2018-06-01

Network Specializations, Symmetries, and Spectral Properties

Dallas C. Smith
Brigham Young University

Network Specializations, Symmetries, and Spectral Properties

Dallas C. Smith

A dissertation submitted to the faculty of
Brigham Young University
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

Benjamin Webb, Chair
Emily Evans
Todd Fisher
Christopher Grant
Jared Whitehead

Department of Mathematics

Brigham Young University

ABSTRACT

Network Specializations, Symmetries, and Spectral Properties

Dallas C. Smith
Department of Mathematics, BYU
Doctor of Philosophy

In this dissertation, we introduce three techniques for network sciences. The first of these techniques is a series of new models for describing network growth. These models, called *network specialization models*, are built with the idea that networks grow by specializing the function of subnetworks. Using these models we create theoretical networks which exhibit well-known properties of real networks. We also demonstrate how the spectral properties are preserved as the models grow. The second technique we describe is a method for decomposing networks that contain automorphisms in a way that preserves the spectrum of the original graph. This method for graph (or equivalently matrix) decomposition is called an *equitable decomposition*. Previously this method could only be used for particular classes of automorphisms, but in this dissertation we have extended this theory to work for every automorphism. Further we explain a number of applications which use equitable decompositions. The third technique we describe is a generalization of network symmetry, called *latent symmetry*. We give numerous examples of networks which contain latent symmetries and also prove some properties about them.

Keywords: Networks Growth Model, Specialization, Equitable Partition, Automorphism, Network Symmetry, Isospectral Network Reduction

Acknowledgments

First I would like to thank my advisor Dr. Benjamin Webb for coaching me through the program, having brilliant ideas and always being so positive.

I am also so grateful for the hard work of all the other co-authors on the papers I have been a part of including: Amanda Francis, Derek Sorensen, and Leonid Bunimovich. Also I am very grateful to Wayne Barrett for his helpful suggestions for the equitable decomposition papers.

I am especially grateful for my sweet wife, Amy. I could not have done any of this without her love, devotion and patience.

Contents

List of Figures

Chapter 1. Introduction

Today networks can be found in just about every field of study, from food webs to the Internet, from social sciences to statistical physics, from facebook to bibliographies. Though graph theory has been around since Leonhard Euler solved the *Seven Bridges of Königsburg* problem in 1736 [1], network theory has only recently become a field of study in its own right. This is partly due to the large amount of network data that has become available via the Internet and social networks. When scientists analyzed this new data, they found a number of common properties which were very different from the properties expected in a randomly generated graph (for instance the Erdös-Rényi graph [2]). These observations sparked an interest in understanding how and why networks form and function.

Early on, many network scientists recognized that spectral properties were useful in analyzing networks. There are a number of ways that spectral properties of a network can be used to analyze the structure of a network. Spectral properties are quantities related to the eigenvalues and eigenvectors of a matrix associated with the network. An example of the application of a spectral property is a centrality measure called *eigenvector centrality*. Eigenvector centrality ranks vertices at a level of higher importance depending on the level of importance of their neighbors. Eigenvector centrality is calculated by ranking vertices according to their corresponding entry in the leading eigenvector associated with the network's adjacency matrix. Eigenvector centrality is widely used. For example, Google uses a variation of eigenvector centrality, called 'Page-rank centrality', when ranking web-pages in a World Wide Web search.

Another example of analyzing a graph via its spectral properties is community detection. Determining which members of a network are most tightly connected and/or alike is a significant and often difficult problem to solve. There are multiple methods used in community detection which utilize spectral properties. For instance, the most basic algorithm for detecting communities relies on looking at the eigenvector of the graph Lapacian matrix which is associated with the smallest

non-zero eigenvalue [3]. It is remarkable that we can use these spectra to find potential communities or natural clusters in a network.

A third example of using spectral properties of a graph to analyze a network is stability analysis of a dynamical network. This example is discussed in section 2.6, and explains that the spectral radius of a network (the largest magnitude of all the eigenvalues) can be used to determine whether particular dynamics on a network are stable.

This dissertation is a collection of three tools for analyzing various types of networks and graphs. The common thread which weaves each of these topics together is that each is concerned with the graph's spectral properties. In Chapter 2, we develop a new model for network growth which adds eigenvalues to the graph's spectrum in a controlled way as the network grows. In Chapter 3, we develop a technique to decompose a network which contains a symmetry in a way which preserves the eigenvalues of the network. Finally, in Chapter 4, we develop a generalization of symmetry which is related to automorphisms in an *isospectral reduction* of the graph. An isospectral graph reduction is a smaller graph with essentially the same spectrum (see Section 2.5).

One of the most important features observed in real networks is that, as a network's topology evolves so does the network's ability to perform various complex tasks. To explain this, it has also been observed that as a network grows certain subnetworks begin to specialize the function(s) they perform. In Chapter 2, we use this notion of *specialization* to design a new model of network expansion. This new model is significant as it has the property that as a network is grown using this specialization method its topology becomes increasingly sparse, modular, and hierarchical, each of which are important properties observed in real networks. The procedure outlined is also highly flexible in that a network can be specialized over any subset of its elements. This flexibility allows those studying specific networks the ability to search for mechanisms that describe their growth. For example, we find that by randomly selecting these elements a network's topology acquires some of the most well-known properties of real networks including the small-world property, disassortativity, and a right-skewed degree distribution. Beyond this, we show how this

model can be used to generate networks with real-world like clustering coefficients and power-law degree distributions, respectively.

In addition to creating graphs with many real-world properties, this method of specialization also preserves basic spectral properties of a network. In Section 2.6, we show that the network maintains certain dynamic properties, specifically stability, as its structure evolves, under mild conditions, which links this model of specialization to the robustness of network function.

The theory of equitable decompositions introduced in [4] shows that if a graph (network) has a particular type of symmetry, i.e., a uniform or basic automorphism ϕ, it is possible to use ϕ to decompose a matrix M appropriately associated with the graph. The result is a number of strictly smaller matrices whose collective eigenvalues are the same as the eigenvalues of the original matrix M. It is significant that this method is based on network symmetries as having many symmetries is a hallmark of real networks [5]. In Chapter 3, we extend the theory to include all automorphisms, not just those that are uniform and basic. We also show that not only can a matrix M be decomposed but that the eigenvectors of M can also be equitably decomposed. After proving in Sections 3.3, 3.4 and 3.5 that this can always be done, we go through a number of applications of equitable decompositions. For instance, we show that under mild conditions, if a matrix M is equitably decomposed the resulting divisor matrix, which is the divisor matrix of the associated equitable partition, will have the same spectral radius as the original matrix M. Thus this process could be used to reduce the complexity in finding a matrix's spectral radius. We also describe how an equitable decomposition affects the Gershgorin region $\Gamma(M)$ of a matrix M, which can be used to localize the eigenvalues of M. We show that the Gershgorin region of an equitable decomposition of M is contained in the Gershgorin region $\Gamma(M)$ of the original matrix. We demonstrate on a real-world network that by a sequence of equitable decompositions it is possible to significantly reduce the size of a matrix' Gershgorin region. Thus, given a graph with a symmetry, there is a process which could improve simple estimates for a network's eigenvalues and spectral radius. At the end of Chapter 3 we show how graphs can be decomposed over separable automorphisms to give a more visual interpretation of an equitable decomposition of a graph.

We know symmetries are ubiquitous in real networks and often characterize network features and functions. The last topic explored in this dissertation in Chapter 4 is a new notion of symmetry. Here we present a generalization of network symmetry called *latent symmetry*, which is an extension of the standard notion of symmetry. Latent symmetries are defined in terms of standard symmetries in a reduced version of the network. One unique aspect of latent symmetries is that each one is associated with a *size*, which provides a way of discussing symmetries at multiple scales in a network. We demonstrate several examples of networks (graphs) which contain latent symmetry, including a number of real networks. Further we use numerical experiments to show that latent symmetries are found more frequently in graphs which were built using preferential attachment, a standard model of network growth, when compared to non-network like (Erdős-Rényi) graphs. We also prove that if vertices in a network are latently symmetric, then they must have the same eigenvector centrality, similar to vertices which are symmetric in the standard sense. This suggests that the latent symmetries present in real-networks may serve the same structural and functional purpose standard symmetries do in these networks. We conclude from these facts and observations that *latent symmetries* are present in real networks and provide useful information about the network. Because latent symmetries can appear at multiple scales, information derived from latent symmetries is potentially more useful than information derived from only standard symmetries. Finally, at the end of this chapter, we prove that latently symmetric vertices are also co-spectral.

Throughout this dissertation, we will use the terms "network" and "graph" interchangeably, as a graph is the only model we use in each chapter to study the structure of a network.

1.1 Notation

In this section we will describe some basic concepts connected to networks and graph theory. We will also establish the notation which will be used through the dissertation.

The standard method used to describe the topology of a network is a graph. Here, a *graph* $G = (V, E, \omega)$ is composed of a *vertex set* V, an *edge set* E, and a function ω used to weight the

edges E of the graph. The vertex set V represents the *elements* of the network, while the edges E represent the links or *interactions* between these network elements. The vertices of a graph are typically represented by points in the plane and an edge by a line or curve in the plane that connects two vertices.

In some networks, it is useful to define a direction to each interaction. This is the case in which an interaction between two network elements influences one but not the other. For instance, in a citation network, in which network elements are papers and edges represent whether one paper cites another, papers can only cite papers that have already been written. Thus each edge has a clearly defined direction. We model this type of network as a *directed* graph in which each edge is directed from one network element to another. If this does not apply, the edges are not directed and we have an *undirected* graph. The *degree* of a vertex is the number of edges which connect to the vertex. For a directed graph, we use the terms *in-degree* and *out-degree*, to refer to the number of edges pointing into the vertex and out of the vertex, respectively.

The weights of the edges given by ω measure the *strength* of these interactions. Some examples of weighted networks include: social networks where weights corresponds to the frequency of interaction between actors, food web networks where weights measure energy flow, or traffic networks where weights measure how often roads are used [6]. Throughout this work we will mostly consider networks with real-valued edge weights because they represent the majority of weighted networks considered in practice. Though it is worth mentioning that much of the theory we present throughout the chapter is valid for more general edge weights, e.g. complex-valued or more complicated weights (see for instance [7]).

Let $G = (V, E, \omega)$ be a weighted graph on n vertices representing a network. Its *weighted adjacency matrix*, $M = M(G)$, is an $n \times n$ matrix whose entries are given by

$$M_{ij} = \begin{cases} \omega(e_{i,j}) \neq 0 & \text{if } e_{i,j} \in E \\ 0 & \text{otherwise} \end{cases}$$

where $e_{i,j}$ is the edge from vertex i to vertex j. An *unweighted* graph can be considered to be a special case of a weighted graph where all edge weights are equal to 1. Moreover, the weighted adjacency matrix of an undirected graph is symmetric since each edge can be thought of as a directed edge oriented in both directions. Figure 4.1 gives an example of an unweighted, undirected graph with its corresponding adjacency matrix. In practice there are a number of matrices that are often associated with a given graph G.

Another common matrix associated with a graph G is the Laplacian matrix $L = L(G)$. To define the Laplacian matrix of a *simple graph* G, i.e., an unweighted undirected graph without loops, let $D_G = \text{diag}[\deg(1), \ldots, \deg(n)]$ denote the degree matrix of G, where $\deg(i)$ denotes the degree of vertex i. Then the Laplacian matrix $L(G)$ is the matrix $L(G) = D_G - M(G)$.

For an $n \times n$ matrix $M = M(G)$ associated with a graph G we let $\sigma(M)$ denote the *eigenvalues* of M. For us $\sigma(M)$ is a multiset with each eigenvalue in $\sigma(M)$ listed according to its multiplicity.

It is also worth noting that there is a one-to-one relation between weighted graphs (networks) and their corresponding weighted adjacency matrices $M \in \mathbb{R}^{n \times n}$ meaning that there is no more information presented in one than the other. Often it is more convenient to work with matrices instead of graphs, though both are useful ways to represent network structure. Graphs are typically used for network visualization while matrices are better suited for network analysis [6]. Throughout this work we will use graphs and matrices without ambiguity to refer to the "graph of the network" and the "matrix associated with the network."

Chapter 2. Specialization Models for Network Growth

2.1 Overview and Background

Networks studied in the biological, social, and technological sciences perform various tasks, which are determined by both the network's topology as well as the network's dynamics. In the biological setting gene regulatory and metabolic networks allow cells to organize into tissues, and tissues

into organs whose dynamics are essential to the network's function [8, 9], e.g. a beating heart in a circulatory network. Neuronal networks are responsible for complicated processes related to cognition and memory, which are based on the network's structure of connections as well as the electrical dynamics of the network's neurons [10]. Social networks such as Facebook, Twitter, the interactions of social insects [11, 12], and professional sports teams [13] function as a collection of overlapping communities or a single unified whole based on the underlying topology and hierarchies present within the network's social interactions. Technological networks such as the Internet together with the World Wide Web allow access to information based on the topology of network links and the network's dynamic ability to route traffic.

In the study of networks a network's *topology* refers to the network's structure of interactions while a network's *dynamics* is the pattern of behavior exhibited by the network's elements [10]. It is worth emphasizing that real-world networks are not only dynamic in terms of the behavior of their elements but also in terms of their topology, both of which affect the network's function. For example, the World Wide Web has an ever changing structure of interactions as web pages and the hyperlinks connecting these pages are updated, added, and deleted (see [14] for a review of the evolving topology of networks).

These models are devised to create networks that exhibit some of the most widely observed features found in real networks. This includes (i) having a degree distribution that is right-skewed and monotonically decreasing beyond a certain point, e.g. power-law distributions, (ii) having a *disassortative neighbor* property where vertices with high (low) degree have neighbors with low (high) degree, (iii) having a high *clustering coefficient* indicating the presence of many triangles within the network, and (iv) having the *small-world property*, meaning that the average distance between any two network elements is logarithmic in the size of the network (see [6] for more details on these properties).

Aside from these structural features, one of the hallmarks of a real network is that, as its topology evolves, so does its ability to perform complex tasks. This happens, for instance, in neural networks, which become more modular in structure as individual parts of the brain become

increasingly specialized in function [15]. Similarly, gene regulatory networks can specialize the activity of existing genes to create new patterns of gene behavior [16]. In technological networks such as the Internet, this differentiation of function is also observed and is thought to be driven by the need to handle and more efficiently process an increasing amount of information.

In this section we propose a very different class of models than those described above. These models are built on the notion of specialization. We refer to these as *specialization models* of network growth. These models are based on the fundamental idea that as a network specializes the function of one or more of its *components*, i.e. a subnetwork(s) that performs a specific task, the network first creates a number of copies of this component. These copies are attached to the network in ways that reflect the original connections the component had within the network only sparser. The new copies "specialize" the function of the original component in that they carry out only those functions requiring these specific connections (cf. Figure 2.1).

The components which are specialized in this growth process form *motifs*, i.e. statistically significant structures in which particular network functions are carried out. As copies of these motifs are placed throughout the network via the process of specialization the result is an increase in the network's modularity [17]. Moreover, repeated application of this process results in a *hierarchical topology* in which this modular structure appear at multiple scales. Because new components are far less connected to the network than the original components the result is an increasingly sparse network topology. Hence the network acquires a more modular [18], hierarchical [19, 20], and sparse topology [21, 22], each of which is a distinguishing feature of real networks when compared, for instance, to random graphs (see [14] Section 6.3.2.1).

Importantly, our model of network growth is extremely flexible in that a network can be uniquely specialized over any subset B of its elements. We refer to any such subset B as a network *base*. Since B can be any subset of a network's elements there is a significant number of ways in which a network can be specialized. This is why we refer to multiple network specialization models. An obvious application is, given a particular network, to find a base over which this network can be specialized that evolves the network's topology in a way that models its ob-

served growth. Finding a *rule* τ that generates this base is a natural objective of a network scientist who wishes to use this model to investigate their particular network(s) of interests. The reason is that finding such a rule suggests a mechanism for the particular network's growth that can then be tested against the growth of the actual network.

A particularly simple rule we consider in section 2.3 is the rule $r = r_p$ for $p \in (0, 1)$ that uniformly selects a random network base consisting of p percent of the network's elements. Under this specialization rule we find that an initial network evolves under a sequence of specializations into a network that has (i) a right-skewed degree distribution that is monotonically decreasing, (ii) a *disassortative degree* property, and (iii) the *small-world property*. Hence, this random variant of the specialization model appears to capture a number of the well-know properties observed in real networks (see Example 2.5). To our knowledge this is the only such class of models to capture these properties along with creating an increasingly sparse, modular, and hierarchical network topology.

Since one of the main points of this chapter is that different specialization rules lead to different growth models we consider two other rules in Section 2.3. The first is the rule that selects $p \in (0, 1)$ percent of those vertices that are not part of a triangle. The result of this choice is a network whose clustering coefficient is similar to many real-world networks in that it remains relatively large as the network evolves under this rule (see example 2.6). The second additional rule chooses the top and bottom twenty-five percent of the network's vertices that have the largest in-degree and a random fraction of the remaining vertices. The result of repeatedly using this specialization rule is a network with a power-law like degree distribution, i.e. a *scale-free* network, (see example 2.7).

Additionally, we show how specialization rules can be used to compare the topology of different networks. Specifically, two graphs G and H are considered to be *similar* to each other with respect to a rule τ if they specialize to the same graph under τ. This notion of similarity, which we refer to as *specialization equivalence*, can be used to partition any set of networks into those that are similar, i.e. are specialization equivalent, with respect to a given rule τ and those that are not (see Section 2.4, Theorem 2.8). One reason for designing such a rule τ is that typically it is

not obvious that two different graphs are in some sense equivalent. That is, two networks may be similar but until both are specialized with respect to τ this may be difficult to see. Here we show that by choosing an appropriate rule τ one can discover this similarity (see example 2.9). Of course, it is important that this rule be designed by the particular biologist, chemist, physicist, etc. to have some significance with respect to the nature of the networks under consideration.

Beyond the structural properties of specialized networks we also consider how specialization affects the spectral properties of a network. This is done by using the theory of isospectral network transformations [23, 7], which describes how certain changes in a network's structure affect the network's *spectrum*, i.e. the eigenvalues associated with the network's weighted or unweighted adjacency matrix (see Section 2.5). We show that if a network is specialized then the eigenvalues of the resulting network are those of the original network together with the eigenvalues of the specialized components (see Section 2.5, Theorem 2.10). Additionally, using the theory of isospectral transformations we show that the eigenvector centrality of the core vertices of a network remain unchanged as the network is specialized (see Section 2.5, Theorem 2.15).

As a network's dynamics can be related to the network's spectrum we can in certain cases determine how specialization of a network will affect the network's dynamics. Specifically, we consider stability, a property observed in a number of important systems including neural networks [24, 25, 26, 27, 28], network epidemic models [29], and in the study of congestion in computer networks [30].

In a *stable* dynamical network the network's state always evolves to the same unique state irrespective of the network's initial state. We show that if a dynamical network is intrinsically stable, which is a stronger form of this standard notion of stability (see definition 2.18 or [31] for more details), then any specialized version of this network will also be intrinsically stable (see Section 2.6, Theorem 2.21). Hence, network growth, at least via specialization will not destabilize the network's dynamics if the network has this stronger version of stability. This is important in many real world applications since network growth can have a destabilizing effect on a network. A

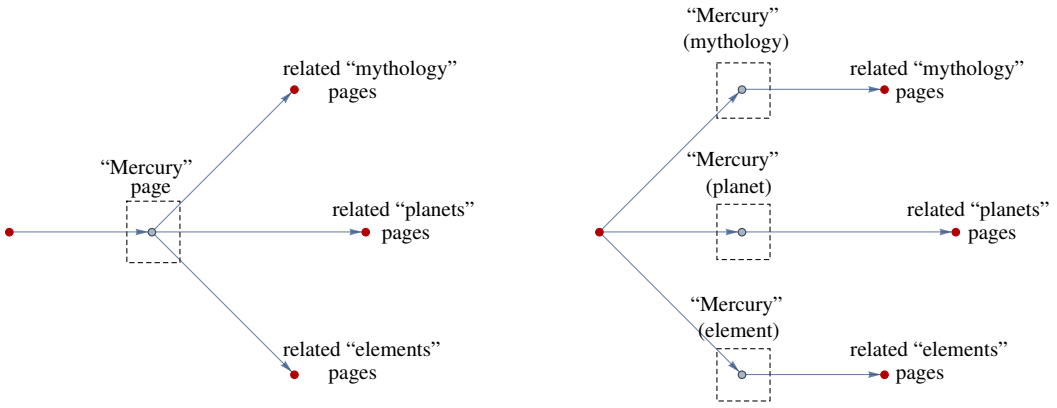
well-known example of this phenomena is cancer, which is the abnormal growth of cells that can impair the function and ultimately lead to the failure of specific biological networks.

Although networks exhibit many other types of dynamics the reason we study stability is because of its simplicity and use it as a first step to understanding the interplay of network dynamics and network growth. Importantly, our results suggest that if the network's growth is due to specialization and the network has a strong form of dynamics, e.g. intrinsic stability, the network can maintain this type of dynamics. The ability to maintain dynamics is an important feature found in real networks that maintain a specific function as their structure evolves, e.g. the cellular network of a beating heart.

The chapter is organized as follows. In Section 2.2 we introduce the specialization model and the notion of specialization rules. In Section 2.3 we show that if a network is randomly specialized over a fixed percentage of its elements the result is a network that has many properties found in real-world networks including the small-world property, disassortativity, and right-skewed degree distributions. We show that by using slightly more sophisticated rules we can also create networks that have other properties such as real-world like clustering coefficients and power-law degree distributions, respectively. In Section 2.4 we describe the notion of specialization equivalence and how this notion can be used to compare the structure of different networks. In Section 2.5 we explain our main results regarding the spectral properties of network specializations, which describe the effect that specialization has on the eigenvalues and eigenvectors associated with a network. In Section 2.6 we use graph specializations to evolve the structure of a dynamical system used to model network dynamics. We show that if such a network is intrinsically stable then any specialized version of the network is also intrinsically stable. Finally Section 2.7 contains some concluding remarks.

2.2 Specialization Model of Network Growth

In this section we will model networks using the notation set forth in Section 1.1. In this section edges E of a graph can either be *directed* or *undirected*, *weighted* or *unweighted*. Without loss



Undifferentiated "Mercury" Wikipedia Page

Disambiguated "Mercury" Wikipedia Pages

Figure 2.1: This figure shows the effect of disambiguating the Wikipedia page on "Mercury" into three distinct webpages, which are respectively Mercury the mythological figure, Mercury the planet, and mercury the element.

in generality, we consider those graphs that have weighted directed edges. The reason is that an undirected edge is equivalent to two directed edges pointing in opposite directions and any unweighted edge can be weighted by giving the edge unit weight.

As mentioned in the introduction, one of the hallmarks of real networks is that as a network evolves so does its ability to perform various tasks. It has been observed that to accomplish this a network will often specialize the tasks performed by one or more of its components, i.e. subnetworks. As motivation for our model of network growth we give the following example of network specialization.

Example 2.1. (Wikipedia Disambiguation) The website Wikipedia is a collection of webpages consisting of articles that are linked by topic. The website evolves as new articles are either added, linked, or modified within the existing website. One of the ways articles are added, linked, or modified is that some article within the website is disambiguated. That is, if an article's content is deemed to refer to a number of distinct topics then the article can be *disambiguated* by separating the article into a number of new articles, each on a more specific or *specialized* topic than the original.

Wikipedia's own page on disambiguation gives the example that the word "Mercury" can refer to either Mercury the *mythological figure*, Mercury the *planet*, or mercury the *element* [32]. To emphasize these differences the Wikipedia page on Mercury has since been disambiguated into three pages on Mercury; one for each of these subcategories. Users arriving at the Wikipedia "Mercury" page [33] are redirected to these pages (among a number of other related pages).

The result of this disambiguation is shown in Figure 2.1. In the original undifferentiated Mercury page users arriving from other pages could presumably find links to other "mythology", "planet", and "element" pages (see Figure 2.1, left). After the page was disambiguated users were linked to the same pages but only those relevant to the particular "Mercury" page they had chosen (see Figure 2.1, right). In terms of the topology of the network, this disambiguation results in the creation of a number of new "Mercury" pages each of which is linked to a subset of pages that were linked to the original "Mercury" page. Growth via disambiguation is a result of the new "copies" of the original webpage.

However, what is important to the functionality of the new specialized network is that the way in which these new copies are linked to the unaltered pages reflects the topology of the original network. In our model the way in which we link these new components, which can be much more complex than single vertices, is by separating out the paths and cycles on which these components lie, in a way that mimics the original network structure.

Hence, to describe our different models of network specialization and their consequences we first need to consider the paths and cycles of a graph. A *path* P in the graph $G = (V, E, \omega)$ is an ordered sequence of distinct vertices $P = v_1, \ldots, v_m$ in V such that $e_{i,i+1} \in E$ for $i = 1, \ldots, m-1$. If the vertices v_1 and v_m are the same then P is a *cycle*. If it is the case that a cycle contains a single vertex then we call this cycle a *loop*.

Another fundamental concept that is needed is the notion of a strongly connected component. A graph $G = (V, E, \omega)$ is *strongly connected* if for any pair of vertices $v_i, v_j \in V$ there is a path from v_i to v_j or G consists of a single vertex. A *strongly connected component* of a graph G is a subgraph that is strongly connected and is maximal with respect to this property.

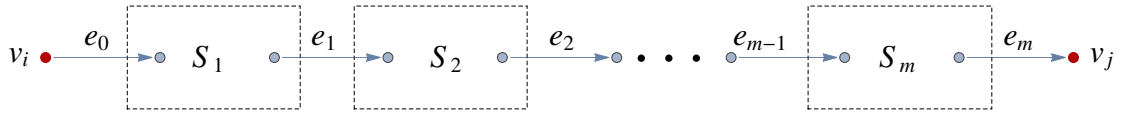


Figure 2.2: A representation of a path of components is shown, consisting of the sequence $S_1, \ldots, S_m$ of components beginning at vertex $v_i \in B$ and ending at vertex $v_j \in B$. From S_k to S_{k+1} there is a single directed edge e_{k+1}. From v_i to S_1 and from S_m to v_j there is also a single directed edge.

Because we are concerned with evolving the topology of a network in ways that preserve, at least locally, the network's topology we will also need the notion of a graph restriction. For a graph $G = (V, E, \omega)$ and a subset $B \subseteq V$ we let $G|_B$ denote the *restriction* of the graph G to the vertex set B, which is the subgraph of G on the vertex set B along with any edges of the graph G between vertices in B. We let $\bar{B}$ denote the *complement* of B, so that the restriction $G|_{\bar{B}}$ is the graph restricted to those vertices not in B.

The key to specializing the structure of a graph is to look at the strongly connected components of the restricted graph $G|_{\bar{B}}$. If $S_1, \ldots, S_m$ denote these strongly connected components then we will need to find paths or cycles of these components, which we refer to as *component branches*.

Definition 2.2. (Component Branches) For a graph $G = (V, E, \omega)$ and vertex set $B \subseteq V$ let $S_1, \ldots, S_m$ be the strongly connected components of $G|_{\bar{B}}$. If there are edges $e_0, e_1, \ldots, e_m \in E$ and vertices $v_i, v_j \in B$ such that

(i) e_k is an edge from a vertex in S_k to a vertex in S_{k+1} for $k = 1, \ldots, m-1$;

(ii) e_0 is an edge from v_i to a vertex in S_1; and

(iii) e_m is an edge from a vertex in S_m to v_j, then we call the ordered set

$$\beta = \{v_i, e_0, S_1, e_1, S_2, \ldots, S_m, e_m, v_j\}$$

a *path of components* in G with respect to B. If $v_i = v_j$ then β is a *cycle of components*. We call the collection $\mathcal{B}_B(G)$ of these paths and cycles the *component branches* of G with respect to the base set of vertices B.

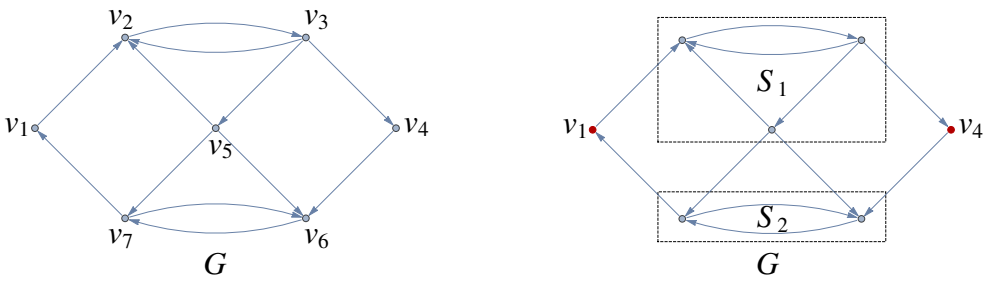


Figure 2.3: The unweighted graph $G = (V, E)$ is shown left. The components S_1 and S_2 of G with respect to the vertex set $B = \{v_1, v_4\}$ are shown right. These components are the subgraphs $S_1 = G|_{\{v_2,v_3,v_5\}}$ and $S_2 = G|_{\{v_6,v_7\}}$, indicated by the dashed boxes, which are the strongly connected components of the restricted graph $G|_{\bar{B}}$.

A representation of the path of components is shown in Figure 2.2. The sequence of components $S_1, \ldots, S_m$ in this definition can be empty in which case $m = 0$ and β is the path $\beta = \{v_i, v_j\}$ or loop if $v_i = v_j$. It is worth emphasizing that each branch $\beta \in \mathcal{B}_B(G)$ corresponds to a subgraph of G. Consequently, the edges of β inherit the weights they had in G if G is weighted. If G is unweighted then its component branches are likewise unweighted.

Once a graph has been decomposed into its various branches we construct the specialized version of the graph by merging these branches as follows.

Definition 2.3. (Graph Specialization) Suppose $G = (V, E, \omega)$ and $B \subseteq V$. Let $\mathcal{S}_B(G)$ be the graph which consists of the component branches $\mathcal{B}_B(G) = \{\beta_1, \ldots, \beta_\ell\}$ in which we *merge*, i.e. identify, each vertex $v_i \in B$ in any branch β_j with the same vertex v_i in any other branch β_k. We refer to the graph $\mathcal{S}_B(G)$ as the *specialization* of G over the *base* vertex set B.

A specialization of a graph G over a base vertex set B is a two step process. The first step is the construction of the component branches. The second step is the merging of these components into a single graph. We note that, in a component branch $\beta \in \mathcal{B}_B(G)$ only the first and last vertices of β belong to the base B. The specialized graph $\mathcal{S}_B(G)$ is therefore the collection of branches $\mathcal{B}_B(G)$ in which we identify an endpoint of two branches if they are the same vertex. This is demonstrated in the following example.

Example 2.4. (Constructing Graph Specializations) Consider the *unweighted* graph $G = (V, E)$ shown in Figure 2.3 (left). For the base vertex set $B = \{v_1, v_4\}$ the specialization $\mathcal{S}_B(G)$ is constructed as follows.

Step 1: Construct the branch components of G with respect to B. The graph $G|_{\bar{B}}$ has the strongly connected components $S_1 = G|_{\{v_2,v_3,v_5\}}$ and $S_2 = G|_{\{v_6,v_7\}}$, which are indicated in Figure 2.3 (right). The set $\mathcal{B}_B(G)$ of all paths and cycles of components beginning and ending at vertices in B consists of the component branches

$$\beta_1 = \{v_1, e_{12}, S_1, e_{34}, v_4\} \qquad \beta_2 = \{v_4, e_{46}, S_2, e_{71}, v_1\}$$
$$\beta_3 = \{v_1, e_{12}, S_1, e_{56}, S_2, e_{71}, v_1\} \quad \beta_4 = \{v_1, e_{12}, S_1, e_{57}, S_2, e_{71}, v_1\};$$

which are shown in Figure 2.4 (left).

Step 2: Merging the branch components. By merging each of the vertices $v_1 \in B$ in all branches of $\mathcal{B}_B(G) = \{\beta_1, \beta_2, \beta_3, \beta_4\}$ shown in Figure 2.4 (left) and doing the same for the vertex $v_4 \in B$, the result is the graph $\mathcal{S}_B(G)$ shown in Figure 2.4 (right), which is the specialization of G over the base vertex subset B.

To summarize, our model of network growth consists in evolving the topology of a given network by selecting some base subset of the network's elements and specializing the graph associated with the network over the corresponding vertices. We refer to this process as the *specialization model* of network growth.

The idea is that in an information network, such as the World Wide Web, this model of specialization can model the differentiation and inclusion of new information (cf. Example 2.1). In a biological model this can be used to describe various developmental processes, for instance, the specialization of cells into tissue. In a social network this model can similarly be used to model

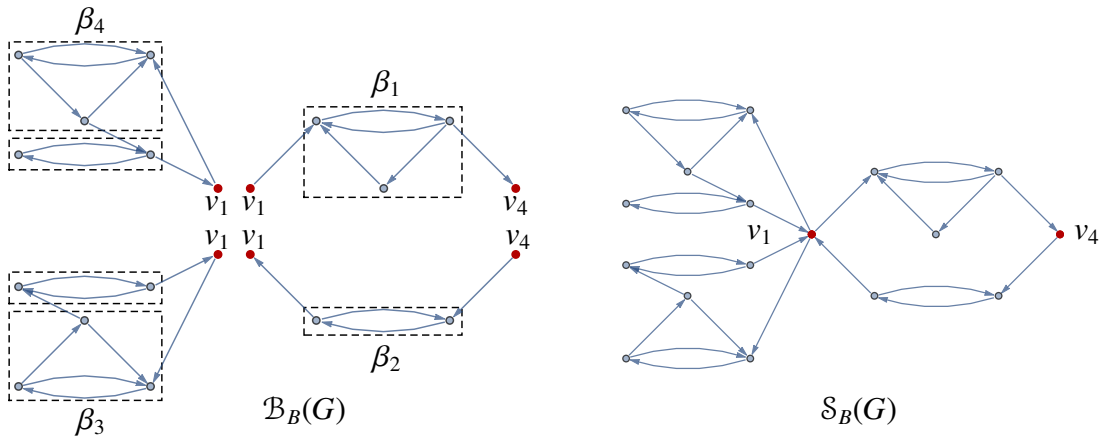


Figure 2.4: The component branches $\mathcal{B}_B(G) = \{\beta_1, \beta_2, \beta_3, \beta_4\}$ of the graph $G = (V, E)$ from figure 2.3 over the base vertex set $B = \{v_1, v_4\}$ are shown (left). The specialized graph $\mathcal{S}_B(G)$ is shown (right), which is made by merging each of the vertices v_1 and v_4 respectively in each of the branches of $\mathcal{B}_B(G)$. The edge labels and vertex labels are omitted, except for those vertices in B, to emphasize which vertices are identified.

how new relationships are formed when, for example, an individual is introduced by someone to their immediate group of friends.

It is also worth mentioning that a network specialization evolves the network's topology by maintaining the interactions between its base elements B and by differentiating the other network functions into sequences of components. The result is a network with many more of these components. These components are important for a number of reasons. The first is that they form network *motifs*, which are statistically important subgraphs within the network that typically perform a specific network function [34]. Second, because there are very few connections between these components the resulting network has a far more modular structure, which is a feature found in many real networks (see [18] for a survey of modularity). Third, because of the number of copies of the same component, the specialized graph has a certain amount of redundancy in its topology, which is another feature observed in real networks [35, 36]. Last, specializations results in *sparser* graphs, i.e. graphs in which the ratio of edges to vertices is relatively small, which is again a characteristic found in real networks [21, 22].

Additionally, many networks exhibit hierarchical organization, in which network vertices divide into groups or components that further subdivide into smaller groups of components, and so

on over multiple scales. It has been observed that these components often come from the same functional units, e.g. ecological niches in food webs, modules in biochemical networks including protein interaction networks, metabolic networks or genetic regulatory networks or communities in social networks [19, 20]. Because new components are created each time a graph is specialized a network becomes increasingly hierarchial as this process of specialization is repeated.

2.3 Specialization Rules

As a significantly large number of bases are possible for any reasonably sized network a natural question is, given a particular real-world network (or class of networks) can we find a base or sequence of bases that can be used to model this network's growth via specialization. Another way of stating this is: is it possible to find a specialization rule that selects network base(s) that can be used to specialize the network in a way that mimics its actual growth?

Here a *specialization rule* τ is a rule that selects for any graph $G = (V, E, \omega)$ a base vertex subset $V_\tau(G) \subseteq V$. For simplicity of notation, we let

$$\tau(G) = \mathcal{S}_{V_\tau(G)}(G) \text{ and } \tau^k(G) = \tau(\tau^{k-1}(G)) \text{ for } k > 0$$

denote the specialization of G with respect to τ and the kth specialization of G with respect to τ, respectively. Each rule τ generates a different type of growth and as such can be thought of as inducing a different specialization model of network growth.

The specialization $\tau(G)$ is *unique* if τ selects a unique base vertex subset of G. Not all rules produce a unique outcome as τ can be a rule that selects vertices of G in some random way. Here the first example of specialization uses a random specialization rule. We chose this because it is a very simple rule which produces graphs which exhibit properties found in many real world networks.

Example 2.5. (Random Specializations) For $p \in (0, 1)$ let $r = r_p$ be the specialization rule that uniformly selects a random network base consisting of p percent of the network's elements rounded

to the nearest whole number. To understand how this rule evolves the topology of a network we start by considering its effect on a random undirected graph with ten vertices and fifteen undirected edges. Specifically, we sequentially specialize a thousand such graphs G and analyze the statistics of the first six iterates $G, r(G), \ldots, r^6(G)$ in this process.

In Figure 2.5 (first row) we show a single realization of this process using the rule $r = r_p$ for $p = .9$. The iterates that are shown beyond the initial graph G are the first, second, and sixth. Below this the degree-distribution of each iterate is shown averaged over the thousand realizations we generate. Moving from left to right these distributions become increasing monotonic and right-skewed with each succesive specialization. This increase in skewness can be seen in Figure 2.5 (a) where the average, median, first and third quartiles of the graphs skewness is shown over each iterate.

Similarly, for each iterate the average, median, first and third quartiles for the graph's (b) size, (d) density, (e) clustering coefficient, and (f) assortativity is shown. These suggest that under iteration the graph's size is exponentially increasing and its density is exponentially decreasing, where *density* is $\rho = m/2(n(n-1))$ for a graph with m edges and n vertices. In part (e) the network's clustering coefficient per iteration is plotted (blue) along with the clustering coefficient of the corresponding configuration model (red). That is, the clustering coefficient we would expect for a random graph with the same degree distribution. In both cases we note that the clustering coefficients appear to be approaching zero.

In (f) the network's assortativity is shown for each iterate. The network is disassortative at each iterate but this dissassortativity has an interesting nonmonotonic behavior in which the dissasortativity initially makes a large jump then eventually relaxes back into less negative values.

In (c) the mean-distance between any two vertices vs. the network's size is shown for the sequence of graphs in the top row. Each dot from left to right indicates the next iterate in this sequence of specializations. In this particular plot, we display data for 12 iterations. This data is well fit by the indicated line which has the form $L(k) = \log_\beta(ck)$, for some constants $c > 0$ and

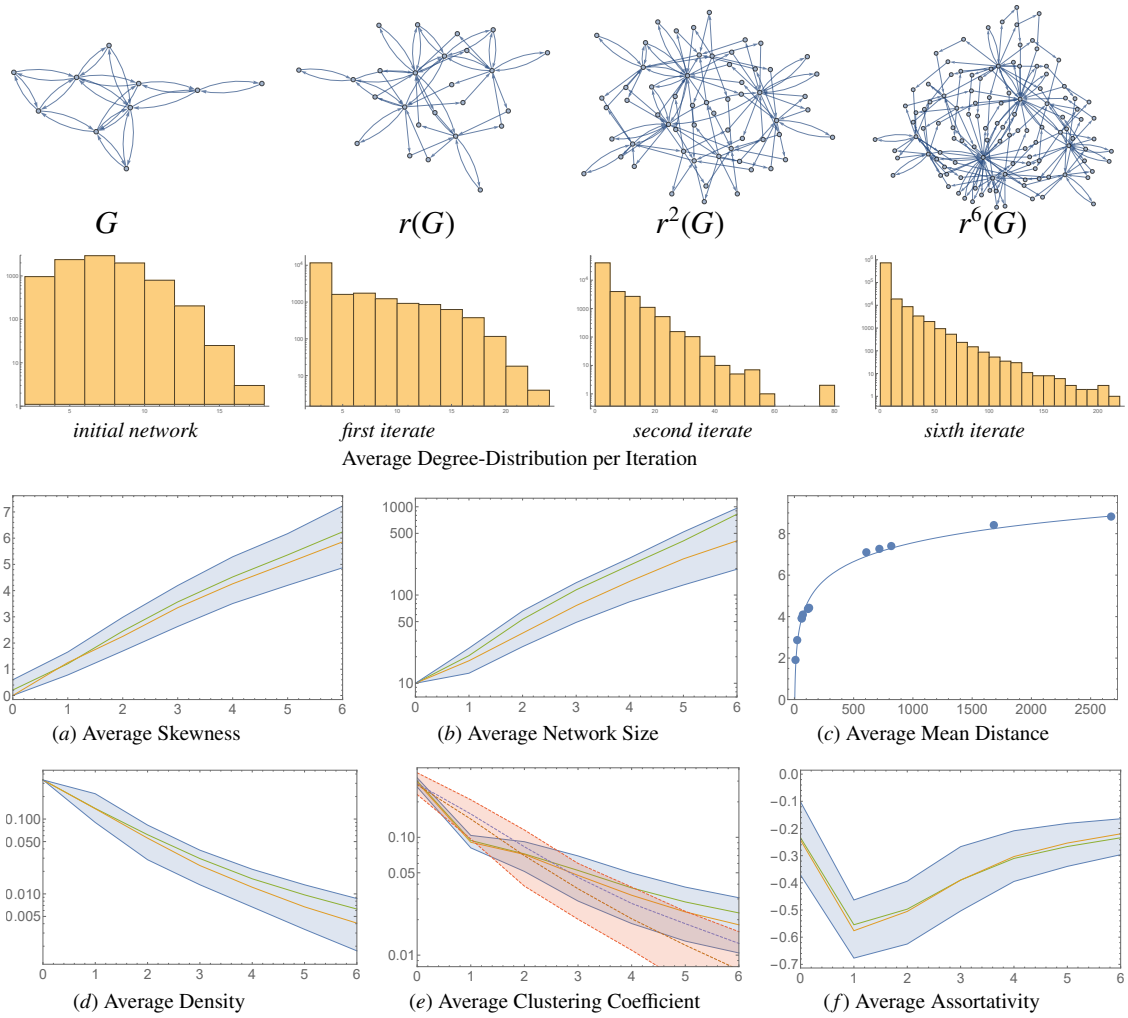


Figure 2.5: A realization of the specialization process using the rule $r = r_{.9}$ is show in the first row. In the second the average degree distribution over a thousand realization of this process per iteration is shown. For these thousand realizations we plot the average, median, first and third quartiles of the (a) degree distribution skewness vs. iteration, (b) graph size vs. iteration, (d) density vs. iteration, (e) global clustering coefficient vs iteration of the network (blue) and of the associated configuration model (red), (f) degree assortativity vs. iteration. In (c) the mean-distance between any two vertices vs. number of vertices is shown for the sequence shown in the first row.

