocks
        block = blkdiag(block, obj.irreducibles(j)(1:k, :));
      end
    end
  end
  obj.irred(:, :, i) = block;
end
end

% This method finds the Irreducible Decomposition Transformation
% Given a specific dimension of the subsystems of the "Pattern"
function T = IrredTrans(obj, subSysDim)

% SubSysDim - the dimension of the subsystems exhibiting the
% "Pattern"
% We find the solution space by intersecting the solution spaces of the
% individual Sylvester equations. Then we find a matrix in the solution
% Space.
for i = 1:obj.numGen % first numGen elements are generators (we only need to use these)
  M = kron(eye(obj.n), obj.perm(:, :, i)) - kron(obj.irred(:, :, i)', eye(obj.n));
  tempNullM = null(M);
  if i == 1
    kerM = tempNullM; % define starting space
  else
    kerM = intersect(kerM, tempNullM); % intersect space with new space to update
  end
end

% Alternate Version of the above algorithm
% Unvectorize the basis vectors; dim is the number of basis matrices
[dummy, dimM] = size(kerM);
kerM = reshape(kerM, obj.n, obj.n, dimM);

% Define Irreducible Transformation Matrix T
T = zeros(obj.n, obj.n);
while (rank(T) < obj.n) % check if T is invertible
  T = zeros(obj.n, obj.n); % reset T
  for i = 1:dimM
    T = T + rand(); % add matrix basis elements with random multipliers
  end
end

% AS OF YET UNCLEAR WHY THIS IS NECESSARY (but if not here the
% imaginary components of the decomposed permutation rep are
% flipped)
T = conj(T);

% We now increase the dimension of the irreducible
% decomposition matrix to account for larger subsystems.
if subSysDim > 1
  % define perfect shuffle matrix for each irreducible that
  % has non zero multiplicity
  for j = 1:obj.numChars
    if (obj.eta(j) > 0)
      n_j = obj.charTable(j, 1); % size of irreducible
      s = zeros(subSysDim + n_j * obj.eta(j)); % initialize current perfect shuffle block
      % define the perfect shuffle of this irreducible
      % block
      for i = 1:subSysDim
        a = zeros(subSysDim, 1);
        a(i) = 1;
        s = s + kron(a', kron(eye(n_j * obj.eta(j)), a));
      end
      % check if a has been defined yet
      if ~exist('S', 'var')
function [FSI] = FrobSchurInd(obj)
% Calculate the Frobenius-Schur Indicators of the Irreducible Representations
FSI = zeros(obj.numChars, 1);
for i = 1:obj.numChars
    for j = 1:obj.numChars
        FSI(i) = FSI(i) + trace(obj.irreducibles{i}(:,j)^2);
    end
end
FSI = FSI / obj.o;
end

This method decomposes given matrix A based on the underlying pattern of pattern1 and pattern2 which represent the input and output spaces of the matrix respectively.

% If the inputs Tleft and Tright are not zero, they are used to decompose the matrix (this allows a decomposition defined in a given space to be used for multiple matrices).
% OUTPUT: Ahat - cell containing the Ahat matrices - empty cell indicates a multiplicity of zero
% ASSUMPTIONS:
% It is assumed that pattern1 and pattern2 represent the same group but possibly on different spaces. Furthermore, the generator order is assumed to be the same between the two representations.
% It is assumed that matrix A either has the same dimensions as pattern1 and pattern2, or, the dimensions of A are an integer multiple of these dimensions (indicates an increase in the dimension of subsystems of the system).

function [Ahat, Atilde, Tleft, Tright] = Decompose(pattern1, pattern2, A, Tleft, Tright)
% Check for differences between pattern1 and pattern2
if (pattern1.o˜=pattern2.o || isequal(pattern1.charTable, pattern2.charTable))
    error('Mismatch between input patterns...input patterns must represent the same Group and have the same character table');
end
% Check for dimension errors
[dimOut, dimIn] = size(A);
if mod(dimOut, pattern1.n)˜=0
    error('Output dimension mismatch');
end
subDimOut = dimOut/pattern1.n; % Subsystem dimension on output space
if mod(dimIn, pattern2.n)˜=0
    error('Input dimension mismatch');
end
subDimIn = dimIn/pattern2.n; % Subsystem dimension on input space
% Define Irreducible Decomposition Transformations
% If already defined from input, use input matrices
if(Tleft == 0)
    Tleft = pattern1.IrredTrans(subDimOut);
else
    Tleft = Tleft;
end
if(Tright==0)
    % If the patterns are equal (as dictated by their generators), we don't want to make a new transformation
    if isequal(pattern1.perm(:,1), pattern2.perm(:,1)) && isequal(subsDimOut, subsDimIn))
        Tright = Tleft;
    else
        % Otherwise, if they are different, make a new transformation
        Tright = pattern2.IrredTrans(subsDimOut);
end
Appendix B. Pattern Decomposition Code

285 end
286 else
287 Trt = Tright;
288 end
289
290 \% Perform transformation
291 Atilde = inv(Tlft) * A * Trt;
292
293 \% Create Ahat output - partition and reduce Astar:
294 Ahat = cell(pattern1.numChars,1);
295 markRow = 0; \% Marks the current block position in Astar
296 markCol = 0;
297 for i = 1:pattern1.numChars
298 \% Get the size of the current irreducible representation
299 ni = pattern1.charTable(i,1);
300 \% Next get multiplicity of output and input spaces (note that
301 \% multiplicity must be multiplied by the increase in
302 \% dimension
303 etaIn = pattern2.eta(i) * subsDimIn;
304 etaOut = pattern1.eta(i) * subsDimOut;
305 a = zeros(etaOut, etaIn); \% Initialize matrix (Ahat_i)
306 for j = 1:etaOut
307 for k = 1:etaIn
308 a(j,k) = Atilde(markRow+(j-1)*ni+1,markCol+(k-1)*ni+1); \% Get element skipping over redundant values
309 end
310 end
311 markRow = markRow + etaOut * ni;
312 markCol = markCol + etaIn * ni;
313 Ahat{i} = a;
314 end
315 end
316 end
317 end
318

Additionally, there are some simple helper functions to make computations more simple. The \texttt{intersect} function gives the basis vectors of the intersection of two vector spaces using native MATLAB commands.

function \[ V \] = intersect( V1, V2 )
% This function finds the intersection of the two vector spaces V1 and V2
% note that matlab function null() returns an orthonormal basis of the
% nullspace of a given matrix, and we have matlab function orth()
% we use the formula perp(V1 inter V2) = perp(V1) + perp(V2)
7 V = null(orth([null(V1) null(V2)]))
8 end

The \texttt{permMatrix} function generates a permutation matrix based on a input vector of the permutation indices.

function \[ \text{perm} \] = permMatrix( n, v )
% This Function generates the nxn permutation matrix corresponding to the
% input vector of permutations \text{v}
4
5 \% Get size of \text{n}
6 [numPerms, vecCheck] = size(v);
7 if vecCheck ~= 2
8 error(‘Input dimension of permutation vector not correct\nExpected numPerms by 2 vector’);
9 end
10
11 \% Generate Permutation Matrices
12 perm = eye(n); \% Start with identity
13 for i = 1: numPerms \% For each permutation
14 perm(v(i,1), v(i,1)) = 0; \% Zero diagonal on terms that change/permute
15 perm(v(i,2), v(i,2)) = 0; \% Add permutation terms
16 perm(v(i,1), v(i,2)) = 1; \% ...and add non-diagonal terms
17 end
18
The basic procedure to perform the full decomposition of a system is given in the next section.

B.1 Sample Decomposition

Example B.1. We illustrate all of the steps of the pattern decomposition in this example, including all of the computational steps that are required. We are investigating the block circulant system given by the following matrices.

\[
A = \begin{bmatrix}
-1 & 5 & 3 & 1 & -9 & -3 & 3 & 1 \\
0 & -3 & 0 & -5 & 0 & 1 & 0 & -5 \\
3 & 1 & -1 & 5 & 3 & 1 & -9 & -3 \\
0 & -5 & 0 & -3 & 0 & -5 & 0 & 1 \\
-9 & -3 & 3 & 1 & -1 & 5 & 3 & 1 \\
0 & 1 & 0 & -5 & 0 & -3 & 0 & -5 \\
3 & 1 & -9 & -3 & 3 & 1 & -1 & 5 \\
0 & -5 & 0 & 1 & 0 & -5 & 0 & -3
\end{bmatrix},
B = \begin{bmatrix}
1 & 0 & 1 & 0 \\
1 & 0 & -1 & 0 \\
0 & 1 & 0 & 1 \\
0 & 1 & 0 & -1 \\
1 & 0 & 1 & 0 \\
-1 & 0 & 1 & 0 \\
0 & 1 & 0 & 1 \\
0 & -1 & 0 & 1
\end{bmatrix},
C = \begin{bmatrix}
7 & 5 & -1 & 1 & -5 & -3 & -1 & 1 \\
-1 & 1 & 7 & 5 & -1 & 1 & -5 & -3 \\
-5 & -3 & -1 & 1 & 7 & 5 & -1 & 1 \\
-1 & 1 & -5 & -3 & -1 & 1 & 7 & 5
\end{bmatrix},
D = \begin{bmatrix}
5 & 0 & -1 & 0 & 1 & 0 & -1 & 0 \\
-1 & 0 & 5 & 0 & -1 & 0 & 1 & 0 \\
1 & 0 & -1 & 0 & 5 & 0 & -1 & 0 \\
-1 & 0 & 1 & 0 & -1 & 0 & 5 & 0
\end{bmatrix}.
\]

This example is drawn from (ADAMS EXAMPLE).