$\beta > 2$. This logarithmic fit is on average quite good which indicates that as a graph evolves under this rule, it has the *small-world property*.

Similar behavior to those shown in Figure 2.5 is observed for a wide range of p-values. It is worth noting that a number of the properties shown in this figure mimic those found in real-world networks. These include monotonic right-skewed degree distributions, disassortativity, and the small-world property [6].

Example 2.6. (Specialization and Clustering Coefficients) Many real-world networks have a large number of triangles. This property is often found in social networks, where we expect with a high probability that two connected individuals have common contacts [6]. One of the properties not found in the specializations of the previous example is the presence of many triangles, i.e. these graphs do not have a high *clustering coefficient*. In fact, Figure 2.5 (e) suggests that as a network evolves under the rule r the probability of finding a triangle in the network tends to zero.

What is observed in many real networks is that as the network evolves the networks clustering coefficient is much higher than is expected for a randomly generated network with the same degree distribution [6]. In this example we consider a rule $\Delta = \Delta_p$ that chooses a base that causes a network to maintain a large number of its triangles as the network is specialized. This rule is similar to the rule r_p with the important exception that Δ_p only chooses from p percent of the network's vertices *which are not part of a triangle*. In our directed graphs, we define a triangle to be any three vertices where each pair is connected by at least one directed edge.

The top row of Figure 2.6 shows a realization of the process of sequentially specializations a graph using the rule $\Delta_{.9}$. For simplicity, the initial graphs we use are the undirected graphs on fifteen vertices with fifteen edges that have a single triangle. As can be seen in the figure, since the vertices of any triangle cannot be part of a base under the rule Δ the motif of a triangle is copied throughout the network in each successive iteration.

As in the previous example we randomly generate a thousand initial graphs and sequentially evolve each using the rule $\Delta_{.9}$. The number of iterates we consider here is five. Figure 2.6 shows the same statistic for these graphs and their iterations as it did in the previous example. Again we

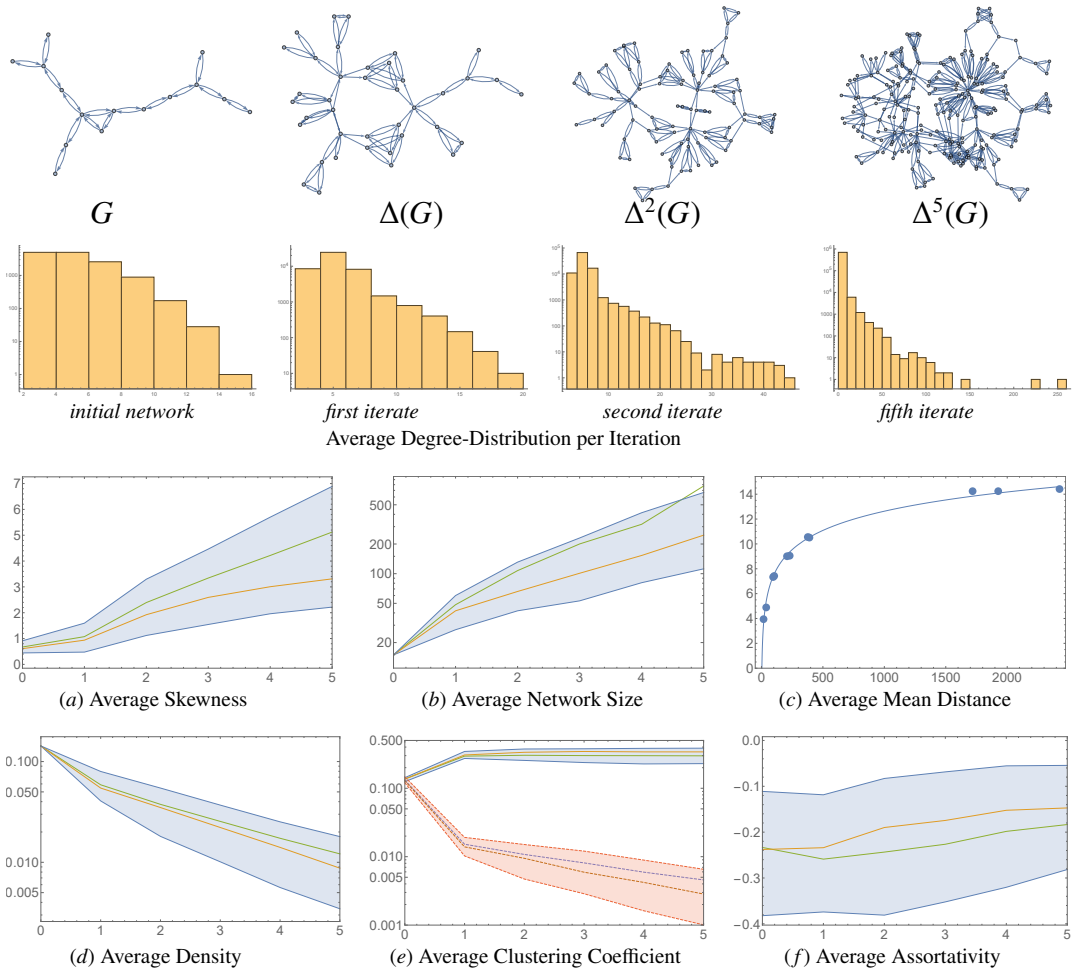


Figure 2.6: Similar to the previous figure a realization of the specialization process using the rule $\Delta = \Delta_9$ is show in the first row. In the second the average degree distribution over a thousand realization of this process per iteration is shown. For these thousand realizations we plot the average, median, first and third quartiles of the (a) degree distribution skewness vs. iteration, (b) graph size vs. iteration, (d) density vs. iteration, (e) global clustering coefficient vs iteration of the network (blue) and of the associated configuration model (red), (f) degree assortativity vs. iteration. In (c) the mean-distance between any two vertices vs. number of vertices is shown for the sequence shown in the first row.

see that this rule results in a sequence of networks whose degree-distributions becomes increasingly right-skewed, whose average density decreases, and whose average assortativity is negative. Moreover, the network has the small-world property as evidenced by the Figure 2.6 (c), which shows the growth of the average mean distance between vertices in the sequence of graphs shown in the top row of the figure.

However, the point of this example is that the network's clustering coefficient is nearly constant after a few iterates. This is shown in Figure 2.6 (c) in blue. In the same figure the clustering coefficient of the associated configuration model is shown in red, which appears to tend to zero with increasing iterates. Hence, by singling out the "triangle" as a motif that is preserved under the rule Δ our network maintains a high number of triangles as it is specialized, similar to many real networks.

Example 2.7. (Specialization and Power-Law Degree Distributions) As a final example, we chose a rule which is designed to generate networks whose degree distributions tend toward a power-law. That this can be done is significant as many real world networks seem to exhibit this behavior including the World Wide Web, actor networks, metabolic networks, citation networks, etc. [37].

The specialization rule we consider is the rule $\Xi_{p,q,s}$ which at each iterate i chooses vertices with the highest $p - 1 - \sum_{k=0}^{i-1} q^i$ percent and lowest $p - 1 + \sum_{k=0}^{i-1} q^i$ percent in-degree from the network, as well as a uniformly random s percent of the remaining vertices. The specific rule we use is $\Xi = \Xi_{.25,.09,.02}$. As before we sequentially apply this rule a thousand times to measure the statistical properties of this process. The difference from the previous examples is that here we start with the single undirected graph G shown in the upper left-hand corner of Figure 2.7. The statistics derived from the first seven iterations in this procedure are shown in the same figure.

Similar to the rules r and Δ we find that this rule creates networks that are increasingly sparse, disassortative, have increasingly right-skewed degree distribution, as well as clustering coefficients that tend towards a fixed constant. Somewhat surprisingly though, this rule does not lead to networks that have the small-world property as can be seen in Figure 2.7 (c). In this example there is

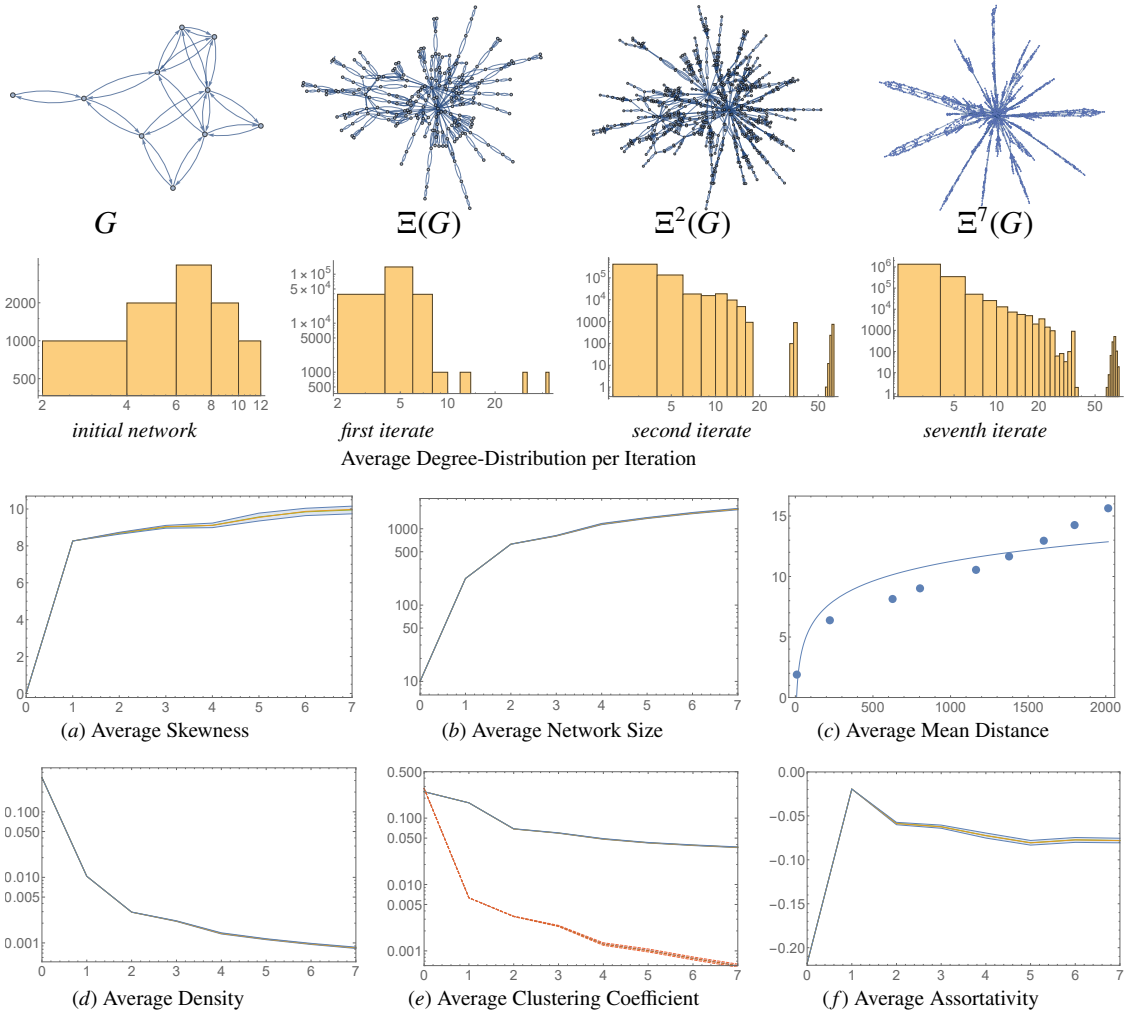


Figure 2.7: As in the previous two figures a realization of the specialization process using the rule Ξ is show in the first row. In the second the average degree distribution over a thousand realization of this process per iteration is shown. For these thousand realizations we plot the average, median, first and third quartiles of the (a) degree distribution skewness vs. iteration, (b) graph size vs. iteration, (d) density vs. iteration, (e) global clustering coefficient vs iteration of the network (blue) and of the associated configuration model (red), (f) degree assortativity vs. iteration. In (c) the mean-distance between any two vertices vs. number of vertices is shown for the sequence shown in the first row.

also much less variance in our statistics. This is likely due to the fact that we start with the same initial graph and at most two percent of the network's vertices are chosen at random at any stage to be part of the base.

The most striking feature and the point of this example is the particular type of degree distribution these networks have. In the second row of Figure 2.7 the average degree-distributions of the networks we generate using this rule together with initial graph G are shown on a log-log plot with logarithmic binning. The shape of these distributions suggest that G may evolve under the specialization rule Ξ into a scale-free network, at least on average.

In order to verify that we do indeed get a power-law distribution after some cut-off point, we followed the procedure outlined in [38]. In this method one first finds the optimal degree x_{min} beyond which the degree-distribution of a single specialized network is most power-law like. This is done by finding the x_{min} which minimizes

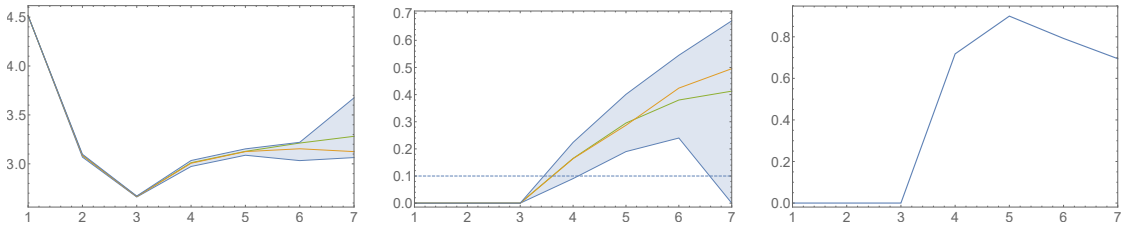
$$D = \max_{x \geq x_{min}} |S(x) - P(x)| \tag{2.1}$$

where $S(x)$ is the CDF of the data of our observations and $S(x)$ the CDF for the power-law model that best fits our data in the region $x \geq x_{min}$. Before finding $P(x)$ we calculate its power-law *exponent* α given by

$$\alpha = 1 + n \left[\sum_{i=1}^{n} \ln \frac{x_i}{x_{min} - \frac{1}{2}} \right]^{-1}.$$

Thus by minimizing D in equation (2.1), which is our *goodness-of-fit* value, we get both an x_{min} and an α-value.

To determine how power-law like our distribution is we generate a large number of power-law distributed synthetic data sets with the same x_{min} and α values. Following the method in [38] we find a goodness of fit for each of these synthetic data sets. We let ϕ be the percent of these data sets for which our goodness-of-fit D of our degree distribution is better (smaller) than the goodness-of-fit of these synthetic data sets. According to the method described in [38] we have a power-law like distribution if $\phi \geq 0.1$.



(*a*) Average α-Exponent (*b*) Average ϕ-Value (*c*) Percent that are Power-Laws

Figure 2.8: The average, median, first and third quartiles of the (a) power-law exponents α and (b) the ϕ-values is plotted per iterate for the process described in Example 2.7. In (c) the percent of those specialization that have ϕ-values greater than 0.1 is shown per iterate.

In Figure 2.8 (*a*) and (*b*) the average, median, first and third quartiles of the power-law exponents α and ϕ-values are shown per iterate for the thousand sequential specializations we consider of the graph G under Ξ. The percent of those specializations for which $\phi \geq 0.1$ is shown in (*c*) for each iterate. It is worth noting that before the fourth iterate none of the specialized networks passed the test of having $\phi \geq 0.1$ but thereafter a large percentage did indicating that these networks evolved into scale-free networks. Moreover, the exponent α for these iterates is on average between 3.0 and 3.5, which is typical of many real networks including citation networks, collaboration networks, the Internet, and email networks [37].

2.4 Specialization Equivalence

As mentioned before Example 2.5 there are two types of specialization rules; those that select a unique base vertex subset and those that do not. For instance, the random specialization rule in Example 2.5 does not select unique bases. We refer to τ as a *structural rule* if it does select a unique nonempty subset of vertices from any graph G. For instance, τ could be the rule that selects all vertices with a certain number of neighbors, or eigenvector centrality, etc. (cf. example 2.9). An important property of structural rules is that they give us a way of comparing the topologies of two distinct networks. In particular, any such rule allows us to determine which networks are similar and dissimilar with respect to the rule τ.

To make this precise we say two graphs $G = (V_1, E_1, \omega_1)$ and $H = (V_2, E_2, \omega_2)$ are *isomorphic* if there is a relabeling of the vertices of V_1 such that $G = H$ as weighted digraphs. If this is the case, we write $G \simeq H$. The idea is that two graphs are similar with respect to a rule τ if they both evolve to the *same*, i.e. isomorphic graph, under this rule. This allows us to partition all graphs, and therefore networks, into classes of similar graphs with respect to a structural rule τ. This can be stated as the following result.

Theorem 2.8. ***(Specialization Equivalence)*** *Suppose τ is a structural rule. Then τ induces an equivalence relation $\sim$ on the set of all weighted directed graphs where $G \sim H$ if $\tau(G) \simeq \tau(H)$. If this holds, we call G and H specialization equivalent with respect to τ.*

Proof. For any $G = (V, E, \omega)$ and structural rule τ the set $\tau(V) \subseteq V$ is unique implying the graph $\tau(G) = \mathcal{S}_{\tau(V)}(G)$ is unique up to a labeling of its vertices. Clearly, the relation of being specialization equivalent with respect to τ is reflexive and symmetric. Also, if $\tau(G) \simeq \tau(H)$ and $\tau(H) \simeq \tau(K)$ then there is a relabeling of the vertices of $\tau(G)$ such that $\tau(G) = \tau(H)$ and of $\tau(K)$ such that $\tau(K) = \tau(H)$. Hence, $\tau(G) = \tau(K)$ under some relabeling of the vertices of these graphs implying $\tau(G) \simeq \tau(K)$. This completes the proof. □

Theorem 2.8 states that every structural rule τ can be used to partition the set of graphs we consider, and by association all networks, into subsets. These subsets, or more formally *equivalence classes*, are those graphs that share a common topology with respect to τ. By *common topology* we mean that graphs in the same class have the same set of component branches and therefore evolve into the same graph under τ.

One reason for studying these equivalence classes is that it may not be obvious, and most often is not, that two different graphs belong to the same class. That is, two graphs may be structurally similar but until both graphs are specialized this similarity may be difficult to see. One of the ideas introduced here is that by choosing an appropriate rule τ one can discover this similarity as is demonstrated in the following example.

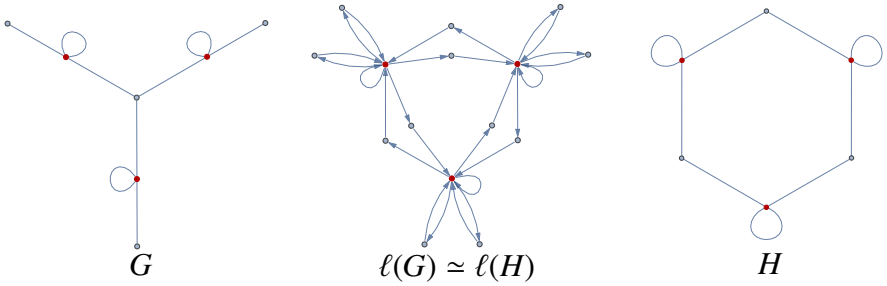


Figure 2.9: The graph G and the graph H are specialization equivalent with respect to the rule ℓ that selects those vertices of a graph that have loops. That is, the graphs $\ell(G)$ and $\ell(H)$ are isomorphic as is shown.

Example 2.9. (Specialization Equivalent Graphs) Consider the unweighted graphs $G = (V_1, E_1)$ and $H = (V_2, E_2)$ shown in Figure 2.9. Here, we let ℓ be the rule that selects all vertices of a graph that have loops, or all vertices if the graph has no loops. The vertices of G and H selected by the rule ℓ are the vertices highlighted (red) in Figure 2.9 in G and H, respectively. Although G and H appear to be quite different, the graphs $\ell(G)$ and $\ell(H)$ are isomorphic as is shown in Figure 2.9 (center). Hence, G and H belong to the same equivalence class of graphs with respect to the structural rule ℓ.

It is worth mentioning that two graphs can be equivalent under one rule but not another. For instance, if w is the structural rule that selects vertices *without* loops then $w(G) \not\simeq w(H)$ although $\ell(G) \simeq \ell(H)$.

From a practical point of view, a specialization rule τ allows those studying a particular class of networks a way of comparing the *specialized topology* of these networks and drawing conclusions about both the specialized and original networks. Of course, the rule τ should be designed by the particular biologist, chemist, physicist, etc. to have some significance with respect to the networks under consideration.

2.5 Spectral Properties of Specializations

To understand how specializing a network's topology affects the network's dynamics and in turn the network's function, we need some notion that relates both structure and dynamics. One of the

most fundamental concepts that does this is the notion of a network's spectrum [39, 23, 31]. The spectrum of a network can be defined in a number of ways since there are a number of ways that a matrix can be associated with a network. Matrices that are often associated with a network include various Laplacian matrices, e.g. the regular Laplacian, combinatorial Laplacian, normalized Laplacian, signless Laplacian, etc. Other matrices include the adjacency and weighted adjacency matrix of a graph, the distance matrix of a graph, etc.

The type of matrix we consider is the weighted adjacency matrix of a graph. The reason we consider the adjacency matrix of a graph G versus any one of the other matrices that can be associated with G is that there is a one-to-one relationship between the matrices $M \in \mathbb{R}^{n\times n}$ and the weighted directed graphs we consider.

Because we are concerned with the spectrum of a graph, which is a set that includes multiplicities, the following is important for our discussion. First, the element α of the set A has *multiplicity* m if there are m elements of A equal to α. If $\alpha \in A$ with multiplicity m and $\alpha \in B$ with multiplicity n then

(i) the *union* $A \cup B$ is the set in which α has multiplicity $m + n$; and

(ii) the *difference* $A - B$ is the set in which α has multiplicity $m - n$ if $m - n > 0$ and where $\alpha \notin A - B$ otherwise.

For ease of notation, if A and B are sets that include multiplicity then we let $B^k = \cup_{i=1}^{k} B$ for $k \geq 1$. That is, the set B^k is k copies of the set B where we let $B^0 = \emptyset$. For $k = -1$ we let $A \cup B^{-1} = A - B$. With this notation in place, the spectrum of a graph G and the spectrum of the specialized graph $\mathcal{S}_B(G)$ are related by the following result.

Theorem 2.10. ***(Spectra of Specialized Graphs)*** *Let $G = (V, E, \omega)$, $B \subseteq V$, and let $S_1, \ldots, S_m$ be the strongly connected components of $G|_{\bar{B}}$. Then*

$$\sigma(\mathcal{S}_B(G)) = \sigma(G) \cup \sigma(S_1)^{n_1-1} \cup \sigma(S_2)^{n_2-1} \cup \cdots \cup \sigma(S_m)^{n_m-1}$$

where n_i is the number of branches in $\mathcal{B}_B(G)$ that contain S_i.

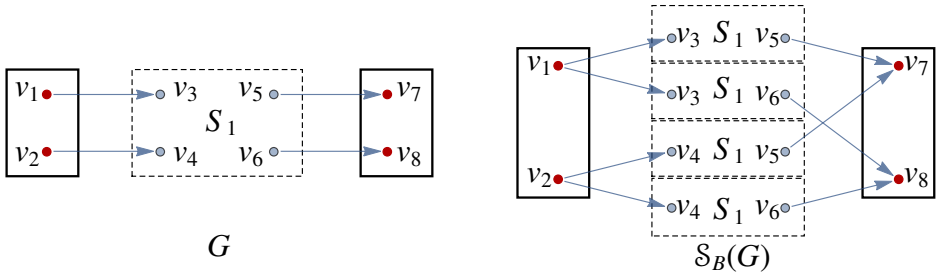


Figure 2.10: An example of a graph $G = (V, E, \omega)$ with the single strongly connected component $S_1 = G|_{\bar{B}}$ is shown (left), where solid boxes indicate the graph $G|_B$. As there are two edges from $G|_B$ to S_1 and two edges from S_1 to $G|_B$ there are 2×2 branches in $\mathcal{B}_B(G)$ containing S_1. These are merged together with $G|_B$ to form $\mathcal{S}_B(G)$ (right).

Proof. For the graph $G = (V, E, \omega)$ without loss of generality let $B = \{v_1, \ldots, v_\ell\}$ where $V = \{v_1, \ldots, v_n\}$. By a slight abuse in notation we let B denote the index set $B = \{1, \ldots, \ell\}$ that indexes the vertices in B.

For the moment we assume that the graph $G|_{\bar{B}}$ has the single strongly connected component S_1. The weighted adjacency matrix $M \in \mathbb{R}^{n \times n}$ then has the block form

$$M = \begin{bmatrix} U & W \\ Y & Z \end{bmatrix}$$

where $U \in \mathbb{R}^{\ell \times \ell}$ is the matrix $U = M_{BB}$, which is the weighted adjacency matrix of $G|_B$. The matrix $Z \in \mathbb{R}^{(n-\ell)\times(n-\ell)}$ is the matrix $Z = M_{\bar{B}\bar{B}}$, which is the weighted adjacency matrix of $G|_{\bar{B}} = S_1$. The matrix $W = M_{B\bar{B}} \in \mathbb{R}^{\ell \times (n-\ell)}$ is the matrix of edges weights of edges from vertices in $G|_B$ to vertices in S_1 and $Y = M_{\bar{B}B} \in \mathbb{R}^{n-\ell \times \ell}$ is the matrix of edge weights of edges from vertices in S_1 to vertices in $G|_B$.

The specialization $\mathcal{S}_B(G)$ is the graph in which all component branches of the form

$$\beta = v_i, e_{ip}, S_1, e_{qj}, v_j \text{ for all } v_i, v_j \in S, v_p, v_q \in \bar{B} \text{ and } e_{ip}, e_{qj} \in E$$

are merged together with the graph $G|_B$ (cf. Figure 2.10). The weighted adjacency matrix $\hat{M} = M(\mathcal{S}_B(G))$ has the block form

$$\hat{M} = \begin{bmatrix} U & \begin{bmatrix} W_1 & \cdots & W_1 \end{bmatrix} & \cdots & \begin{bmatrix} W_w & \cdots & W_w \end{bmatrix} \\ \begin{bmatrix} Y_1 \\ \vdots \\ Y_y \end{bmatrix} & \begin{bmatrix} Z & & \\ & \ddots & \\ & & Z \end{bmatrix} & & 0 \\ \vdots & & \ddots & \\ \begin{bmatrix} Y_1 \\ \vdots \\ Y_y \end{bmatrix} & 0 & & \begin{bmatrix} Z & & \\ & \ddots & \\ & & Z \end{bmatrix} \end{bmatrix} = \begin{bmatrix} U & \hat{W} \\ \hat{Y} & \hat{Z} \end{bmatrix},$$

where each $W_i \in \mathbb{R}^{\ell \times (n-\ell)}$, each $Y_j \in \mathbb{R}^{(n-\ell)\times \ell}$, $\sum_{i=1}^{w} W_i = W$ and $\sum_{j=1}^{y} Y_j = Y$. Here, $w \geq 0$ is the number of directed edges from $G|_B$ to S_1. The matrix W_i has a single nonzero entry corresponding to exactly one edge from this set of edges. Similarly, $y \geq 0$ is the number of directed edges from S_1 to $G|_B$. The matrix Y_i has a single nonzero entry corresponding to exactly one edge from this set of edges. Since there are $w \cdot y$ component branches in $\mathcal{S}_B(G)$ containing S_1 then the matrix $\hat{M} \in \mathbb{R}^{(\ell + p(n-\ell))\times(\ell+p(n-\ell))}$ where $p = w \cdot y$.

To prove the result of Theorem 2.10 we will use the Schur complement formula, which states that a matrix with the block form of M has the determinant

$$\det[M] = \det[Z]\det[U - WZ^{-1}Y]$$

if Z is invertible. Here we apply this formula to the block matrix $M - \lambda I$, which results in the determinant

$$\begin{aligned} \det[M - \lambda I] &= \det[Z - \lambda I]\det[(U - \lambda I) - W(Z - \lambda I)^{-1}Y] \\ &= \det[Z - \lambda I]\det[(U - \lambda I) - \sum_{i=1}^{w}\sum_{j=1}^{y} W_i(Z - \lambda I)^{-1}Y_j], \end{aligned}$$

where the matrix $Z - \lambda I$ is invertible as a result of the proof of Theorem 1.2 in [7]. Similarly, applying the Schur complement formula to the matrix $\hat{M} - \lambda I$ we have the determinant

$$\begin{aligned}\det[\hat{M} - \lambda I] &= \det[\hat{Z} - \lambda I] \det[(\hat{U} - \lambda I) - \hat{W}(Z - \lambda I)^{-1}\hat{Y}] \\ &= \det[Z - \lambda I]^p \det[(U - \lambda I) - \sum_{i=1}^{w}\sum_{j=1}^{y} W_i(Z - \lambda I)^{-1}Y_j],\end{aligned}$$

where the equality $\hat{W}(Z - \lambda I)^{-1}\hat{Y} = \sum_{i=1}^{w}\sum_{j=1}^{y} W_i(Z - \lambda I)^{-1}Y_j$ can be seen by direct multiplication of the block entries in the matrix $\hat{M}$. Hence,

$$\det[M - \lambda I] \det[Z - \lambda I]^{p-1} = \det[\hat{M} - \lambda I]$$

implying

$$\sigma(\mathcal{S}_B(G)) = \sigma(G) \cup \sigma(S_1)^{p-1}, \tag{2.2}$$

so that Theorem 2.10 holds if $G|_{\bar{B}}$ has a single strongly connected component S_1.

In order to prove Theorem 2.10 in full generality, we will first need to describe an alternative process which generates the specialized graph $\mathcal{S}_B(G)$, which we refer to as "stepwise specialization." We start this process by defining L_1 to be the set of the strongly connected components of $G|_{\bar{B}}$, and label the components of this set as $L_1 = \{S_{1,1}, S_{2,1}, \ldots, S_{m,1}\}$. Next we set $G_1 = G$ and randomly choose a strongly connected component $S_{i_1,1} \in L_1$ with the property that specializing G around $S_{i_1,1}$ generates a *non-trivial specialization*, i.e. $\mathcal{S}_{\bar{S}_{i,1}}(G_1) \neq G_1$. That is, when G, is specialized over $\bar{S}_{i,1}$, the result is a larger graph, meaning that there is more than one directed edge into or out of $S_{i,1}$. We then let $G_2 = \mathcal{S}_{\bar{S}_{i,1}}(G_1)$. Because $S_{i_1,1}$ is a single strongly connected component, we showed in the beginning of this proof that the only new vertices which appear in G_2 as G_1 is specialized must be $w \cdot y$ copies of $S_{i_1,1}$, where w is the number of edges which are directed into $S_{i_1,1}$ and y is the number of edges which are directed out of $S_{i_1,1}$. Now we relabel the collection of all copies of $S_{i_1,1}$ in the specialized graph G_2 as $\{S_{i_1,1}, S_{i_1,2}, \ldots, S_{i_1,r_1}\}$.

Next we define L_2 to be the collection of all strongly connected components of $G_2|_{\bar{B}}$, which includes all the elements of L_1, plus the newly relabeled collection of copies of $S_{i_1,1}$. Thus $L_2 = \{S_{1,1}, S_{2,1}, \ldots, S_{i_1-1,1}, S_{i_1,1}, S_{i_1,2}, \ldots, S_{i_1,r_1}, S_{i_1+1,1}, \ldots, S_{m,1}\}$. We then define G_3 by selecting a random element S_{i_2,j_2} of L_2 which does not induce a trivial specialization of G_2 and set $G_3 = \mathcal{S}_{\bar{S}_{i_2,j_2}}(G_2)$. Again we relabel all of the strongly connected component copies of S_{i_2,j_2} which appear in G_3 (this includes new copies and any copies which previously existed in the graph) and define L_3 to be the collection of all strongly connected components of $G_3|_{\bar{B}}$. We continue this process inductively, at each step defining $G_{k+1} = \mathcal{S}_{\bar{S}_{i_k,j_k}}(G_k)$ where S_{i_k,j_k} is randomly chosen from L_k and has the property that $\mathcal{S}_{\bar{S}_{i_k,j_k}}(G_k) \neq G_k$. Then we relabel all copies of $S_{i_k,1}$ in G_k as $\{S_{i_k,1}, \ldots S_{i_k,p_k}\}$ and define L_{k+1} as the strongly connected components of G_{k+1} with all the newly defined labels.

Note that at each step we only specialize over one strongly connected component of $G_{\bar{B}}$. An important observation about this process is that the set of component branches with respect to base B is invariant over each step. This can be seen by recognizing that there is a one-to-one correspondence between the component branches $\mathcal{B}_B(G_k)$ and $\mathcal{B}_B(G_{k+1})$. From this fact we can conclude that

$$\mathcal{S}_B(G_k) = \mathcal{S}_B(G) \tag{2.3}$$

for all $k \geq 1$, since we defined specialization to be the graph built by identifying elements of B in all component branches of the graph.

We will continue the process described above until all strongly connected components of $G_K|_{\bar{B}}$ induce a trivial specialization. At this point, the process terminates and we are left with the final graph G_K. We claim that eventually this process must reach this condition and terminate. Suppose the process did not terminate, then the specialized graph in each step would be strictly larger than the graph in the previous step. Thus the number of vertices in G_k must diverge to infinity, which we denote as $\lim_{k\to\infty} |G_k| = \infty$. However, we know that by construction $|G_k| \leq |\mathcal{S}_B(G_k)$. Using Equation (2.3) we know that $|\mathcal{S}_B(G_k)| = |\mathcal{S}_B(G)| < \infty$. Therefore, at some point this process must terminate.

When the process does terminate, we claim the final graph $G_K = \mathcal{S}_B(G)$. To see this, first recall that we defined $\mathcal{S}_B(G)$ to be the graph which consists of the component branches of $\mathcal{B}_B(G)$ in which all elements of B are individually identified (see Definition 2.3). Thus, if S is a strongly connected component of $G|_{\bar{B}}$, then each copy of S found in $\mathcal{S}_B(G)$ will have exactly one edge pointing into it and exactly one edge pointing out of it by the definition of a component branch. In the final graph G_K, there are no more strongly connected components for which specializing gives a larger graph. Therefore, each strongly connected component in L_K has exactly one edge directing into it and one edge directing out of it. Since this is true for each strongly connected component, then G_K must be composed of a collection of component branches where all corresponding vertices in B are identified. We already showed G_K has the same component branches with respect to B as $\mathcal{S}_B(G)$. Using this fact and because strongly connected components of $G_K|_{\bar{B}}$ are each their own component branch, we can conclude that $G_K = \mathcal{S}_B(G)$.

Hence, we can repeatedly apply the result given in Equation (2.2) for a single strongly connected component at each step in our "stepwise specialization process." Thus the spectrum of the specialized graph G_k is the spectrum of the graph G_{k-1} from the previous step together with the eigenvalues $\sigma(S_{i_k})^{p_k}$ where p_k is the number of copies of S_{i_k} which were added. Therefore the eigenvalues of the final graph will be the eigenvalues of the original graph together with the eigenvalues of all the copies of the strongly connected components we added at each step. That is

$$\sigma(\mathcal{S}_B) = \sigma(G) \cup \sigma(S_1)^{n_1-1} \cup \cdots \cup \sigma(S_m)^{n_m-1}$$

where n_i is the number of component branches in $\mathcal{B}_B(G)$ containing S_i. This completes the proof.

□

For now, we consider an example of Theorem 2.10. In Section 2.6 we consider some consequences and applications of Theorem 2.10 where we study the interplay of network dynamics and network growth.

Example 2.11. Continuing example 2.4, we can calculate the graph G in Figure 2.3 (left) has eigenvalues

$$\sigma(G) \approx \{1.38, -1.18, 0.57 \pm 0.93i, -0.67 \pm 0.74i, 0\}.$$

For $B = \{v_1, v_4\}$ the graph $G|_{\bar{B}}$ has the strongly connected components S_1 and S_2 with eigenvalues $\sigma(S_1) \approx \{1.32, -0.66 \pm 0.56i, \}$ and $\sigma(S_2) = \{\pm 1\}$. Since three branches of $\mathcal{B}_B(G)$ contain S_1 and S_2 respectively, Theorem 2.10 implies that

$$\sigma(\mathcal{S}_B(G)) = \sigma(G) \cup \sigma(S_1)^2 \cup \sigma(S_2)^2.$$

A simple eigenvalue calculation of $\mathcal{S}_B(G)$ will demonstrate that this is true.

If a network has an evolving structure that can be modeled via a graph specialization, or more naturally a sequence of specializations, then Theorem 2.10 allows us to effectively track the changes in the network's spectrum, resulting from the components $S_1, \ldots, S_m$.

Not only are the eigenvalues of a graph G preserved in a specific way as the graph is specialized but so are its eigenvectors. An *eigenvector* $\mathbf{v}$ of a graph G corresponding to the eigenvalue λ is a vector such that $M(G)\mathbf{v} = \lambda\mathbf{v}$, in which case $(\lambda, \mathbf{v})$ an *eigenpair* of G. If B is a subset of the vertices of G then we let $\mathbf{v}_B$ denote the eigenvector $\mathbf{v}$ restricted to those entries indexed by the set B.

The remainder of the results in this section rely on the concept of *isospectral graph reduction*, which is a way of reducing the size of a graph while essentially preserving its set of eigenvalues. Since there is a one-to-one relation between the graphs we consider and the matrices $M \in \mathbb{R}^{n\times n}$, there is also an equivalent theory of *isospectral matrix reductions*. Both types of reductions will be useful to us.

For the sake of simplicity we begin by defining an isospectral matrix reduction. For these reductions we need to consider matrices of rational functions. The reason is that, by the Fundamental Theorem of Algebra, a matrix $A \in \mathbb{R}^{n\times n}$ has exactly n eigenvalues including multiplicities. In order to reduce the size of a matrix while at the same time preserving its eigenvalues we need something

that carries more information than scalars. The objects we will use are rational functions. The specific reasons for using rational functions can be found in [7], chapter 1.

We let $\mathbb{W}^{n\times n}$ be the set of $n \times n$ matrices whose entries are rational functions $p(\lambda)/q(\lambda) \in \mathbb{W}$, where $p(\lambda)$ and $q(\lambda) \neq 0$ are polynomials with real coefficients in the variable λ. The eigenvalues of the matrix $M = M(\lambda) \in \mathbb{W}^{n\times n}$ are defined to be solutions of the *characteristic equation*

$$\det(M(\lambda) - \lambda I) = 0,$$

which is an extension of the standard definition of the eigenvalues for a matrix with complex entries.

For $M \in \mathbb{R}^{n\times n}$ let $N = \{1, \ldots, n\}$. If the sets $T, U \subseteq N$ are proper subsets of N, we denote by M_{TU} the $|T| \times |U|$ *submatrix* of M with rows indexed by T and columns by U. The isospectral reduction of a square real valued matrix is defined as follows.

Definition 2.12. (Isospectral Matrix Reduction) The *isospectral reduction* of a matrix $M \in \mathbb{R}^{n\times n}$ over the proper subset $B \subseteq N$ is the matrix

$$\mathcal{R}_B(M) = M_{BB} - M_{B\bar{B}}(M_{\bar{B}\bar{B}} - \lambda I)^{-1} M_{\bar{B}B} \in \mathbb{W}^{|B|\times|B|}.$$

One of the most important properties of an isospectral reduction is the following result which relates the eigenvalues and eigenvectors of a graph G and its reduction $\mathcal{R}_B(G)$.