We first note the repetition of blocks in this system and partition it accordingly as follows.

\[
A = \begin{bmatrix}
A_1 & A_2 & A_3 & A_2 \\
A_2 & A_1 & A_2 & A_3 \\
A_3 & A_2 & A_1 & A_2 \\
A_2 & A_3 & A_2 & A_1
\end{bmatrix},
B = \begin{bmatrix}
B_1 & B_2 & B_3 & B_2 \\
B_2 & B_1 & B_2 & B_3 \\
B_3 & B_2 & B_1 & B_2 \\
B_2 & B_3 & B_2 & B_1
\end{bmatrix},
C = \begin{bmatrix}
C_1 & C_2 & C_3 & C_2 \\
C_2 & C_1 & C_2 & C_3 \\
C_3 & C_2 & C_1 & C_2 \\
C_2 & C_3 & C_2 & C_1
\end{bmatrix},
D = \begin{bmatrix}
D_1 & D_2 & D_3 & D_2 \\
D_2 & D_1 & D_2 & D_3 \\
D_3 & D_2 & D_1 & D_2 \\
D_2 & D_3 & D_2 & D_1
\end{bmatrix},
\]

where \(A_1 = \begin{bmatrix}
-1 & 5 \\
0 & -3
\end{bmatrix}, A_2 = \begin{bmatrix}
3 & 1 \\
0 & -5
\end{bmatrix}, A_3 = \begin{bmatrix}
-9 & -3 \\
0 & 1
\end{bmatrix}, B_1 = \begin{bmatrix}
1 \\
1
\end{bmatrix}, B_2 = \begin{bmatrix}
0 \\
0
\end{bmatrix}, B_3 = \begin{bmatrix}
1 \\
-1
\end{bmatrix}, C_1 = \begin{bmatrix}
7 & 5 \\
-1 & 1
\end{bmatrix}, C_2 = \begin{bmatrix}
-5 & -3
\end{bmatrix}, C_3 = \begin{bmatrix}
-5 & -3
\end{bmatrix}, D_1 = \begin{bmatrix}
5 & 0 \\
-1 & 0
\end{bmatrix} \text{ and } D_3 = \begin{bmatrix}
1 & 0
\end{bmatrix}.
\]
Appendix B. Pattern Decomposition Code

We use the NAUTY software to determine the permutations that compose the automorphism group of the system graph given above. Note that in this case we can simply concern ourselves with the graph of the state matrix. We start the software and enter the commands to analyze the permissible permutations.

```
Microsoft Windows [Version 10.0.10240]
(c) 2015 Microsoft Corporation. All rights reserved.
C:\Users\Deborah\Documents>bash
Deborah@Dell /cygdrive/c/Users/Deborah/Documents
dreadnaut
Dreadnaut version 2.5 (64 bits).
> n=4
> g
0 : 1;
1 : 2;
2 : 3;
3 : 0.
> x
(1 3)
level 2: 3 orbits; 1 fixed; index 2
(0 1)(2 3)
level 1: 1 orbit; 0 fixed; index 4
1 orbit; gsize=8; 2 gens; 6 nodes; maxlev=3
cpu time = 0.00 seconds
> n=4
> g
0 : 2;
1 : 3.
> x
(1 3)
level 2: 3 orbits; 1 fixed; index 2
(0 1)(2 3)
level 1: 1 orbit; 0 fixed; index 4
1 orbit; gsize=8; 2 gens; 6 nodes; maxlev=3
cpu time = 0.00 seconds
```

Note that we are applying Lemma B.1 above to account for the fact that NAUTY does not support edge labels. We see that the permutations (1, 3) and (0, 1), (2, 3) are the generators of the automorphism group of the system graph. We proceed to enter this information into the GAP software to extract the necessary information (character table, and irreducible representations). We load the library used for representations in GAP:

```
gap> LoadPackage("repsn");
Repns for Constructing Representations of Finite Groups
Version 3.0.2
Written by
Vahid Dabbaghian

Next, we create the group and obtain its character table:
```
gap> g:=Group([ (2,4), (1,2)(3,4) ]); Group([ (2,4), (1,2)(3,4) ])
gap> Display(CharacterTable(g));
CT1
2 3 2 2 2 3
1a 2a 2b 4a 2c
2P 1a 1a 1a 2c 1a
```
We note that there are five irreducible representations (inferred from character theory) and proceed to
discover these representations:

\[
\begin{align*}
\text{gap} &\rangle \text{IrreducibleAffordingRepresentation(Irr(g)[1]);} \\
&\text{[ (2, 4), (1, 2)(3, 4) ] -> [ [ 1 ] , [ 1 ] ]} \\
\text{gap} &\rangle \text{IrreducibleAffordingRepresentation(Irr(g)[2]);} \\
&\text{[ (2, 4), (1, 2)(3, 4) ] -> [ [ -1 ] , [ -1 ] ]} \\
\text{gap} &\rangle \text{IrreducibleAffordingRepresentation(Irr(g)[3]);} \\
&\text{[ (2, 4), (1, 2)(3, 4) ] -> [ [ -1 ] , [ 1 ] ]} \\
\text{gap} &\rangle \text{IrreducibleAffordingRepresentation(Irr(g)[4]);} \\
&\text{[ (2, 4), (1, 2)(3, 4) ] -> [ [ 1 ] , [ -1 ] ]} \\
\text{gap} &\rangle \text{IrreducibleAffordingRepresentation(Irr(g)[5]);} \\
&\text{[ (2, 4), (1, 2)(3, 4) ] -> [ [ -1, 0 ] , [ 0, 1 ] ] , [ 0, 1 ], [ 1, 0 ] ]} \\
\end{align*}
\]

The output of the function "IrreducibleAffordingRepresentaion()" shows us exactly how the generating
permutations of the automorphism group map to the irreducible representations. We see that four of
the irreducible representation subspaces are one dimensional and the fifth is two dimensional. We are
now ready to insert this information into the Pattern Toolbox in MATLAB and retrieve the pattern of
the system:

```matlab
% Define Permutations
perms = zeros(4, 4, 2);
perms(:,:,1) = [1 0 0 0;
0 0 0 1;
0 0 1 0;
0 1 0 0];
perms(:,:,2) = [0 1 0 0;
1 0 0 0;
0 0 0 1;
0 0 1 0];

% Define Irreducible Representations
irred = cell(5, 1);
irred{1} = zeros(1, 1, 2);
irred{2} = zeros(1, 1, 2);
irred{3} = zeros(1, 1, 2);
irred{4} = zeros(1, 1, 2);
irred{5} = zeros(2, 2, 2);
irred{1}(:,:,1) = 1;
irred{2}(:,:,2) = 1;
irred{2}(:,:,1) = -1;
irred{2}(:,:,2) = -1;
irred{3}(:,:,1) = 1;
irred{3}(:,:,2) = 1;
irred{4}(:,:,1) = 1;
irred{4}(:,:,2) = 1;
irred{5}(:,:,1) = [-1 0 0 1];
irred{5}(:,:,2) = [0 1 1 0];

% Define Character Table
char = [1 1 1 1 1;
1 -1 -1 1 1;
1 -1 1 -1 1;
1 1 -1 -1 1;
2 0 0 0 -2];

circ = Pattern(8, perms, irred, char);
```
We note that the pattern is defined as if each subsystem is one dimensional, i.e. a vertex of the system graph. We take a closer look at the irreducible subspace multiplicities $\eta_i$ of the pattern:

```
>> circ.eta
ans =
1
0
0
1
1
```

We note that the irreducible representations of interest are the first, fourth and fifth, which each have a multiplicity of one. Furthermore, since each subsystem of the system is the same size, we note that by Lemma A.2 the non-zero multiplicities will all increase to two. If we were dealing with a different system with differently sized subsystems, though the irreducible representations and the character table would remain the same, the permutation generators would need to be altered/expanded accordingly. Continuing with the investigation of the system, we can decompose the system matrices with the following function:

```matlab
% Decompose matrices
[Ahat, Atilde, Tx, Tx] = Decompose(circ, circ, A, 0, 0);
[Bhat, Btilde, Ty, Tx] = Decompose(circ, circ, B, Ty, 0);
[Chat, Ctilde, Ty, Tx] = Decompose(circ, circ, C, 0, Tx);
[Dhat, Dtilde, Tz, Tx] = Decompose(circ, circ, D, 0, Tx);
```

The decomposed system matrices take on the form expressed in Theorem 4.1. For example, we show the decomposed form of the state matrix $A$:

\[
\tilde{A} = \begin{bmatrix}
0.69 & -0.69 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 2.08 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & -30.13 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 15.07 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 21.21 & 0 & 21.21 & 0 \\
0 & 0 & 0 & 0 & 0 & 21.21 & 0 & 21.21 \\
0 & 0 & 0 & 0 & 0 & 0 & 10.61 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & -10.61 \\
\end{bmatrix},
\]

\[
\hat{A}_1 = \begin{bmatrix}
0.69 & -0.69 \\
0 & 2.08 \\
\end{bmatrix}, \hspace{1cm} \hat{A}_2 = \emptyset, \hspace{1cm} \hat{A}_3 = \emptyset, \hspace{1cm} \hat{A}_4 = \begin{bmatrix}
-30.13 & 0 \\
0 & 15.07 \\
\end{bmatrix}, \hspace{1cm} \hat{A}_5 = \begin{bmatrix}
21.21 & 21.21 \\
0 & -10.61 \\
\end{bmatrix}.
\]

We may now proceed with the feedback design in the decomposed and reduced form of the matrices.
In this section of the Appendix we give the code used to generate and solve the examples seen throughout this document.

C.1 Stabilization Example Code

The stabilization example code given here was written during the developmental phase of the Pattern Toolbox and does not use it directly. However, much of the code and methodology is the same.
Appendix C. Sample Code