Theorem 2.13. ***(Eigenvectors of Reduced Matrices)*** *Suppose $M \in \mathbb{R}^{n\times n}$ and $B \subseteq N$. If $(\lambda, \mathbf{v})$ is an eigenpair of M and $\lambda \notin \sigma(M_{\bar{B}\bar{B}})$ then $(\lambda, \mathbf{v}_B)$ is an eigenpair of $\mathcal{R}_B(M)$.*

Proof. Suppose $(\lambda, \mathbf{v})$ is an eigenpair of M and $\lambda \notin \sigma(M_{\bar{B}\bar{B}})$. Then without loss in generality we may assume that $\mathbf{v} = (\mathbf{v}_B^T, \mathbf{v}_{\bar{B}}^T)^T$. Since $M\mathbf{v} = \lambda\mathbf{v}$ then

$$\begin{bmatrix} M_{BB} & M_{B\bar{B}} \\ M_{\bar{B}B} & M_{\bar{B}\bar{B}} \end{bmatrix} \begin{bmatrix} \mathbf{v}_B \\ \mathbf{v}_{\bar{B}} \end{bmatrix} = \lambda \begin{bmatrix} \mathbf{v}_B \\ \mathbf{v}_{\bar{B}} \end{bmatrix},$$

which yields two equations the second of which implies that

$$M_{\bar{B}B}\mathbf{v}_B + M_{\bar{B}\bar{B}}\mathbf{v}_{\bar{B}} = \lambda \mathbf{v}_{\bar{B}}.$$

Solving for $\mathbf{v}_{\bar{B}}$ in this equation yields

$$\mathbf{v}_{\bar{B}} = -(M_{\bar{B}\bar{B}} - \lambda I)^{-1} M_{\bar{B}B}\mathbf{v}_B, \tag{2.4}$$

where $M_{\bar{B}\bar{B}} - \lambda I$ is invertible given that $\lambda \notin \sigma(M_{\bar{B}\bar{B}})$.

Note that

$$\begin{aligned}
(M - \lambda I)\mathbf{v} &= \begin{bmatrix} (M - \lambda I)_{BB}\mathbf{v}_B + (M - \lambda I)_{B\bar{B}}\mathbf{v}_{\bar{B}} \\ (M - \lambda I)_{\bar{B}B}\mathbf{v}_B + (M - \lambda I)_{\bar{B}\bar{B}}\mathbf{v}_{\bar{B}} \end{bmatrix} \\
&= \begin{bmatrix} M_{BB}\mathbf{v}_B - M_{B\bar{B}}(M_{\bar{B}\bar{B}} - \lambda I)^{-1} M_{\bar{B}B}\mathbf{v}_B \\ M_{\bar{B}B}\mathbf{v}_B - (M_{\bar{B}\bar{B}} - \lambda I)(M_{\bar{B}\bar{B}} - \lambda I)^{-1} M_{\bar{B}B}\mathbf{v}_B \end{bmatrix} \\
&= \begin{bmatrix} (\mathcal{R}_B(M) - \lambda I)\mathbf{v}_B \\ 0 \end{bmatrix}.
\end{aligned}$$

Since $(M - \lambda I)\mathbf{v} = 0$ it follows that $(\lambda, \mathbf{v}_B)$ is an eigenpair of $\mathcal{R}_B(M)$.

Moreover, we observe that if $(\lambda, \mathbf{v}_B)$ is an eigenpair of $\mathcal{R}_B(M)$ then by reversing this argument, $(\lambda, (\mathbf{v}_B^T, \mathbf{v}_{\bar{B}}^T)^T)$ is an eigenpair of M where $\mathbf{v}_{\bar{B}}$ is given by (2.4). □

Using the previous theorem, we can now prove that the eigenvectors of the specialized graph are related to the original eigenvectors.

Proposition 2.14. ***(Eigenvectors of Evolved Graphs)*** *Let $(\lambda, \mathbf{v})$ be an eigenpair of the graph $G = (V, E, \omega)$. If $B \subseteq V$ and $\lambda \notin \sigma(G|_{\bar{B}})$ then there is an eigenpair $(\lambda, \mathbf{w})$ of the specialized graph $\mathcal{S}_B(G)$ such that $\mathbf{w}_B = \mathbf{v}_B$.*

Proof. Let $M = M(G)$ and $\hat{M} = M(\mathcal{S}_B(G))$ where $B \subseteq N$. If $(\lambda, \mathbf{v})$ is an eigenpair of M and $\lambda \notin \sigma(M_{\bar{B}\bar{B}})$ then Theorem 2.13 implies that $(\lambda, \mathbf{v}_B)$ is an eigenpair of $\mathcal{R}_B(M)$. The claim is that by reducing both M and $\hat{M}$ over B the result is the same matrix. To see this we note that by Definition 2.12 the reduced matrix $\mathcal{R}_B(M)$ is

$$\mathcal{R}_B(M) = U - W(Z - \lambda I)^{-1} Y \in \mathbb{W}^{|B|\times|B|}.$$

For the matrix $\hat{M}$ its reduction over B is the matrix

$$\begin{aligned}
\mathcal{R}_B(\hat{M}) &= U - \hat{W}(\hat{Z} - \lambda I)^{-1}\hat{Y} \\
&= U - \hat{W}\ \mathrm{diag}[(Z - \lambda I)^{-1}, \ldots, (Z - \lambda I)^{-1}]\hat{Y} \\
&= U - \sum_{i=1}^{w}\sum_{j=1}^{y} W_i (Z - \lambda I)^{-1} Y_j \\
&= U - (\sum_{i=1}^{w} W_i)(Z - \lambda I)^{-1}(\sum_{j=1}^{y} Y_j) \\
&= U - W(Z - \lambda I)^{-1} Y \in \mathbb{W}^{|S|\times|S|}.
\end{aligned}$$

Hence, $\mathcal{R}_B(M) = \mathcal{R}_B(\hat{M})$.

Using the fact that $\mathcal{R}_B(\mathcal{S}_B(M)) = \mathcal{R}_B(M)$ (see the proof of Theorem 2.10) and the observation in the last line of the proof of Theorem 2.13 it follows that $(\lambda, \hat{\mathbf{v}})$ is an eigenpair of $\hat{M}$ where

$$\hat{\mathbf{v}} = \begin{bmatrix} \hat{\mathbf{v}}_B \\ \hat{\mathbf{v}}_{\bar{B}} \end{bmatrix} = \begin{bmatrix} \mathbf{v}_B \\ -(\hat{M}_{\bar{B}\bar{B}} - \lambda I)^{-1}\hat{M}_{\bar{B}B}\mathbf{v}_B \end{bmatrix}.$$

Note that $\mathbf{v}_B = \hat{\mathbf{v}}_B$, which completes the proof. □

Proposition 2.14 states that the graphs G and $\mathcal{S}_B(G)$ have the same eigenvectors if we restrict our attention to those entries that correspond to the base vertices B and to those eigenvectors with corresponding eigenvalues in $\sigma(G) - \sigma(G|_{\bar{B}}) \subset \sigma(\mathcal{S}_B(G))$. One consequence of this fact is that the eigenvector centrality of the vertices in B remain the same as the graph is specialized.

To describe eigenvector centrality note that by the Perron-Frobenius theorem, if the unweighted graph $G = (V, E)$ is strongly connected then G has a unique eigenvalue ρ, which is the *spectral radius* of G, i.e. $\rho = \max\{|\lambda| : \lambda \in \sigma(G)\}$. Moreover, ρ is a simple eigenvalue and the eigenvector $\mathbf{p}$ associated with ρ has nonnegative entries. The vector $\mathbf{p}$, which is unique up to a constant, gives the relative ranking p_i to each vertex $v_i \in V$. This value p_i is referred to as the *eigenvector centrality* of the vertex v_i (see [6] for more details). Here we refer to the vector $\mathbf{p}$ as an *eigencentrality vector* of the graph G. A graph specialization of G preserves its vertices' eigenvector centrality in the following way.

Theorem 2.15. ***(Eigenvector Centrality of Specialized Graphs)*** *Let $G = (V, E)$ be strongly connected with eigencentrality vector $\mathbf{p}$. If $B \subset V$ then $\mathcal{S}_B(G)$ has an eigencentrality vector $\mathbf{q}$ where $\mathbf{p}_B = \mathbf{q}_B$. Hence, the relative eigenvector centrality of the vertices in the base B is preserved as the graph is specialized.*

Proof. Suppose that $G = (V, E)$ is strongly connected and $B \subset V$. Since the specialization $\mathcal{S}_B(G)$ preserves the path structure of G, i.e. there is a path from v_i to v_j in $\mathcal{S}_B(G)$ if and only if there is a path from the corresponding v_i to v_j in G, then $\mathcal{S}_B(G)$ must be strongly connected. Therefore, both G and $\mathcal{S}_B(G)$ have eigencentrality vectors.

Given that $M(G)$ is a nonnegative matrix, Theorem 2.10 together with Corollary 8.1.20 in [40] imply that G and $\mathcal{S}_B(G)$ have the same spectral radius ρ. Since ρ is a simple eigenvalue of both G and $\mathcal{S}_B(G)$, Proposition 2.14 implies that given an eigencentrality vector $\mathbf{p}$ of G there is an eigencentrality vector $\mathbf{q}$ of $\mathcal{S}_B(G)$ such that $\mathbf{p}_B = \mathbf{q}_B$ completing the proof. □

The graph G in Figure 2.3 (left) and its specialization $\mathcal{S}_B(G)$ in Figure 2.4 (right) have eigencentrality vectors $\mathbf{p} \in \mathbb{R}^7$ and $\mathbf{q} \in \mathbb{R}^{17}$, which correspond to the spectral radius $\rho \approx 1.38$ of both graphs. Since G is strongly connected it follows from Theorem 2.15 that $\mathbf{p}_B = \mathbf{q}_B \approx \{1.27, 1\}$. That is, the vertices v_1 and v_4 of B have the same relative eigenvalue centrality in G and in $S_B(G)$.

2.6 Network Growth, Dynamics, and Function

In this chapter thus far we have been primarily concerned with the dynamics *of* a network's topology. This evolution determines to a certain extent the network's function and how well the network preforms this function. However, the network's performance also depends on the type of dynamics that emerges from the interactions between the network elements, i.e. the dynamics *on* the network.

One of the more complicated processes to model is the growth of a network that needs to maintain a specific type of dynamics. Some of the most natural examples come from the biological sciences. As mentioned in the introduction, the network of cells in a beating heart will attempt to maintain this function even as this network grows. Similarly, in the technological sciences electrical grids are designed to consistently carry power to consumers even as new lines, plants, etc. are added to the grid.

The *dynamics* of a network with a fixed structure of interactions can be modeled by iterating a map $F : X \to X$ on a product space $X = \bigoplus_{i \in N} X_i$ where $N = \{1, \ldots, n\}$ and each local phase space (X_i, d) is a metric space. Here the dynamics of the ith network element is given by the ith component of F given by

$$F_i : \bigoplus_{j \in I_i} X_j \to X_i, I_i \subseteq N,$$

where the set I_i indexes the elements that *interact* with the ith element. We refer to the system (F, X) as a *dynamical network*.

The dynamics of the network (F, X) is generated by iterating the function F where we let $\mathbf{x}(k) \in X$ be the state of the network at time $k \geq 0$ and $\mathbf{x}(k+1) = F(\mathbf{x}(k))$ be the state of the network at time $k+1$. The ith component $x_i(k)$ represents the state of the ith network element at time $k+1$.

The specific type of dynamics we consider here is global stability, which is observed in a number of important systems including neural networks [24, 25, 26, 27, 28], in epidemic models [29], and is also important in the study of congestion in computer networks [30] to name just a few. In a *globally stable* network, which we will simply refer to as *stable*, the state of the network

tends towards equilibrium irrespective of its present state. That is, there is an $\bar{\mathbf{x}} \in X$ such that for any $\mathbf{x}(0) \in X$, $\mathbf{x}(k) \to \bar{\mathbf{x}}$ as $k \to \infty$.

This globally attracting equilibrium is typically a state in which the network can carry out a specific task. Whether or not this equilibrium stays stable depends on a number of factors including external influences such as changes in the environment the network is in. However, it is worth emphasizing that not only can outside influences destabilize a network but potentially the network's own growth. As mentioned, an important example is cancer, which is the abnormal growth of cells that can lead to failure in a biological networks.

Here we consider how network specialization affects the stability of the network. We find that if the network is stable then, as the network specializes, the network can, in fact, lose stability (cf. example 2.24). However, if the network has a stronger form of stability, which we refer to as intrinsic stability, the network will remain stable as its topology is specialized.

For simplicity in our discussion we assume that the map $F : X \to X$ is differentiable and that each X_i is some closed interval of real numbers, although this can be done in more generality (see [23, 7]). Under this assumption we define the following matrix which can be used to investigate the stability of a dynamical network.

Definition 2.16. (Stability Matrix) For the dynamical network (F, X) suppose there exist finite constants

$$\Lambda_{ij} = \sup_{\mathbf{x} \in X} \left| \frac{\partial F_i}{\partial x_j}(\mathbf{x}) \right| \text{ for } 1 \leq i, j \leq n. \tag{2.5}$$

Then we call the matrix $\Lambda \in \mathbb{R}^{n \times n}$ the *stability matrix* of (F, X).

The stability matrix Λ can be thought of as a global linearization of the typically nonlinear dynamical network (F, X). The matrix Λ can also be used to describe the topology of the dynamical network (F, X) in that the graph we associate with (F, X) is the graph G where $M(G) = \Lambda$.

The original motivation for defining the matrix Λ is that if the linear dynamical network (Λ, X) given by $\mathbf{x}(k+1) = \Lambda\mathbf{x}(k)$ is stable then the same is true of the original network (F, X) (see [23]). To make this precise we let $\rho(\Lambda)$ denote the *spectral radius* of Λ.The fact that stability of

the linearized network (Λ, X) implies stability of the original network (F, X) is summarized in the following result, the proof of which can be found in [23].

Theorem 2.17. ***(Network Stability)*** *Suppose* Λ *is the stability matrix of the dynamical network* (F, X)*. If* $\rho(\Lambda) < 1$ *then the dynamical network* (F, X) *is stable.*

An important aspect of the dynamic stability described in Theorem 2.17 is that it is not the standard notion of stability. In [31] it is shown that if $\rho(\Lambda) < 1$ then the dynamical network (F, X) is not only stable but remains stable even if time-delays are introduced into the network's interactions. Since the introduction of time-delays can have a destabilizing effect on a network, the type of stability considered in Theorem 2.17 is a stronger version of the standard notion of stability. To distinguish between these two types of stability, the stability described in Theorem 2.17 is given the following name (see [31] for more details).

Definition 2.18. (Intrinsic Stability) If $\rho(\Lambda) < 1$, where Λ is a stability matrix of the dynamical network (F, X), then we say that this network is *intrinsically stable*.

The goal in this section is to describe how intrinsic stability is a natural notion for stability of a network with a topology that evolves via specialization. The idea is that, not only is it possible to specialize the graph structure G of a network with respect to a specialization rule τ but it is also possible to specialize a dynamical network (F, X) with respect to τ.

Consider the class of dynamical networks (F, X) having components of the form

$$F_i(\mathbf{x}) = \sum_{j=1}^{n} A_{ij} f_{ij}(x_j), \quad i = 1, \ldots, n \tag{2.6}$$

where the *interaction matrix* $A \in \{0, 1\}^{n \times n}$ is an $n \times n$ matrix of zeros and ones and $f_{ij} : X_j \to \mathbb{R}$ are functions with bounded derivatives for all $1 \leq i, j \leq n$. It is worth noting that the matrix A in equation (2.6) could be absorbed into the functions f_{ij}. The reason we use A is for convenience as it will be the means by which we will evolve (F, X). The reason is that the theory of graph specializations presented here applies equally well to matrices. That is, given a matrix $A \in \mathbb{R}^{n \times n}$ there is a unique graph G for which $M(G) = A$. Hence, for a given rule τ we can define

$$\tau(A) = M(\tau(G)) \text{ where } A = M(G)$$

to be the *specialization* of A with respect to τ. This allows us to define the specialization of the dynamical network (2.6).

Definition 2.19. (Specializations of Dynamical Networks) Let τ be a structural rule. Then the *specialization* of (F, X) in (2.6) with respect to τ is the dynamical network (F_τ, X_τ) with components

$$(F_\tau(\mathbf{x}))_i = \sum_{j=1}^{m} \tau(A)_{ij} f_{\tau(ij)}(x_j), \quad i = 1, \ldots, m$$

where $X_\tau = \mathbb{R}^m$ for $\tau(A) \in \mathbb{R}^{m \times m}$. Here $\tau(ij) = pq$ is the index such that $A_{\tau(ij)} = \tau(A)_{pq}$. We let Λ_τ denote the stability matrix of the specialized network (F_τ, X_τ).

The connection to real-world networks is the idea that, under certain conditions the topology of a dynamical network will specialize over time according to some fixed rule τ.

To give an example of a well known class of dynamical networks that can be specialized according to Definition 2.19 we consider the class of dynamical networks known as discrete-time recurrent neural networks (DRNN). The stability of such systems has been the focus of a large number of studies, especially time-delayed versions of these systems [41]. The class of DRNN we consider is the dynamical network (R, X), which has the form

$$R_i(\mathbf{x}) = a_i x_i + \sum_{j=1, j \neq i}^{n} b_{ij} g_j(x_j) + c_i, \quad i = 1, \ldots, n. \tag{2.7}$$

Here the component R_i describes the dynamics of the ith neuron of the network in which each $|a_i| < 1$ are the *feedback coefficients*, the matrix $B \in \mathbb{R}^{n \times n}$ with $b_{ii} = 0$ is the *connection weight matrix*, and the constants c_i are the *exogenous inputs* to the network. In the general theory of recurrent neural networks the functions $g_j : \mathbb{R} \to \mathbb{R}$ are typically assumed to be differentiable, monotonically increasing, and bounded. Here, for the sake of illustration we make the additional assumption that each g_j has a bounded derivative.

Before continuing we note that equation (2.7) can be written in the form of equation (2.6) by setting

$$f_{ij}(x_j) = \begin{cases} a_j x_j + c_j, & \text{for } i = j \\ b_{ij} g_j(x_j), & \text{for } i \neq j \end{cases} \quad \text{and} \quad A_{ij} = \begin{cases} 0 \text{ if } f_{ij}(x_j) \equiv 0, \\ 1 \text{ otherwise.} \end{cases} \tag{2.8}$$

In the following example we describe how a DRNN is specialized with respect to a given rule.

Example 2.20. (Topological Specialization of a DRNN) Consider the recurrent dynamical network $(R, \mathbb{R}^4)$ given by equation (2.7) where each $a_j = \alpha$, $b_{ij} = \beta$ for $i \neq j$ and is zero otherwise, $c_j = \gamma$, and $g_j(x) = \tanh(x)$ for $1 \leq i, j \leq 4$ where $|\alpha| < 1$ and $\beta, \gamma \in \mathbb{R}$.

Let A be the interaction matrix

$$A = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 \end{bmatrix}.$$

We choose the function $g_j(x) = \tanh(x)$ as this is a standard activation function used to model neural interactions in network science as it converts continuous inputs to binary outputs. Under these conditions the dynamical network $(R, \mathbb{R}^4)$ is given by

$$R(\mathbf{x}) = \begin{bmatrix} \alpha x_1 + \beta[\tanh(x_2)] + \gamma \\ \alpha x_2 + \beta[\tanh(x_1) + \tanh(x_3)] + \gamma \\ \alpha x_3 + \beta[\tanh(x_4)] + \gamma \\ \alpha x_4 + \beta[\tanh(x_1) + \tanh(x_3)] + \gamma \end{bmatrix}.$$

We evolve the topology of the network $(R, \mathbb{R}^4)$ using the rule $\delta(V) = \{v \in V : deg_{in}(v) = 1\}$, where $deg_{in}(v)$ is the in-degree of vertex v. This rule specializes the matrix A and the dynamical network $(R, \mathbb{R}^4)$ into $\delta(A)$ and $(R_\delta, \mathbb{R}^6)$ where

$$\delta(A) = \begin{bmatrix} 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix} \text{ and } R_\delta(\mathbf{x}) = \begin{bmatrix} \alpha x_1 + \beta[\tanh(x_4) + \tanh(x_5)] + \gamma \\ \alpha x_2 + \beta[\tanh(x_3) + \tanh(x_6)] + \gamma \\ \alpha x_3 + \beta[\tanh(x_1)] + \gamma \\ \alpha x_4 + \beta[\tanh(x_2)] + \gamma \\ \alpha x_5 + \beta[\tanh(x_1)] + \gamma \\ \alpha x_6 + \beta[\tanh(x_2)] + \gamma \end{bmatrix},$$

respectively. Using the fact that $\sup_{x\in\mathbb{R}} |\frac{d}{dx}\tanh(x)| = 1$, the stability matrix Λ of $(R, \mathbb{R}^4)$ and the stability matrix Λ_δ of $(R_\delta, \mathbb{R}^6)$ are

$$\Lambda = \begin{bmatrix} |\alpha| & |\beta| & 0 & 0 \\ |\beta| & |\alpha| & |\beta| & 0 \\ 0 & 0 & |\alpha| & |\beta| \\ |\beta| & 0 & |\beta| & |\alpha| \end{bmatrix} \text{ and } \Lambda_\delta = \begin{bmatrix} |\alpha| & 0 & 0 & |\beta| & |\beta| & 0 \\ 0 & |\alpha| & |\beta| & 0 & 0 & |\beta| \\ |\beta| & 0 & |\alpha| & 0 & 0 & 0 \\ 0 & |\beta| & 0 & |\alpha| & 0 & 0 \\ |\beta| & 0 & 0 & 0 & |\alpha| & 0 \\ 0 & |\beta| & 0 & 0 & 0 & |\alpha| \end{bmatrix}.$$

The graphs G and $\delta(G)$ associated with the stability matrices Λ and Λ_δ are shown in Figure 2.11 left and center, respectively. Importantly, $\delta(\Lambda) = \Lambda_\delta$, meaning that if the stability matrix of $(R, \mathbb{R}^4)$ is specialized by δ the result is the stability matrix of the expanded network $(R_\delta, \mathbb{R}^6)$.

The fact that $\delta(\Lambda) = \Lambda_\delta$ in this example is not a coincidence but is a simple consequence of how specializations of dynamical networks are defined. That is, if a dynamical network (F, X) given by (2.6) with stability matrix Λ is specialized with respect to the rule τ, the resulting dynamical network (F_τ, X_τ) has the stability matrix $\Lambda_\tau = \tau(\Lambda)$. It is therefore possible to test the stability of a specialized version of (F, X) with stability matrix Λ by specializing Λ.

It is worth noting that in previous studies of dynamical networks including DRNN the goal has been to determine under what condition(s) a given network has stable dynamics (see for instance the references in [41]). Here we consider a different but related question which is, under what con-

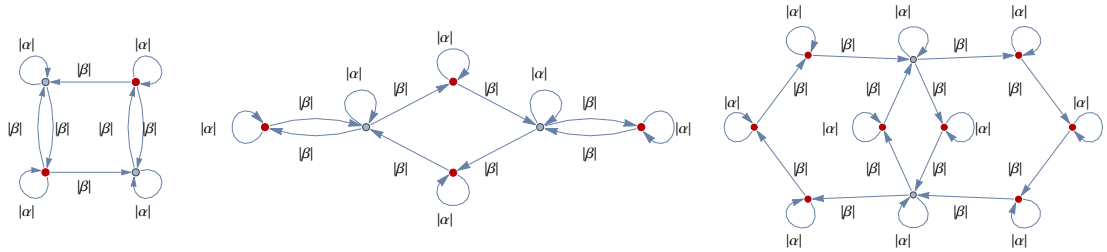


Figure 2.11: The sequence G, $\delta(G)$, and $\delta^2(G)$ represents the topology of the recurrent dynamical network $(R, \mathbb{R}^4)$ and its sequence of specializations $(R_\delta, \mathbb{R}^6)$, and $(R_{\delta^2}, \mathbb{R}^{10})$ respectively, considered in example 2.20. The vertices selected by the rule δ are highlighted in each graph. The parameters $\alpha, \beta \in \mathbb{R}$.

dition(s) does a dynamical network with an evolving structure of interactions maintain its stability as it evolves. As a partial answer to this general question, we show that if a dynamical network is not only stable but intrinsically stable and evolves via specialization then its stability is preserved as its topology evolves.

Theorem 2.21. ***(Stability of Specialized Networks)*** *Let* (F, X) *be a dynamical network given by equation* (2.6) *and* τ *a specialization rule. The specialized dynamical network* (F_τ, X_τ) *is intrinsically stable if and only if* (F, X) *is intrinsically stable.*

The importance of Theorem 2.21 is that it describes, as far as we know, the first general mechanism for evolving the structure of a network that preserves the network's stability. A proof of Theorem 2.21 is the following.

Proof. Suppose the dynamical network (F, X) is intrinsically stable so that in particular $\rho(\Lambda) < 1$. Given a specialization rule τ, Theorem 2.10 states, in terms of matrices, that

$$\sigma(\tau(\Lambda)) = \sigma(\Lambda) \cup \sigma(\Lambda_1)^{n_1 - 1} \cup \cdots \cup \sigma(\Lambda_m)^{n_m - 1} \tag{2.9}$$

where each Λ_i is a square submatrices of Λ and $n_i \geq 0$ for all $1 \leq i \leq m$.

Since Λ is a nonnegative matrix then $\rho(\Lambda_i) \leq \rho(\Lambda)$ for all $1 \leq i \leq m$ (see, for instance, corollary 8.1.20 in [40]). Hence, $\rho(\tau(\Lambda)) = \rho(\Lambda) < 1$ by equation (2.9). Since $\tau(\Lambda) = \Lambda_\tau$ is the

stability matrix of (F_τ, X_τ) then this implies that $\rho(\Lambda_\tau) < 1$ so that the specialized network (F_τ, X_τ) is intrinsically stable. Conversely, if (F_τ, X_τ) is intrinsically stable then $\rho(\Lambda_\tau) < 1$ and equation (2.9) likewise implies that $\rho(\Lambda) < 1$, completing the proof. □

Theorem 2.21 is built on the fact that the spectral radius of a graph is preserved under a graph specialization if the graph has nonnegative weights. This follows directly from the proof of Theorem 2.21 and is stated as the following corollary.

Corollary 2.22. *Suppose $G = (V, E, \omega)$ is a graph with nonnegative weights. Then for any base vertex set $B \subseteq V$ the spectral radius $\rho(\mathcal{S}_B(G)) = \rho(G)$.*

Note that the recurrent network $(R, \mathbb{R}^4)$ in example 2.20 has the stability matrix Λ with eigenvalues $\sigma(\Lambda) = \{|a| \pm \sqrt{2}|b|, |a|, |a|\}$. Hence, $(R, \mathbb{R}^4)$ is intrinsically stable if $||a| + \sqrt{2}|b|| < 1$. If this condition holds then, Theorem 2.21 implies that the specialized network $(R_\delta, \mathbb{R}^6)$ is intrinsically stable. Here, one can directly compute that $\sigma(\Lambda_\delta) = \{|a| \pm \sqrt{2}|b|\} \cup \{|a|\}^4$ verifying the result.

It is also worth noting that if (F, X) is given by (2.6) then its specialization (F_τ, X_τ) is also of the same form. Hence, (F_τ, X_τ) can also be specialized by τ, resulting in the dynamical network (F_{τ^2}, X_{τ^2}). As a direct consequence to Theorem 2.21, if (F, X) is intrinsically stable then any sequence of specializations of (F, X) results in an intrinsically stable network.

Corollary 2.23. *Suppose (F, X) is a dynamical network given by* (2.6) *and τ is a specialization rule. If (F, X) is intrinsically stable then (F_{τ^j}, X_{τ^j}) is intrinsically stable for all $j \geq 0$.*

Continuing the sequence of network specializations in example 2.20, if we evolve the network $(R_\delta, \mathbb{R}^6)$ again with respect to δ the result is the dynamical network $(R_{\delta^2}, \mathbb{R}^{10})$ whose stability matrix Λ_{δ^2} is represented by the graph $\delta^2(G)$ shown in Figure 2.11 (right). One can again check that

$$\sigma(\Lambda_{\delta^2}) = \{|a| \pm \sqrt{2}|b|\} \cup \{|a|\}^8.$$

As guaranteed by corollary 2.23, $(R_{\tau^2}, \mathbb{R}^{10})$ is intrinsically stable if and only if the original network $(R, \mathbb{R}^4)$ is intrinsically stable.

In contrast, if a dynamical network (F, X) is stable but not intrinsically stable, it can fail to maintain its stability as its topology is specialized even if it is a simple linear dynamical network as is illustrated in the following example.

Example 2.24. (Loss of Stability) Let w be the rule that selects all vertices of a graph without loops. Consider the linear dynamical network $(F, \mathbb{R}^3)$ given by

$$F(\mathbf{x}) = \begin{bmatrix} 0 & -1 & 3/4 \\ 0 & 0 & 1/2 \\ -1/2 & 0 & 3/2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}. \tag{2.10}$$

Its specialization $(F_w, \mathbb{R}^4)$ with respect to w is given by

$$F_w(\mathbf{x}) = \begin{bmatrix} 0 & -1 & 0 & 3/4 \\ 0 & 0 & 1/2 & 0 \\ -1/2 & 0 & -3/2 & 0 \\ -1/2 & 0 & 0 & -3/2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}.$$

If $M \in \mathbb{R}^{3\times 3}$ is the matrix in (2.10) such that $F(\mathbf{x}) = M\mathbf{x}$ then $F_w(\mathbf{x}) = w(M)\mathbf{x}$ where $w(M) \in \mathbb{R}^{4\times 4}$. Here one can compute that $\rho(M) \approx .938$ whereas $\rho(w(M)) = 3/2$. Since both systems are linear, it follows immediately that $(F, \mathbb{R}^3)$ is a stable dynamical network whereas $(F_w, \mathbb{R}^4)$ is unstable.

The reason $(F, \mathbb{R}^3)$ can lose its stability as it is specialized is that it is not intrinsically stable. That is, the stability matrix of (F, X) is the matrix $|M|$, which is the matrix with entries $|M|_{ij} = |M_{ij}|$ with spectral radius $\rho(|M|) = 1.787$. Since this is greater than one, the system is stable but not intrinsically stable. Therefore, it is possible, as it is demonstrated here, for the network to lose stability as its topology is specialized.

Example 2.24 is meant to emphasize the fact that more than the standard notion of stability is needed to guarantee a network's stability as a network evolves under some rule τ. A natural and open question is whether this is the case for other types of dynamics exhibited by networks, i.e.

what conditions are required for other dynamical behaviors such as multistability, periodicity, and synchronization, etc. to be maintained as the network grows.

2.7 Concluding Remarks

In this chapter we introduce a class of models of network formation, which we refer to as *specialization models* of network growth. These models are based on the observation that most, if not all, real networks specialize the function of their components as they evolve. As a first observation we note that by specializing a network via this model the result is a network whose topology becomes sparser, more modular, and more hierarchical. This is particularly relevant since each of these properties is found throughout networks studied in the biological, technological, and social sciences.

Our method of specialization is highly flexible in that a network can be specialized over any subset of its elements, i.e. any network base. Since so many bases are possible, a natural question is, "given a particular real-world network can we find a rule that generates a base that can be used to accurately model this network's growth?"

To give evidence that this is possible, we first consider the simple rule that randomly selects a certain percentage of a network's elements. Numerically we show that this rule evolves the topology of a network in ways that are consistent with the properties widely observed in real networks. This includes the small-world property, disassortativity, and a right-skewed degree distribution. We then consider two other rules that give us networks with real-world like clustering coefficients and power-law like degree distributions, respectively. So far as we know this is the only such model to capture these properties that also creates an increasingly sparse, modular, and hierarchical network topology.

Additionally, we show how certain specialization rules, which we refer to as *structural rules*, can be used to compare the topology of different networks. This notion of similarity, which we refer to as *specialization equivalence*, can be used to partition all networks into those that are similar and dissimilar with respect to a given rule τ. It is worth emphasizing that in practice it

is important that this rule be designed by the particular biologist, chemist, physicist, etc. to have some significance with respect to the nature of the networks under consideration.

Additionally, we show that as a network is specialized the network maintains a number of important spectral properties (see Theorems 2.10, 2.14 and 2.15). These spectral properties are related to the dynamics *on* the network, which in turn is related to the network's function. This suggests an interdependence between the network's ability to maintain its primary function and the topological properties of modularity, etc. that results from this process of specialization.

With the network's function in mind, we show that if a network's dynamics is *intrinsically stable* then the network will remain intrinsically stable even as the network's topology becomes more specialized (see Theorem 2.21). As the standard notion of stability is not always preserved under specialization this suggests that real networks may need to have stronger versions of behavior such as stability, multistability, periodicity, etc. that are robust to changes in network structure to maintain their function rather than those that are typically studied in the theory of dynamical systems.

The notion of specialization and the associated specialization growth model introduced in this chapter also leads to a number of open questions, a few of which we mention here. The first, and likely most important, is whether specific specialization rules can be designed to model the growth of specific networks in a way that captures the network's finer more specific structure. Such rules will likely be very network dependent and therefore need to be devised and examined again by the particular biologist, chemist, physicist, etc. who has some expertise with the nature of the particular network.

Related to this, the properties observed as consequences of specializing using the rules introduced in examples 2.5 2.6 and 2.7 are only observed numerically. It is currently unknown whether these properties can be proven rigorously.

Last, it is worth reiterating the question posed at the end of section 2.6 regarding other types of intrinsic dynamics. Specifically, whether other types of network dynamics can be established, analogous to intrinsic stability, that are robust to changes in network growth, specifically special-

ization. The hope is that by discovering such new types of dynamics we can better understand the interplay of network topology, dynamics, and growth.

Chapter 3. Equitable Decompositions

3.1 Overview and Background

Spectral graph theory is the study of the relationship between two objects, a graph G and an associated matrix M. The goal of this theory is to understand how spectral properties of the matrix M can be used to infer structural properties of the graph G and vice versa.

The particular structures we consider in this chapter are graph symmetries. A graph is said to have a *symmetry* if there is a permutation $\phi : V(G) \to V(G)$ of the graph's vertices $V(G)$ that preserves (weighted) adjacencies. The permutation ϕ is called an *automorphism* of G, hence the symmetries of the graph G are characterized by the graph's set of automorphisms. Intuitively, a graph automorphism describes how parts of a graph can be interchanged in a way that preserves the graph's overall structure. In this sense these *smaller parts*, i.e., subgraphs, are symmetrical and together these subgraphs constitute a graph symmetry.

In a previous paper [4] it was shown that if a graph G has a particular type of automorphism ϕ then it is possible to decompose any matrix M that respects the structure of G into a number of smaller matrices $M_\phi, B_1, \ldots, B_{k-1}$. Importantly, the eigenvalues of M and the collective eigenvalues of these smaller matrices are the same, i.e.

$$\sigma(M) = \sigma(M_\phi) \cup \sigma(B_1) \cup \cdots \cup \sigma(B_{k-1}).$$

This method of decomposing a matrix into a number of smaller pieces over a graph symmetry is referred to as an *equitable decomposition* due to its connection with the theory of equitable partitions. An *equitable partition* of the adjacency matrix A associated with a graph G is a partition of the graph's set of vertices, which may arise from an automorphism ϕ of G, yielding a smaller matrix A_ϕ whose eigenvalues form a subset of the spectrum of A (Theorem 9.3.3 of [42] , Theorem 3.9.5 of [43]).

In [4] the notion of an equitable partition is extended to other matrices beyond the adjacency matrix of a graph to include various Laplacian matrices, distance matrices, etc. (see Proposition 3.4). This class of matrices, referred to as *automorphism compatible* matrices, are those matrices associated with a graph G that can be equitably decomposed over an automorphism ϕ of G. In particular, the matrix M_ϕ in the resulting decomposition is the same as the matrix that results from an equitable decomposition of G if $M = A$ is the adjacency matrix of G.

The particular types of automorphisms considered in [4] are referred to as uniform and basic automorphisms. A *uniform* automorphism ϕ is one in which all orbits have the same cardinality (see Remark 3.2). A *basic* automorphism ϕ is an automorphism for which all *nontrivial orbits*, i.e. orbits of size greater than one, have the same cardinality. Hence, any uniform automorphism is a basic automorphism.

Since many graph automorphisms are not basic, a natural question is whether an automorphism compatible matrix M can be decomposed over a nonbasic automorphism. In this chapter we first show in Section 3.3 that if an automorphism is separable, i.e. is an automorphism whose order is the product of distinct primes, then there are basic automorphisms $\psi_0, \psi_1, \ldots, \psi_h$ that induce a sequence of equitable decompositions on M. The result is a collection of smaller matrices $M_\phi, B_1, \ldots, B_k$ such that

$$\sigma(M) = \sigma(M_\phi) \cup \sigma(B_1) \cup \cdots \cup \sigma(B_k)$$

where $k = p_0 p_1 \ldots p_h - 1$ and M_ϕ is again the matrix associated with the equitable partition induced by ϕ (see Theorem 3.4). That is, the theory of equitable decompositions can be extended to any separable automorphism of a graph G.

In section 3.4 we develop a more sophisticated decomposition process which allows us to decompose matrices with prime-power automorphism ϕ, i.e. an automorphism whose order is p^N for some prime p and integer $N > 1$. This more advanced method is needed as it guarantees that there exists a sequence of automorphisms to completely decompose a matrix with a prime-power automorphism using a process which is similar to the one described in Section 3.3. Example 3.9

describes an instance where just using the method of Section 3.3 cannot be used to complete the equitable decomposition.

Finally in section 3.5, we show how it is possible to completely equitably decompose a matrix with *any* automorphism, by finding a sequence of automorphisms to decompose the matrix iteratively. For each of the decomposition processes described above an explicit algorithm is given that describes in detail the steps to complete the decomposition.

After establishing how to do equitable decompositions, we then show how the eigenvectors (and generalized eigenvectors) of M can be decomposed over an automorphism ϕ. More specifically, if M can be decomposed into the matrices $M_\phi, B_1, \ldots, B_k$ over ϕ then the eigenvectors of M can be explicitly constructed from the eigenvectors of $M_\phi, B_1, \ldots, B_k$. That is, the eigenvectors of these smaller matrices form the building blocks of the larger eigenvectors of the original matrix M (see Theorem 3.18), which we refer to as an equitable decomposition of the eigenvectors (and generalized eigenvectors) of M.

Importantly, an equitable decomposition of M, as opposed to its spectral decomposition, does not require any knowledge of the matrix' eigenvalues or eigenvectors. Only the knowledge of a symmetry of G is needed. In fact, if an automorphism describes a graph symmetry that involves only part of the graph i.e. a *local symmetry*, this local information together with the theory presented here can be used to determine properties of the graph's associated eigenvalues and eigenvectors, which in general depend on the entire graph structure!

This method of using local symmetries to determine spectral properties of a graph is perhaps most useful in analyzing the spectral properties of real-world networks. One reason is that many networks have a high degree of symmetry [5] when compared, for instance, to randomly generated graphs [44, 6, 45, 46]. From a practical point of view, the large size of these networks limit our ability to quickly compute their associated eigenvalues and eigenvectors. However, their high degree of symmetry suggests that it may be possible to effectively estimate a network's spectral properties by equitably decomposing the network over local symmetries, which is a potentially much more feasible task.

For instance, we show that in a network given by the graph G with automorphism compatible matrix M, the spectral radius of M and its divisor matrix M_ϕ are equal if M is both nonnegative and irreducible (see Proposition 3.21). This result is of interest by itself since the spectral radius can be used to study stability properties of a network's dynamics [23, 31].

Finally, we show that the Gershgorin region associated with an equitable decomposition is contained in the Gershgorin region associated with the original matrix (see Theorem 3.25). Since the eigenvalues of a matrix are contained in its Gershgorin region [47], then by equitably decomposing a matrix over some automorphism that is either basic or separable it is possible to gain improved eigenvalue estimates of the matrix' eigenvalues. Again, this result is potentially useful for estimating the eigenvalues associated with a real network as such networks often have a high degree of symmetry.

This Chapter is organized as follows. In Section 3.2 we summarize the theory of equitable decompositions found in [4]. In Section 3.3 we describe how the theory of equitable decompositions can be extended to separable automorphisms by showing that a decomposition over such an automorphism ϕ can be realized as a sequence of decompositions over basic automorphisms $\psi_0, \psi_1, \ldots, \psi_h$ (Corollary 3.7). We also present an algorithm describing how these automorphisms can be generated and used to equitably decompose an associated matrix.

In Section 3.4 we give an algorithm for decomposing a graph over any automorphism of order p^N. Finally, in Section 3.5 we give a general algorithm for equitably decomposing a graph over *any* of its automorphisms.

In Section 3.6 we introduce the notion of an equitable decomposition of a matrix' eigenvectors and generalized eigenvectors (Theorem 3.18). We also demonstrate that M and M_ϕ have the same spectral radius if M is both nonnegative and irreducible (Proposition 3.21).

In Section 3.7 we show that we gain improved eigenvalue estimates using Gershgorin's theorem when a matrix is equitably decomposed (Theorem 3.25), which we demonstrate on a large social network from the pre-American revolutionary war era.

Though much of this theory is stated in term of matrices, in Section 3.8 we show how the theory of equitable decompositions can be directly applied to graphs, at least for those with separable automorphisms. This provides a visual way of thinking about equitable decompositions for graphs with separable automorphisms. Section 3.9 contains some closing remarks including a few open questions regarding equitable decompositions.

3.2 Graph Symmetries and Equitable Decompositions

One of our main concerns in this chapter is understanding how symmetries in a graph's structure (i) affect the eigenvalues and eigenvectors of a matrix $M = M(G)$ and (ii) how these symmetries can be used to decompose the matrix M into a number of smaller matrices in a way that preserves the eigenvalues of M. Such graph symmetries are formally described by the graph's set of automorphisms.

Definition 3.1. (Graph Automorphism) An *automorphism* ϕ of an unweighted graph G is a permutation of $V(G)$ such that the adjacency matrix $A = A(G)$ satisfies $A_{ij} = A_{\phi(i)\phi(j)}$ for each pair of vertices i and j. Note that this is equivalent to saying i and j are adjacent in G if and only if $\phi(i)$ and $\phi(j)$ are adjacent in G. For a weighted graph G, if $w(i, j) = w(\phi(i), \phi(j))$ for each pair of vertices i and j, then ϕ is an automorphism of G.

The set of all automorphisms of G is a group, denoted by $\mathrm{Aut}(G)$. The *order* of ϕ is the smallest positive integer ℓ such that ϕ^ℓ is the identity.

Remark. For a graph G with automorphism ϕ, we define the relation $\sim$ on $V(G)$ by $u \sim v$ if and only if $v = \phi^j(u)$ for some nonnegative integer j. It follows that $\sim$ is an equivalence relation on $V(G)$, and the equivalence classes are called the *orbits* of ϕ. The orbit associated with the vertex i is denoted $\mathcal{O}_\phi(i)$.

Here, as in [4] we consider those matrices $M = M(G)$ associated with a graph G whose structure mimics the symmetries of the graph.

Definition 3.2. (Automorphism Compatible) Let G be a graph on n vertices. An $n \times n$ matrix M is *automorphism compatible* on G if, given any automorphism ϕ of G and any $i, j \in \{1, 2, \ldots, n\}$, $M_{\phi(i)\phi(j)} = M_{ij}$.

Some of the most well-known matrices that are associated with a graph are automorphism compatible. This includes the adjacency matrix, combinatorial Laplacian matrix, signless Laplacian matrix, normalized Laplacian matrix, and distance matrix of a simple graph. Additionally, the weighted adjacency matrix of a weighted graph is automorphism compatible. (See Proposition 3.4, [4].)

If $M = M(G)$ is an automorphism compatible matrix, M can be decomposed over an automorphism ϕ of G into a number of smaller matrices if ϕ is a basic automorphism.

Definition 3.3. (Basic Automorphism) If ϕ is an automorphism of a graph G with all orbits of size $k > 1$ and possibly 1, then ϕ is a *basic automorphism* of G with orbit size k. Any vertices with orbit size 1 are said to be *fixed* by ϕ.

Given a basic automorphism ϕ with orbit size k, we form a set by choosing one vertex from each orbit of size k. We call this set $\mathcal{T}_0$ of vertices a *semi-transversal* of the orbits of ϕ. Further we define the set

$$\mathcal{T}_\ell = \{\phi^\ell(v) \mid v \in \mathcal{T}_0\} \tag{3.1}$$

for $\ell = 0, 1, \ldots, k-1$ to be the ℓth power of $\mathcal{T}_0$ and we let $M[\mathcal{T}_i, \mathcal{T}_j]$ be the submatrix of M whose rows are indexed by $\mathcal{T}_i$ and whose columns are indexed by $\mathcal{T}_j$. This notion of a semi-transversal allows us to decompose an automorphism compatible matrix $M = M(G)$ in the following way.

Theorem 3.4. ***(Basic Equitable Decomposition)*** *[4] Let G be a graph on n vertices, let ϕ be a basic automorphism of G of size $k > 1$, let $\mathcal{T}_0$ be a semi-transversal of the k-orbits of ϕ, let $\mathcal{T}_f$ be the vertices fixed by ϕ, and let M be an automorphism compatible matrix on G. Set $F = M[\mathcal{T}_f, \mathcal{T}_f]$, $H = M[\mathcal{T}_f, \mathcal{T}_0]$, $L = M[\mathcal{T}_0, \mathcal{T}_f]$, $M_m = M[\mathcal{T}_0, \mathcal{T}_m]$, for $m = 0, 1, \ldots, k-1$, $\omega = e^{2\pi i/k}$, and*

$$B_j = \sum_{m=0}^{k-1} \omega^{jm} M_m, \quad j = 0, 1, \ldots, k-1. \tag{3.2}$$

Then there exists an invertible matrix S that can be explicitly constructed such that

$$S^{-1}MS = M_\phi \oplus B_1 \oplus B_2 \oplus \cdots B_{k-1} \tag{3.3}$$

where $M_\phi = \begin{bmatrix} F & kH \\ L & B_0 \end{bmatrix}$. *Thus* $\sigma(M) = \sigma\left(M_\phi\right) \cup \sigma(B_1) \cup \sigma(B_2) \cup \cdots \cup \sigma(B_{k-1})$.

The decomposition in Equation (3.3) is referred to as an *equitable decomposition* of M associated with the automorphism ϕ. The reason for this is that this decomposition is related to an equitable partition of the graph G.

Definition 3.5. (Equitable Partition) An *equitable partition* of a graph G and a matrix M associated with G, is a partition π of $V(G)$, $V(G) = V_1 \cup \ldots \cup V_k$ which has the property that for all i, $j \in \{1, 2, \ldots, k\}$

$$\sum_{t \in V_j} M_{st} = D_{ij} \tag{3.4}$$

is a constant D_{ij} for any $s \in V_i$. The $k \times k$ matrix $M_\pi = D$ is called the *divisor matrix* of M associated with the partition π.

Definition 3.5 is, in fact, an extension of the standard definition of an equitable partition, which is defined for *simple graphs*. For such graphs the requirement that π be an equitable partition is equivalent to the condition that any vertex $\ell \in V_i$ has the same number of neighbors in V_j for all $i, j \in \{1, \ldots, k\}$ (for example, see p. 195-6 of [42]).

An important fact noted in [4] is that, if ϕ is a basic automorphism of G and M is an automorphism compatible matrix associated with G, the orbits of ϕ form an equitable partition of $V(G)$ (see Proposition 3.2, [4]). If M is equitably decomposed over the basic automorphism ϕ as in Equation (3.3), the matrix M_ϕ in the resulting decomposition is in fact the divisor matrix D associated with the equitable partition induced by ϕ (see Theorem 4.4, [4]), which is the reason this decomposition is referred to as an equitable decomposition.

If ϕ is an automorphism in which every orbit has the same size $k > 1$ then ϕ is referred to as a *uniform automorphism* of *size k*. Any uniform automorphism is clearly a basic automorphism

in the sense that it is a basic automorphism that fixes no vertices. Thus, Theorem 3.4 holds for uniform automorphisms as well, in which case the divisor matrix $M_\phi = B_0$.

If a graph G has a non-basic automorphism ϕ, the current theory of equitable decompositions does not directly allow us to decompose a matrix $M = M(G)$ over ϕ. In the following section we show that an automorphism compatible matrix M can be decomposed with respect to any separable automorphism ϕ of G via a sequence of basic automorphisms.

3.3 Equitable Partitions using Separable Automorphisms

Many graph automorphisms are not basic automorphisms. In this section we will demonstrate how to equitably decompose a matrix with respect to an arbitrary separable automorphism by repeated use of Theorem 3.4. Here, a *separable automorphism* ϕ of a graph G is an automorphism whose order $|\phi| = p_0 p_1 \dots p_h$ where $p_0, p_1, \dots, p_h$ are distinct primes. Before we can describe an equitable decomposition over a separable automorphism we first need the following propositions and algorithm.

Remark. Notice that if $B = M_\phi \oplus B_1 \oplus \cdots \oplus B_{k-1}$ is the equitable decomposition of a matrix M with respect to ϕ, then we may view B as the weighted adjacency matrix for a new graph $\tilde{G}$ with the same vertex set as G. In the proofs of Theorems 3.8 and 4.4 of [4] the rows and columns of the matrix M are labeled in the order $\mathcal{U}, \mathcal{T}_0, \dots, \mathcal{T}_{k-1}$. We continue this *row/column labeling*, so that the labeling for the divisor matrix M_ϕ follows the ordering $\mathcal{U}, \mathcal{T}_0$, and for all remaining matrices B_m in the decomposition the labeling follows $\mathcal{T}_m$.

Proposition 3.6. *Let ϕ be an automorphism of order pq with p prime and $p \nmid q$ of a graph $G = (V, E, w)$ with automorphism compatible matrix M. Then $\psi = \phi^q$ is a basic automorphism of G with order p and we can construct an automorphism $\tilde{\phi}$ associated with the equitable decomposition of M over ψ of order q such that the divisor matrix $(M_\psi)_{\tilde{\phi}} = M_\phi$.*

Proof. Let M and ϕ be as described in Proposition 3.6. For ease of notation, let $M(i, j) = M_{ij}$, the ij^{th} element of M. Certainly, the automorphism $\psi = \phi^q$ must have order p implying that ψ is a basic automorphism.

In order to perform an equitable decomposition with respect to ψ, we choose a semi-transversal $\mathcal{T}_0$ in the following way: For each orbit of ϕ that is not fixed by ψ, pick an element a. Then $|a| = pq_a$ for some $q_a \in \mathbb{Z}_{>0}$. We add the elements of the set $\{a, \phi^p(a), \phi^{2p}(a), \ldots, \phi^{(q_a-1)p}(a)\}$ to $\mathcal{T}_0$. We let $\mathcal{U}$ denote the set of vertices fixed by ψ.

To see that $\mathcal{T}_0$ is a semi-transversal, notice that the element a gives q_a orbits under ψ and there are q_a elements in the set listed above. We now show that the elements in the above set must come from different orbits. Suppose that $\phi^{\eta p}(a)$ and $\phi^{\eta' p}(a)$ (with $0 < |\eta - \eta'| < q_a$) are in the same ψ-orbit, then for some integer $s < p$, $\phi^{\eta p}(a) = \psi^s \phi^{\eta' p}(a) = \phi^{q_a s + \eta' p}(a)$. Thus, $q_a \mid (\eta - \eta')$, a contradiction.

Now we define a map $\tilde{\phi} = \phi^p$, and notice that $\tilde{\phi}(\mathcal{T}_m) \subseteq \mathcal{T}_m$. Recall that the decomposed matrix $B = M_\psi \oplus B_1 \oplus \cdots \oplus B_{p-1}$ guaranteed by Theorem 4.4 of [4], will have row and column order agreeing with the vertex order $\mathcal{U}, \mathcal{T}_0, \mathcal{T}_1, \ldots, \mathcal{T}_{p-1}$. Thus, to show that $\tilde{\phi}$ is an automorphism of B, we need only demonstrate that each $\tilde{\phi}|_{\mathcal{T}_m}$ is an automorphism on B_m (see Theorem 3.4). Recall that

$$B_m = \sum_{j=0}^{p-1} \omega^{mj} M[\mathcal{T}_0, \mathcal{T}_j],$$

Thus, if $a, b \in \mathcal{T}_m$, and we wish to calculate the (a, b) entry in B_m, we must examine the corresponding entries in M which come from $\mathcal{T}_0$ and $\mathcal{T}_j$. This is expressed in the first equality below:

$$\begin{aligned} B_m(\tilde{\phi}|_{\mathcal{T}_m}(a), \tilde{\phi}|_{\mathcal{T}_m}(b)) &= \sum_{j=0}^{p-1} \omega^{mj} M(\psi^{-m}\tilde{\phi}(a), \psi^{j-m}\tilde{\phi}(b)) = \sum_{j=0}^{p-1} \omega^{mj} M(\phi^p\psi^{-m}(a), \phi^p\psi^{j-m}(b)) \\ &= \sum_{j=0}^{p-1} \omega^{mj} M(\psi^{-m}(a), \psi^{j-m}(b)) = B_m(a, b) \end{aligned}$$

where the second equality holds because $\psi = \phi^q$, and the third equality is the defining property of automorphism compatible matrices. Thus, $\tilde{\phi}$ is an automorphism on each B_m and subsequently, $\tilde{\phi}$ is an automorphism on B, the decomposition of M.

The equality below similarly shows that $\tilde{\phi}$ is an automorphism of the vertices (a, b) that appear in M_ψ (those in $\mathcal{U} \cup \mathcal{T}_0$).

$$M_\psi(\tilde{\phi}(a), \tilde{\phi}(b)) = M_\psi(\phi^p(a), \phi^p(b)) = \sum_{m=1}^{|\mathcal{O}_\psi(b)|} M(\phi^p(a), \phi^{qm+p}(b))$$
$$= \sum_{m=1}^{|\mathcal{O}_\psi(b)|} M((a), \phi^{qm}(b)) = M_\psi(a, b).$$

To show the final equality in Proposition 3.6, we consider a semi-transversal $\widetilde{\mathcal{T}}_0$ of $\tilde{\phi}$ and the set of vertices fixed by $\tilde{\phi}$ (which we call $\widetilde{\mathcal{U}}$). Then for vertices a and b in $\widetilde{\mathcal{U}} \cup \widetilde{\mathcal{T}}_0$ we find that

$$M_\phi(a, b) = \sum_{s \in \mathcal{O}_\phi(b)} M(a, s) = \sum_{m \in \mathcal{O}_{\tilde{\phi}}(b)} \left(\sum_{s \in \mathcal{O}_\psi(m)} M(a, s) \right) = \sum_{s \in \mathcal{O}_{\tilde{\phi}}(b)} M_\psi(a, s) = (M_\psi)_{\tilde{\phi}}(a, b).$$

The second equality holds because

$$\mathcal{O}_\phi(s) = \bigcup_{t \in \mathcal{O}_{\tilde{\phi}}(S)} \mathcal{O}_\psi(t)$$

Hence, $M_\phi = (M_\psi)_{\tilde{\phi}}$. □

Thus, for any automorphism $\phi \in Aut(G)$ of order $pq = \ell$ where p is prime and $p \nmid q$, we can equitably decompose an automorphism compatible matrix M over ϕ^q and subsequently create another automorphism $\tilde{\phi}$ associated with the decomposed matrix. In fact, if ϕ is separable then we can repeat this process until we exhaust each of the distinct prime factors $p_0, p_1, \ldots, p_h$ of ℓ where $\ell = p_0 p_1 \cdots p_h$ is the order of ϕ. This decomposition of the matrix M, is summarized in the following theorem.

Theorem 3.7. ***(Equitable Decompositions Over Separable Automorphisms)*** *Let ϕ be any separable automorphism of a graph G with automorphism compatible matrix M. Then there exists basic automorphisms $\psi_1, \ldots, \psi_h$ that induce a sequence of equitable decompositions on M, such that the divisor matrix satisfies*

$$M_\phi = (\ldots (M_{\psi_0})_{\psi_1} \ldots)_{\psi_h}.$$

Proof. This follows from repeated use of Proposition 3.6. □

We now give an algorithm for decomposing a graph with respect to any of its automorphisms.

Performing Equitable Decompositions using Separable Automorphisms

For a graph G with automorphism compatible matrix M and separable automorphism ϕ of order ℓ with prime factorization $\ell = p_0 p_1 \cdots p_h$, set $M(0) = M$, $\ell_0 = \ell$, and $\phi_0 = \phi$. We perform $h + 1$ sequential decompositions of M, one for each prime in our factorization. Thus we will run through Steps a-c $h + 1$ times to fully decompose the matrix. To begin we start with $i = 0$, and move to *Step a.*

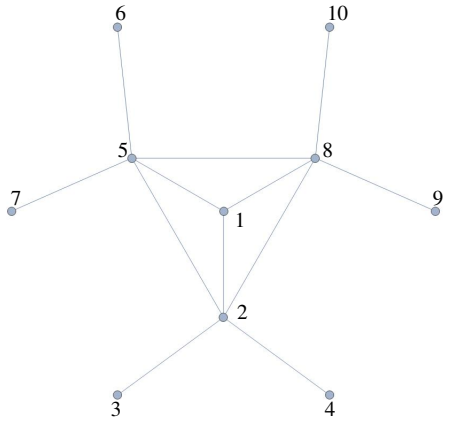
Step a: *Let $\ell_{i+1} = \ell_i / p_i$. Form the basic automorphism $\psi_i = \phi_i^{\ell_{i+1}}$ of order p_i.*
Step b: *Perform an equitable decomposition of $M(i)$ over ψ_i as in Theorem 3.4 by choosing a semi-transversal $\mathcal{T}_0$ of the p_i-orbits of ψ_i according the the method set out in Proposition 3.6 and setting $\mathcal{U}$ to be the set of all vertices fixed by ψ_i. Let $\widetilde{M}(i)$ be the matrix obtained from $M(i)$ by permuting the rows and columns to agree with the new vertex order: $\mathcal{U}, \mathcal{T}_0, \mathcal{T}_1, \ldots, \mathcal{T}_{p_i - 1}$. Then define*

$$M(i+1) = S\,\widetilde{M}(i)S^{-1} = \widetilde{M}(i)_{\psi_i} \oplus B(i)_1 \oplus B(i)_2 \oplus \cdots \oplus B(i)_{p_i - 1}$$

Step c: *Define $\phi_{i+1} = \tilde{\phi}_i = (\phi_i)^{p_i}$ as described in the proof of Proposition 3.6. If $i < h$, then set $i = i + 1$ and return to Step a. Otherwise, the decomposition is complete.*

Remark. Each occurrence of *Step b* requires choosing a semi-transversal $\mathcal{T}_0$ and setting up a new fixed set $\mathcal{U}$ (determined by ψ_i). By slight abuse of notation we will simply reuse the same notation for each 'round', forgetting the previously used semi-transversals and fixed vertex sets.

The procedure described in Steps *a–c* allows one to sequentially decompose a matrix M over any of its separable automorphisms. By extension we refer to the resulting matrix as an *equitable*



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

Figure 3.1: The graph G considered in Example 3.8 with automorphism $\phi = (2,5,8)(3,6,9,4,7,10)$ and adjacency matrix $A = A(G)$.

decomposition of M over ϕ. The following example illustrates an equitable decomposition over a separable automorphism that is not basic.

Example 3.8. Consider the graph G in Figure 3.1 whose adjacency matrix $A = A(G)$ is also shown, which has the separable automorphism

$$\phi = (2,5,8)(3,6,9,4,7,10). \tag{3.5}$$

The automorphism ϕ has order $\ell = 6 = 3 \cdot 2$. Since ℓ factors into two primes we will proceed through *Steps a-c* two times to equitably decompose the adjacency matrix A with respect to ϕ.

Round 1: Let $A(0) = A$, $\phi_0 = \phi$, $\ell_0 = 6$, and $p_0 = 3$.

Step a: Note that $\ell_1 = 2$ and $\psi_0 = \phi_0^2 = (2,8,5)(3,9,7)(4,10,6)$, which is a basic automorphism of order $p_0 = 3$.

Step b: To choose a semi-transversal $\mathcal{T}_0$ of ψ_0, we select vertices 2 and 3 from the orbits of ϕ_0, and add $\phi_0^3(3) = 4$ to $\mathcal{T}_0$ as well (following the method for choosing semi-transversals set out in Proposition 3.6. Then $\mathcal{T}_1 = \{8,9,10\}$ and $\mathcal{T}_2 = \{5,7,6\}$, with $\mathcal{U} = \{1\}$. We let $\tilde{A}$ be the matrix obtained from A by permuting the rows and columns to agree with the vertex order $\mathcal{U}, \mathcal{T}_0, \mathcal{T}_1, \mathcal{T}_2 = 1,2,3,4,8,9,10,5,7,6$ (in this case $\tilde{A} = A$). Using Theorem 3.4, we have

$$\tilde{A}(0)_0 = \tilde{A}\,[\mathcal{T}_0, \mathcal{T}_0] = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix},\ \tilde{A}(0)_1 = \tilde{A}\,[\mathcal{T}_0, \mathcal{T}_1] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} = \tilde{A}(0)_2,$$

$$\text{and } F(0) = \tilde{A}(0)\left[\mathcal{T}_f, \mathcal{T}_f\right] = \begin{bmatrix} 0 \end{bmatrix},\ H(0) = L(0)^T = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}$$

from which the matrices

$$\tilde{A}(0)_{\psi_0} = \begin{bmatrix} F(0) & p_0 H(0) \\ L(0) & B(0)_0 \end{bmatrix} = \begin{bmatrix} 0 & 3 & 0 & 0 \\ 1 & 2 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix},\ B(0)_1 = B(0)_2 = \begin{bmatrix} -1 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \tag{3.6}$$

$$A(1) = \begin{bmatrix} \tilde{A}(0)_{\psi_0} & 0 & 0 \\ 0 & B(0)_1 & 0 \\ 0 & 0 & B(0)_2 \end{bmatrix}$$

can be constructed.

Step c: Next, we derive $\phi_1 = \tilde{\phi}_0 = (\phi_0)^{p_0} = (3,4)(6,7)(9,10)$. And notice that

$$\phi_1|_{\mathcal{T}_0} = (3,4), \quad \phi_1|_{\mathcal{T}_1} = (9,10), \quad \text{and} \quad \phi_1|_{\mathcal{T}_2} = (6,7).$$

where ϕ_1 is the automorphism associated with $A(1)$ guaranteed by Proposition 3.6. Since ℓ factors into two primes we proceed to Round 2.

Round 2: $A(1)$ and ϕ_1 have been computed, $\ell_1 = 2$, and $p_1 = 2$.

Step a: Since $\ell_2 = 1$ then $\psi_1 = \phi_1 = (3,4)(6,7)(9,10)$, which is a basic automorphism of order $p_1 = 2$.

Step b: We choose the semi-transversal $\mathcal{T}_0 = \{3, 6, 9\}$ which causes $\mathcal{T}_1 = \{4, 7, 10\}$. Note that the set of fixed points is $\mathcal{U} = \{1, 2, 5, 8\}$. Now, we create the matrix $\tilde{A}(1)$ from $A(1)$ by reordering the rows and columns to agree with the order $\mathcal{U}, \mathcal{T}_0, \mathcal{T}_1 = 1, 2, 5, 8, 3, 6, 9, 4, 7, 10$. By decomposing the matrix $\tilde{A}(1)$ as is Theorem 3.4 we have

$$\tilde{A}(1)_0 = \tilde{A}(0)\,[\mathcal{T}_0, \mathcal{T}_0] = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix},\ \tilde{A}(1)_1 = \tilde{A}(0)\,[\mathcal{T}_0, \mathcal{T}_1] = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

$$F(1) = \tilde{A}(0)\left[\mathcal{T}_f, \mathcal{T}_f\right] = \begin{bmatrix} 0 & 3 & 0 & 0 \\ 1 & 2 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}, \text{and } H(1) = L(1)^T = \tilde{A}(0)[\mathcal{T}_0, \mathcal{T}_f] = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

from which we can construct the matrices

$$B(1)_0 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix},\ B(1)_1 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \text{ and } A(2) = \begin{bmatrix} F(1) & p_1 H(1) & 0 \\ L(1) & B(1)_0 & 0 \\ 0 & 0 & B(1)_1 \end{bmatrix}.$$

Step c: The matrix $A(2)$ is now decomposed into blocks, and in this final step there is no need to find ϕ_2 since the decomposition is complete.

Thus, our final decomposition is the matrix $A(2)$. To see the block diagonal form of $A(2)$ we permute the rows and columns with a permutation matrix P to put the associated vertices back in the original order. The result is the matrix $PA(2)P^{-1}$ which equals

$$P\left[\begin{array}{cccc|ccc|ccc} 0 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 2 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 2 & 0 & 0 & 0 \\ \hline 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{array}\right] P^{-1} = \left[\begin{array}{ccc|c|cc|c|cc|c} 0 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 2 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & -1 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{array}\right]$$

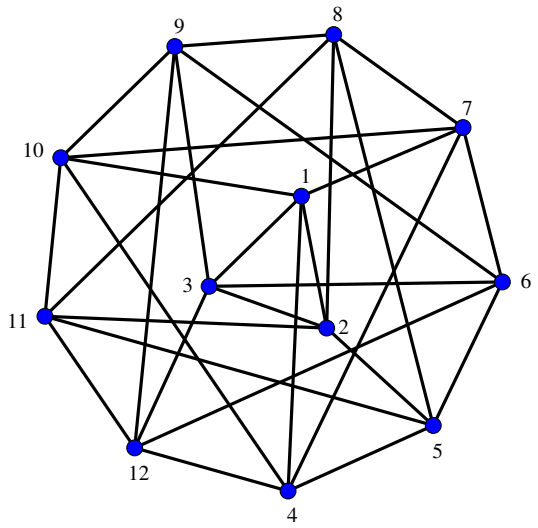
We can see in the final decomposition, that the (twice) decomposed divisor matrix is found in the first block $(A_{\psi_0})_{\psi_1} = \begin{bmatrix} 0 & 3 & 0 \\ 1 & 2 & 2 \\ 0 & 1 & 0 \end{bmatrix}$, which is the divisor matrix A_ϕ associated with the equitable partition of A induced by ϕ.

3.4 Equitable Decompositions over Prime-Power Automorphisms

One might naively believe that we could also use the method described in the previous section to completely equitably decompose a matrix with an automorphism whose order is p^N, by repeating the process as was outlined but using the same prime number on subsequent steps. The following is an example showing this method does not work in general and demonstrating a need for a more sophisticated method to decompose matrices over automorphisms of order p^N.

Example 3.9. Consider the following matrix and its corresponding adjacency graph in Figure 3.2.

We attempt to follow the recursive method of equitable decompositions for separable automorphisms found in the previous section by first forming a new automorphism



$$\left[\begin{array}{ccc||ccc|ccc|ccc}
0&1&1&1&0&0&1&0&0&1&0&0\\
1&0&1&0&1&0&0&1&0&0&1&0\\
1&1&0&0&0&1&0&0&1&0&0&1\\
\hline\hline
1&0&0&0&1&0&1&0&0&1&0&1\\
0&1&0&1&0&1&0&1&0&0&1&0\\
0&0&1&0&1&0&1&0&1&0&0&1\\
\hline
1&0&0&1&0&1&0&1&0&1&0&0\\
0&1&0&0&1&0&1&0&1&0&1&0\\
0&0&1&0&0&1&0&1&0&1&0&1\\
\hline
1&0&0&1&0&0&1&0&1&0&1&0\\
0&1&0&0&1&0&0&1&0&1&0&1\\
0&0&1&1&0&1&0&0&1&0&1&0
\end{array}\right]$$

Figure 3.2: The graph G on 12 vertices and its adjacency matrix with automorphism $\phi = (1,2,3)(4,5,6,7,8,9,10,11,12)$.

$\psi = \phi^3 = (4,7,10)(5,8,11)(6,9,12)$. The first decomposition gives a direct sum of smaller matrices with the associated digraphs found in Figure 3.3.

$$\left[\begin{array}{ccc|ccc}
0&1&1&3&0&0\\
1&0&1&0&3&0\\
1&1&0&0&0&3\\
\hline
1&0&0&2&1&1\\
0&1&0&1&2&1\\
0&0&1&1&1&2
\end{array}\right] \oplus \begin{bmatrix} \omega+\omega^2 & 1 & \omega^2 \\ 1 & \omega+\omega^2 & 1 \\ \omega & 1 & \omega+\omega^2 \end{bmatrix} \oplus \begin{bmatrix} \omega+\omega^2 & 1 & \omega \\ 1 & \omega+\omega^2 & 1 \\ \omega^2 & 1 & \omega+\omega^2 \end{bmatrix}$$

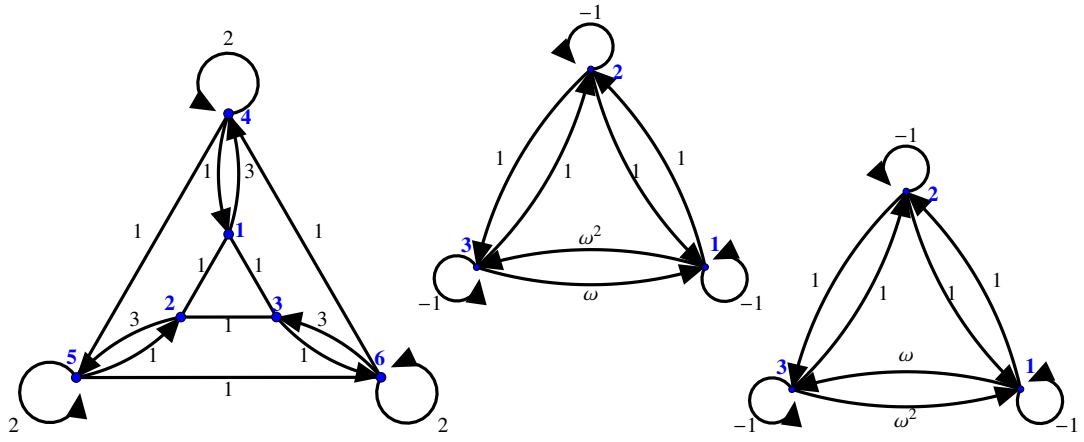


Figure 3.3: The decomposition of the graph G on 12 vertices using the basic automorphism $\psi = (4,7,10),(5,8,11),(6,9,12)$.

While there is an automorphism of order three which acts on the first "piece" of the decomposed matrix, there is no automorphism permuting all of the vertices of G as ϕ does. Thus, if we continue the recursive process in this example, we fail to account for part of the symmetry found in ϕ. This decomposition was done using the transversal $\mathcal{T}_0 = \{4, 8, 12\}$. One may wonder if a different choice of semi-transversal could give better results. However, simple computations demonstrate that any choice of transversal yields a similarly unsatisfying conclusion. Thus we conclude that the methods contained in Section 3.3 cannot be used to *completely* decompose examples like this.

In this section we give a step-by-step method for decomposing a graph over any of its automorphisms ϕ of order p^N for some prime p and $N \geq 1$, which we refer to as a *prime-power automorphism*. (We note that if $N = 1$ then ϕ is a basic automorphism.) This result will allow us in the following section to describe the general case of an equitable decomposition of a graph over any of its automorphisms.

To show how a graph can be equitably decomposed over a prime-power automorphism we require the following lemma. The lemma describes a special case in which it is possible to completely equitably decompose a matrix given it has a particular form. Later in Proposition 3.11 we will show that, in fact, the situation prescribed in this lemma is the general case. That is any graph with a prime powered automorphism can be labeled in such a way that its corresponding adjacency matrix has the special form prescribed in Lemma 3.10 . By combining these results we will be able to give an algorithm in this section for equitably decomposing a graph over any prime-powered automorphism.

Lemma 3.10. *For a prime p and $N \geq 2$ let M be the $(f + rp^N) \times (f + rp^N)$ block matrix*

$$M = \begin{bmatrix} F & H & H & H & \cdots & H \\ L & & & & & \\ L & & & & & \\ L & & & C & & \\ \vdots & & & & & \\ L & & & & & \end{bmatrix}, \tag{3.7}$$

where F is an $f \times f$ matrix, H is an $f \times rp^{N-1}$ matrix, L is an $rp^{N-1} \times f$ matrix, and C is an $rp^N \times rp^N$ matrix. Suppose that the matrix C can be partitioned in two ways:

$$C = \begin{bmatrix} C_0 & C_1 & C_2 & \cdots & C_{p^N-1} \\ C_{p^N-1} & C_0 & C_1 & \cdots & C_{p^N-2} \\ C_{p^N-2} & C_{p^N-1} & C_0 & \cdots & C_{p^N-3} \\ \vdots & \vdots & \vdots & & \vdots \\ C_1 & C_2 & C_3 & \cdots & C_0 \end{bmatrix} = \begin{bmatrix} D_0 & D_1 & D_2 & \cdots & D_{p-1} \\ D_{p-1} & D_0 & D_1 & \cdots & D_{p-2} \\ D_{p-2} & D_{p-1} & D_0 & \cdots & D_{p-3} \\ \vdots & \vdots & \vdots & & \vdots \\ D_1 & D_2 & D_3 & \cdots & D_0 \end{bmatrix} \tag{3.8}$$

where each D_i block is of size $rp^{N-1} \times rp^{N-1}$, and each C_j of is size $r \times r$. Then there exists an $(f + rp^N) \times (f + rp^N)$ invertible matrix T such that

$$T^{-1}MT = \tilde{M} \oplus B_1 \oplus B_2 \oplus \cdots \oplus B_{p^N - p^{N-1}}, \tag{3.9}$$

with

$$\tilde{M} = \begin{bmatrix} F & pH \\ L & B_0 \end{bmatrix}, \; B_0 = \sum_{m=0}^{p-1} D_m, \text{ and } B_j = \sum_{m=0}^{p^N-1} \omega^{\gamma_j m} C_m \text{ for } j = 1, 2, \ldots, p^N - p^{N-1}, \tag{3.10}$$

where γ_j are elements of $\{1, 2, \ldots, p^N - 1\}$ which are not multiples of p. Consequently,

$$\sigma(M) = \sigma\left(\tilde{M}\right) \cup \sigma(B_1) \cup \sigma(B_2) \cup \cdots \cup \sigma(B_{p^N - p^{N-1}}).$$

We refer to any matrix which has the form of the first equality in Equation (3.8) *block-circulant*. A matrix which has the form of C which is block-circulant for two different sized block partitions is called *double block-circulant*.

Proof. Before we begin let us establish some useful identities involving roots of unity. Let $\omega = e^{2\pi i/p^N}$, and γ and γ' be integers that are both relatively prime to p with $0 < \gamma < \gamma' < p^N$. Thus ω^γ is a primitive p^N-root of unity. Using the fact that $\sum_{k=0}^{p^N-1}(\omega^\gamma)^{km} = 0$ if p^N does not divide m it is trivial to prove the following identities for any integer a

$$\sum_{m=0}^{p-1} \omega^{\gamma(mp^{N-1}+a)} = 0, \text{ and } \sum_{m=0}^{p^N-1} \omega^{m(\gamma-\gamma')} = 0. \tag{3.11}$$

Let S be the $rp^N \times rp^{N-1}(p-1)$ block matrix written below in $r \times r$ blocks:

$$S = \begin{bmatrix} I_r & I_r & \dots & I_r \\ \omega^{\gamma_1} I_r & \omega^{\gamma_2} I_r & \dots & \omega^{\gamma_\rho} I_r \\ \omega^{2\gamma_1} I_r & \omega^{2\gamma_2} I_r & \dots & \omega^{2\gamma_\rho} I_r \\ \vdots & \vdots & & \vdots \\ \omega^{(p^N-1)\gamma_1} I_r & \omega^{(p^N-1)\gamma_2} I_r & \dots & \omega^{(p^N-1)\gamma_\rho} I_r \end{bmatrix}, \tag{3.12}$$

where $\omega = e^{2\pi i/p^N}$, and $\omega^{\gamma_1}, \omega^{\gamma_2}, \dots, \omega^{\gamma_\rho}$ are the $p^N - p^{N-1}$ generators of the cyclic group of the p^N-roots of unity. We let T be the block matrix

$$T = \left[\begin{array}{c|c|c} I_f & 0 & 0 \\ \hline & I_{rp^{N-1}} & \\ 0 & I_{rp^{N-1}} & S \\ & \vdots & \\ & I_{rp^{N-1}} & \end{array}\right], \tag{3.13}$$

let $R = \begin{bmatrix} I_{rp^{N-1}} & I_{rp^{N-1}} & \cdots & I_{rp^{N-1}} \end{bmatrix}$, and consider the product

$$\begin{bmatrix} I_f & 0 \\ 0 & \frac{1}{p}R \\ 0 & \frac{1}{p^N}S^* \end{bmatrix} \begin{bmatrix} I_f & 0 & 0 \\ 0 & R^T & S \end{bmatrix} = \begin{bmatrix} I_f & 0 & 0 \\ 0 & \frac{1}{p}RR^T & \frac{1}{p}RS \\ 0 & \frac{1}{p^N}S^*R^T & \frac{1}{p^N}S^*S \end{bmatrix},$$

where S^* is the conjugate transpose of S. The matrix RS is a $p^{N-1} \times p^N$ r-block matrix where the $(a,b)^{th}$ block is given by

$$I_r \sum_{m=0}^{p-1} \omega^{\gamma_b(mp^{N-1}+a-1)} = 0,$$

using Equation (3.11). Therefore $RS = 0$ and similarly, $S^*R^T = 0$.

Next we consider S^*S as a block matrix with $r \times r$ blocks where the $(a,b)^{th}$ block has the form

$$I_r \sum_{m=0}^{p^N-1} \omega^{m(\gamma_b-\gamma_a)} = \left\{ \begin{array}{ll} p^N I_r & \text{if } a = b \\ 0_r & \text{if } a \neq b. \end{array} \right\}$$

So, $S^*S = p^N I$, and therefore

$$T^{-1} = \begin{bmatrix} I_f & 0 \\ 0 & \frac{1}{p}R \\ 0 & \frac{1}{p^N}S^* \end{bmatrix}.$$

Next we show that performing a similarity transformation using T gives the equitable decomposition in Equation (3.9). Let M be the matrix given in Equation (3.7). Then with $P = [H \quad H \quad \cdots \quad H]$ and $Q = [L^T \quad L^T \quad \cdots \quad L^T]^T$, we have

$$T^{-1}MT = \begin{bmatrix} I_f & 0 \\ 0 & \frac{1}{p}R \\ 0 & \frac{1}{p^N}S^* \end{bmatrix} \begin{bmatrix} F & P \\ Q & C \end{bmatrix} \begin{bmatrix} I_f & 0 & 0 \\ 0 & R^T & S \end{bmatrix} = \begin{bmatrix} F & PR^T & PS \\ \frac{1}{p}RQ & \frac{1}{p}RCR^T & \frac{1}{p}RCS \\ \frac{1}{p^N}S^*Q & \frac{1}{p^N}S^*CR^T & \frac{1}{p^N}S^*CS \end{bmatrix}.$$

It is straightforward to verify that

$$PR^T = pH, \quad RQ = pL, \quad \text{and} \quad RCR^T = p\sum_{m=0}^{p-1} D_m = pB_0.$$

To show that $PS = 0$, we break P into $f \times r$ blocks H_i and observe that $H_i = H_{i+p^{N-1}}$ for $0 \leq i \leq p^N - p^{N-1} - 1$. The kth $f \times r$ block in the product PS is given by

$$\sum_{i=0}^{p^{N-1}-1} H_i \sum_{m=0}^{p-1} \omega^{\gamma_k(mp^{N-1}+i)} = 0,$$

as in Equation (3.11). A similar calculation shows that $S^*Q = 0$. Next we consider the matrix product RCS. The $(a,b)^{th}$ block has the form

$$\sum_{m=0}^{p^N-1} C_m \omega^{(m+a-1)\gamma_b} \sum_{j=0}^{p-1} \omega^{\gamma_b(jp^{N-1})} = 0$$

using Equation (3.11) (with $a = 0$). Thus $RCS = 0$ and similarly $S^*CR^T = 0$.

We then calculate S^*CS. The $(a,b)th$ $r \times r$ block of S^*CS is given by

$$\sum_{n=0}^{p^N-1}\sum_{m=0}^{p^N-1} C_m \omega^{(m+n)\gamma_a}\omega^{-n\gamma_b} = \begin{cases} p^N \sum_{m=0}^{p^N-1} C_m\omega^{m\gamma_a} = p^N B_a & \text{if } a = b \\ \sum_{m=0}^{p^N-1} C_m\omega^{m\gamma_a} \sum_{n=0}^{p^N-1} \omega^{n(\gamma_a-\gamma_b)} = 0 & \text{if } a \neq b, \end{cases}$$

using Equation (3.11) for entries where $a \neq b$. Thus,

$$S^*CS = B_1 \oplus B_2 \oplus \ldots \oplus B_{p^N-p^{N-1}}.$$

Finally, we have

$$T^{-1}MT = \begin{bmatrix} F & pH \\ L & B_0 \end{bmatrix} \oplus B_1 \oplus B_2 \oplus \ldots \oplus B_{p^N-p^{N-1}}$$

and the result follows.

□

Our goal is to show that if a graph has a prime-power automorphism ϕ then any automorphism compatible matrix M of the graph has the form given in Equation (3.7) if we choose the transversal of the automorphism correctly.

Proposition 3.11. *Let G be a graph with automorphism ϕ of order p^N for some prime p and integer $N > 0$. Let $\mathcal{T}_0$ be a transversal of the orbits of length p^N of ϕ, and let $\tilde{\mathcal{T}}_0$ be a transversal of the orbits of ϕ^p when restricted to only vertices contained in orbits of maximal size given by $\tilde{\mathcal{T}}_0 = \bigcup_{m=0}^{p^{N-1}-1} \mathcal{T}_m$. Let M be an automorphism compatible matrix on G and set*

$$\mathcal{T}_F = \{v \in V(G) \mid |\mathscr{O}_\phi(v)| < p^N\},\ f = |\mathcal{T}_F|,\ F = M[\mathcal{T}_F, \mathcal{T}_F],$$
$$H = M[\mathcal{T}_F, \tilde{\mathcal{T}}_0],\ L = [\tilde{\mathcal{T}}_0, \mathcal{T}_F],\ C_m = M[\mathcal{T}_0, \mathcal{T}_m],\ \textit{and}\ D_s = M[\tilde{\mathcal{T}}_0, \tilde{\mathcal{T}}_s].$$

Then there is a permutation similarity transformation of M which is the matrix in Equation (3.7) *satisfying the conditions of Lemma 3.10.*

Proof. Let r be the number of orbits of length p^N, thus $|\mathcal{T}_k| = r$, and $|\tilde{\mathcal{T}}_k| = rp^{N-1}$. Permute the rows and columns of M so that they are labeled in the order $\mathcal{T}_F, \mathcal{T}_0, \mathcal{T}_1, ..., \mathcal{T}_{p^N-1}$ and let C be the principal submatrix consisting of the last rp^N rows and columns of M. Also let

$$\mathcal{T}_C = \mathcal{T}_0 \cup \mathcal{T}_1 \cup \ldots \mathcal{T}_{p^N-1}. \tag{3.14}$$

Notice that $\phi|_{\mathcal{T}_C} : \mathcal{T}_C \to \mathcal{T}_C$. Hence, C is compatible with $\phi|_{\mathcal{T}_C}$, since

$$M[\mathcal{T}_s, \mathcal{T}_t] = M[\phi(\mathcal{T}_s), \phi(\mathcal{T}_t)] = \begin{cases} M[\mathcal{T}_{s+1}, \mathcal{T}_{t+1}] & \text{if } s,\ t \neq p^N - 1 \\ M[\mathcal{T}_0, \mathcal{T}_{t+1}] & \text{if } s = p^{N-1}, t \neq p^N - 1 \\ M[\mathcal{T}_{s+1}, \mathcal{T}_0] & \text{if } s \neq p^{N-1},\ t = p^N - 1 \\ M[\mathcal{T}_0, \mathcal{T}_0] & \text{if } s,\ t = p^N - 1 \end{cases}$$

Thus C is a block-circulant matrix made up of $r \times r$ blocks.