```matlab
36    zero zero zero I zero zero zero;
37    zero zero zero zero zero I zero;
38    zero zero zero zero zero zero I ;
39    \texttt{a2x} = [ I zero zero zero zero zero ;] ; \texttt{class} 2b : \texttt{char} \neq 1
40    zero I zero zero zero zero zero ;
41    zero I zero zero zero zero zero ;
42    zero zero zero zero zero zero I zero ;
43    zero zero zero zero zero I zero ;
44    zero zero zero zero zero zero I zero ;
45    zero zero zero zero zero zero I zero ;
46
47 \% Define other elements within the group
48
49 \texttt{a2} = \texttt{ax} \ast \texttt{a3x} ; \texttt{class} 2a : \texttt{char} \neq 3
50 \texttt{x} = \texttt{a2x+a2} ; \texttt{class} 2b : \texttt{char} \neq 1
51 \texttt{a} = \texttt{a2x+ax} ; \texttt{class} 2c : \texttt{char} \neq 1
52 \texttt{a3} = \texttt{a2x+a3x} ; \texttt{class} 2c : \texttt{char} \neq 1
53
54 \% Test A for commutativity with Group
55
56 Group = \texttt{zeros}(14,14,8) ;
57 Group(:,1,:) = a ;
58 Group(:,2,:) = ax ;
59 Group(:,3,:) = a3x ;
60 Group(:,4,:) = a2x ;
61 Group(:,5,:) = a2 ;
62 Group(:,6,:) = x ;
63 Group(:,7,:) = a ;
64 Group(:,8,:) = a3 ;
65
66
67 \texttt{for} i=1:8
68 \texttt{b} = Group(:,i,:)\ast\texttt{A} ;
69 \texttt{c} = \texttt{A}\ast\texttt{Group}(:,i,:) ;
70 \texttt{if \{\texttt{isequal(b,c)}\}}
71 \texttt{disp(’Matrix A not Commutative with Group’)} ;
72 \texttt{break} ;
73 \texttt{end}
74 \texttt{if \{i==8\}}
75 \texttt{disp(’Matrix A Commutes with Group’)} ;
76 \texttt{end}
77 \texttt{end}
78
80 \% Pattern Decomposition Matrix
81
82 \% Character Table for Reference:
83 \% e | a 3 | a2 | ax | a3x | a | a2 | a3 |
84 \% 1 | a 4 | a 2 | 2b | 2c |
85 \% 1
86 \% 2 | 1 1 1 -1 -1
87 \% 3 | 1 -1 1 -1 1
88 \% 4 | 1 -1 1 1 -1
89 \% 5 | 2 0 -2 0 0
90 \%\%
92 \% begin to generate Irreducible Representations
94
95 \texttt{c1a} = \texttt{e} ;
96 \texttt{c2c} = \texttt{a2x+ax} ;
97 \texttt{c2a} = \texttt{a2} ;
98 \texttt{c2b} = \texttt{ax+a3x} ;
99 \texttt{c4a} = \texttt{a+a3} ;
100 \% canonical decomposition:
101
102 V1 = \texttt{ima(\texttt{c1a+c4a+c2a+c2b+a2} \ast \texttt{c2c})} ;
103 V2 = \texttt{ima(\texttt{c1a+c4a+c2a-a2b+c2e})} ;
104 V3 = \texttt{ima(\texttt{c1a-c4a+c2a-c2b+c2e})} ;
105 V4 = \texttt{ima(\texttt{c1a-c4a+c2a+c2b+c2e})} ;
106 V5 = \texttt{ima(2*a1a-2*a2a)} ;
```
Appendix C. Sample Code

% Irreducible form of group:
% Give irred decopm enumerated as in group above
% v, x, a2x, a2x, a2x, x, a2
triv = [1 1 1 1 1 1 1 1];
kern = [1 1 1 -1 1 -1 -1]; % a2 kernel
dim2 = zeros(2,2,8);

dim2(:,:,1) = [1 0;0 1];
dim2(:,:,2) = [0 1;1 0];
dim2(:,:,3) = [0 -1;-1 0];
dim2(:,:,4) = [-1 0;0 -1];
dim2(:,:,5) = [1 0;0 -1];
dim2(:,:,6) = [0 -1;1 0];
dim2(:,:,7) = [0 1;-1 0];
dim2(:,:,8) = [0 1;0 -1];

% Find the canonical representation of the group
irred = zeros(14,14,8);
canon = zeros(14,14,8);
for i = 1:8
    irred(:,:,i) = blkdiag(triv(i),triv(i),triv(i),triv(i),triv(i),triv(i),kern(i),kern(i),kern(i),kern(i),dim2(:,:,i),dim2(:,:,i),dim2(:,:,i),dim2(:,:,i),dim2(:,:,i),dim2(:,:,i),dim2(:,:,i),dim2(:,:,i),dim2(:,:,i));
canon(:,:,i) = V^(-1)*Group(:,:,i)*V;
end

% Intersection Method:
for i = 1:8
    M = kron(eye(14),canon(:,:,i)')-kron(irred(:,:,i)',eye(14));
    vecP = null(M);
    if i == 1
        vecPspace = vecP; % define starting space
    else
        vecPspace = intspace(vecPspace,vecP); % intersect space with new space
    end
end

[a,b]=size(vecPspace);

% Solution Space Matrix Basis:
Pspace = reshape(vecPspace,[14,14,b]);

% Some magic... look at the space
p = Pspace([11:14,11:14,54]+Pspace([11:14,11:14,55]));

% Define the S
P = blkdiag(eye(10),p);

text = zeros(14,14,8);
for i = 1:8
    text(:,:,i) = P'-1*V^(-1)*Group(:,:,i)*V*P;
end

% Text should be equal to irred

%%% Patterned Decomposition
% Construct pattern decom matrix
T = V*P;

% Find A in patterned space
Apat = T'-1*A*T;

% Find Abuts
Ahat1 = Apat(1:6,1:6);
Ahat2 = Apat(7:10,7:10);
Ahat3 = [Apat(11,11) Apat(11,13);
        Apat(13,11) Apat(13,13)];

%% Input

% define input matrix

B = [0 1 0 0 0 0 0 0 0 0 0 0 0 0;
     0 0 0 0 0 0 1 0 0 0 0 0 0 0;
     0 0 0 0 0 0 0 1 0 0 0 0 0 0;
     0 0 0 0 0 0 0 0 1 0 0 0 0 0;
     0 0 0 0 0 0 0 0 0 1 0 0 0 0;
     0 0 0 0 0 0 0 0 0 0 1 0 0 0;
     0 0 0 0 0 0 0 0 0 0 0 0 1 0;
     0 0 0 0 0 0 0 0 0 0 0 0 0 1];
B = B';

% Check Standard Controllability Notions:
% controllability matrix