Since C is automorphism compatible with $\phi|_{\mathcal{T}_C}$, C must also be automorphism compatible with $\phi^{p^{N-1}}|_{\mathcal{T}_C}$. Thus

$$M[\tilde{\mathcal{T}}_s, \tilde{\mathcal{T}}_t] = M[\phi^{p^{N-1}}(\tilde{\mathcal{T}}_s), \phi^{p^{N-1}}(\tilde{\mathcal{T}}_t)] = \begin{cases} M[\tilde{\mathcal{T}}_{s+1}, \tilde{\mathcal{T}}_{t+1}] & \text{if } s,\ t \neq p-1 \\ M[\tilde{\mathcal{T}}_0, \tilde{\mathcal{T}}_{t+1}] & \text{if } s = p-1,\ t \neq p-1 \\ M[\tilde{\mathcal{T}}_{s+1}, \tilde{\mathcal{T}}_0] & \text{if } s \neq p-1, t = p-1 \\ M[\tilde{\mathcal{T}}_0, \tilde{\mathcal{T}}_0] & \text{if } s,\ t = p-1 \end{cases}$$

implying C is also block circulant with $rp^{N-1} \times rp^{N-1}$ blocks.

Notice that $\phi^{p^{N-1}}$ fixes $\mathcal{T}_f$, so that

$$H = M[\mathcal{T}_F, \tilde{\mathcal{T}}_0] = M[\phi^{mp^{N-1}}(\mathcal{T}_F), \phi^{mp^{N-1}}(\tilde{\mathcal{T}}_0)] = M[\mathcal{T}_F, \tilde{\mathcal{T}}_m],\ \textit{and}$$
$$L = M[\tilde{\mathcal{T}}_0, \mathcal{T}_F] = M[\phi^{mp^{N-1}}(\tilde{\mathcal{T}}_0), \phi^{mp^{N-1}}(\mathcal{T}_F)] = M[\tilde{\mathcal{T}}_m, \mathcal{T}_F].$$

Thus, M has the form as required in Lemma 3.10. □

Given a graph G with a prime-powered automorphism ϕ our goal is to equitably decompose this graph, or equivalently the associated automorphism compatible matrix M, by sequentially decomposing M into smaller and smaller matrices. The way we do this is to first use Lemma 3.10 and Proposition 3.11 to decompose M into the product

$$\tilde{M} \oplus B_1 \oplus B_2 \oplus \cdots \oplus B_{p^N - p^{N-1}}$$

as in Lemma 3.10. By virtue of the way in which this decomposition is carried out the smaller matrix $\tilde{M}$ also has a "smaller" automorphism ψ that can similarly be used to decompose the matrix $\tilde{M}$.

Proposition 3.12. *Assume the graph G, the matrix M, and the automorphism ϕ satisfy the conditions in Proposition 3.11. Then there exists an automorphism $\psi \in Aut(\tilde{M})$ of order p^{N-1} where*

$$\widetilde{M} = \begin{bmatrix} F & pH \\ L & B_0 \end{bmatrix}$$

is the matrix described in Lemma 3.10.

Proof. Let M be the automorphism compatible matrix with the correct ordering: $\mathcal{T}_F, \mathcal{T}_0, \mathcal{T}_1, \ldots, \mathcal{T}_{p^{N-1}}$. Recall that under this vertex ordering, each orbit of maximal length of ϕ looks like

$$\mathscr{O}_\phi(i) = (i, i + r, i + 2r, \ldots, i + rp^N).$$

We define a map ψ on $\mathcal{T}_F \cup \tilde{\mathcal{T}}_0$ (where $\tilde{\mathcal{T}}_0$ is defined in Proposition 3.11) by

$$\psi(i) = \begin{cases} \phi(i) & i \notin \mathcal{T}_{p^{N-1}-1} \\ \phi^{1-p^{N-1}}(i) & \textit{otherwise.} \end{cases}$$

It is straightforward to verify that $\phi(\mathcal{T}_F) = \mathcal{T}_F$, $\phi^k(\mathcal{T}_m) = \mathcal{T}_{k+m}$, so that $\psi : \mathcal{T}_F \cup \tilde{\mathcal{T}}_0 \to \mathcal{T}_F \cup \tilde{\mathcal{T}}_0$.

We wish to show for all $i, j \in \mathcal{T}_F \cup \tilde{\mathcal{T}}_0$ that $\widetilde{M}(\psi(i), \psi(j)) = \widetilde{M}(i, j)$. To do this we first consider the case where i or j is in $\mathcal{T}_F$. For $\epsilon_1, \epsilon_2 \in \{0, 1\}$, $\widetilde{M}(\psi(i), \psi(j))$ is equal to the quantities below in the cases indicated.

$$\begin{aligned}
&M(\phi(i), \phi(j)) = M(i, j) = \widetilde{M}(i, j) && i \in \mathcal{T}_F \; j \in \mathcal{T}_F \\
&\widetilde{M}(\phi(i), \phi^{1-\epsilon_1 p^{N-1}}(j)) = \widetilde{M}(\phi(i), \phi(j)) = pM(\phi(i), \phi(j)) = pM(i, j) = \widetilde{M}(i, j) && i \in \mathcal{T}_F \; j \notin \mathcal{T}_F \\
&\widetilde{M}(\phi^{1-\epsilon_2 p^{N-1}}(i), \phi(j)) = \widetilde{M}(\phi(i), \phi(j)) = M(\phi(i), \phi(j)) = M(i, j) = \widetilde{M}(i, j) && i \notin \mathcal{T}_F \; j \in \mathcal{T}_F
\end{aligned}$$

The first equalities in the second and third cases are valid because of the block circulant nature of M, as described in Proposition 3.11 and Lemma 3.10.

Finally, we consider the case where neither i nor j is in $\mathcal{T}_F$. Then for $\epsilon_1, \epsilon_2 \in \{0, 1\}$,

$$\widetilde{M}(\psi(i), \psi(j)) = \widetilde{M}(\phi^{1-\epsilon_1 p^{N-1}}(i), \phi^{1-\epsilon_2 p^{N-1}}(j)) = \sum_{m=0}^{p-1} M(\phi^{1-\epsilon_1 p^{N-1}}(i), \phi^{1+(m-\epsilon_2)p^{N-1}}(j))$$
$$= \sum_{m=0}^{p-1} M(i, \phi^{(m+\epsilon_1-\epsilon_2)p^{N-1}}(j)) = \sum_{m=0}^{p-1} M\left(i, \left(\phi^{p^{N-1}}\right)^m(j)\right) = \widetilde{M}(i, j),$$

where the second to last equality is true because the sum passes through all p distinct powers of ϕ^{N-1} exactly once, and the addition of ϵ_1 and ϵ_2 only changes the order in which this happens. □

Each time we use Propositions 3.11 and 3.12 on a matrix M with automorphism ϕ of order $|\phi| = p^N$ we obtain a smaller matrix $\tilde{M}$ with automorphism ψ of order $|\psi| = p^{N-1}$. It is in fact possible to sequentially repeat this process until we "run out" of powers of p. The result is the equitable decomposition of the graph G over ϕ.

Theorem 3.13. ***(Equitable Decompositions over Prime-Powered Automorphisms)*** *Suppose G is a graph with automorphism ϕ where $|\phi| = p^N$ for some prime p and $N \geq 1$. If M is an automorphism compatible matrix of G then by repeated application of Propositions 3.11 and 3.12, we obtain the equitable decomposition*

$$M_\phi \oplus \hat{M}_1 \oplus \hat{M}_2 \oplus \cdots \oplus \hat{M}_N,$$

where M_ϕ is the divisor matrix associated with ϕ and $\hat{M}_i = \left({}_iB_1 \oplus {}_iB_2 \oplus \cdots \oplus {}_iB_{p^{N-i+1}-p^{N-i}}\right)$ where ${}_iB_j$ has size $r_i \times r_i$, where r_i is the number of orbits of ϕ with length greater than or equal to p^{N-i+1} and for $1 \leq j \leq p^{N-i+1} - p^{N-i}$

Proof. By Proposition 3.11 and Lemma 3.10 we can decompose M into $M_1 = \widetilde{M} \oplus \hat{M}$, where $\widetilde{M}_1 = \widetilde{M}$ and $\hat{M}_1 =_1 B_1 \oplus_1 B_2 \oplus \cdots \oplus_1 B_{p^N-p^{N-1}}$. Also let ψ be the automorphism of $\widetilde{M}_1$ as in Proposition 3.12.

Next we will use ψ to decompose $\widetilde{M}_1$. We pick a transversal $\mathcal{T}_0$ of the orbits *of maximal length* of ψ, and $\mathcal{T}_F$ which will contain all the indices belonging to orbits of length less than p^{N-1}. If we perform a permutation similarity transformation on M_1 so that our indices now appear in

the order $\mathcal{T}_f, \mathcal{T}_0, \ldots, \mathcal{T}_{p^{N-1}}$, we can use Proposition 3.11, and Lemma 3.10 to complete another decomposition. We repeat this process N times.

We need to show that the block matrix $\widetilde{M}_N$ appearing in the upper left portion of the final decomposition satisfies $\widetilde{M}_N = M_\phi$, where M_ϕ is the divisor matrix obtained from an equitable partition of the original matrix M using the orbits of ϕ as the partition set. To do so, recall that

$$M_\phi(i, j) = \sum_{r \in \mathcal{O}_\phi(j)} M_{i,r}.$$

It is easy to verify that each of the indices of the rows and columns of $\widetilde{M}_N$ correspond to a distinct orbit of length p^k.

Suppose that j is an index appearing in $\widetilde{M}_N$ and in an orbit of length p^k for some $k > 0$. Then in the first $N - k$ decompositions, j will be placed in the set $\mathcal{T}_F$. Thus, if $\widetilde{M}_{\kappa-1}$ is the matrix created at the end of the $(\kappa - 1)$th decomposition and $\tilde{M}$ is the matrix created at the end of the κth decomposition, and $1 \leq \kappa \leq N - k$, then $\widetilde{M}_\kappa(i, j) = \widetilde{M}_{\kappa-1}(i, j) = M(i, j)$.

We consider the $(N - k + 1)$th decomposition, which begins with an automorphism of order p^k. Examining the formulas in Equation (3.9) gives

$$\widetilde{M}_{N-k+1}(i, j) = \sum_{m=0}^{p-1} M(i, \phi^{mp^{k-1}}(j)).$$

The next decomposition will yield

$$\widetilde{M}_{N-k+2}(i, j) = \sum_{m_2=0}^{p-1} \widetilde{M}_{N-k+1}(i, \phi^{m_2 p^{k-2}}(j)) = \sum_{m_2=0}^{p-1} \sum_{m_1=0}^{p-1} M(i, \phi^{m_1 p^{k-1} + m_2 p^{k-2}}(j)).$$

Thus, the entries in $\widetilde{M}_N$ are determined by

$$\widetilde{M}_N = \sum_{m_1, m_2, \ldots, m_k = 0}^{p-1} M(i, \phi^{m_1 p^{k-1} + m_2 p^{k-2} + \ldots + m_k}(j)) = \sum_{r \in \mathcal{O}_\phi(j)} M_{i,r}.$$

□

We note here that in the case that $N = 1$, this process is exactly the same as the algorithm discussed in Section 3.3 for equitably decomposing over a basic automorphisms.

According to Theorem 3.13 it is possible to equitably decompose a graph over any of its prime-power automorphisms. Although this is true it is likely unclear at this point how this type of decomposition can be carried out. What follows is an algorithm detailing the steps involved in this process.

Performing Equitable Decompositions Using Prime-Power Automorphisms

For a graph G with automorphism compatible matrix M and automorphism ϕ with $|\phi| = p^N$, set $M_0 = M$, and $\phi_1 = \phi$. To begin we start with $i = 1$, and start with *Step a.*

Step a: Choose $\mathcal{T}_F$ to be all elements of the graph G which are contained in orbits of ϕ_i with length less than p^{N-i+1}. Choose $\mathcal{T}_0$ to be a transversal of all orbits of ϕ_i with length equal to p^{N-i+1}, and let

$$\tilde{\mathcal{T}}_0 = \{\mathcal{T}_0, \phi_i(\mathcal{T}_0), \phi_i^2(\mathcal{T}_0), \ldots, \phi_i^{p^{n-i}}(\mathcal{T}_0)\}$$

and use this $\mathcal{T}_0$ and $\tilde{\mathcal{T}}_0$ to order your matrix M_i as in Proposition 3.11.

Step b: Form the T matrix as described in Lemma 3.10. Perform the equitable decomposition of M_i via a similarity transformation and define

$$M_{i+1} = TM_iT^{-1}.$$

Step c: Extending the definition of ψ in the method described in the proof of Proposition 3.3, we define ϕ_{i+1} by

$$\phi_{i+1}(k) = \begin{cases} \phi_i(k) & k \in \mathcal{T}_f \cup \mathcal{T}_0 \cup \mathcal{T}_1 \cup \cdots \cup \mathcal{T}_{\ell_{i+1}-1} \\ \phi_i^{1-\ell_{i+1}}(k) & k \in \mathcal{T}_{\ell_{i+1}} \\ k & otherwise. \end{cases} \tag{3.15}$$

If $i < N$, then set $i = i + 1$ and return to step (a), otherwise the decomposition is complete.

To demonstrate how this algorithm is applied we consider the following example. The example we consider is the graph and automorphism shown in Figure 3.2, which we are now able to fully decompose.

Example 3.14. Consider the graph shown in Figure 3.2 previously considered in Example 3.9. Here, $\phi = \phi_1$ is the automorphism $\phi_1 = (1,2,3)(4,5,6,7,8,9,10,11,12)$, which has order $9 = 3^2$. Thus we will run through steps (a)-(c) in our algorithm twice.

Note that the matrix M has the form guaranteed by Proposition 3.11. Following the initial steps to equitably decompose M we set $i = 1$.

Round 1

Step (a): We choose $\mathcal{T}_F = \{1,2,3\}$ and $\mathcal{T}_0 = \{4\}$, which gives us $\tilde{\mathcal{T}}_0 = \{4,5,6\}$. Thus the adjacency matrix M for G given in Figure 3.2 is already ordered appropriately.

Step (b): Following Proposition 3.11, we can perform an equitable decomposition, which gives

$$M_2 = \left[\begin{array}{ccc|ccc} 2 & 1 & 1 & 1 & 0 & 0 \\ 1 & 2 & 1 & 0 & 1 & 0 \\ 1 & 1 & 2 & 0 & 0 & 1 \\ \hline 3 & 0 & 0 & 0 & 1 & 1 \\ 0 & 3 & 0 & 1 & 0 & 1 \\ 0 & 0 & 3 & 1 & 1 & 0 \end{array}\right] \oplus [\lambda_1] \oplus [\lambda_2] \oplus [\lambda_3] \oplus [\lambda_4] \oplus [\lambda_5] \oplus [\lambda_6]$$

where (with $\omega = e^{\frac{2\pi i}{9}}$)

$$\begin{aligned} \lambda_1 &= \omega + \omega^3 + \omega^6 + \omega^8 \approx -0.652, \\ \lambda_2 &= \omega^2 + \omega^6 + \omega^{12} + \omega^{16} \approx -2.879, \\ \lambda_3 &= \omega^4 + \omega^{12} + \omega^{24} + \omega^{32} \approx 0.532, \\ \lambda_4 &= \omega^5 + \omega^{15} + \omega^{30} + \omega^{40} \approx 0.532, \\ \lambda_5 &= \omega^7 + \omega^{21} + \omega^{42} + \omega^{56} \approx -2.879, \\ \lambda_6 &= \omega^8 + \omega^{24} + \omega^{48} + \omega^{64} \approx -0.652, \end{aligned}$$

which are six eigenvalues of the adjacency matrix M. The associated adjacency graph is shown in Figure 3.4.

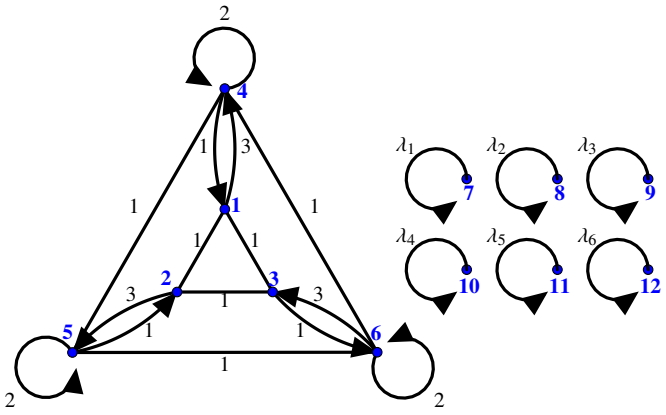


Figure 3.4: The decomposition of the graph G on 12 vertices using the method outlined in Proposition 3.11 after Round 1.

Step (c): Now we form our new automorphism to act on the matrix $\tilde{M}_2$. This is the automorphism $\phi_2 = (1,2,3)(4,5,6)$ constructed using Equation 3.15. Now, we then set $i = 2$ and return to Step (a).

Round 2

Step (a): First we choose $\mathcal{T}_f = \emptyset$ and $\mathcal{T}_0 = \{1,4\}$. In this step we have $\tilde{\mathcal{T}}_0 = \{1,4\}$.

Step (b): We reorder the rows and columns of $\tilde{M}_2$ to correspond to our transversals. The relevant block matrices are

$$C_0 = \begin{bmatrix} 0 & 3 \\ 1 & 2 \end{bmatrix}, C_1 = C_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Performing the equitable decomposition as described in Proposition 3.11 gives

$$M_3 = \begin{bmatrix} 2 & 3 \\ 1 & 4 \end{bmatrix} \oplus \begin{bmatrix} -1 & 3 \\ 1 & 1 \end{bmatrix} \oplus \begin{bmatrix} -1 & 3 \\ 1 & 1 \end{bmatrix} \oplus [\lambda_1] \oplus [\lambda_2] \oplus [\lambda_3] \oplus [\lambda_4] \oplus [\lambda_5] \oplus [\lambda_6] \tag{3.16}$$

with associated adjacency graph in Figure 3.5.

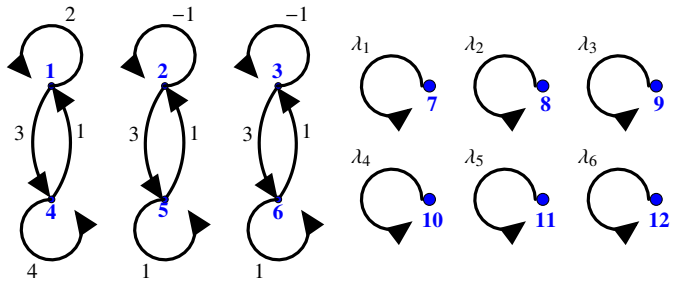


Figure 3.5: The decomposition of the graph G on 12 vertices using the method outlined in Proposition 3.11 after Round 2.

Step (c): Because $i = 2$, the decomposition is complete and there is no need to find ϕ_2.

It is worth noting that in this equitable decomposition we have recovered the divisor matrix of the equitable partition associated with the prime-power automorphism ϕ. This is the matrix $\begin{bmatrix} 2 & 3 \\ 1 & 4 \end{bmatrix}$ seen in Equation (3.16).

3.5 General Equitable Decompositions

Typically the order of an automorphism ϕ of a graph G will not be prime-powered. That is it will have the order $\phi = p^N \ell$ where the prime p does not divide ℓ. In this case Theorem 3.13 nor any previous result guarantees that it is possible to create an equitable decomposition of G with respect to ϕ. In this section we show that this can, in fact, be done.

Remark. When performing an equitable decomposition of a graph using an automorphism ϕ whose order is not prime-powered (say, $|\phi| = p^N \ell$), our strategy will be to create an automorphism $\psi = \phi^\ell$ of order p^N and follow the procedure set out in Theorem 3.13. In order to guarantee that the resulting decomposed matrix still has an automorphism of order ℓ, we must restrict our method of choosing transversals. This is done using the following rules:

- If $a \in V(G)$ was chosen to be in $\mathcal{T}_0$ in a previous round, a must appear in $\mathcal{T}_0$ in the next round as well.

- If $a \in V(G)$ is chosen to be in $\mathcal{T}_0$ in a certain round, and $|\mathcal{O}_\phi(a)| = p^k m$, then $\phi^{p^k}(a), \phi^{2p^k}(a), \ldots, \phi^{(m-1)p^k}(a)$ must also be in $\mathcal{T}_0$.

Proposition 3.15. *Let ϕ be an automorphism of a graph G with automorphism compatible matrix M. Suppose that ϕ has order $p^N\ell$, with p a prime which does not divide ℓ. Then $\psi = \phi^\ell$ is an automorphism of G of order p^N. Moreover, it is possible to construct an automorphism $\tilde{\phi}$ associated with the equitable decomposition of M over ψ of order ℓ such that the divisor matrix $M_\phi = (M_\psi)_{\tilde{\phi}}$.*

Proof. Let M be an automorphism compatible matrix of the graph G and $\phi \in Aut(G)$ of order $|\phi| = p^N\ell$ where p is a prime that does not divide ℓ. Then $\psi = \phi^\ell$ has order p^N, which allows us to use Theorem 3.13 to decompose M with respect to this automorphism.

To carry out this decomposition we follow the procedure outlined in the previous propositions and theorems, choosing our transversals according to the guidelines in Remark 3.5.

Note that since p^k and ℓ are relatively prime, there exist integers α and β such that $1 = \ell\alpha + p^N\beta$. We define our automorphism $\tilde{\phi} : V(G) \to V(G)$ by

$$\tilde{\phi}(a) = \phi^{(1-\ell\alpha)}(a) = \phi^{p^k(p^{N-k})\beta}(a).$$

The second equality above demonstrates that $\tilde{\phi}$ is closed on each $\mathcal{T}_m$.

We now show that $\tilde{\phi}$ is an automorphism of $\tilde{M}_k$ for each $k = 0, \ldots, N$.

By hypothesis, $M = \tilde{M}_0$ is compatible with $\phi^{1-\ell\alpha}$. We then assume that $\widetilde{M}_{k-1}$ is as well. Let $\mathcal{T}_F, \mathcal{T}_0, \ldots, \mathcal{T}_{p^k-1}$ be defined in Round k of the decomposition. Note that since $\tilde{\phi}$ is closed on each transversal, a and $\phi^{1-\ell\alpha}(a)$ must both be in $\mathcal{T}_F$ or neither, and if $a \in \mathcal{T}_m$, then $\phi^{1-\ell\alpha}(a)$ must also be an element of $\mathcal{T}_m$. Thus, $\phi^{1-\ell\alpha}$ is a closed map on the indices of $\widetilde{M}_k$.

We wish to show that for any $a, b \in \mathcal{T}_F \cup \tilde{\mathcal{T}}_0$,

$$\widetilde{M}_k(\phi^{1-\ell\alpha}(a), \phi^{1-\ell\alpha}(b)) = \widetilde{M}_k(a, b). \tag{3.17}$$

Lemma 3.10 and Proposition 3.11 give

$$\widetilde{M}_k(\phi^{1-\ell\alpha}(a),\phi^{1-\ell\alpha}(b)) = \begin{cases} \widetilde{M}_{k-1}(\phi^{1-\ell\alpha}(a),\phi^{1-\ell\alpha}(b)) & \text{if } b \in \mathcal{T}_F \\ p\widetilde{M}_{k-1}(\phi^{1-\ell\alpha}(a),\phi^{1-\ell\alpha}(b)) & \text{if } b \notin \mathcal{T}_F,\ a \in \mathcal{T}_F \\ \sum_{m=0}^{p-1} \widetilde{M}_{k-1}(\phi^{1-\ell\alpha}(a),\phi^{m\ell p^{k-1}}\phi^{1-\ell\alpha}(b)) & \text{if } b \notin \mathcal{T}_F,\ a \notin \mathcal{T}_F \end{cases}$$

$$= \begin{cases} \widetilde{M}_{k-1}(a,b) = \widetilde{M}_k(a,b) & \text{if } b \in \mathcal{T}_F \\ p\widetilde{M}_{k-1}(a,b) = \widetilde{M}_k(a,b) & \text{if } b \notin \mathcal{T}_F,\ a \in \mathcal{T}_F \\ \sum_{m=0}^{p-1} \widetilde{M}_{k-1}(a,\phi^{m\ell p^{k-1}}(b)) = \widetilde{M}_k(a,b) & \text{if } b \notin \mathcal{T}_F,\ a \notin \mathcal{T}_F. \end{cases}$$

All three cases give the desired result in Equation (3.17) because, by assumption, $\widetilde{M}_{k-1}$ is compatible with $\tilde{\phi}$.

We now consider the block matrices ${}_kB_j$ in $\hat{M}_k$ (cf. Theorem 3.13), setting $i = N - k + 1$ in the kth decomposition and ω to be a primitive p^ith root of unity. Note that

$$\begin{aligned} {}_kB_j(\tilde{\phi}(a),\tilde{\phi}(b)) &= \sum_{m=0}^{p^i-1} \omega^{m\gamma_j} \widetilde{M}_{k-1}(\tilde{\phi}(a),\phi^{\ell m}(\tilde{\phi}(b)) = \sum_{m=0}^{p^i-1} \omega^{m\gamma_j} \widetilde{M}_{k-1}(\tilde{\phi}(a),\tilde{\phi}\circ\phi^{\ell m}(b) \\ &= \sum_{m=0}^{p^i-1} \omega^{m\gamma_j} \widetilde{M}_{k-1}(a,\phi^{\ell m}(b) =_k B_j(a,b) \end{aligned}$$

since $\widetilde{M}_{k-1}$ is compatible with $\tilde{\phi}$. Thus, $\tilde{\phi}$ is an automorphism on the decomposed matrix $M_k = \widetilde{M}_k + \hat{M}_k$ for each $k = 0, 1, \ldots, N$. It is straightforward to verify that $\tilde{\phi}$ has order ℓ on M_N, since the row and column indices in this matrix contain representatives from each orbit of ϕ.

Finally, we need to show by first decomposing M using ψ then using $\tilde{\phi}$ the result is the divisor matrix M_ϕ. To see this we assume that $|\mathcal{O}_\phi(b)| = p^k m$ and note that

$$\left(M_{\phi^\ell}\right)_{\tilde{\phi}}(a,b) = \sum_{r\in\mathcal{O}_{\tilde{\phi}}(b)} M_{\phi^\ell}(a,r) = \sum_{r\in\mathcal{O}_{\tilde{\phi}}(b)} \sum_{s\in\mathcal{O}_{\phi^\ell}(r)} M(a,s)$$

Since $\mathscr{O}_{\tilde{\phi}}(b) = \{\phi^{(1-\ell\alpha)t}(b) = \phi^{\beta p^{N-k} p^k t}(b) \mid t = 0, \ldots, m\}$ (since β is relatively prime to both p and m),

$$\bigcup_{r \in \mathscr{O}_{\tilde{\phi}}(b)} \mathscr{O}_{\phi^\ell}(r) = \{\phi^{\ell x} \circ \phi^{(1-\ell\alpha)t}(b) \mid t = 0, \ldots, m,\ x = 0, \ldots, p^k\}.$$

Thus we obtain $\left(M_{\phi^\ell}\right)_{\tilde{\phi}}(a, b) = M_\phi(a, b)$, as desired.

□

By repeated application of Theorem 3.13 and Proposition 3.15 we can finally state the general theorem of equitable decompositions.

Theorem 3.16. ***(Equitable Decompositions over Arbitrary Automorphisms)*** *Let G be a graph, ϕ be any automorphism of G, and M be an automorphism compatible matrix of G. Then there exists an invertible matrix T that can be explicitly constructed such that*

$$T^{-1} M T = M_\phi \oplus B_1 \oplus B_2 \oplus \cdots B_{h-1} \tag{3.18}$$

where M_ϕ is the divisor matrix associated with ϕ. Thus $\sigma(M) = \sigma\left(M_\phi\right) \cup \sigma(B_1) \cup \sigma(B_2) \cup \cdots \cup \sigma(B_{h-1})$.

We now give an algorithm for completely decomposing a graph with respect to any of its automorphisms.

Performing Equitable Decompositions of General Type

Let G be a graph with automorphism compatible matrix M and ϕ of order ℓ with prime factorization $\ell = p_0^{N_0} p_1^{N_1} \cdots p_{h-1}^{N_{h-1}}$. Initially set $M_0 = M$, $\ell_0 = \ell$, and $\phi_0 = \phi$. We perform h sequential decompositions of M, one for each prime in the factorization of ℓ. To begin we start with $i = 0$, and move to *Step (a).*

Step a: *Let $\ell_{i+1} = \ell_i / p_i^{N_i}$. Form the prime-power automorphism $\psi_i = \phi_i^{\ell_{i+1}}$,* which has order $p_i^{N_i}$.

Step b: *Perform the N_i equitable decompositions of M_i using the algorithm described in Section 3.4. Throughout this process we choose a semi-transversal as prescribed in Remark 3.5. Finally we define M_{i+1} to be the resulting matrix of the above algorithm.*

Step c: *Define $\phi_{i+1} = \tilde{\phi}_i = \phi_i^{1-\ell_i\alpha}$, where α is the integer chosen so that $1 = \ell_{i+1}\alpha + p_i^{N_i}\beta$ as described in the proof of Proposition 3.15. If $i < h$, then set $i = i + 1$ and return to Step a. Otherwise, the decomposition is complete.*

The procedure described in Steps (a)–(c) allows one to sequentially decompose a matrix M over any of its automorphisms. By extension we refer to the resulting matrix as an *equitable decomposition* of M over ϕ. The following example illustrates an equitable decomposition over an automorphism that is neither basic, separable, nor prime-powered, i.e. an equitable partition that cannot be done by any previously given algorithm.

Example 3.17. Consider the graph G shown in Figure 3.6. Here we consider its adjacency matrix M.

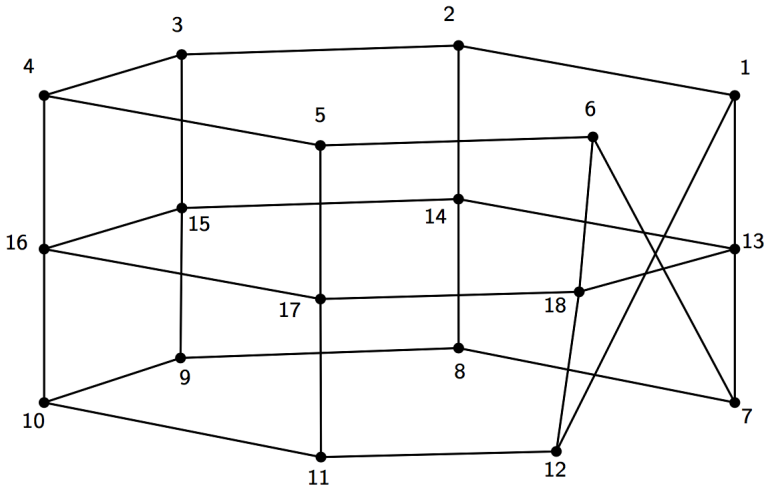


Figure 3.6: The graph G with automorphism of order 12.

This graph has the automorphism

$$\phi = (1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12)(13, 14, 15, 16, 17, 18),$$

which has order $12 = 2^2 \cdot 3$. Thus in the prime decomposition of $|\phi|$, $p_0 = 2, N_0 = 2$ and $p_1 = 3, N_1 = 1$. Because there are two distinct prime factors in this prime decomposition we will go through the steps (a)-(c) in our algorithm two times in order to fully decompose the graph G. We start with $i = 0$, $M_0 = M$, $\ell_0 = 12$, and $\phi_0 = \phi$.

Round 1

Step (a): We start with $\ell_1 = \ell_0/p_0^{N_0} = 12/2^2 = 3$ so that

$$\psi_0 = \phi_0^3 = (1, 4, 7, 10)(2, 5, 8, 11)(3, 6, 9, 12)(13, 16)(14, 17)(15, 18).$$

Step (b): Now we run though the algorithm given in Section 3.4 for decomposing a graph over a prime-power automorphism. In this case we will require two rounds for the automorphism ψ_0.

In Round 1 of this decomposition, we have three orbits of maximal length. When choosing our semi-transversal $\mathcal{T}_0$, we are free to choose any element from the first orbit, but based on that choice the other two elements of $\mathcal{T}_0$ are determined by the rules in Remark 3.5. We choose vertex 1 to be in $\mathcal{T}_0$. Therefore $\mathcal{T}_0 = \{1, \phi_0^4(1), \phi_0^{2\cdot 4}(1)\} = \{1, 5, 9\}$. Thus,

$$\mathcal{T}_0 = \{1, 5, 9\},\ \mathcal{T}_1 = \{4, 8, 12\},\ \mathcal{T}_2 = \{7, 11, 3\},\ \text{and } \mathcal{T}_3 = \{10, 2, 6\}.$$

The remaining vertices are put into $\mathcal{T}_F = \{13, 14, 15, 16, 17, 18\}$ since they are contained in orbits of ψ_0 whose order is not maximal. The relevant block matrices for this stage of the decomposition are

$$F = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}, \; H = L^T = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$C_0 = C_2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \; C_1 = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \; C_3 = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}$$

$$D_0 = \left[\begin{array}{c|c} C_0 & C_1 \\ \hline C_3 & C_0 \end{array}\right], \; D_1 = \left[\begin{array}{c|c} C_2 & C_3 \\ \hline C_1 & C_2 \end{array}\right]$$

The first round of this algorithm will result in a decomposed matrix M_1 whose adjacency matrix has the form,

$$\left[\begin{array}{c|cc} F & 2H & \\ \hline L & 0 & C_1 + C_3 \\ \hline & C_1 + C_3 & 0 \end{array}\right] \oplus \begin{bmatrix} & -i & i \\ i & 0 & -i \\ -i & i & 0 \end{bmatrix} \oplus \begin{bmatrix} & i & -i \\ -i & 0 & i \\ i & -i & 0 \end{bmatrix}.$$

The decomposed graph is shown in Figure 3.7. In the second iteration of the sub-algorithm, we use Equation (3.15) to find the new automorphism

$$\psi_1 = \psi_0^2 = (1,4)(5,8)(9,12)(13,16)(14,17)(15,18),$$

which has six cycles of length two. We now must choose the semi-transversals for the next step. We begin with $\mathcal{T}_0 = \{1, 5, 9\}$ from the previous round. Now we are free to choose any element from the orbits of length 2^1 of ϕ_0, so we choose 13, and then add $\{13, \phi_0^2(13), \phi_0^{2\cdot 2}(13)\} = \{13, 15, 17\}$ to

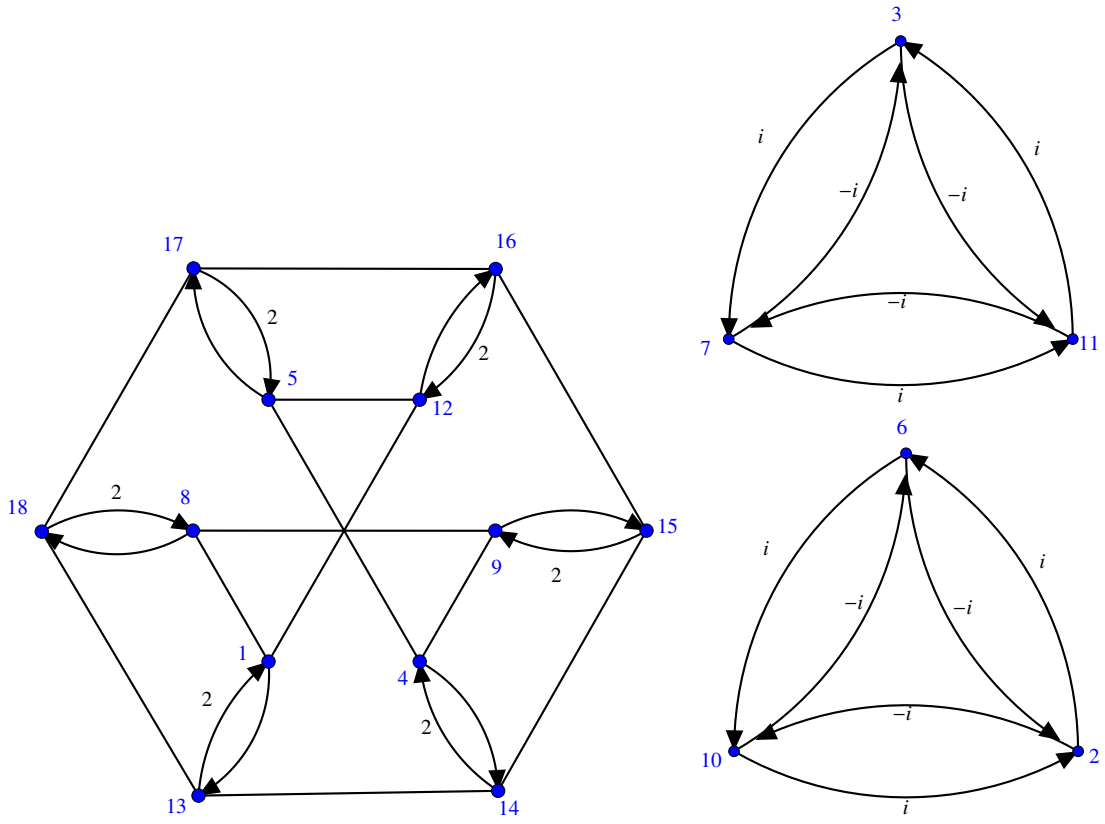


Figure 3.7: The decomposed graph G after one round of decomposing using automorphism $\psi_0 = (1,4,7,10)\,(2,5,8,11)(3,6,9,12)(13,16)(14,17)(15,18)$. The weights of unidirectional and bidirectional edges are equal to one unless otherwise stated.

$\mathcal{T}_0$. Thus we have

$$\mathcal{T}_0 = \{1,5,9,13,15,17\} \text{ and } \mathcal{T}_1 = \{4,8,12,16,18,14\}.$$

Using these transversals, the relevant block matrices are

$$F = H = L = \emptyset,\ C_0 = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 \end{bmatrix},\ C_1 = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 \end{bmatrix}$$

So, the second round results in an adjacency matrix M_2 with the corresponding weighted adjacency matrix

$$\begin{bmatrix} 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 1 & 0 \\ 2 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 2 & 1 & 0 & 1 \\ 0 & 2 & 0 & 1 & 1 & 0 \end{bmatrix} \oplus \begin{bmatrix} 0 & -1 & -1 & 1 & 0 & 0 \\ -1 & 0 & -1 & 0 & 0 & 1 \\ -1 & -1 & 0 & 0 & 1 & 0 \\ 2 & 0 & 0 & 0 & -1 & -1 \\ 0 & 0 & 2 & -1 & 0 & -1 \\ 0 & 2 & 0 & -1 & -1 & 0 \end{bmatrix} \oplus \begin{bmatrix} 0 & -i & i \\ i & 0 & -i \\ -i & i & 0 \end{bmatrix} \oplus \begin{bmatrix} 0 & i & -i \\ -i & 0 & i \\ i & -i & 0 \end{bmatrix} \quad (3.19)$$

with decomposed graphs given in Figure 3.8

Step (c): Now we have exhausted the decompositions that can be accomplished using $p_0 = 2$. We move to the prime $p_1 = 3$, and notice that

$$1 = 1 \cdot 4 + (-1) \cdot 3.$$

Thus we now use the automorphism

$$\phi_1 = \tilde{\phi} = \phi_0^{(1+3)} = (1,5,9)(2,6,10)(3,7,11)(4,8,12)(13,17,15),(14,18,16).$$

Round 2

Step (a): Now $\ell_2 = 1$ and $\psi_1 = \phi_1^1$. We begin with the matrix M_0 equal to the matrix given in equation (3.19).

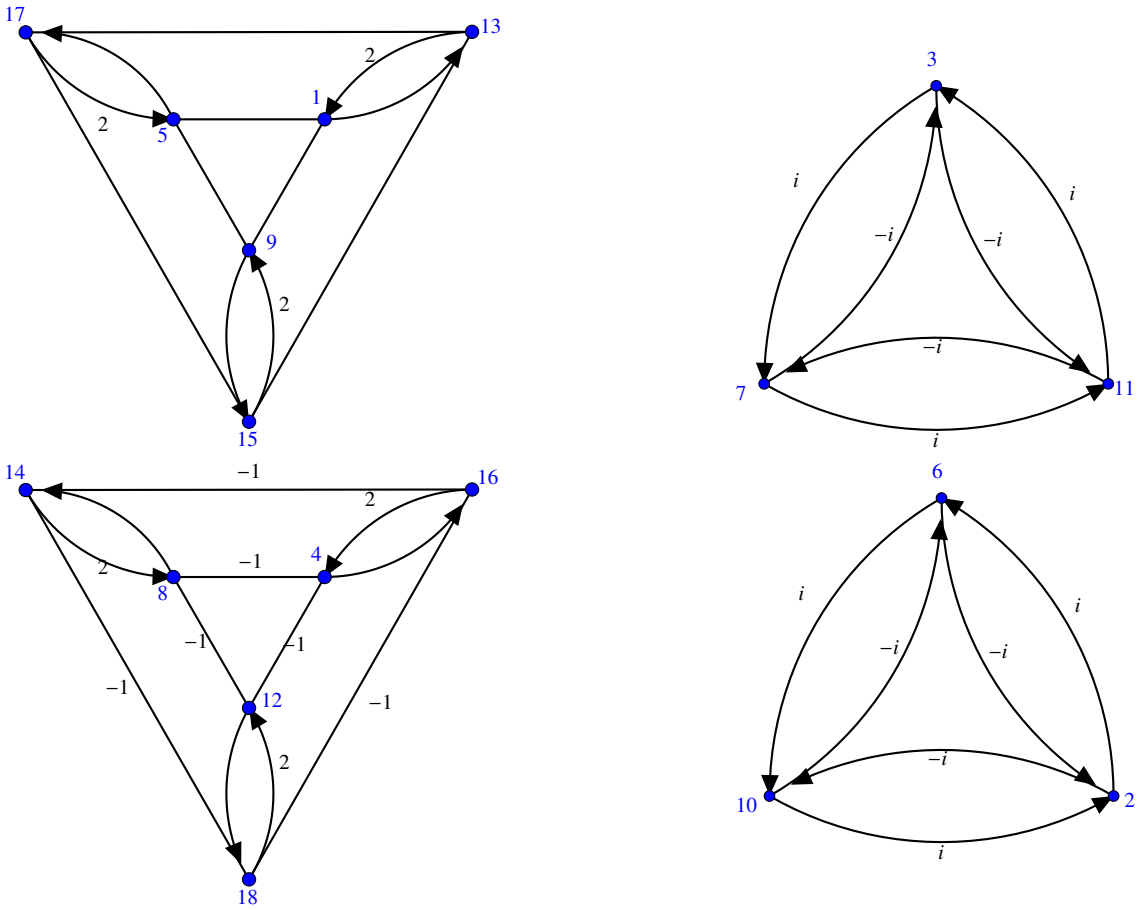


Figure 3.8: The decomposed graph G after the second round using $\psi_0 = (1,4,7,10)(2,5,8,11)(3,6,9,12)(13,16)(14,17)(15,18)$.The weights of unidirectional and bidirectional edges are equal to one unless otherwise stated.

Step (b): On this step we only need to run through the above algorithm once since the order of ϕ_1 is three. It is worth mentioning that during this step we do not need to be worried about how we choose the transversal for this decomposition because this is the final step and the transversal only need to be carefully chosen to guarantee that the resulting decomposed graph contains a symmetry for the next round. We choose the transversal $\mathcal{T}_0 = \{1,2,3,4,13,14\}$, thus

$$\mathcal{T}_0 = \{1,13,4,14,2,3\},\ \mathcal{T}_1 = \{5,17,8,18,6,7\},\ \text{and}\ \mathcal{T}_2 = \{9,15,12,16,10,11\}$$

The relevant block matrices for this decomposition are

$$C_0 = \left[\begin{array}{cc|cc|cc} 0 & 1 & 0 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 \end{array}\right], \; C_1 = \left[\begin{array}{cc|cc|cc} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 2 & -1 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & i & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & i \end{array}\right] C_2 = \left[\begin{array}{cc|cc|cc} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & -i & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & -i \end{array}\right]$$

So, setting $\omega = e^{\frac{2\pi i}{3}}$ the final matrix decomposition is

$$\begin{array}{l} \left[\begin{array}{cc} 2 & 1 \\ 2 & 2 \end{array}\right] \oplus \left[\begin{array}{cc} -2 & 1 \\ 2 & -2 \end{array}\right] \oplus [0] \oplus [0] \oplus \\ \quad \left[\begin{array}{cc} -1 & 1 \\ 2 & -1 \end{array}\right] \oplus \left[\begin{array}{cc} 1 & \omega^2 \\ 2\omega & 1 \end{array}\right] \oplus [\sqrt{3}] \oplus [\sqrt{3}] \oplus \\ \quad\quad \left[\begin{array}{cc} -1 & 1 \\ 2 & -1 \end{array}\right] \oplus \left[\begin{array}{cc} 1 & \omega \\ 2\omega^2 & 1 \end{array}\right] \oplus [-\sqrt{3}] \oplus [-\sqrt{3}] \end{array}$$

Step (c): There is no need to find ϕ_2 since we cannot decompose this matrix any further.

Notice that the first matrix appearing in the above decomposition $\left[\begin{array}{cc} 2 & 1 \\ 2 & 2 \end{array}\right]$ is precisely the divisor matrix associated with the original automorphism ϕ.

3.6 Eigenvectors and Spectral Radii Under Equitable Decompositions

Our first result in this section introduces the notion of an equitable decomposition of the eigenvectors and generalized eigenvectors of a matrix M associated with a graph G. The following theorem tells us how the eigenvectors of a matrix can be constructed from the collection of eigenvectors of the smaller matrices which result from the similarity transformation described in Lemma 3.10.

Theorem 3.18. ***(Eigenvector Decomposition)*** *Let M be an $n \times n$ automorphism compatible matrix of the graph G. For ϕ a prime-powered automorphism of G with f vertices in orbits with length less than p^N. Thus $n = f + rp^N$. Let $\widetilde{M} \oplus B_1 \oplus \cdots \oplus B_{p^N - p^{N-1}}$ be the result of one step of an equitable decomposition of M as described in Lemma 3.10. Suppose $\{\mathbf{u}_{m,\ell} : 1 \leq \ell \leq r\}$ is a (generalized) eigenbasis for B_m for $1 \leq m \leq p^N - p^{N-1}$ and $\{\mathbf{u}_{0,i} : 1 \leq i \leq f + rp^{N-1}\}$ is a (generalized) eigenbasis for $\widetilde{M}$ and where each $\mathbf{u}_{0,i} = \mathbf{w}_i \oplus \mathbf{v}_i$ with $\mathbf{w}_i \in \mathbb{C}^f$ and $\mathbf{v}_i \in \mathbb{C}^{rp^{N-1}}$. Then a (generalized) eigenbasis of M is the set*

$$\left\{ \mathbf{0}_f \oplus \left[\bigoplus_{j=0}^{p^N-1} \omega^{mj} \mathbf{u}_{m,\ell} \right], \mathbf{w}_i \oplus \left[\bigoplus_{j=0}^{p-1} \mathbf{v}_i \right] : 1 \leq m \leq p^N - p^{N-1}, 1 \leq \ell \leq r, \right. \tag{3.20}$$
$$\left. 1 \leq i \leq f + rp^{N-1}, \omega = e^{2\pi i / p^N} \right\}.$$

Moreover, if $\mathbf{x}_{m,l} = \mathbf{0}_f \oplus \left[\bigoplus_{j=0}^{p^N-1} \omega^{mj} \mathbf{u}_{m,l} \right]$ and $\mathbf{x}_{0,i} = \mathbf{w}_i \oplus \left[\bigoplus_{j=0}^{p-1} \mathbf{v}_i \right]$ then the following hold.
(i) If $\lambda_{m,\ell}$ is the ℓ^{th} eigenvalue of B_m then $\lambda_{m,\ell} \in \sigma(M)$ corresponds to the (generalized) eigenvector $\mathbf{x}_{m,l}$.
(ii) If $\lambda_{0,i}$ is an eigenvalue of $\widetilde{M}$ then $\lambda_{0,i} \in \sigma(M)$ corresponds to the (generalized) eigenvector $\mathbf{x}_{0,i}$.

Proof. Now according to Lemma 3.10, we can decompose M as

$$T^{-1} M T = \widetilde{M} \oplus B_1 \oplus B_2 \oplus \cdots \oplus B_{p^N - p^{N-1}} = B,$$

where $\widetilde{M} = \begin{bmatrix} F & kH \\ L & B_0 \end{bmatrix}$. Now let $\mathbf{u}$ be a (generalized) eigenvector of B corresponding to the eigenvalue λ, so that $(B - \lambda I)^t \mathbf{u} = \mathbf{0}$ for some positive integer t, where $t > 1$ if $\mathbf{u}$ is a generalized eigenvector. We now consider the vector $T\mathbf{u}$ which has the property that

$$
\begin{aligned}
(M-\lambda I)^t\, T\mathbf{u} &= (TBT^{-1}-\lambda I)^t T\mathbf{u} \\
&= (T(B-\lambda I)T^{-1})^t T\mathbf{u} \\
&= (T(B-\lambda I)^t T^{-1})T\mathbf{u} \\
&= T(B-\lambda I)^t\mathbf{u} \\
&= 0.
\end{aligned}
$$

Thus, $T\mathbf{u}$ is a (generalized) eigenvector for M.

Because B is block diagonal, the (generalized) eigenvectors of B are either $(\mathbf{0}_f^T\ \mathbf{0}_{rp^{N-1}}^T, \mathbf{0}_r^T \dots \mathbf{u}_{m,\ell}^T \dots \mathbf{0}_r^T)^T$ or $(\mathbf{w}_i^T\ \mathbf{v}_i^T\ \mathbf{0}_r^T \dots \mathbf{0}_r^T)^T$ where $\mathbf{u}_{m,\ell}$ is the m^{th} component in this block vector and represents the ℓ^{th} (generalized) eigenvector of B_m associated with eigenvalue $\lambda_{m,\ell}$ and $\mathbf{w}_i \oplus \mathbf{v}_i = \mathbf{u}_{0,i}$, ($\mathbf{w}_i \in \mathbb{C}^f$ and $\mathbf{v}_i \in \mathbb{C}^{rp^{N-1}}$), the i^{th} (generalized) eigenvector of M_ϕ. Thus the (generalized) eigenvectors of M are represented by

$$
T\begin{pmatrix} \mathbf{0}_f \\ \mathbf{0}_{rp^{N-1}} \\ \mathbf{0}_r \\ \vdots \\ \mathbf{u}_{m,l} \\ \vdots \\ \mathbf{0}_r \end{pmatrix} = \left[\begin{array}{cc|cccc} I_f & 0 & 0 & 0 & \cdots & 0 \\ \hline 0 & I_{rp^{N-1}} & & & & \\ 0 & I_{rp^{N-1}} & & & & \\ 0 & I_{rp^{N-1}} & & & S & \\ \vdots & & & & & \\ 0 & I_{rp^{N-1}} & & & & \end{array}\right] \begin{pmatrix} \mathbf{0}_f \\ \mathbf{0}_{rp^{N-1}} \\ \mathbf{0}_r \\ \vdots \\ \mathbf{u}_{m,\ell} \\ \vdots \\ \mathbf{0}_r \end{pmatrix} = \begin{pmatrix} \mathbf{0}_f \\ \mathbf{u}_{m,\ell} \\ \omega^m \mathbf{u}_{m,\ell} \\ \omega^{2m} \mathbf{u}_{m,\ell} \\ \vdots \\ \omega^{m(p^N-1)} \mathbf{u}_{m,\ell} \end{pmatrix} = \mathbf{0}_f \oplus \bigoplus_{j=0}^{k-1} \omega^{mj} \mathbf{u}_{m,l}
$$

and

$$T\begin{pmatrix}\mathbf{w}_i\\ \mathbf{v}_i\\ \mathbf{0}_r\\ \vdots\\ \mathbf{0}_r\end{pmatrix} = \left[\begin{array}{cc|ccccc} I_f & 0 & 0 & 0 & \cdots & 0 \\ \hline 0 & I_{rp^{N-1}} & & & & \\ 0 & I_{rp^{N-1}} & & & & \\ 0 & I_{rp^{N-1}} & & & S & \\ \vdots & & & & & \\ 0 & I_{rp^{N-1}} & & & & \end{array}\right]\begin{pmatrix}\mathbf{w}_i\\ \mathbf{v}_i\\ \mathbf{0}_r\\ \vdots\\ \mathbf{0}_r\end{pmatrix} = \begin{pmatrix}\mathbf{w}_i\\ \mathbf{v}_i\\ \mathbf{v}_i\\ \vdots\\ \mathbf{v}_i\end{pmatrix} = \mathbf{w}_i \oplus \bigoplus_{j=0}^{p-1} \mathbf{v}_i.$$

Thus we have found n (generalized) eigenvectors of the original matrix M. In order to show this is a complete (generalized) eigenbasis, we need to show that (3.20) is a set of linearly independent vectors. To do so let E_0 be the $(f + rp^{N-1}) \times (f + rp^{N-1})$ matrix formed from the eigenbasis vectors of the divisor matrix M_ϕ and let E_i for $1 \leq i \leq r$ be the $r \times r$ matrices formed from the eigenbasis vectors of B_i, i.e $E_0 = \left[\mathbf{u}_{0,1}\ \mathbf{u}_{0,2} \ldots \mathbf{u}_{f+rp^{N-1}}\right]$, $E_i = [\mathbf{u}_{i,1}\ \mathbf{u}_{i,2} \ldots \mathbf{u}_{i,r}]$. Let E denote the matrix built from the vectors in Equation (3.20) as the columns. Thus we can write E in the following block form

$$E = \begin{bmatrix} E_0 & 0 & 0 & 0 & \ldots & 0 \\ V & E_1 & E_2 & E_3 & \ldots & E_{p^N - p^{N-1}} \\ V & \omega E_1 & \omega^2 E_2 & \omega^3 E_3 & \ldots & \omega^{p^N - p^{N-1}} E_{p^N - p^{N-1}} \\ V & \omega^2 E_1 & \omega^4 E_2 & \omega^6 E_3 & \ldots & \omega^{2(p^N - p^{N-1})} E_{p^N - p^{N-1}} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ V & \omega^{k-1} E_1 & \omega^{2(k-1)} E_2 & \omega^{3(k-1)} E_3 & \ldots & \omega^{(p^N - p^{N-1})^2} E_{p^N - p^{N-1}} \end{bmatrix},$$

where $V = [\mathbf{v}_1\ \mathbf{v}_2\ \ldots\ \mathbf{v}_{N+r}]$. Showing that the vectors in (3.20) are linearly independent, is equivalent to showing that $\det E \neq 0$. Here, we notice that

$$\det(E) = \det\left(T\begin{bmatrix} E_0 & 0 & 0 & \dots & 0 \\ 0 & E_1 & 0 & \dots & 0 \\ 0 & 0 & E_2 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & 0 \\ 0 & 0 & 0 & \dots & E_{k-1} \end{bmatrix}\right) = \det(T)\det\begin{bmatrix} E_0 & 0 & 0 & \dots & 0 \\ 0 & E_1 & 0 & \dots & 0 \\ 0 & 0 & E_2 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & 0 \\ 0 & 0 & 0 & \dots & E_{k-1} \end{bmatrix}$$
$$= \det(T)\prod_{j=0}^{k-1}\det(E_j).$$

We have shown previously that the columns of T are orthogonal, thus $\det(T) \neq 0$. Also we chose the columns of E_i, for $0 \leq i \leq p^N - p^{N-1}$, to be generalized eigenbases. Thus we can guarantee that, for every i, $\det E_i \neq 0$ and thus $\det E \neq 0$. This proves that that the set of n vectors we have found actually constitutes a (generalized) eigenbasis for the matrix M.

One can check that the (generalized) eigenvectors correspond to the eigenvalues, as stated in the theorem, by showing that

$$(M - I\lambda_{m,l})^{t_{m,l}}\left[\mathbf{0}_N \oplus \left[\bigoplus_{j=0}^{p^N-1}\omega^{mj}\mathbf{u}_{m,l}\right]\right] = \mathbf{0} \text{ for } t_{m,l} = (\text{rank of } \mathbf{x}_{m,l})$$

and also that

$$(M - I\lambda_{0,i})^{t_{0,i}}\left[\mathbf{w}_i \oplus \left[\bigoplus_{j=0}^{p-1}\mathbf{v}_i\right]\right] = \mathbf{0} \text{ for } t_{0,i} = (\text{rank of } \mathbf{x}_{0,i}).$$

□

In this proof we only showed how to construct the eigenvectors after a single decomposition step, i.e. a single similarity transformation. Because we were able to show in Section 3.4 that equitably decomposing a matrix over prime-power automorphism can be done via a sequence of such similarity transformations, we can use this formula iteratively to reconstruct the eigenvectors of a matrix M starting with the eigenvectors of M_N. Further, because we were able to describe an equitable decomposition over a general automorphism in Section 3.5 as a sequence of decomposi-

tions over prime-power automorphisms, we can again use this formula iteratively to construct the eigenvectors for a matrix starting with the eigenvectors of the decomposition associated with any general automorphism.

Example 3.19. As an illustration of Theorem 3.18 we consider the graph G with a basic automorphism shown in Figure 3.1. In Example 3.8 we found the decomposition $A(1) = A(0)_\psi \oplus B(0)_1 \oplus B(0)_2$ of the adjacency matrix $A = A(G)$ over the basic automorphism $\psi = \phi^2 = (2,8,5)(3,9,7)(4,10,6)$ (see Equation (3.6)). Eigenbases corresponding to A_ψ, B_1, and B_2, respectively, are given by the vectors $\mathbf{u}_{0,i}$, $\mathbf{u}_{1,i}$, and $\mathbf{u}_{2,i}$ where

$$A_\psi = \begin{bmatrix} 0 & 3 & 0 & 0 \\ 1 & 2 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \qquad \begin{array}{llll} \mathbf{u}_{0,1} = (3,\ 1+\sqrt{6},\ 1,\ 1)^T & \mathbf{w}_1 = (3) & \mathbf{v}_1 = (1+\sqrt{6},\ 1,\ 1)^T \\ \mathbf{u}_{0,2} = (3,\ 1-\sqrt{6},\ 1,\ 1)^T & \mathbf{w}_2 = (3) & \mathbf{v}_2 = (1-\sqrt{6},\ 1,\ 1)^T \\ \mathbf{u}_{0,3} = (-1,\ 0,\ 0,\ 1)^T & \mathbf{w}_3 = (-1) & \mathbf{v}_3 = (0,\ 0,\ 1)^T \\ \mathbf{u}_{0,4} = (-1,\ 0,\ 1,\ 0)^T & \mathbf{w}_4 = (-1) & \mathbf{v}_4 = (0,\ 1,\ 0)^T \end{array}$$

$$B_1 = B_2 = \begin{bmatrix} -1 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \qquad \begin{array}{l} \mathbf{u}_{1,1} = \mathbf{u}_{2,1} = (-2,\ 1,\ 1)^T \\ \mathbf{u}_{1,2} = \mathbf{u}_{2,2} = (1,\ 1,\ 1)^T \\ \mathbf{u}_{1,3} = \mathbf{u}_{2,3} = (0,\ -1,\ 1)^T \end{array} .$$

Note that $\mathbf{w}_i$ has only one component since the first basic automorphism only fixed one vertex, i.e. $N = 1$. Using the formula in Theorem 3.18 an eigenbasis of the original matrix A is given by

$$\mathbf{w}_1 \oplus \mathbf{v}_1 \oplus \mathbf{v}_1 \oplus \mathbf{v}_1 = (3,\ 1+\sqrt{6},\ 1,\ 1,\ 1+\sqrt{6},\ 1,\ 1,\ 1+\sqrt{6},\ 1,\ 1)^T$$

$$\mathbf{w}_2 \oplus \mathbf{v}_2 \oplus \mathbf{v}_2 \oplus \mathbf{v}_2 = (3,\ 1-\sqrt{6},\ 1,\ 1,\ 1-\sqrt{6},\ 1,\ 1,\ 1-\sqrt{6},\ 1,\ 1)^T$$

$$\mathbf{w}_3 \oplus \mathbf{v}_3 \oplus \mathbf{v}_3 \oplus \mathbf{v}_3 = (-1,\ 0,\ 0,\ 1,\ 0,\ 0,\ 1,\ 0,\ 0,\ 1)^T$$

$$\mathbf{w}_4 \oplus \mathbf{v}_4 \oplus \mathbf{v}_4 \oplus \mathbf{v}_4 = (-1,\ 0,\ 1,\ 0,\ 0,\ 1,\ 0,\ 0,\ 1,\ 0)^T$$

$$\mathbf{0}_N \oplus \mathbf{u}_{1,1} \oplus \omega\mathbf{u}_{1,1} \oplus \omega^2\mathbf{u}_{1,1} = (0,\ -2,\ 1,\ 1,\ -2\omega,\ \omega,\ \omega,\ -2\omega^2,\ \omega^2,\ \omega^2)^T$$

$$\mathbf{0}_N \oplus \mathbf{u}_{1,2} \oplus \omega\mathbf{u}_{1,2} \oplus \omega^2\mathbf{u}_{1,2} = (0,\ 1,\ 1,\ 1, \omega,\ \omega,\ \omega,\ \omega^2,\ \omega^2,\ \omega^2)^T$$

$$\mathbf{0}_N \oplus \mathbf{u}_{1,3} \oplus \omega\mathbf{u}_{1,3} \oplus \omega^2\mathbf{u}_{1,3} = (0,\ 0,\ -1,\ 1,\ 0,\ -\omega,\ \omega,\ 0,\ -\omega^2,\ \omega^2)^T$$

$$\mathbf{0}_N \oplus \mathbf{u}_{2,1} \oplus \omega^2\mathbf{u}_{2,1} \oplus \omega^4\mathbf{u}_{2,1} = (0,\ -2,\ 1,\ 1,\ -2\omega^2,\ \omega^2,\ \omega^2,\ -2\omega,\ \omega,\ \omega)^T$$

$$\mathbf{0}_N \oplus \mathbf{u}_{2,2} \oplus \omega^2\mathbf{u}_{2,2} \oplus \omega^4\mathbf{u}_{2,2} = (0,\ 1,\ 1,\ 1, \omega^2,\ \omega^2,\ \omega^2,\ \omega,\ \omega,\ \omega)^T$$

$$\mathbf{0}_N \oplus \mathbf{u}_{2,3} \oplus \omega^2\mathbf{u}_{2,3} \oplus \omega^4\mathbf{u}_{2,3} = (0,\ 0,\ -1,\ 1,\ 0,\ -\omega^2,\ \omega^2,\ 0,\ -\omega,\ \omega)^T$$

where $\omega = e^{2\pi i/3}$.

The process carried out in Example 3.19 of constructing eigenvectors of a matrix from the eigenvectors of its decomposition over a basic automorphism can also be done for separable automorphisms. This is done by finding the eigenvectors of the matrices in the final decomposition and working backwards, as in this example, until the eigenvectors of the original matrix have been fully reconstructed.

If ϕ is a uniform automorphism of a graph G then the eigenvectors of an automorphism compatible matrix M can also be decomposed as is shown in (3.20) for $N = 0$. Specifically, if $M_\phi \oplus B_1 \oplus B_2 \oplus \cdots \oplus B_{k-1}$ is an equitable decomposition of M with respect to ϕ in which $\mathbf{u}_{m,\ell}$ is the ℓ^{th} eigenvector of B_m then the eigenvectors of M are given by the set

$$\left\{ \bigoplus_{j=0}^{k-1} \omega^{mj}\mathbf{u}_{m,\ell} : 1 \le m \le k-1, 1 \le \ell \le r, \omega = e^{2\pi i/k} \right\}.$$

It is worth noting that both the eigenvalues and eigenvectors of M are *global* characteristics of the matrix M in the sense that they depend, in general, on all entries of the matrix or equivalently on the entire structure of the graph G. In contrast, many symmetries of a graph G are inherently *local*, specifically when they correspond to an automorphism that fixes some subset of the vertex set of G, e.g. a basic automorphism.

This difference is particularly important in the case where we are interested in deducing spectral properties associated with the graph structure of a network. Reasons for this include the fact that most real networks are quite large, often having either thousands, hundreds of thousands, or more vertices. Second, real networks are on average much more structured and in particular have more symmetries than random graphs (see [5]). Third, there is often only partial or local information regarding the structure of many of these networks because of the complications in obtaining network data (see, for instance, [19]).

The implication, with respect to equitable decompositions, is that by finding a local graph symmetry it is possible to gain information regarding the graph's set of eigenvalues and eigenvectors, which is information typically obtained by analyzing the entire structure of the network. This information, although incomplete, can be used to determine key spectral properties of the network.

One of the most important and useful of these characteristics is the spectral radius associated with the graph structure G of a network. The spectral radius $\rho(M)$ of a network, or, more generally, a dynamical system, is particularly important for studying the system's dynamics. For instance, the matrix M associated with a network may be a global or local linearization of the system of equations that govern the network's dynamics. If the network's dynamics are modeled by a discrete-time system, then stability of the system is guaranteed if $\rho(M) < 1$ and local instability results when $\rho(M) > 1$ [23, 31, 48].

Using the theory of equitable decompositions, it is possible to show not only that $\sigma(M_\phi) \subset \sigma(M)$, but also that the spectral radius $\rho(M)$ of M is an eigenvalue of M_ϕ if M is both nonnegative and irreducible.

Proposition 3.20. ***(Spectral Radius of Separable Equitable Partition)*** *Let ϕ be a basic or separable automorphism of a graph G with M an automorphism compatible matrix. If M is nonnegative, then $\rho(M) = \rho(M_\phi)$. If M is both irreducible and nonnegative then the spectral radius, $\rho(M)$, is an eigenvalue of M_ϕ.*

Proof. We begin by proving the result for basic (and uniform) automorphisms and then extending the result to separable automorphisms. Assume M is nonnegative and with basic automorphism ϕ. To prove that $\rho(M) = \rho(M_\phi)$, we first claim that $\rho(M_\phi) \geq \rho(B_j)$ for $1 \leq j \leq k-1$ where $M_\phi \oplus B_1 \oplus \cdots \oplus B_{k-1}$ is an equitable decomposition of M with respect to ϕ.

To verify this claim, we first need Corollary 8.1.20 in [40] which states that if N is a principal submatrix of M then $\rho(N) \leq \rho(M)$ if M is nonnegative. Recall that $M_\phi = \begin{bmatrix} F & kH \\ L & B_0 \end{bmatrix}$ if ϕ is a basic automorphism that fixes some positive number of vertices. Thus, for a basic automorphism, B_0 is a principal submatrix of M_ϕ. Since M is nonnegative, Equation (3.2) shows that M_ϕ is nonnegative, and we can conclude that $\rho(B_0) \leq \rho(M_\phi)$. In the case that ϕ is a uniform automorphism, $M_\phi = B_0$.

Next, for a matrix $P \in \mathbb{C}^{n\times n}$, let $|P| \in \mathbb{R}^{n\times n}_{\geq 0}$ denote the matrix with entries $|P|_{ij} = |P_{ij}|$, i.e. $|P|$ is the entrywise absolute value of P. Moreover, if $P, Q \in \mathbb{R}^{n\times n}$ let $P \leq Q$ if $P_{ij} \leq Q_{ij}$ for all $1 \leq i, j \leq n$. Theorem 8.1.18 in [40], states that if $|P| \leq Q$ then $\rho(P) \leq \rho(Q)$. Because

$$\left|B_j\right| = \left|\sum_{m=0}^{k-1} \left(\omega^j\right)^m M_m\right| \leq \sum_{m=0}^{k-1} \left|\left(\omega^j\right)^m M_m\right| = \sum_{m=0}^{k-1} M_m = B_0$$

we can conclude that $\rho(B_j) \leq \rho(B_0)$ for all $1 \leq j \leq k-1$. Therefore,

$$\rho(B_j) \leq \rho(B_0) \leq \rho(M_\phi) \text{ for all } 1 \leq j \leq k-1, \tag{3.21}$$

which verifies our claim. Using this claim and the fact that $\sigma(M) = \sigma(M_\phi) \cup \sigma(B_1) \cup \cdots \cup \sigma(B_{k-1})$ we can immediately conclude that $\rho(M) = \rho(M_\phi)$.

Now we assume that M is both nonnegative and irreducible. The Perron-Frobenius Theorem implies that $r = \rho(M)$ is a simple eigenvalue of M.

Next we claim that if M is irreducible, then M_ϕ is also irreducible. Let M be an $n \times n$ nonnegative matrix with an associated basic automorphism ϕ with order k, and let M_ℓ be the submatrix of M associated with the ℓ^{th} power of the semitransversal in an equitable decomposition of M over ϕ. Recall that a matrix is reducible if and only if its associated weighted digraph is *strongly connected*, meaning for any two vertices in the graph there is a directed path between them. Suppose G is the strongly connected graph with weighted adjacency matrix M. Also we suppose that G_ϕ is the graph whose weighted adjacency matrix is M_ϕ and let a and b be vertices of G_ϕ. Note that every vertex fixed by ϕ in G directly corresponds to a vertex in G_ϕ, and all other vertices correspond to a collection of k vertices in G. Choose a' and b' in G to be any vertices corresponding to a and b, respectively. Now because G is strongly connected it contains a path $a' = v'_0, v'_1, v'_2, \ldots, b' = v'_m$ from a' to b'. Consider the sequence of vertices $a = v_0, v_1, v_2, \ldots, b = v_m$ where v_i is the unique vertex in G_ϕ to corresponding v'_i in G. To prove this is a path we must show each of the entries in matrix M_ϕ corresponding to the edges $v_i \to v_{i+1}$ are positive. If v'_i or v'_{i+1} are fixed by ϕ then the entry in M corresponding to the $v'_i \to v'_{i+1}$ edge is either equal to the entry in M_ϕ corresponding to the $v_i \to v_{i+1}$ edge, or is a positive multiple thereof. If v'_i and v'_{i+1} are not fixed by ϕ, then suppose the $v'_i \to v'_{i+1}$ edge corresponds to $M_\ell(r, s)$ for some ℓ and for some indices r and s. By hypothesis, $M_\ell(r, s) > 0$. The $v_i \to v_{i+1}$ edge corresponds to $B_0(r, s)$, and $B_0 = \sum_\ell M_\ell$, where each $M_\ell(r, s)$ is nonnegative. Therefore this entry must also be positive in M_ϕ. Thus we can conclude that G_ϕ is strongly connected and therefore M_ϕ is irreducible.

Because M_ϕ is irreducible, we can apply the Perron-Frobenius Theorem to M_ϕ. This implies that $\rho(M_\phi)$ is an eigenvalue of M_ϕ, but from the first part of this theorem, we already showed that $\rho(M) = \rho(M_\phi)$, thus we conclude that $\rho(M)$ must be an eigenvalue of M_ϕ.

This completes the proof if ϕ is a basic automorphism. If ϕ is separable then Proposition 3.7 guarantees that there are basic automorphisms $\psi_0, \ldots, \psi_h$ that induce a sequence of equitable decompositions on M such that $M_\phi = (\ldots(M_{\psi_0})_{\psi_1}\ldots)_{\psi_h}$. By induction each subsequent decomposition results in a nonnegative divisor matrix $(\ldots(M_{\psi_0})_{\psi_1}\ldots)_{\psi_i}$ for $i \leq h$ with the same spectral radius $r = \rho(M)$ implying that $\rho(M_\phi) = \rho(M)$ for any $\phi \in Aut(G)$. □

This result can be generalized in that the spectral radius $\rho(M)$ of an automorphism compatible matrix M is the same as the spectral radius of the divisor matrix M_ϕ for any automorphism ϕ if M is both nonnegative and irreducible.

Proposition 3.21. ***(Spectral Radius of a General Equitable Partition)*** *Let ϕ be any automorphism of a graph G with M an automorphism compatible matrix. If M is nonnegative and irreducible, then $\rho(M) = \rho(M_\phi)$.*

Before we can prove the proposition we need the following Lemma.

Lemma 3.22. *If irreducible, non-negative matrix M has block circulant form*

$$M = \begin{bmatrix} A_1 & A_2 & A_3 & \dots & A_n \\ A_n & A_1 & A_2 & \dots & A_{n-1} \\ A_{n-1} & A_n & A_1 & \dots & A_{n-2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ A_2 & A_3 & A_4 & \dots & A_1 \end{bmatrix}, \text{ and } N = \sum_{m=1}^{n} A_m$$

then

$$\rho(M) = \rho(N).$$

Proof. Because M is non-negative and irreducible, then N must also be non-negative and irreducible. Thus the Perron-Frobenius Theorem guarantees that the spectral radius of N, $\rho(N)$, is a positive eigenvalue of N. It also guarantees that the eigenvector $\mathbf{v}$ associated to $\rho(N)$ can be chosen to have all positive entries. Now consider the vector $\mathbf{w} = \mathbf{v} \oplus \mathbf{v} \oplus \cdots \oplus \mathbf{v}$ (a total of m $\mathbf{v}$'s in the direct sum). We can see that $\mathbf{w}$ is an eigenvector of M since

$$M\mathbf{w} = \begin{bmatrix} A_1 & A_2 & A_3 & \dots & A_n \\ A_n & A_1 & A_2 & \dots & A_{n-1} \\ A_{n-1} & A_n & A_1 & \dots & A_{n-2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ A_2 & A_3 & A_4 & \dots & A_1 \end{bmatrix} \begin{bmatrix} \mathbf{v} \\ \mathbf{v} \\ \mathbf{v} \\ \vdots \\ \mathbf{v} \end{bmatrix} = \begin{bmatrix} \sum A_m\mathbf{v} \\ \sum A_m\mathbf{v} \\ \sum A_m\mathbf{v} \\ \vdots \\ \sum A_m\mathbf{v} \end{bmatrix} = \begin{bmatrix} \rho(N)\mathbf{v} \\ \rho(N)\mathbf{v} \\ \rho(N)\mathbf{v} \\ \vdots \\ \rho(N)\mathbf{v} \end{bmatrix} = \rho(N)\mathbf{w}.$$

Thus $\mathbf{w}$ is an eigenvector of M with only positive entries and with eigenvalue $\rho(N)$. Because M is irreducible and non-negative, the Perron-Frobenius Theorem tell us the only eigenvector of M that is all positive must correspond the the largest eigenvalue, which is the spectral radius. Therefore we conclude that $\rho(N) = \rho(M)$. □

Now we prove Proposition 3.21.

Proof. Suppose we have matrix M which is irreducible and non-negative with an associated automorphism ϕ. Using the process outlined in the algorithm found in Section 3.5, we can decompose M through a process of similarity transformations. Throughout this proof we will follow all the notation found in Lemma 3.10. We will first show after the similarity transformation in Lemma 3.10 that $\rho(M) = \rho(\tilde{M})$. At the final step, $\tilde{M} = M_\phi$, the divisor matrix associated with ϕ

In order to perform the similarity transformation, we first need to reorder the rows and columns of M as prescribed in Proposition 3.11 and construct the T matrix. From Lemma 3.10, we have

$$T^{-1}MT = \tilde{M} \oplus B_1 \oplus B_2 \oplus \cdots \oplus B_{p^N - p^{N-1}}.$$

where $\tilde{M} = \begin{bmatrix} F & kH \\ L & B_0 \end{bmatrix}$. First we will prove the claim that $\rho(\tilde{M}) \geq \rho(B_j)$ for $1 \leq j \leq p^N - p^{N-1}$.

To begin, we use Corollary 8.1.20 in [40] which states for a non-negative matrix Q, if P is a principal submatrix of Q then $\rho(P) \leq \rho(Q)$. Since M is nonnegative, and $B_0 = \sum_{m=0}^{p-1} D_m$, we know $\tilde{M}$ is nonnegative. Because B_0 is a principal submatrix of $\tilde{M}$, using this corollary 8.1.20 we conclude that $\rho(B_0) \leq \rho(\tilde{M})$.

Next we must show that $\rho(B_j) \leq \rho(B_0)$ for $1 \leq j \leq p^N - p^{N-1}$. Because

$$\left|B_j\right| = \left|\sum_{m=0}^{p^N-1} (\omega^{m\gamma_j}) C_m\right| \leq \sum_{m=0}^{p^N-1} |(\omega^{m\gamma_j})\, C_m| = \sum_{m=0}^{p^N-1} C_m,$$

we can use Theorem 8.1.18 in [40] to conclude that

$$\rho(B_j) \leq \rho\left(\sum_{m=1}^{p^N-1} C_m\right)$$

for all $1 \leq j \leq p^N - p^{N-1}$, similar to the proof of Proposition 3.20. We note that

$$B_0 = \sum_{i=0}^{p-1} D_i = \begin{bmatrix} \sum_{t=0}^{p-1} C_{tp^{N-1}} & \sum_{t=0}^{p-1} C_{tp^{N-1}+1} & \cdots & \sum_{t=0}^{p-1} C_{tp^{N-1}+p^N-1} \\ \sum_{t=0}^{p-1} C_{tp^{N-1}+p^N-1} & \sum_{t=0}^{p-1} C_{tp^{N-1}} & \cdots & \sum_{t=0}^{p-1} C_{tp^{N-1}+p^N-2} \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{t=0}^{p-1} C_{tp^{N-1}+1} & \sum_{t=0}^{p-1} C_{tp^{N-1}+2} & \cdots & \sum_{t=0}^{p-1} C_{tp^{N-1}}, \end{bmatrix}$$

which has block-circulant form. Now we apply Lemma 3.22, which shows that $\rho(B_0) = \rho\left(\sum C_m\right)$. Therefore,

$$\rho(B_j) \leq \rho(B_0) \leq \rho(\tilde{M}) \text{ for all } 1 \leq j \leq p^N - p^{N-1}, \tag{3.22}$$

which verifies our claim.

Using this claim and the fact that $\sigma(M) = \sigma(\tilde{M}) \cup \sigma(B_1) \cup \cdots \cup \sigma(B_{p^N - p^{N-1}})$ we can immediately conclude that $\rho(M) = \rho(\tilde{M})$.

We have showed after one step, the spectral radius is always maintained in $\tilde{M}$. We can now just consider decomposing $\tilde{M}$ at each step of the algorithm since all other elements are fixed by the decomposition. In order to use the same argument on the subsequent decompositions, we just need to prove that $(\tilde{M})$ is also irreducible and non-negative. Because M is non-negative, and $\tilde{M}$ is built from elements of M and sums of elements from M (see equation 3.10), we know $\tilde{M}$ is also always non-negative. Also if M is irreducible, then we claim $\tilde{M}$ must also be irreducible. This is proven in the proof of Proposition 3.20.

Now we can repeat the above argument for each decomposition of this algorithm. If ϕ has order $p_1^{N_1} \cdot p_2^{N_2} \cdot \cdots \cdot p_h^{N_h}$, then Proposition 3.15 shows that it is possible to decompose the matrix using a sequence of automorphisms that induce a sequence of equitable decompositions on M. By induction each subsequent decomposition results in a nonnegative divisor matrix $(\dots(M_{\psi_0})_{\psi_1}\dots)_{\psi_i}$ for $i \leq h$ with the same spectral radius $r = \rho(M)$ implying that $\rho(M)$ must be the largest eigenvalue for M_ϕ for any $\phi \in Aut(G)$. □

It is worth noting that many matrices typically associated with real networks are both nonnegative and irreducible. This includes the adjacency matrix as well as other weighted matrices [6]; although, there are some notable exceptions, including Laplacian matrices. Moreover, when analyzing the stability of a network, a linearization M of the network's dynamics inherits the symmetries of the network's structure. Hence, if a symmetry of the network's structure is known then this symmetry can be used to decompose M into a smaller divisor matrix M_ϕ. As M and M_ϕ have the same spectral radius, under the conditions stated in Proposition 3.21, then one can use the smaller matrix M_ϕ to either calculate or estimate the spectral radius of the original unreduced network as is demonstrated in the following example.

Example 3.23. Returning to Example 3.17, we can calculate that the eigenvalues of the graph's adjacency matrix M in this example are

$$\sigma(M) = \{-2 \pm \sqrt{2}, 2 \pm \sqrt{2}, \pm 1 \pm \sqrt{2}, 0\}$$

not including multiplicities. Similarly, we can compute that the eigenvalues of M_ϕ are

$$\sigma(M_\phi) = \{2 + \sqrt{2}, 2 - \sqrt{2}\}.$$

Thus, $\rho(M) = \rho(M_\phi) = 2 + \sqrt{2}$.

3.7 Equitable Decompositions and Improved Eigenvalues Estimates

Beginning in the mid-19th century a number of methods were developed to approximate the eigenvalues of general complex valued matrices. These included the results of Gershgorin [47], Brauer [49], Brualdi [50], and Varga [51]. The main idea in each of these methods is that for a matrix $M \in \mathbb{C}^{n\times n}$ it is possible to construct a bounded region in the complex plane that contains the matrix' eigenvalues. This region serves as an approximation for the eigenvalues of M.

In this section we investigate how equitable decompositions affect, in particular, the approximation method of Gershgorin. Our main result is that the Gershgorin region associated with an equitable decomposition of a matrix M is contained in the Gershgorin region associated with M. That is, by equitably decomposing a matrix over a separable automorphism it is possible to gain improved eigenvalue estimates by use of Gershgorin's theorem.

To describe this result we first give the following classical result of Gershgorin.

Theorem 3.24. ***(Gershgorin's Theorem)*** *[52] Let $M \in \mathbb{C}^{n\times n}$. Then all eigenvalues of M are contained in the set*

$$\Gamma(M) = \bigcup_{i=1}^{n} \left\{ \lambda \in \mathbb{C} : |\lambda - M_{ii}| \leq \sum_{j=1, j\neq i}^{n} |M_{ij}| \right\}$$

Geometrically this theorem states that all eigenvalues of a given matrix $M \in \mathbb{C}^{n\times n}$ must lie in the union of n disks in the complex plane, where the i^{th} disk is constructed from the i^{th} row of M. Specifically, the i^{th} disk is centered at $M_{ii} \in \mathbb{C}$ and has the radius $\sum_{j=1, j\neq i}^{n} |M_{ij}|$. The union of these disks forms the *Gershgorin region* $\Gamma(M)$ of the matrix M. The following theorem describes the effect that an equitable decomposition has on a matrix' Gershgorin region.