Q = ctrb(A,B);

% obtain bases for controllable decom
P1 = ima(Q);
P2 = ortco(P1);
if (rank(P1) == 14)
P = [P1];
else
P = [P1 P2];
end
end

Pcanon = P;
[a,ncLen] = size(P2);
% decompose
A_cd = P'*A*P;
ncSpec = diag(A_cd(14-ncLen+1:14,14-ncLen+1:14));
% check spectrum
for i = 1:ncLen
    disp('System is not Stabilizable');
    break;
end
if (i == ncLen)
disp('System is Stabilizable');
end
end
end

%% Automorphism of input space

eu = eye(5);
axu = blkdiag(eye(3), [0 1 1 0]);
asu = [1 0 0 0 0;
      0 1 0 0;
      0 0 1 0;
      0 0 0 1];
as2u = [1 0 0 0 0;
       0 0 1 0;
       0 0 0 1;
       0 1 0 0;
       0 0 1 0];

% Define other elements within the group
s2u = axu*a3s_u;
su = axu*a3s_u;
a3s_u = a2s_u*a3s_u;
a3u = a2s_u*a3s_u;
gGroup = zeros(5,5,8);
gGroup(:,:,1) = eu;
gGroup(:,:,2) = axu;
gGroup(:,:,3) = asu;
gGroup(:,:,4) = a2s_u;
gGroup(:,:,5) = a2u;
%begin to generate Irreducible Representations

c1a = e; 
c2c = a2 + a3; 
c2a = a2; 
c2b = a2 + a3; 
c4a = a + a3; 

% canonical decomposition: 

V1u = ima(c1a+c4a+c2a+c2b+c2c); 
V2u = ima(c1a+c4a+c2a−c2b−c2c); 
V3u = ima(c1a−c4a+c2a−c2b+c2c); 
V4u = ima(c1a−c4a+c2a+c2b−c2c); 
V5u = ima(2∗c1a−2∗c2a); 
Vu = [V1u V4u V5u]; 

%find the canonical representation of the group 

irred_u = zeros(5,5,8); 
canon_u = zeros(5,5,8); 
for i = 1:8 
    irred_u(:, :, i) = blkdiag(triv(i),triv(i),kern(i),dim2(:, :, i)); 
    canon_u(:, :, i) = Vu−1*Group(:, :, i)*Vu; 
end 

% Intersection Method: 

for i = 1:8 
    M = kron(eye(5),canon_u(:, :, i))−kron(irred_u(:, :, i)',eye(5)); 
    vecP = null(M); 
    if i == 1 
        vecPspace = vecP; %define starting space 
    else 
        vecPspace = intS(vecPspace, vecP); %intersect space with new space 
    end 
end 

[a, b] = size(vecPspace); 

% Solution Space Matrix Basis: 
Pspace = reshape(vecPspace,5,5,b); 

%more magic... give a Deus 
P = blkdiag(eye(3), Pspace(4:5,4:5,6)); 

%define Pattern Decomposition for U space 
Tu = Vu*P; 

%decompose B 
Bpat = T−1*B*Tu; 

%reduced form: 
Bhat1 = Bpat([1 6 1 2]); 
Bhat2 = Bpat([7 10 3]); 
Bhat3 = [Bpat([11 4]); Bpat([13 4])]; 

%%% Pole Placement 
Khat1 = place(Ahat1,Bhat1, [−1 −2 −3 −4 −5 −6]); 
Khat2 = place(Ahat2,Bhat2, [−7 −8 −9 −10]); 
Khat3 = place(Ahat3,Bhat3, [−11 −12]); 
K = Tu*bkldiag(Khat1,Khat2,kron(Khat3,eye(2)))*T−1; 
F = A*inv(K); 
for i = 1:8 
    b = Group(:, :, i)*A;
c = A*Group(:, :, i);
if (~isequal(b, c))
    disp('Feedback System not Commutative with group');
    break;
end
if (i==8)
    disp('Feedback System Commutes with Group');
end
end
disp('The eigenvalues of the feedback system are as follows: ');
disp(eig(F))

%% Test Random Matrix Search

% Irreducible form of group:
% give irred decemp enumerated as in group above
% e, ax, a2x, ax, a3, a2, x, a, a3
triv = [1 1 1 1 1 1 1 1];
kern = [1 1 1 -1 1 1 1 -1];
% <a2x> kernel
dim2 = zeros(2, 2, 8);
dim2(:, :, 1) = [1 0; 0 1];
dim2(:, :, 2) = [0 1; 1 0];
dim2(:, :, 3) = [0 -1; -1 0];
dim2(:, :, 4) = [-1 0; 0 1];
dim2(:, :, 5) = [-1 0; 0 -1];
dim2(:, :, 6) = [1 0; 0 -1];
dim2(:, :, 7) = [0 -1; 1 0];
dim2(:, :, 8) = [0 1; -1 0];

% find the canonical representation of the group
irred = zeros(14, 14, 8);
for i = 1:8
    irred(:, :, i) = blkdiag(triv(i), triv(i), triv(i), triv(i), triv(i), triv(i), kern(i), kern(i), kern(i), kern(i), dim2(:, :, i), dim2(:, :, i));
end

% now need P for Group*P = P*irred, sylvester equation for all group
% elements. we use a little tensor calculus to come up with a matrix
% solution space and intersect the spaces to find a final solution space

% Intersection Method:
for i = 1:8
    M = kron(eye(14), Group(:, :, i)) - kron(irred(:, :, i)', eye(14));
    vecP = null(M);
    if i == 1
        vecPspace = vecP; % define starting space
    else
        vecPspace = intsect(vecPspace, vecP); % intersect space with new space
    end
end
[a, b] = size(vecPspace);
% Solution Space Matrix Basis:
Pspace = reshape(vecPspace, 14, 14, b);
% Add all of the elements of the matrix basis with random coefficients
Prand=rand()*Pspace(:, :, 1);
for i = 2:b
    Prand = Prand + rand()*Pspace(:, :, i);
end
% double check stuff
test = zeros(14, 14, 8);
for i = 1:8
    test(:, :, i) = Prand' * Group(:, :, i) * Prand;
end
% test should be equal to irred
Appendix C. Sample Code

C.2 SMFP Example Code

The following is the MATLAB code used to solve the SMFP example:

```matlab
%% SMFP - Five subsystem ring
clear all

% setup system
A = [-2 2000 5 0000 -0.3708 0.0000 0 0 0 0 -0.3708 -0.0000;...
    -0.3708 0.0000 -2 2000 5 0000 -0.3708 -0.0000 0 0 0 0;...
    0 -0.4944 0 5 4000 0 -0.4944 0 0 0 0;...
    0 0 -0.3708 -0.0000 -2 2000 5 0000 -0.3708 -0.0000 0 0;...
    0 0 0 0 -0.4944 0 5 4000 0 -0.4944 0 0;...
    0 0 0 0 0 -0.4944 0 5 4000 0 -0.4944 0 0;...
    -0.3708 -0.0000 0 0 0 0 0 -0.3708 0.0000 -2 2000 5 0000;...
    0 -0.4944 0 0 0 0 0 0 -0.4944 0 5 4000];
C = [0 1 0 0 0 0 0 0 0 0;...
     0 0 0 1 0 0 0 0 0 0;...
     0 0 0 0 0 1 0 0 0 0;...
     0 0 0 0 0 0 0 1 0 0;...
     0 0 0 0 0 0 0 0 1];
B = [0 0 0 0 0;...
     1 0 0 0 0;...
     0 1 0 0 0;...
     0 0 0 0 0;...
     0 0 1 0 0;...
     0 0 0 0 0;...
     0 0 0 1 0;...
     0 0 0 0 0;...
     0 0 0 0 1];

% Setup Pattern
perm = zeros(5,5,2);
perm(:,:,1) = [0 0 0 1 0;0 0 1 0 0;0 1 0 0 0;1 0 0 0 0;0 0 0 0 1];
perm(:,:,2) = [0 1 0 0 0;1 0 0 0 0;0 0 0 0 0;1 0 0 0 0;0 0 0 0 0];
irred = cell(4,1);
irred{1} = zeros(1,1,2);
irred{2} = zeros(1,1,2);
irred{3} = zeros(2,2,2);
irred{4} = zeros(2,2,2);
irred{1}(:,:,1) = 1;
irred{1}(:,:,2) = 1;
irred{2}(:,:,1) = -1;
irred{2}(:,:,2) = -1;
irred{3}(:,:,1) = [1;0;0;1];
irred{3}(:,:,2) = [cos(2*pi/5);sin(2*pi/5);sin(2*pi/5);cos(2*pi/5)];
irred{4}(:,:,1) = [1;0;0;1];
irred{4}(:,:,2) = [cos(2*pi/5);sin(2*pi/5);sin(2*pi/5);cos(2*pi/5)];
A3 = (-1+sqrt(5))/2;
A4 = (-1-sqrt(5))/2;
charTable = [1 1 1 1 1;1 2 0 A3 A4;0 A4 2 A3];
FiveRing=Pattern(10,perm,irred,charTable);

% Decompose
[Ahat,Atilde,Tx,Tz] = Decompose(FiveRing,FiveRing,A,0,0);
[Bhat,Btilde,Tx,Tz] = Decompose(FiveRing,FiveRing,B,Tx,0);
[Chat,Ctilde,Ty,Tx] = Decompose(FiveRing,FiveRing,C,Tx);