Theorem 3.25. *Let ϕ be a basic or separable automorphism of a graph G with M an automorphism compatible matrix. If $B = M_\phi \oplus B_1 \oplus \cdots \oplus B_k$ is an equitable decomposition of M with respect to ϕ then $\Gamma(B) \subseteq \Gamma(M)$.*

Proof. First, suppose for simplicity that ϕ is a uniform automorphism of G. The i^{th} row of the matrix M defines a disk in the complex plane centered at M_{ii} with radius equal to $\sum_{j\neq i} \left|M_{ij}\right|$. So we

want every disk generated by each B_t matrix to be contained in some disk generated by M. We can achieve this if the distance between disk centers is less than the difference in the two disk's radii. Thus we need to show that for every i and t that $\left|M_{qq} - [B_t]_{ii}\right| \leq \left|\sum_{j\neq q}\left|M_{qj}\right| - \sum_{j\neq i}\left|[B_t]_{ij}\right|\right|$ for some q. Let $q = i$. Using Equation (3.2) and rearranging terms we see that

$$\begin{aligned}
\left|\sum_{j\neq q}\left|M_{qj}\right| - \sum_{j\neq i}\left|[B_t]_{ij}\right|\right| &\geq \sum_{l\neq i}^{n}|M_{il}| - \sum_{j\neq i}^{r}\left|[B_t]_{ij}\right| \\
&= \left[\sum_{m=0}^{k-1}\sum_{j=1}^{r}\left|[M_m]_{ij}\right| - \sum_{j\neq i}^{r}\left|\sum_{m=0}^{k-1}(\omega^t)^m[M_m]_{ij}\right|\right] - |[M_0]_{ii}| \\
&= \left[\sum_{m=0}^{k-1}\sum_{j=1}^{r}\left|(\omega^t)^m[M_m]_{ij}\right| - \sum_{j\neq i}^{r}\left|\sum_{m=0}^{k-1}(\omega^t)^m[M_m]_{ij}\right|\right] - |[M_0]_{ii}| \\
&\geq \left[\sum_{m=0}^{k-1}\sum_{j=1}^{r}\left|(\omega^t)^m[M_m]_{ij}\right| - \sum_{j\neq i}^{r}\sum_{m=0}^{k-1}\left|(\omega^t)^m[M_m]_{ij}\right|\right] - |[M_0]_{ii}| \\
&= \sum_{m=0}^{k-1}\left[\sum_{j=1}^{r}\left|(\omega^t)^m[M_m]_{ij}\right| - \sum_{j\neq i}^{r}\left|(\omega^t)^m[M_m]_{ij}\right|\right] - |[M_0]_{ii}| \\
&= \sum_{m=0}^{k-1}\left[\left|(\omega^t)^m[M_m]_{ii}\right|\right] - |[M_0]_{ii}| \\
&= \sum_{m=1}^{k-1}\left|(\omega^t)^m[M_m]_{ii}\right| \\
&\geq \left|\sum_{m=1}^{k-1}(\omega^t)^m[M_m]_{ii}\right| \\
&= |[B_t]_{ii} - M_{ii}| \\
&= |M_{ii} - [B_t]_{ii}|
\end{aligned} \tag{3.23}$$

where the M_m matrices are the block matrices found in the block circulant form of M. Therefore, every disk generated by B_m for each m is contained in some disk generated by M and we conclude that $\Gamma(B) \subset \Gamma(M)$ for a uniform automorphism.

Next we assume our graph has a basic automorphism ϕ that fixes a positive number of vertices

of G. Then the matrix M decomposes in the following way

$$\begin{bmatrix} F & H & H & H & \cdots & H \\ L & M_0 & M_1 & M_2 & \cdots & M_{k-1} \\ L & M_{k-1} & M_0 & M_1 & \cdots & M_{k-2} \\ L & M_{k-2} & M_{k-1} & M_0 & \cdots & M_{k-3} \\ \vdots & \vdots & \vdots & \vdots & & \vdots \\ L & M_1 & M_2 & M_3 & \cdots & M_0 \end{bmatrix} \rightarrow \begin{bmatrix} F & kH & 0 & 0 & \cdots & 0 \\ L & B_0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & B_1 & 0 & \cdots & 0 \\ 0 & 0 & 0 & B_2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & 0 & \cdots & B_{k-1} \end{bmatrix}$$

Clearly the first N rows, which correspond to vertices fixed by the automorphism, have exactly the same Gershgorin region in both matrices. For the next block row, the previous argument implies that the Gershgorin disks associated with these rows are contained in the Gershgorin disks of the corresponding rows in M if we disregard the first block column. Including the first block column in the Gershgorin region calculation increases the radii of the corresponding Gershgorin disks for both matrices by the same amount. Thus we still get containment. For all remaining rows the same argument applies except we only increase the radii of the disks from the original matrix when considering the first block column. Thus, for the basic automorphism ϕ we have $\Gamma(B) \subset \Gamma(M)$.

We showed in Proposition 3.7 that decomposing a matrix over a separable automorphism can be done as a sequence of decompositions of basic automorphisms. After each decomposition of a basic automorphism we get containment. Thus, using an inductive argument we can extend this theorem to all separable automorphisms. □

Example 3.26. Again we consider the graph G shown in Figure 3.1 with basic automorphism $\psi_0 = \phi^2 = (2, 8, 5)(3, 9, 7)(4, 10, 6)$. The equitable decomposition of the adjacency matrix $A = A(G)$ with respect to ψ_0 is given by $B = A_{\psi_0} \oplus B_1 \oplus B_2$ where A_{ψ_0}, B_1, B_2 are given in Equation (3.6). The Gershgorin regions of both A and B are shown in Figure 3.9 where $\Gamma(B) \subset \Gamma(A)$. That is, the equitable decomposition of A over ψ_0 results in an improved Gershgorin estimate of its eigenvalues.

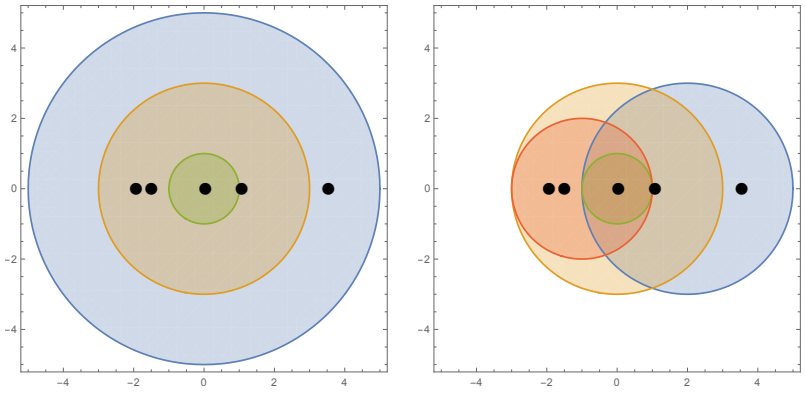


Figure 3.9: The Gershgorin regions $\Gamma(A)$ and $\Gamma(B)$ each made up of a union of disks corresponding to the adjacency matrix $A = A(G)$ of the graph G in Figure 3.1 and its equitable decomposition B over the automorphism $\psi_0 = (2, 8, 5)(3, 9, 7)(4, 10, 6)$, respectively. Black points indicate the eigenvalues $\sigma(A) = \sigma(B)$.

The effectiveness of Gershgorin's theorem in estimating a matrix's eigenvalues depends heavily on whether the row sums $\sum_{j=1, j\neq i}^{n} |M_{ij}|$ are large or small. For example, if the matrix M is the adjacency matrix of a graph G then the i^{th} *disk* of $\Gamma(M)$ has a radius equal to the number of neighbors the vertex v_i has in G. Real networks often contain a few vertices that have an abnormally high number of neighbors. These vertices, which are referred to as *hubs*, greatly reduce the effectiveness of Gershgorin's theorem.

One application of the theory of equitable decompositions is that it can be used to potentially reduce the size of the Gershgorin region associated with a graph's adjacency matrix $A = A(G)$. Note that this region is made up of disks where the radius of the i^{th} disk is equal to the degree of the i^{th} vertex. Hence, the graph's hubs generate the graph's largest Gershgorin disks. The strategy we propose is to find an automorphism ϕ of a graph G that permutes the vertices adjacent to the graph's largest hub. After decomposing the adjacency matrix over ϕ, the resulting decomposed matrix will potentially have a smaller row sum associated with this largest hub and consequently a smaller Gershgorin region. This process can be continued by finding automorphisms of the decomposed graph to further decompose and potentially further reduce the Gershgorin region associated with the decomposed matrix.

This is demonstrated in the following example.

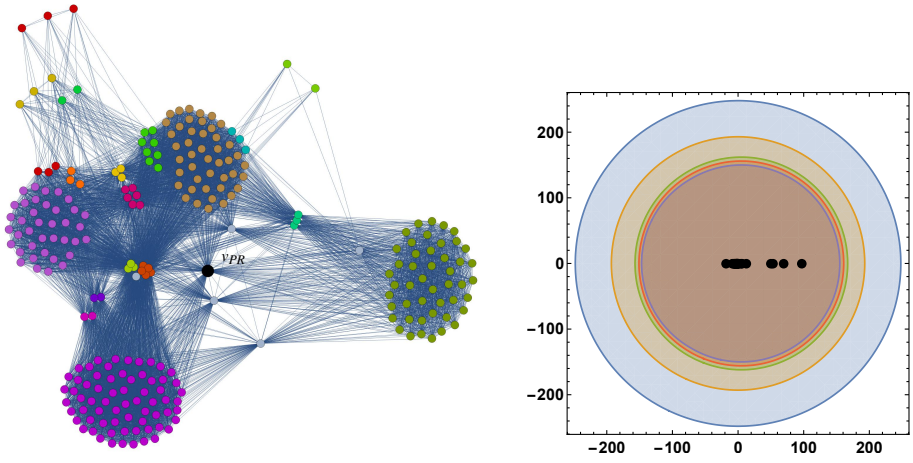


Figure 3.10: Left: A graph representing 254 individuals belonging to seven different organizations in the Boston area prior to the American revolutionary war. Edges represent whether two individuals belonged to the same organization. The black vertex v_{PR} represents Paul Revere. Right: The transposed Gersgorin regions corresponding to a sequence of equitable decompositions performed on the network's adjacency matrix, in which each subsequent decomposition results in a smaller region contained in the previous. Black points indicate the eigenvalues $\sigma(A)$.

Example 3.27. Figure 3.10 (left) shows a social network of 254 members of seven different organizations in the Boston area prior to the American revolutionary war [53]. Each vertex represents a person and an edge between two people appears when these two people belong to the same organization. The colors of the graph represent symmetries present in the graph, i.e. two vertices are colored the same if they are *automorphic*, by which we mean that there is an automorphism ϕ of the graph such that these two vertices belong to the same orbit under ϕ. The black central vertex v_{PR} with the most neighbors (Paul Revere) is connected to 248 of the 254 vertices.

The Gershgorin region $\Gamma(A)$ where A is the adjacency matrix of the this social network is shown as the largest disk in Figure 3.10 (right). The region consist of the union of 254 concentric disks each centered at the origin the largest of which is the disk of radius 248, corresponding to the vertex v_{PR}. Hence, the spectral radius of A is less than or equal to 248.

Using equitable decompositions we can decrease this Gershgorin estimate. The largest contributor to the size of the Gershgorin region is, in fact, the central vertex v_{PR}, since it has the highest degree of any vertex in the network. In order to reduce the associated Gershgorin disk, we look for

an automorphism ψ that permutes a subset of the vertices neighboring this hub, where our goal is to decompose the network's adjacency matrix A over ψ. The issue we run into is that if ψ fixes the vertex v_{PR} then, as is shown in the proof of Theorem 3.25, the Gershgorin disk associated with v_{PR} will not decrease in size as A is decomposed over ψ. Consequently there will be no improvement in the Gershgorin region.

To actually improve this Gershgorin estimate, we note that finding the eigenvalues of a matrix is equivalent to finding the eigenvalues of its transpose. Thus, making Gershgorin regions from columns instead of rows, i.e. a *transposed Gershgorin region* $\Gamma(M^T)$ of a matrix M could potentially improve the Gershgorin region, even when the region is dominated by the fixed portion of the graph. One complication is that we are not guaranteed the Gershgorin region formed from the *transpose* of the decomposed matrix columns will be contained in the Gershgorin region of the original matrix. Thus, this method of using a matrix' transposed Gershgorin region for gaining improved eigenvalue estimates may not be effective for all networks.

In this example we are, in fact, able to decrease the transposed Gershgorin region associated with the network by decomposing the matrix A over certain automorphisms where points in non-trivial orbits are adjacent to v_{PR}. In fact, we are able to find a number of automorphisms that allow us to sequentially decompose the matrix A such that at each step we gain an improved estimate of the network's eigenvalues.

This process results in a significant improvement in the original (transposed) Gershgorin estimate of the network's eigenvalues. This improvement is shown in Figure 3.10 (right), where each disk starting from the largest and moving inward represents the next transposed Gershgorin region after the corresponding equitable decomposition is performed on the network. After 4 equitable decompositions, the final, smallest region is the union of two disks one centered at 5 with radius 150 and one centered at 1 with radius 148, which is roughly thirty-seven percent the size of the original Gershgorin region associated with the network.

3.8 Graphical Realization of Equitable Decompositions

Carrying out an equitable decomposition is much more visual on a graph than on an associated matrix, and the (tedious) ordering of the labels is trivial when working with graphs instead of matrices. To illustrate the process of an equitable decomposition of a graph we introduce the notion of a folded graph. A folded graph can be thought of as a graph G in which we have folded together all the vertices that are in the same orbit under an automorphism of G. The resulting graph can be used to generate a number of smaller graphs $G_\phi, G_1, \ldots, G_{k-1}$ that correspond to an equitable decomposition $M_\phi \oplus B_1 \oplus \cdots \oplus B_{k-1}$ of a matrix M associated with G.

To describe this procedure of folding a graph we first note that a graph G can be either weighted or unweighted. If it is unweighted it is possible to weight the graph's edges by giving each edge unit weight. Under this convention any graph G can be considered to be a weighted graph with weighted adjacency matrix $W = W(G)$.

For simplicity, we assume in this section that the eigenvalues associated with a graph G are the eigenvalues of the graph's weighted adjacency matrix, which we denote by $\sigma(G)$. This can be done without loss in generality since any automorphism compatible matrix M associated with G is the weighted adjacency matrix of a graph H where $Aut(G) = Aut(H)$.

By extending Theorem 3.4 to graphs, an equitable decomposition of a graph G over a basic automorphism ϕ results in a number of smaller matrices $G_0, G_1 \ldots G_{k-1}$ where

$$\sigma(G) = \sigma(G_0) \cup \sigma(G_1) \cdots \cup \sigma(G_{k-1})$$

where $W(G_i) = B_i$ is given by Equation (3.2). Here we describe how the graphs $G_\phi, G_1 \ldots G_{k-1}$ can be generated from a single (folded) weighted graph $\mathcal{G}_\phi(m)$. To show how this is done we let $G = (V, E, w)$ denote the (weighted) graph G with vertices $V = V(G)$ and edges $E = E(G)$. The function $w : E \to \mathbb{C}$ gives the weight of each edge (i, j) in E. The following steps allow one to equitably decompose a graph.

Step 1: *The basic automorphism ϕ partitions the graph's vertices into $V = V_1 \cup \cdots \cup V_r \cup U$, where U is the union of the vertices of orbit size 1. Choose a semi-transversal $\mathcal{T}_0$ of the orbits of ϕ, i.e. the index of one vertex from each set V_i. If this is not the last round, this semi-transversal must be chosen according to the method set out in Proposition 3.15.*

Step 2: *From the graph $G = (V, E, w)$ we construct the folded graph $\mathcal{G}_\phi(m) = (\mathcal{V}_m, \mathcal{E}_m, \nu_m)$ as follows. The vertices of the graph are labeled by $\mathcal{V}_m = \{\phi^m(i) : i \in \mathcal{T}_0 \cup \mathcal{U}\}$. The edge weights of $\mathcal{G}_\phi(m)$ are given by the formula*

$$\nu_m(\phi^m(i), \phi^m(j)) = \begin{cases} \sum_{\ell=0}^{k-1} \omega^{\ell m} w(i, \phi^\ell(j)) & \text{if } j \in \mathcal{T}_0 \\ w(i, j) & \text{if } j \in U. \end{cases}$$

Step 3: *Next we generate the k graphs $G_m = \mathcal{G}_\phi(m)$ for $m = 0, \ldots, k-1$ where $\mathcal{G}_\phi(0) = G_\phi$. If $m = 0$ then $\mathcal{V}_0 = \mathcal{T}_0 \cup U$, otherwise $\mathcal{V}_m = \mathcal{T}_m$. (When drawing $\mathcal{G}_\phi(m)$ we draw the vertices in $\mathcal{U}$ as open circles to indicate that these vertices are only in the graph for $m = 0$, see Figure 3.11).*

If ϕ is a uniform automorphism of G then G can be equitably decomposed over ϕ where steps 1–3 are adjusted such that $\mathcal{U} = \emptyset$. If ϕ is separable then using the method described for decomposing a matrix with respect to a separable automorphism in Section 3.3 we can sequentially decompose G over some sequence of basic automorphisms $\psi_0, \psi_1, \ldots, \psi_h$ associated with ϕ.

One of the main advantages of visualizing equitable decompositions in terms of a folded graph as opposed to the matrix procedure prescribed in Section 3.3, is that it makes the blocks of the decomposition immediately apparent. For instance, recall that in Example 3.8, at the end of Round 2 we were required to reorder the vertices (equivalent to relabeling the graph) in order to see the block diagonal structure. In the graphical approach presented in this section the connected subgraphs are the blocks resulting from the decomposition.

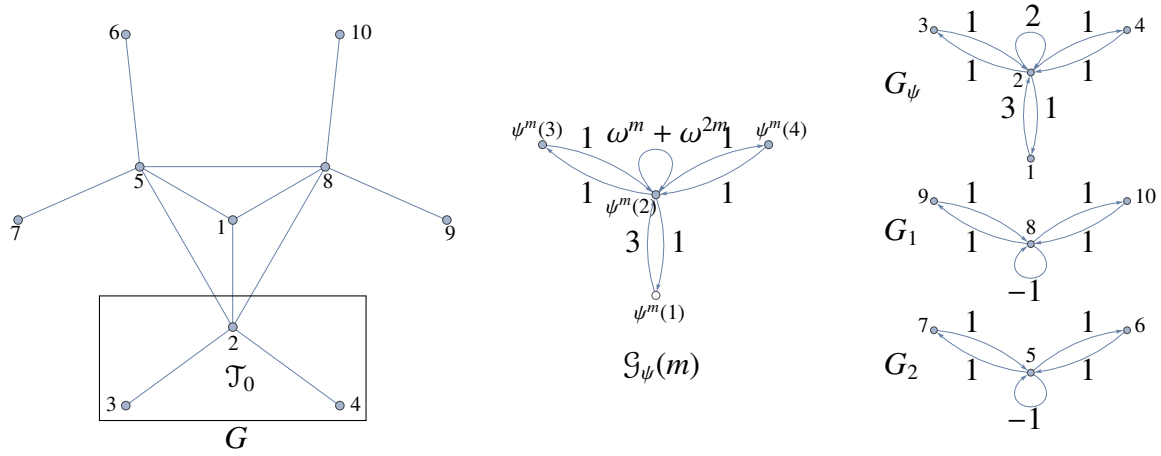


Figure 3.11: The folded graph $\mathcal{G}_\psi(m)$ (center) of the unweighted graph G (left) with basic automorphism $\psi = (2, 8, 5)(3, 9, 7)(4, 10, 6)$ created using the semi-transversal $\mathcal{T}_0 = \{2, 3, 4\}$. The equitable decomposition G_ψ, G_1, G_2 of G is shown (right) where $G_\psi = \mathcal{G}_\psi(0)$ and $G_m = \mathcal{G}_\psi(m)$ for $m = 1, 2$. Here $\omega = e^{2\pi i/3}$.

Example 3.28. Let G be the graph originally introduced in Figure 3.1, which is also shown in Figure 3.11 (left), with the basic automorphism $\psi = (2, 8, 5)(3, 9, 7)(4, 10, 6)$. Following steps 1–3 given in this section, we first choose the semi-transversal $\mathcal{T}_0 = \{2, 3, 4\}$ of ψ. The folded graph $\mathcal{G}_\psi(m)$ that results from this choice is shown in Figure 3.11 (center). The graphs G_ψ, G_1, G_2, which together form an equitable decomposition of the graph G over ψ are also shown in the same figure (right). One can check that $\sigma(G) = \sigma(G_\psi) \cup \sigma(G_1) \cup \sigma(G_2)$ since the matrices $W(G_\psi) = A(0)_{\psi_0}$, $W(G_1) = B(0)_1$, and $W(G_2) = B(0)_2$ given in Equation (3.6).

3.9 Conclusion

The purpose of this chapter is to extend the theory of equitable decomposition as well as to introduce a number of its applications. The theory of equitable decompositions, first presented in [4], describes how an automorphism compatible matrix M can be decomposed over any *uniform* or *basic* automorphism of an associated graph. In this chapter we first extend this result by providing a method for equitably decomposing M over any separable automorphism ϕ by converting any such automorphism into a sequence of basic automorphisms. After establishing the method for decomposing matrices over separable automorphisms, we then further extended this method to equitably

decompose over any automorphism. This further extension required us to describe a more sophisticated way to decompose over a prime-power automorphism than what was described in Section 3.3. These decompositions result in a number of smaller matrices $M_\phi, B_1, \ldots, B_h$ whose collective eigenvalues are the eigenvalues of the original matrix M (see Theorem 3.4, Proposition 3.7 and Theorem 3.16). Importantly, this decomposition relies only on a knowledge of the automorphism ϕ and requires no information regarding any spectral properties of the graph.

Additionally, the algorithms we give for equitably decomposing a graph depends both on the order of the prime factorization of the automorphism's order as well as some choices surrounding the semi-transversals. An open question is whether making these choices differently will result in a fundamentally different decomposition (i.e. not just a simple reordering of resulting block matrices) or if, up to reordering, the decomposition is, in fact, unique.

Beyond preserving the eigenvalues of a matrix, we also show as a direct application of this theory that the eigenvectors of the matrix M are fundamentally related to the eigenvectors of the matrices $M_\phi, B_1, \ldots, B_k$ and give an explicit formula for their construction (see Theorem 3.18). This theorem also extends to generalized eigenvectors in the case where the original matrix does not have a full set of eigenvectors. In this way the concept of an equitable decomposition can be applied not only to matrices but also to their eigenvectors.

A theme throughout this chapter, and in particular in the applications introduced here, is that information regarding spectral properties of a graph can be deduced from knowledge of the graph's local symmetries. One example, shown in Proposition 3.21, is that if M is nonnegative and irreducible then the divisor matrix M_ϕ associated with the automorphism ϕ has the same spectral radius as M. Since M_ϕ is smaller than M, and potentially much smaller if the graph (network) is highly symmetric, then M_ϕ could be a useful tool in determining this spectral radius. It is an open question whether this result can be extended to matrices with negative or complex-valued entries.

Another application mentioned here is concerned with the way in which Gershgorin regions are affected by equitable decompositions. Here, we prove that the Gershgorin region of an equitable decomposition is contained in the Gershgorin region of the original matrix (see Theorem 3.25).

Thus, such decompositions can be used to potentially reduce the size of the resulting Gershgorin region. In particular, Example 3.27 demonstrates this by significantly reducing the size of the Gershgorin region associated with a large network via a series of decompositions.

Lastly, we introduce the analogue of an equitable decomposition for graphs. An equitable graph decomposition, which is built around the notion of a folded graph, has the advantage that it is more visual than an equitable decomposition on a matrix. Moreover, performing a graph decomposition does not require any reordering of the graph's vertices as opposed to the row and column reordering that is required in an equitable decomposition of a matrix. The method of "graph folding" introduced in this section only applies to basic (or separable) automorphism and it is still an open question to understand how this method can be extended to general automorphisms.

Chapter 4. Hidden Symmetries in Real and Theoretical Networks

4.1 Background and Overview

As we discussed in Section 2.1, real networks have a number of properties that distinguish them from many other graphs of interest. For instance, they tend to have right-skewed degree distributions, high clustering coefficients, and the "small-world" property, etc. [6]. In addition to these properties, real networks generally contain a significant number of symmetries [54]. Consequently, many real networks have a large symmetry group [5]. It is important to study these symmetry groups for a number of reasons. First, understanding network symmetry helps us better understand the formation of particular networks [55]. Symmetries can also provide information about vertex function. For instance, it has been observed that two symmetric vertices can play the same role in a network, which is thought to increase network robustness [56]. In the case of networks' dynamics and function, symmetries are known to be important to the processes of synchronization or partial synchronization [57].

Beyond these standard symmetries, "near" symmetries also naturally occur in real networks

[55]. Even though these approximate symmetries are not represented in the symmetry group of the network, they still have an effect on network behavior, both in form and in function [55]. For this reason, there has been a number of attempts to weaken the notion of structural symmetry. By *structural symmetry*, we mean there exists a permutation of the network's vertices that does not change the network structure. A near symmetry can be described in terms of properties invariant under some other network transformation. As an example the network's vertex degree could be maintained as the network's topology is transformed [58].

Notions of stochastic symmetry have also been established [55] to characterize near symmetries in real networks. In this framework one chooses a statistical ensemble of networks which are similar but not exactly identical and assigns a probability measure to them. This allows one to quantify approximate symmetry and associate characteristics of similar networks. This weakening of the notion of symmetry has led to the study of symmetry groupoids, which can be used to create synchronization in dynamical networks [59], [60], [61].

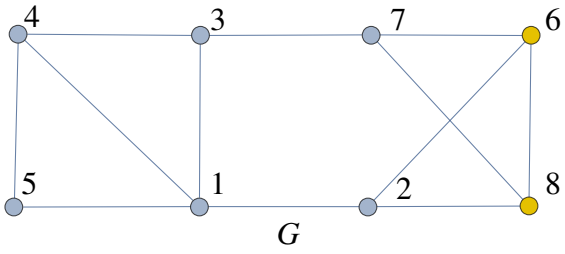
In this chapter we propose a very different extension of the notion of symmetry called *latent symmetry*. Latent symmetries are derived from structural symmetry in a particular reduced version of the network. There are many ways to reduce a network, such as removing edges in a specified way or collapsing chosen subnetworks into single vertices to "coarse grain" the network (for instance, see [62]). When finding latent symmetries, we do not use any of these techniques but use the *isospectral graph reduction* [7] to reduce the size of the network as discussed in Section 2.5. That is, a latent symmetry is a standard structural symmetry in an isospectral reduction of the network. This specific method is chosen because it preserves the spectral properties of a network, i.e. the eigenvalues and eigenvectors associated with the network. The motivation for using a isospectral reduction is that a network's spectral properties encode various structural characteristics, including graph connectivity, vertex centrality, and importantly symmetry (see [4], [63], [64], [65], among others). Additionally, for dynamical networks, stability and other dynamic properties depend on the spectrum of the network [7], [66].

An important property of latent symmetries that is, to the best of our knowledge, not possessed by any other type of symmetry is that it has a sense of *scale*. That is, we can define a *measure of latency* for any latent symmetry, which one can think of as how deep the symmetry is buried within the network. This is of particular interest since many real networks are known to have a *hierarchical structure* in which statistically significant substructures known as *motifs* are repeated at multiple scales throughout the network [67]. Our findings suggest that not only are motifs to be found at multiple levels in a real network, but also symmetries. Thus one can study a network's hierarchical structure of symmetries to better understand the interplay of network structure and function, in particular a network's multiple levels (scales) of redundancy.

This chapter is organized as follows. In Section 4.2, after describing standard network symmetries, we precisely define a *latent symmetry* and give a number of examples. We then define the *measure of latency* of a latent symmetry. In Section 4.3 we give examples of real-world networks which exhibit latent symmetries. In Section 4.4 we prove various spectral properties regarding latent symmetries, including showing that if two vertices are latently symmetric then they have the same eigenvector centrality i.e. the same importance in the network using a particular metric. In section 4.5 we further argue that latent symmetries have relevance in the real world by showing that they are more likely to occur in networks generated using preferential attachment, rather than networks that are randomly generated, e.g. Erdős-Rényi graphs. This is significant because preferential attachment models capture many characteristics of real networks, while Erdős-Rényi graphs do not [68]. In Section 4.6 we demonstrate another application of latent symmetries by combining the notion with equitable decompositions. In Section 4.7 we prove that two vertices in an undirected network are latently symmetric if and only if they are cospectral vetices. Finally, we make some concluding remarks in Section 4.8.

4.2 Network Symmetries

Throughout this chapter, we will use graphs to describe network topology and use the notation outlined in Section 1.1.



$$M(G) = \begin{pmatrix} 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 \end{pmatrix}$$

Figure 4.1: An example of an unweighted, undirected graph G and its corresponding adjacency matrix $M(G)$. The graph has the symmetry given by the automorphism $\phi = (68)$. The symmetric vertices 6 and 8 are highlighted yellow.

We are most interested in *strongly connected* graphs, meaning for any two vertices in the graph there exists a path in the graph which starts at one vertex and ends at the other. For graphs that are not strongly connected, we often only consider the largest strongly connected component of the graph. Our reasons for doing this are essentially pragmatic since many real-world networks are extremely large consisting of a million or more vertices. However, our theory applies to all networks whether strongly connected or not.

The particular type of structure we consider in this chapter is the notion of a graph symmetry, which can be understood via graph automorphisms. These symmetries have received considerable attention in the literature [5], [55], [69]. Intuitively, a graph automorphism describes how parts of a graph can be interchanged in a way that preserves the graph's overall structure. In this sense these *parts*, i.e., subgraphs, are symmetrical and together constitute a graph symmetry. For example, consider the graph in Figure 4.1. Here, it is easy to visually identify the symmetry between the yellow vertices 6 and 8, since transposing them would not change the graph's structure. Formally, a *graph automorphism* was defined in Definition 3.1. In the case of an unweighted graph, this definition is equivalent to saying vertices i and j are adjacent in G if and only if $\phi(i)$ and $\phi(j)$ are adjacent in G. A collection S of vertices in V are *symmetric* if for any two elements a, b in S there exists an automorphism ϕ of G such that $\phi(a) = b$. As an example, the vertices 6 and 8 in Figure 4.1 are symmetric since the permutation ϕ that transposes 6 and 8 and fixes all other vertices of G (written in permutation cycle notation as $\phi = (68)$), is an automorphism of G.

Structural symmetries in networks are important as they can provide information about network robustness as well as the function of specific vertices [56]. Often some of the same type of information can be extracted from a set of vertices which are "nearly" symmetric. There are a number of ways which have been proposed to precisely define a "near" symmetry [55]. Our method involves finding structural symmetries in a reduced version of the network. The network is reduced in a way that preserves spectral properties of the network's adjacency matrix. We are interested in preserving the spectrum of the network since there are a number of important network characteristics which can be determined from its spectrum.

To make this idea more precise, we need the notion of an *isospectral graph reduction*, which is the method we will use to reduce the underlying graph structure of a network. This method was introduced in Section 2.5 and defined in Definition 2.12. Recall that an isospectral graph reduction is a graph operation which produces a smaller graph with essentially the same set of eigenvalues as the original graph. The matrix formula was given by

$$\mathcal{R}_S(M) = M_{SS} - M_{S\bar{S}}(M_{\bar{S}\bar{S}} - \lambda I)^{-1} M_{\bar{S}S} \in \mathbb{W}^{|S|\times|S|}.$$

This method for reducing the graph associated with a network can be formulated both for the graph and equivalently for the adjacency matrix associated with the network, i.e. an *isospectral graph reduction* and an *isospectral matrix reduction*, respectively. Both types of reductions will be useful to us.

Also recall that the eigenvalues of the matrix $M = M(\lambda) \in \mathbb{W}$ are defined to be solutions of the *characteristic equation*

$$\det(M(\lambda) - \lambda I) = 0,$$

which is an extension of the standard definition of the eigenvalues for a matrix with complex entries.

An important aspect of an isospectral reduction is that the eigenvalues of the matrix M and the eigenvalues of its isospectral reduction $\mathcal{R}_S(M)$ are essentially the same, as described by the following theorem [7].

Theorem 4.1. ***(Spectrum of Isospectral Reductions)*** *For $M \in \mathbb{R}^{n\times n}$ and a proper subset $S \subseteq N$, the eigenvalues of the isospectral reduction $\mathcal{R}_S(M)$ are*

$$\sigma(\mathcal{R}_S(M)) = \sigma(M) - \sigma(M_{\bar{S}\bar{S}}).$$

That is, when a matrix M is isospectrally reduced over a set S, the set of eigenvalues of the resulting matrix is the same as the set of eigenvalues of the original matrix M after removing any elements which are eigenvalues of the submatrix $M_{\bar{S}\bar{S}}$.

Phrased in terms of graphs, if the graph $G = (V, E, \omega)$ with adjacency matrix M is isospectrally reduced over some proper subset of its vertices $S \subseteq V$ then the result is the reduced graph $\mathcal{R}_S(G) = (S, \mathcal{E}, \mu)$ with adjacency matrix $\mathcal{R}_S(M)$. Hence,

$$\sigma(\mathcal{R}_S(G)) = \sigma(G) - \sigma(G|\bar{S}).$$

where eigenvalues of a graph are the eigenvalues of a graph's adjacency matrix and where $G|\bar{S}$ denotes the subgraph of G restricted to the vertices not contained in S. It is worth noting that the matrix M and the submatrix $M_{\bar{S}\bar{S}}$ often have no eigenvalues in common, in which case the spectrum is unchanged by the reduction, i.e. $\sigma(\mathcal{R}_S(M)) = \sigma(M)$.

Using isospectral reductions we can define a generalization of the notion of a graph symmetry.

Definition 4.2. (Latent Symmetries) We say a graph G has a *latent symmetry* if there exists a subset of vertices which are symmetric in some isospectral reduction $\mathcal{R}_S(G)$ of G.

It is also worth mentioning here that the definition for latent symmetries works for both directed and undirected graphs. The reason we refer to such symmetries as latent symmetries is that they are difficult to see before the graph reduction is performed, thus they are in some sense *hidden* within the network. For the remainder of the chapter, structural symmetries as defined in Definition 3.1 will be referred to as *standard symmetries* to distinguish them from the latent symmetries defined in Definition 4.2. We note here that technically standard symmetries are a subset of latent

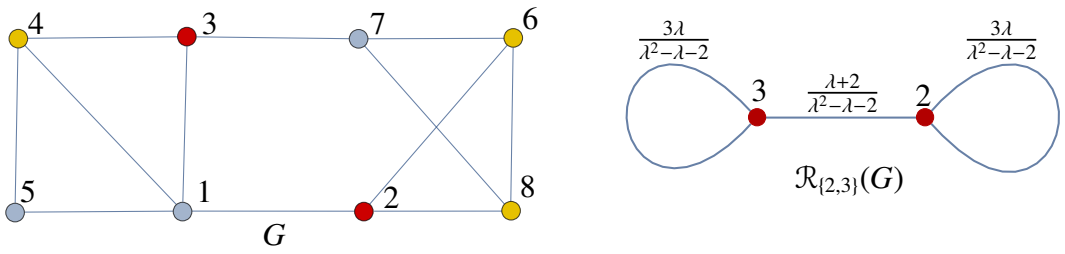


Figure 4.2: (Left) The undirected graph G from Figure 4.1 which has both standard and latent symmetries. Red vertices 2 and 3 are latently symmetric. Yellow vertices 6 and 8 have a standard symmetry between them, but 4 is latently symmetric to both of 6 and 8. (Right) The isospectral reduction of the top graph over vertices 2 and 3, showing the latent symmetry between these two vertices

symmetries since reducing a graph $G = (V, E, \omega)$ over its entire set of vertices preserves the graph, i.e. $\mathcal{R}_V(G) = G$. However, throughout this chapter we will often abuse this terminology by using the term 'latent symmetry' to mean 'latent symmetry which is not also a standard symmetry.'

Example 4.3. The graph in Figure 4.2 is an example of a graph with both standard and latent symmetries. In this figure, colors correspond to groups of vertices which are latently symmetric e.g. vertices 2 and 3. We note that the yellow vertices 6 and 8 form a standard graph symmetry since transposing the two vertices, (i.e. switching their labels), does not change the graph structure of the graph G. When G is reduced over vertices 4 and 6 (or 4 and 8), the resulting reduced graph contains a standard symmetry, i.e. 4 and 6 (4 and 8) are latently symmetric. Also reducing G over the red vertices 2 and 3 results in a graph with symmetry as shown on the right of Figure 4.2.

Some properties of standard symmetries extend to latent symmetries (see, for example, the results in section 4.4). Like standard symmetries, latent symmetries are *transitive*. By this we mean that if there exists a latent symmetry between vertices a and b and a latent symmetry between the vertices b and c in a graph, there must be a latent symmetry between vertices a and c. We note, however, in this scenario there is no guarantee that there exists a subset of vertices S such that a, b, and c are all symmetric in $\mathcal{R}_S(G)$, i.e. a, b and c may not be latently symmetric *as a set*. This is in contrast to standard symmetries where for any set of vertices that are pairwise symmetric, there must exist an automorphism for which all these vertices lie in the same orbit i.e. they are all

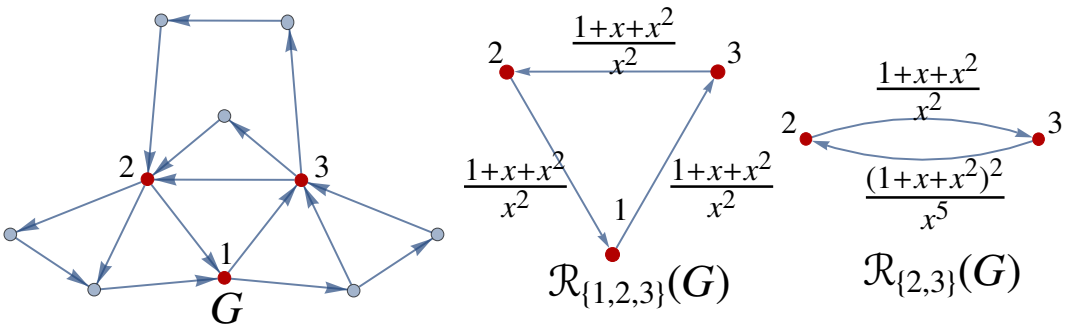


Figure 4.3: In the graph G (left) the vertices labeled 1,2, and 3 are latently symmetric. This is apparent by the three-fold symmetry in $\mathcal{R}_{\{1,2,3\}}(G)$ (middle). However, there is no symmetry in $\mathcal{R}_{\{2,3\}}(G)$ (right).

symmetric as a set. In the setting of standard symmetries this is proved by noting the composition of two automorphisms is an automorphism.

In the above example, as well as in most examples throughout the paper, we reduce the graph to two vertices before discovering the latent symmetry, i.e. the automorphism in the reduced graph is a simple transposition. In general, we may actually find larger groups of vertices with are symmetric after reducing the graph. In a directed graph, it is actually possible to "miss" a symmetry between a set of vertices by reducing too far as is shown in the following example.

Example 4.4. Consider the directed graph in Figure 4.3 (left). When the graph is reduced over the vertices 1,2, and 3 we get Figure 4.3 (middle), which has a symmetry between each of these three vertices. However, if we reduce G over just vertices 1 and 2, we get the graph in Figure 4.3 (right), which has no symmetry. Vertices 1,2 and 3 are latently symmetric as a set, but this symmetry is missed when the graph is reduced to any two of these vertices.

The following is an example of a directed graph with a latent symmetry, which we include here to give the reader some intuition for how latent symmetries can arise. For this we note that a *path* in a directed graph $G = (V, E, \omega)$ is a sequence of distinct vertices, $\{v_0, v_1, \ldots, v_n\} \subseteq V$ such that the set of directed edges $\{(v_i, v_{i+1}) \mid 0 \leq i \leq n-1\}$ is contained in E and a *cycle* is a path for which $v_0 = v_n$.

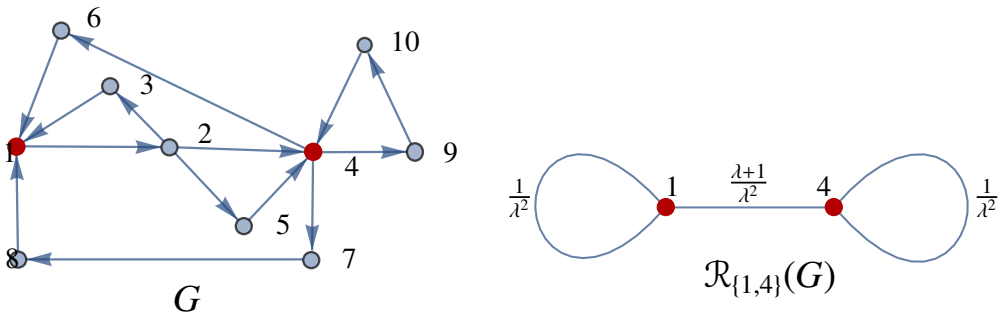


Figure 4.4: In the graph G (left) vertices 1 and 4 are latently symmetric as can be seen by reducing G to $\mathcal{R}_{\{1,4\}}(G)$ (right).