% Design stabilizing K
K1 = [0 -7];

% System 1
C1 = [0 0] => K1 = [0 0]

% System 3
Khat = [0 -7];
```

Appendix C. Sample Code

C.3 OSP Example Code

The following is the MATLAB code used to solve the OSPP example:

```matlab
%% OSP Example – Chain System

% general properties of pattern obtained from GAP
irred = cell(2,1);
irred{1}(:,1) = 1;
irred{2}(:,1) = -1;
char = [1 1; 1 -1];

% setup pattern for X space
permx(:,:,1) = zeros(10,10,1);
b = [0 0 0 0 1; 0 0 0 1; 0 1 0 0; 1 0 0 0];
permx(:,:,1) = kron(b, eye(2));
chain_x = Pattern(2, permx, irred, char);

% setup pattern for v space
permv(:,:,1) = zeros(6,1);
a = [0 0 1; 0 1 0; 1 0 0];
permv(:,:,1) = kron(a, eye(2));
chain_v = Pattern(2, permv, irred, char);

% setup pattern for input space
permu(:,:,1) = zeros(4,1);
c = [0 1; 1 0];
permu(:,:,1) = kron(c, eye(2));
chain_u = Pattern(2, permu, irred, char);

% Set up system
A2 A3 A1 A3 A2
O-------->O<-------- O -------->O<-------O
| A4 | A5 A5 | A4 |
```
Appendix C. Sample Code

```matlab
44 \% B1 D D B1
45
46 A1 = [1 4;
47 0 2];
48 A2 = [3 1;
49 0 4];
50 A3 = [-1 10;
51 0 -5];
52 A4 = eye(2);
53 A5 = [0 1;
54 0 1];
55 I = eye(2);
56 O = zeros(2);
57
58 A = [A3 O O O O;
59 A4 A2 A5 O O;
60 O O A1 O O;
61 O O A5 A2 A4;
62 O O O O A3];
63 B = [I O;
64 O O;
65 O O;
66 O O;
67 O I];
68 D = [O I O O;
69 O O O I];
70 \% V space insertion map
71 S = [I O O;
72 O O O;
73 O I O;
74 O O O;
75 O O I];
76
77 \% Decompose the systems
78 [Ahat, Atilde, Tx, Tx] = Decompose(chain_x, chain_x, A, 0, 0);
79 Tx
80 [Bhat, Btilde, Tx, Tu] = Decompose(chain_x, chain_u, B, Tx, 0);
81 Tx
82 [Shat, Sstilde, Tx, Tv] = Decompose(chain_x, chain_v, S, Tx, 0);
83 Tx
84
85 \% Perform aep on subsystems
86
87 \% Reduced Subsystem 1
88 \% find
89 Vstar1 = mainco(Ahat{1}, Bhat{1}, Shat{1});
90 \% get modal spaces
91 [Xs, Xu, X0] = subsplit(Ahat{1});
92 Xu = ima([Xu X0]);
93 \% controllable space
94 C = mininv(Ahat{1}, Bhat{1});
95 \% display to verify containment
96 Q = ima([C Vstar1]);
97 disp('verify control invariance of unstable subspace, subsystem 1')
98 disp('unstable subspace')
99 disp(Xu);
100 disp('\langle A|B0 + V* ')
101 disp(Q);
102 disp('\langle Xc \langle A|B0 + V* ')
103 if (rank([Q Xu]) == rank(Q));
104 disp('yes')
105 else
106 disp('no');
107 end
108
109 \% Feedback Invariance possible . Generate friend K
110 F1 = effe(Ahat{1}, Bhat{1}, Vstar1);
111
112 \% Now stabilize the other part of the system
113 R1 = [eye(4), zeros(2,4)];
114 T1 = [Vstar1 R1];
115 T1 = inv(T1);
```
Appendix C. Sample Code

P1 = T1(:,3:6,:);
P1 = place(P1*(Ahat{1}+Bhat{1})*P1*R1,P1*Bhat{1},[-2 -3 -6 -4]);

Khat1 = P1*K1*P1;
Vstar2 = mainco(Ahat{2},Bhat{2},Shat{2});

[Xs,Xu,X0] = subsplit(Ahat{2});
Xun2 = ima([Xu X0]);
C2 = mininv(Ahat{2},Bhat{2},[-5 -8 -6 -2]);
Ktilde = blkdiag(Khat1,Khat2);
K = Tu*Ktilde*inv(Tx);
Aclhat1 = Ahat{1}+Bhat{1}*Khat1;
Aclhat2 = Ahat{2}+Bhat{2}*Khat2;

if isequal(round(chain.x.perm(:,1)*(A+B*K).*1000), round((A+B*K)*chain.x.perm(:,1).*1000))
    disp('Closed Loop System is patterned');
else
    disp('Closed Loop System is NOT patterned');
end
x0 = [10 13 4 57 8 32 2 9 43 22];
figure();
title('Non-Output States vs Time');
figure();
title('Output States vs Time');
Vstar = mainco(A,B,Shat{2})
Bibliography