Example 4.5. Consider the directed graph G on the left of Figure 4.4. Because all cycles in this graph contain one of the red vertices (labeled 1 and 4), we can write a finite list of paths and cycles that both begin and end on these vertices. We observe that the paths from 1 to 4 are {1,2,4} and {1,2,5,4}, while all paths from 4 to 1 are {4,6,1} and {4,7,8,1}. Thus, there are the same number of paths with the same lengths from 1 to 4 as there are from 4 to 1. Also there is only one cycle from vertex 1 to itself (not including 4); namely {1,2,3,1}, and only one cycle from vertex 4 to itself (not including 1); namely {4,9,10,4}. Both of these cycles have length 4. This symmetry in number and length of paths and cycles guarantees that a symmetry will appear after reducing the graph over these two vertices. That is, vertices 1 and 4 are latently symmetric. This can be seen in the reduced graph $\mathcal{R}_{\{1,4\}}(G)$ in which there is an automorphism between these two vertices (transposition).

When a graph contains cycles which do not contain any vertices in the reducing set S, it is not possible to write down all the paths and cycles as we did for the graph G. In this case it becomes much more difficult to construct and identify latent symmetries. This is the case for even small undirected graphs. To highlight the unintuitive nature of latent symmetry in this case as well as give a sense of the variety of latent symmetries that occur in undirected graphs, Figure 4.5 depicts six more graphs with examples of latent symmetries. It is worth mentioning that by an exhaustive search we have found the smallest undirected graphs which contain a latent symmetry which is also not a standard symmetry have eight vertices (e.g. Figure 4.5, top left).

Another useful concept we can explore regarding latent symmetries is the scale at which the symmetry is found within the network.

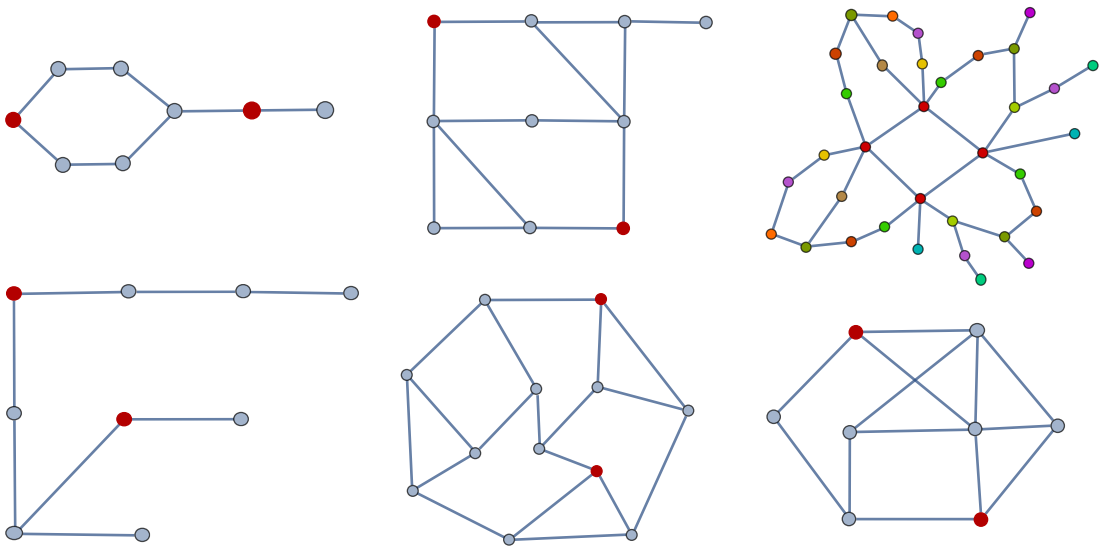


Figure 4.5: Six examples of undirected graphs which contain latent symmetries. Vertices of the same color in a graph correspond to latent symmetries. (Top left) A smallest example of a graph with a latent symmetry but no standard symmetries (fewest edges and vertices), (Top middle) a graph which is symmetric without the pendant vertex on top, which turns the standard symmetry into a latent symmetry, (Top right) a graph where every vertex is latently symmetric to at least one other, though there are no standard symmetries, (Bottom left) a tree, (Bottom middle) a 3-regular graph, (Bottom right) a non-planar graph.

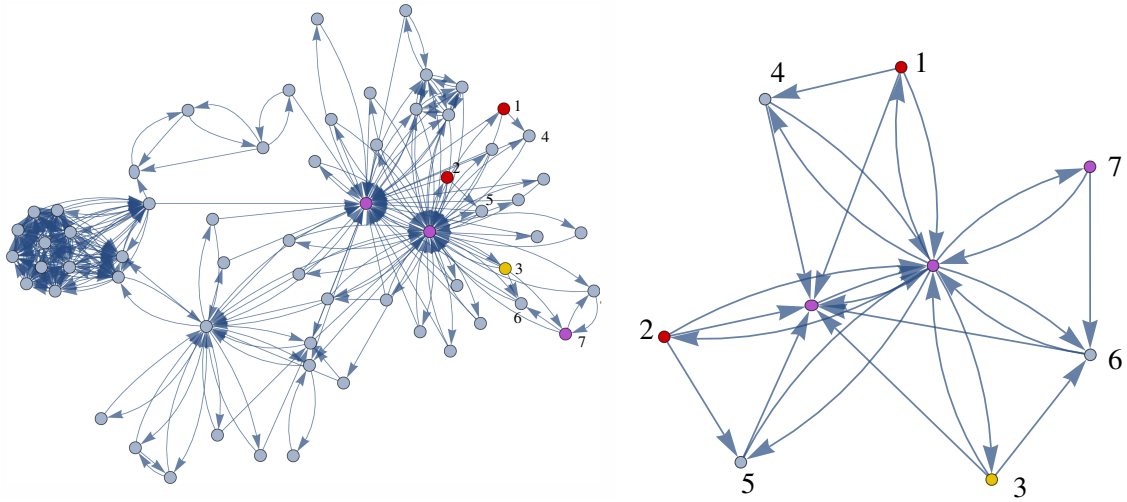


Figure 4.6: (Left) A network representation of the largest strongly connected component of all Wikipedia webpages in the "logic puzzle" category [70]. Vertices represent webpages while the direct edges represent hyperlinks between them. (Right) A subgraph of this network. Red vertices are symmetric and the yellow vertex is latently symmetric with the two red vertices. Purple vertices are vertices which have edges that are not displayed.

Definition 4.6. (Measure of Latency) Let $G = (V, E, \omega)$ be a graph with n vertices and let S be a subset of its vertices which are latently symmetric. This latent symmetry can be said to have a *measure of latency* $\mathcal{M}$, defined as

$$\mathcal{M}(S) = \frac{n - |T|}{n - |S|}$$

where $T \subseteq V$ is a maximal set of vertices such that the vertices S are symmetric in $\mathcal{R}_T(G)$.

From this definition it is clear that $0 \leq \mathcal{M}(S) \leq 1$ since $|T| \geq |S|$. Moreover, if the vertices S are symmetric in the unreduced graph, i.e. are symmetric in the standard sense, then $T = S$ and $\mathcal{M}(S) = 0$. On the other hand, if there is no possible choice of a vertex set T for which $\mathcal{R}_T(G)$ has a symmetry between the vertices in S, except for $S = T$, then $\mathcal{M}(S) = 1$. This is the most hidden a latent symmetry can be since it requires reducing the entire graph to the set S before the symmetry can be seen. We note that although this is an interesting measure, it can be computationally difficult to find since it requires finding the largest possible reducing set under which a symmetry forms.

Example 4.7. For the graph G in Figure 4.2 we have shown that vertices 2 and 3 are latently symmetric as they are symmetric in the reduced graph $\mathcal{R}_{\{2,3\}}(G)$. However, we actually do not

need to reduce to such a small graph to see this symmetry. In fact, 2 and 3 are symmetric in the graph $\mathcal{R}_{\{2,3,4,8\}}(G)$, but are not symmetric in any reduction over five or more vertices. Thus the largest reducing set T in which this symmetry appears must contain four elements. Thus, $\mathfrak{M}(\{2,3\}) = \frac{n-|T|}{n-|S|} = \frac{8-4}{8-2} = 2/3$.

The measure of latency we give to a network symmetry gives the symmetry a size or a scale within the network. This is reminiscent of one of the hallmarks of real networks in which specific structures, known as *motifs*, are found at multiple scales within the network [18], [20]. In the following section we investigate whether latent symmetries occur in real-world networks and specifically whether these occur at different scales.

4.3 Latent Symmetries in Real Networks

The first question one might have concerning latent symmetries is the extent to which they are actually observed in real network data. In this section we present two very different real-world networks which contain latently symmetric vertices.

Example 4.8. Consider the web graph shown in Figure 4.6 (left) which represents all Wikipedia pages contained in the category *"Logic Puzzles"* in August 2017. Each vertex represents a webpage and directed edges represent hyperlinks between webpages [70]. The two red vertices are symmetric in this graph, while the yellow vertex is pairwise latently symmetric with the two red vertices. Figure 4.6 (right) shows a subgraph of the left graph to more clearly demonstrate the path structure causing the latent symmetry. In this second graph purple vertices do not display all of their connections. All vertices labeled 1-7 represent puzzles published by *"Nikoli."* We can see that puzzles 1, 2 and 3 all have an edge pointing to another puzzle (4, 5 and 6 respectively). However, 6 also has vertex 7 pointing to it. This breaks the symmetry between 3 and 1 (also between 3 and 2). The latent symmetry is still present between 3 and 1 (also between 3 and 2) since the same paths are available for traversing the graph from 1 to 3 or from 3 to 1. Thus the latent symmetry is highlighting a common feature in these three puzzles that a standard symmetry search would overlook.

For the vertices which are latently symmetric one can calculate their measure of latency to be $\mathcal{M}(\{1,3\}) = 1/30 \approx 0.033$.

We mention that in the next section will show that by using the metric of eigenvector centrality, vertices 1 and 3 have the same importance to this network. In fact, any set of vertices that are latently symmetric have the same eigenvector centrality (see Theorem 4.10).

Example 4.9. A second example of a latent symmetry in real-world network data is in the metabolic network for the cellular processes in *Arabidopsis thaliana*, a eukaryotic organism [71]. This is a biological network of chemical reactions, which is a very different type of network than the one considered in Example 4.8. Figure 4.7 (left) shows the largest strongly connected component of this network. Here vertices represent cellular substrates (as well as intermediary states) and edges represent metabolic pathways. The red vertices highlighted in the figure (left) have a latent symmetry between them. The right graph is a subgraph of the left where vertex color is preserved. Yellow vertices represent substrates for which all of their edges from the original network are displayed, while purple vertices do not have all of their edges displayed. We can see that many of these vertices are almost symmetric, meaning many of the vertices appear to have a corresponding vertex with a similar local path structure. It is worth noting that between the red vertices there exists the same number of paths of the same length. As in the previous example, this suggests some kind of structural similarity exists between these vertices that a search for standard symmetry would overlook. These vertices are again very close to being symmetric, which is quantized by their measure of latency $\mathcal{M}(S) \approx .0231$.

Currently it is an open question whether latently symmetric vertices have similar or complementary functions within a network. The answer is likely that both are possible and is presumably network dependent. An important point is that once a latent symmetry has been found, an expert in the field which studies the specific network may be able to better answer these questions.

It is also worth emphasizing that both of the real networks we consider in this section have symmetries at different scales. That is, both have standard and latent symmetries which are not standard. We can think of this as symmetries at multiple levels which leads to what one could refer

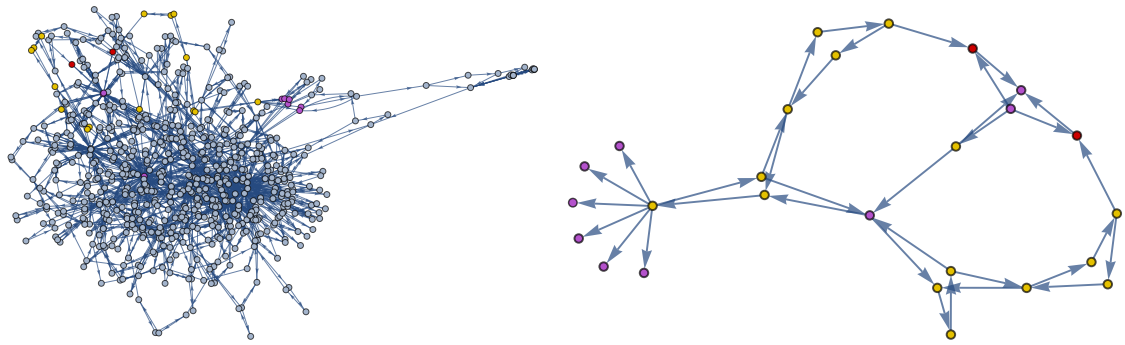


Figure 4.7: Left: Metabolic network of the eukaryotic organism Arabidopsis Thaliana [71]. Right: Subnetwork of left network. In both latently symmetric vertices are colored red. Yellow vertices have been drawn with all their original connections intact whereas purple vertices are missing edges from the original graph.

to as a *hierarchy of symmetries*. It is currently an open question as to how such symmetries might be distributed at various scales through a typical real network.

Going through real-network data, we find it more difficult to find examples of undirected networks than directed networks with latent symmetries. In fact, these symmetries appear to become more rare as the number of vertices in the undirected network gets large. We explore this further using numerical simulations in section 4.5.

4.4 Eigenvector Centrality

In the previous section we presented examples of latent symmetries in real-world networks. We now present evidence to support our claim that latent symmetries capture some type of hidden structure in a network. Specifically we prove that when two vertices are latently symmetric they must have the same *eigenvector centrality*, which is a standard measure of how important a vertex is compared to the other vertices in the network. This result suggests that the notion of a latent symmetry is indeed a natural extension of the standard notion of symmetry since vertex symmetries in the standard sense have the same eigenvector centrality and is therefore an important structural concept that can be used to analyze real networks.

Eigenvector centrality is a widely used metric in network analysis [6]. In fact, it is the basic principle used by "Google" to rank the webpages in the World Wide Web. It is calculated by

ranking the vertices by the value of the corresponding entry in the leading eigenvector of the network's adjacency matrix, where the leading eigenvector is the eigenvector associated with the matrix' largest eigenvalue. Not all graphs have a leading eigenvector. However, essentially all real world networks satisfy the conditions of the Perron-Frobenius theorem which guarantees the existence of a leading eigenvector for the network [6].

To define eigenvector centrality, suppose a network G is represented by a matrix M, and λ_0 is its largest eigenvalue with eigenvector $\mathbf{x}$. The *eigenvector centrality* of a vertex $i \in G$ is the i^{th} entry in $\mathbf{x}$, or x_i.

In order to extend this concept of eigenvector centrality to reduced networks, we first need to extend the notion of eigenvectors to the class of matrices with rational function entries.

If $R(\lambda) = \mathcal{R}_S(M)$ is an $|S| \times |S|$ matrix with rational function entries and $\lambda_0 \in \sigma(M) - \sigma(M_{\bar{S}\bar{S}})$ then $R(\lambda_0)$ is defined, that is each rational function in the matrix $R(\lambda)$ is defined at λ_0. Hence, $R(\lambda_0)$ is simply an $|S| \times |S|$ real-valued matrix [7]. We say the vector $\mathbf{v} \in \mathbb{C}^{n\times 1}$ is an eigenvector corresponding to λ_0 if $(R(\lambda_0) - \lambda_0\mathbf{I})\mathbf{v} = 0$. We note that if the network we are considering is strongly connected with non-negative edge weights then $R(\lambda_0)$ is always defined and $\lambda_0 = \rho(M)$, where $\rho(M) = \max\{|\lambda| : \lambda \in \sigma(M)\}$ is the *spectral radius* of M [40].

It is easy to show that vertices always have the same eigenvector centrality if there is a standard symmetry between them [72]. As previously mentioned, one interesting property of latent symmetries is that if two vertices in a network are latently symmetric, then they will also have the same eigenvector centrality. That is, using the metric of eigenvector centrality, latently symmetric vertices have the same importance in the network. This suggests that latent symmetries reveal something as important as the presence of a standard symmetry in the underlying structure of a network.

In order to show that two latently symmetric vertices have the same eigenvector centrality, we reduce the graph over these vertices to create a graph that contains a symmetry. The leading eigenvector of the reduced graph can then be related to the eigenvectors of the original graph using the following result.

Theorem 4.10. ***(Eignvector Centrality and Latent Symmetries)*** *Let $G = (V, E, \omega)$ be a graph (directed or undirected) with nonnegative edge weights which is strongly connected. If there exists a set $L \subseteq V$ of vertices that are latently symmetric, then these vertices all have the same eigenvector centrality.*

Before we prove this theorem, we need the following lemma.

Lemma 4.11. *If B is a nonzero square submatrix of A where A is nonnegative and irreducible, then $\rho(B) < \rho(A)$.*

Proof. Let A be an irreducible, nonnegative, $n \times n$ matrix with square submatrix B. Now we can write A as $A = A_1 + A_2$ where A_1 is the matrix A with all entries corresponding to B set to zero. Now let $C_\epsilon = \epsilon A_1 + A_2$ and $D_\epsilon = (1 - \epsilon)A_1$ for some $0 < \epsilon < 1$. Thus $A = C_\epsilon + D_\epsilon$ where C_ϵ and D_ϵ are both nonnegative matrices, and C_ϵ is irreducible (this is inherited from A) and D_ϵ is not zero (since B is not zero). Now according to exercise 8.4.P14 in [40], $\rho(C_\epsilon + D_\epsilon) > \rho(C_\epsilon)$. Next we note that $A_2 = C_0$, where C_0 is C_ϵ with ϵ set to 0. But $C_\epsilon > A_2$ for any ϵ, thus we can use Theorem 8.1.18 in [40] to guarantee that $\rho(C_\epsilon) \geq \rho(A_2)$. Finally we notice that A_2 and B only differ by rows and columns of zeros. Thus, except for zero eigenvalues, A_2 and B share the same spectrum. Therefore, $\rho(A_2) = \rho(B)$. Putting these results together we see that

$$\rho(A) = \rho(C_\epsilon + D_\epsilon) > \rho(C_\epsilon) \geq \rho(A_2) = \rho(B).$$

Thus we have shown that $\rho(A) > \rho(B)$. □

Now the proof of Theorem 4.10 which connects latent symmetries and eigenvector centrality.

Proof. Because G is strongly connected, the Perron-Frobenius Theorem, guarantees that $M = M(G)$ has a largest simple eigenvalue, $\lambda_0 \in \mathbb{R}$, which is equal to the spectral radius $\rho(M)$. Lemma 4.11 guarantees that $\lambda_0 \notin \sigma(M_{\bar{S}\bar{S}})$ for any vertex set S. The eigenvector $\mathbf{v}$ associated to λ_0 is by definition the leading eigenvector of M. Also, the isospectral reduction of a strongly connected graph must also be strongly connected, thus $\mathcal{R}_S(G)|_{\lambda=\rho(M)}$ also has a leading eigenvector.

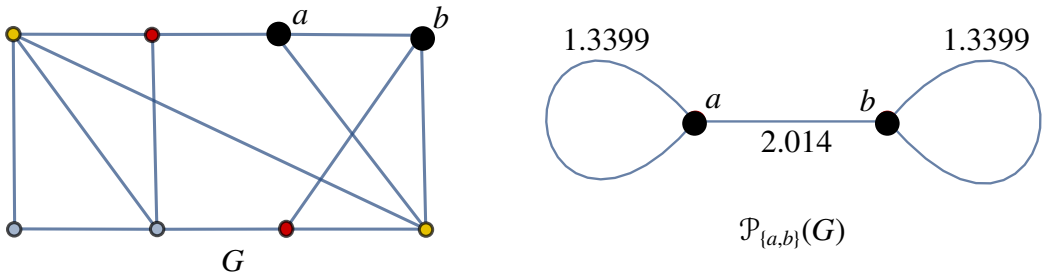


Figure 4.8: (Left) The two large black vertices a and b in the graph G are not latently symmetric, but have the same eigenvector centrality. Red and yellow vertices are represent two latent symmetries in the graph. (Right) The Perron complement graph of G over $\{a, b\}$, $\mathcal{P}_{\{a,b\}}(G)$.

Now recall that a set $L \subseteq V$ of vertices of G is latently symmetric if when G is reduced over a set $S \supseteq L$, the vertices in L are symmetric in the resulting reduced graph. Thus all vertices in L have the same eigenvector centrality in $\mathcal{R}_S(G)$ since symmetric vertices have the same eigenvector centrality, meaning each has the same value in the leading eigenvector, $\mathbf{v}_S$. Next we use Theorem 2.13. Since symmetric vertices correspond to entries in $\mathbf{v}_S$ with the same value, they must also correspond to equal entries in $\mathbf{v}$, the leading eigenvector for the original matrix. This is because $\mathbf{v}_S$ is simply a projection of $\mathbf{v}$. Thus vertices which are latently symmetric must have the same eigenvector centrality. □

If two vertices are latently symmetric then by Theorem 4.10, both vertices have the same eigenvector centrality. However, it is worth emphasizing that even though being latently symmetric is a sufficient condition for having the same eigenvector centrality, it is *not* a necessary condition. For example, consider the graph in Figure 4.8. The two large black vertices have the same eigenvector centrality, although these vertices are *not* latently symmetric. We can, however, strengthen the conclusion of Theorem 4.10 for the case of *undirected* graphs. We do this by searching for symmetry in the graph associated with the *Perron complement* of the networks adjacency matrix (as opposed to the isospectral reduction of the network which was defined in section 2.5). The Perron complement of a matrix is defined as follows.

Definition 4.12. (Perron complement) [73] Let $M \in \mathbb{R}^{n\times n}$ be a nonnegative, irreducible matrix with *spectral radius* $\rho(M)$. If $S \subseteq N$ then the matrix

$$\mathcal{P}_S(M) = M_{SS} - M_{S\bar{S}}(M_{\bar{S}\bar{S}} - \rho(M)I)^{-1}M_{\bar{S}S} \in \mathbb{R}^{|S|\times|S|}$$

is the *Perron complement* of M over S.

Note that $\mathcal{P}_S(M)$ is the isospectral reduction of M over S in which we let $\lambda = \rho(M)$ so that $\mathcal{P}_S(M)$ is a real-valued matrix.

Regarding the Perron complement, the following results hold.

Theorem 4.13. ***(Theorem 2.2 and 3.1 in [73])*** *Let $M \in \mathbb{R}^{n\times n}$ be a nonnegative irreducible matrix and let $S \subseteq N$. Then*
(i) the Perron complement $\mathcal{P}_S(M)$ is also a non-negative and irreducible matrix with the same spectral radius, i.e. $\rho(M) = \rho(\mathcal{P}_S(M))$; and
(ii) if $\mathbf{v}$ is the leading eigenvector of M then its projection $\mathbf{v}_S$ is the leading eigenvector of $\mathcal{P}_S(M)$.

Property (i) in Theorem 4.13 shows that the spectral radius is unaffected by this reduction. Property (ii) is analogous to (and a result of) Theorem 2.13, which tells us that the eigenvectors in the Perron complement are projections of the eigenvectors of the original matrix.

We note here that analogous to isospectral reduction for graphs, we can also define the *Perron complement graph* for a graph G with adjacency matrix M and vertex subset S denoted by $\mathcal{P}_S(G)$, which is the graph whose weighted adjacency matrix is $\mathcal{P}_S(M)$. The Perron complement graph can in some sense be more convenient for analysis since it must have real-valued edge weights, as opposed to the rational function weights which result from isospectral reductions.

However, our interest in the Perron complement in that using the properties of Theorem 4.13, we can give a necessary and sufficient condition under which two vertices of a graph have the same eigenvector centrality.

Theorem 4.14. ***(Eigenvector Centrality in the Perron Complement Graph)*** *Let G be an undirected connected graph. Vertices i, j have the same eigenvector centrality if and only if they are symmetric in $\mathcal{P}_{\{i,j\}}(G)$.*

Proof. Suppose a, b are symmetric in $\mathcal{P}_{\{a,b\}}(G)$; then a, b have the same value in $\mathbf{v}_{\{a,b\}}$, the leading eigenvector of $\mathcal{P}_{\{a,b\}}(A)$. Thus by Theorem 4.13 (ii), a, b must also have the same value in $\mathbf{v}$, the leading eigenvector of A since $\mathbf{v}_{\{a,b\}}$ is just a projection of $\mathbf{v}$. Thus a and b must have the same eigenvector centrality in G.

Now suppose that a and b have the same eigenvector centrality in G. Thus for the unique largest eigenvalue, which in this case is the spectral radius, $\rho(G) = \lambda_0$ with corresponding eigenvector $\mathbf{v}$, we know that $\mathbf{v}_a = \mathbf{v}_b = v$, (where $\mathbf{v}_i$ is the entry in $\mathbf{v}$ corresponding to vertex i). Now using Theorem 2.13, we know that λ_0 is also an eigenvalue of $\mathcal{R}_{\{a,b\}}(G)$ with corresponding eigenvector $\mathbf{v}_{\{a,b\}} = \begin{pmatrix} v \\ v \end{pmatrix}$. Now using the definition of the eigenpair we get

$$\begin{aligned} \mathcal{R}_{\{a,b\}}(A)|_{\lambda=\lambda_0}\mathbf{v}_{\{a,b\}} &= \lambda_0\mathbf{v}_{\{a,b\}} \\ \begin{pmatrix} p_{11}(\lambda_0) & p_{12}(\lambda_0) \\ p_{21}(\lambda_0) & p_{22}(\lambda_0) \end{pmatrix}\begin{pmatrix} v \\ v \end{pmatrix} &= \begin{pmatrix} \lambda_0 v \\ \lambda_0 v \end{pmatrix}. \end{aligned} \tag{4.1}$$

However, because we started with a symmetric matrix (undirected graph) we know that the result of an isospectral matrix reduction is also symmetric (see [7]. Thus $p_{21}(\lambda) = p_{12}(\lambda)$ and

$$\begin{pmatrix} p_{11}(\lambda_0) & p_{12}(\lambda_0) \\ p_{12}(\lambda_0) & p_{22}(\lambda_0) \end{pmatrix}\begin{pmatrix} v \\ v \end{pmatrix} = \begin{pmatrix} \lambda_0 v \\ \lambda_0 v \end{pmatrix}. \tag{4.2}$$

Therefore we conclude that $p_{11}(\lambda_0) = p_{22}(\lambda_0)$ and the reduced matrix $\mathcal{R}_{\{a,b\}}(A)|_{\lambda=\rho(G)}$ has the form

$$\begin{pmatrix} p_{11}(\lambda_0) & p_{12}(\lambda_0) \\ p_{12}(\lambda_0) & p_{11}(\lambda_0) \end{pmatrix}$$

which is symmetric between the two remaining vertices. □

As an example of this theorem the large black vertices a and b in Figure 4.8 (left) have the same eigenvector centrality. In the Perron complement graph, $\mathcal{P}_{\{a,b\}}(G)$, shown in Figure 4.8 (right), we can see a symmetry appears between these two vertices.

Remark. We note here that Theorem 4.14 is stated for two vertices and does not extend to any other set of vertices. Specifically if two vertices of some network G have the same eigenvector centrality then they are symmetric in the Perron complement graph of G over these two vertices. The conclusion does not hold for a larger set of vertices which all have the same eigenvector centrality. By this we mean that if a network G, has a set S of three or more vertices which all have the same eigenvector centrality, then there is no guarantee that $\mathcal{P}_S(G)$ contains a symmetry. We also note here that the above theorem is only true in general for undirected graphs. For instance vertices 2 and 6 for the graph G in Figure 4.4 have the same eigenvector centrality but are not symmetric in $\mathcal{P}_{\{2,3\}}(G)$.

The reason we use isospectral reductions instead of the comparatively simple Perron complement is that by using isospectral reductions we have a smaller class of symmetries. If there is a symmetry in the Perron complement graph $\mathcal{P}_S(G)$ this does not always correspond to a latent symmetry in $\mathcal{R}_S(G)$. That is, the Perron complement cannot be used to find latent symmetries in general. Further, we do not gain any new information regarding network symmetries from the Perron complement graph for directed graphs where Theorem 4.14 does not hold.

4.5 Latent Symmetries and Network Growth Models

Real-world networks are constantly evolving and typically growing (see [14] for a review of the evolving structure of networks). A number of network formation models have been proposed to describe the type of growth observed in these networks. The purpose of this section is to demonstrate that, like standard symmetries, latent symmetries appear to be a hallmark of such networks.

To determine how likely it is to find a latent symmetry in a given network, we perform a number of numerical experiments. In these experiments we count how many latent symmetries occur in

randomly generated graphs. We focus these numerical experiments on directed networks, where latent automorphisms are more likely to occur.

The most well-known class of network growth models are those related to the Barabási-Albert model [74] and its predecessor the Price model [68]. In these models elements are added one by one to a network and are preferentially attached to vertices with high degree, i.e. to vertices with a high number of neighbors or some variant of this process [75, 76, 77]. These models are devised to create networks that exhibit some of the most widely-observed features found in real networks such as scale-free degree distributions. We choose to generate networks using this theoretical model to understand whether latent symmetries are likely to appear in a real-network setting.

We choose a two-parameter growth model which is a variation of the one originally devised by Price (see [6], section 14.1). When generating a graph, we begin with a complete graph on 3 vertices and add vertices and edges iteratively. At each step we add one vertex and two edges. Both of these edges connect to the new vertex on one end. The other end of each new edge attaches to a vertex in the existing graph with a probability proportional to the number of edges already connected to it plus some intrinsic weight α given to each vertex, which is a parameter we vary in our experiment. Vertices that already have many adjacent edges are more likely to attach to the new vertex at each step. The α parameter changes how strongly the new vertices are preferentially attached to vertices with larger degrees. The idea is that as α gets larger the graph is generated in a less preferentially-attached way.

In these experiments, we generate 1000 graphs with 180 vertices at each value of α. We then randomly determined which direction each edge points, where there is a 1/5 probability the edge will point in both directions, otherwise it points in only one direction. We use this to method increases the connectivity of the resulting network. The reason for these probabilities is that this makes each edge half as likely to be directed in both directions compared to being directed in just one direction. That is, an edge is directed in one way with probability 2/5, and the other way with 2/5 probability, and in both ways with probability 1/5. Once a graph is generated, we reduce the graph to its largest strongly connected component since networks are often analyzed at this level.

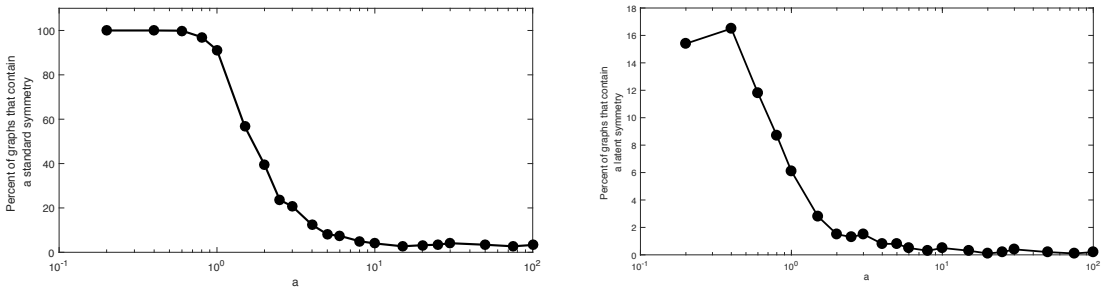


Figure 4.9: The left figure plots the percentage of graphs which were generated using preferential attachment that contain a standard symmetry. The horizontal axis (plotted logarithmically) gives different values of a, the parameter which controls how strongly each edge is attached preferentially. The right is the same figure in which the occurrences of latent symmetries are plotted.

The result is a collection of graphs with a mean number of about 137 vertices. Finally, we count how many of the resulting graphs have at least one standard symmetry and how many have at least one latent symmetry.

Figure 4.9 plots the percentage of graphs generated for each value of α which have a standard symmetry (left) and latent symmetry (right). We first notice that both plots essentially decrease as α increases, demonstrating that as graphs are generated in a way less like preferential attachment, we find less symmetry at any scale. Though there are fewer total latent symmetries than real symmetries, they both follow the same general trend, suggesting the process of creating standard and latent symmetries are correlated. This suggests that mechanisms which allow for a greater number of standard symmetries also allow for the formation of more latent symmetries. Exactly what this mechanism is and how it operates even in these experiments is an interesting and open question. One possibility is that having regions of low edge density among collections of vertices creates an environment where symmetries are more likely to occur randomly. Thus a method which generates a network using preferential attachment concentrates most of the connection around vertices with high degree (hubs), allowing other vertices to have a lower density of edges.

To compare the results of our experiments, we do the same experiment by building graphs via the Erdős-Rényi method [78]. This is a method for generating random graphs where there is a fixed probability α that an edge will exist between any two vertices in the graph. Importantly, it is known that this does not lead to graphs that resemble real-world networks [2]. Starting as

before with 180 vertices, we generate Erdős-Rényi graphs in which there is an edge between every pair of vertices in the graph with probability p. We then follow the same procedure of randomly directing edges and reducing the network to its largest strongly connected component as in the previous experiment where we considered preferential attachment. After generating 1000 graphs for a given value of p we determine what percentage of these graphs have at least one structural symmetry and at least one latent symmetry as before.

The results of this experiments show there is a very narrow range of values for p, between $p = .02$ and $p = .03$, for which we find any symmetries and also generate graphs with a substantial largest connected component. For all values of p we found that 5.9% was the highest percentage of these graphs having a standard symmetry, and 0.3% was the highest percentage of these graphs having a latent symmetry. These numbers are much lower than anything we found using preferential attachment. These experiments were repeated for larger starting vertex numbers and similar results were obtained. This suggests that symmetries, and in particular latent symmetries, are not as likely to be found in nonnetwork-like graphs as they are in graphs that capture some important features of real-world networks.

Remark. We note here that these experiments were only performed on directed networks. We find that latent symmetries are extremely rare in undirected networks generated using either of the above models for a large number of vertices (more than fifteen). Interestingly, when searching for symmetries in small undirected graphs, we were actually able to find more latent symmetries using the Erdős-Rényi model than with preferential attachment. Although most networks are much larger than fifteen vertices, it is not understood why the trend observed for large networks should reverse for small networks.

4.6 Latent Symmetries and Equitable Decompositions

Another potentially useful application of latent automorphisms is that they can be used in conjunction with the equitable decomposition theory discussed in Chapter 3. This provides a way to equi-

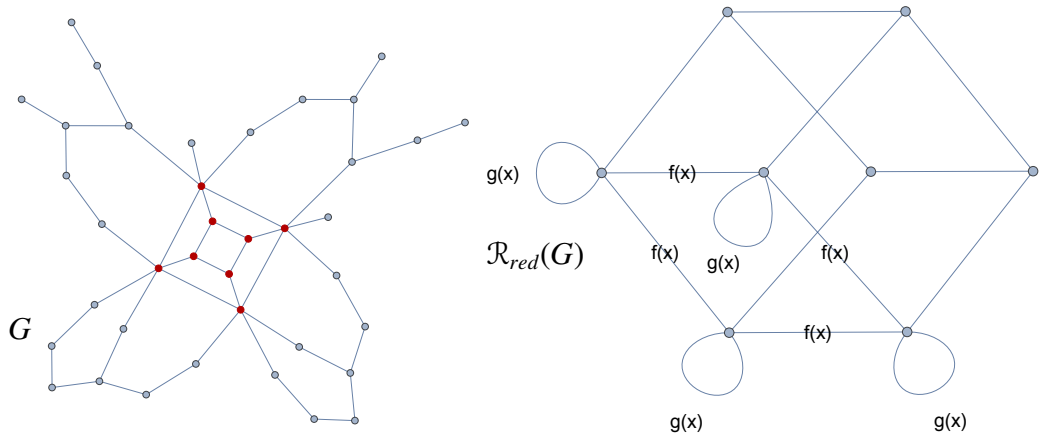


Figure 4.10: (Left) The graph G. (Right) The graph G isospectrally reduced over the red vertices, i.e. $\mathcal{R}_{red}(G)$. The rational functions f and g are given by $f(x) = \frac{x^2-1}{x^2(x^2-3)^2-3} + 1$ and $g(x) = \frac{3x^2(x^4-5x^2+6)-4}{x(x^2(x^2-3)^2-3)}$.

tably decompose networks which only have latent symmetries and no standard symmetries. First we isospectrally reduce the network to create the symmetry, then use the resulting automorphism to decompose the graph. Because both of these steps essentially preserve the graph's spectrum, the collection of the eigenvalues of the resulting small graphs is essentially equal to the spectrum of the original graph.

The only thing we need to verify is that all of the same theory for equitable decompositions still applies to graphs weighted with rational functions. Because the equitable decomposition process only utilizes similarity transformations via complex matrices and permutations, the fact that we use rational functions for entries of the adjacency matrix will have no effect on any of the theory which we have established.

A benefit to using both of these tools is it can simplify the process of finding eigenvalues for a given graph. Because each step preserves the spectrum of the network, we change the eigenvalue problem from solving one large characteristic equation for the graph into solving a series of lower degree characteristic equations for each resulting piece. This method is demonstrated in the following example.

Example 4.15. Consider the graph G in Figure 4.10 (left). The red vertices, are latently symmetric. When G is reduced over the red vertices, the result is $\mathcal{R}_{red}(G)$ in Figure 4.10 (right). Now can we use the theory of equitable decomposition to break this graph around the four-fold symmetry. The result is four graphs whose characteristic equations are:

$$(1)\ -8 - 9x + 50x^2 + 10x^3 - 68x^4 + 6x^5 + 30x^6 - 6x^7 - 4x^8 + x^9 = 0$$

$$(2)\ -4 + 3x + 21x^2 - 9x^3 - 24x^4 + 6x^5 + 9x^6 - x^7 - x^8 = 0$$

$$(3)\ -8 + 15x + 38x^2 - 34x^3 - 32x^4 + 24x^5 + 6x^6 - 8x^7 + x^9 = 0$$

$$(4)\ -4 + 3x + 21x^2 - 9x^3 - 24x^4 + 6x^5 + 9x^6 - x^7 - x^8 = 0.$$

Now the roots of these four polynomials must also be roots of the characteristic polynomial associated the original graph G. Thus by breaking the graph into smaller graphs, we can find roots of the characteristic polynomial of G, which has degree 38, by solving for roots of these smaller polynomials with degree 8 and 9. In this particular example, we do not recover all of the eigenvalues from the original graph because four of them are lost in the isospectral reduction step. We can calculate in this example each of the lost eigenvalues are equal to zero.

This process of reducing a matrix and then decomposing it is especially useful if we are only concerned with finding the spectral radius of a large graph. It is not hard to show that isospectrally reducing a matrix always preserves the spectral radius in the resulting graph. This means that the largest eigenvalue cannot be lost in the reduction process. We showed in Section 3.6 that the spectral radius of an equitable decomposition is always the spectral radius of the divisor matrix. Because both of these steps are guaranteed to preserve the spectral radius, the resulting divisor matrix must always contain the spectral radius. In the above example, finding the spectral radius reduces to finding the largest root of a polynomial with degree 9 instead of a polynomial with degree 38.

4.7 Cospectral Vertices are Latently Automorphic

Given a weighted, undirected graph, two vertices are said to be *cospectral* if the adjacency matrices of $G\backslash a$ and $G\backslash b$ have the same characteristic polynomial (where $G\backslash v$ denotes the graph G after removing vertex v and all of edges adjacent to v). This is an important concept in spectral graph theory [42], and comes up in quantum walks on graphs [79].

In this section we prove that for any weighted undirected graph $G = (V, E, \omega)$, two vertices $a, b \in V$ are *cospectral* if and only if they are latently automorphic. In order to prove this result, we will need the following definition.

Definition 4.16. (Smash Functions)
For a graph $G = (V, E, \omega)$ with a vertex $a \in V$, the *Smash Function of vertex a*, denoted $SF_G(a)$, is the single rational function entry in the matrix which is the result of performing an isospectral reduction on the adjacency matrix $M = M(G)$ over the single vertex a i.e.,

$$SF_G(a) = \mathcal{R}_{\{a\}}(M).$$

In order to prove the following theorem, we will show a number of statements are equivalent. First, we show that if two vertices are latently symmetric, then they must be automorphic when the graph is reduced to only those two vertices. Second, we show two vertices are automorphic when reduced down to just those two vertices if and only if the two vertices have exactly the same smash function. Finally, we show that two vertices have the same smash function if and only if they are cospectral. Thus putting all of these equivalences together proves that being latently symmetric is a necessary and sufficient condition for being cospectral.

Theorem 4.17. *Let $G = (V, E, \omega)$ be a (real or complex weighted) graph (potentially with self-loops). Two vertices $a, b \in V$ are latently automorphic if and only if a and b are cospectral.*

Proof. We will prove the following three equivalences.

$$a \text{ and } b \text{ are latently automorphic in } G \iff a \text{ and } b \text{ are automorphic in } \mathcal{R}_{\{a,b\}}(G) \tag{1}$$

$$a \text{ and } b \text{ are automorphic in } \mathcal{R}_{\{a,b\}}(G) \iff SF_G(a) = SF_G(b) \tag{2}$$

$$SF_G(a) = SF_G(b) \iff a \text{ and } b \text{ are } \textit{cospectral} \tag{3}$$

Proof of (1)

We first assume that a and b are automorphic in $\mathcal{R}_{\{a,b\}}$. By definition, two vertices are latently automorphic if a and b are automorphic in $\mathcal{R}_S(G)$ for some $S \subset V$. Thus this direction is trivial since we can let $S = \{a, b\}$.

Assume a and b are latently automorphic in G and let M be the adjacency matrix of G. Saying that two vertices are automorphic in a graph is equivalent to saying that the associated adjacency matrix is unchanged after permuting the associated rows and columns. We first suppose a and b are automorphic vertices of G, or in other words interchanging the rows and columns of M associated with a and b leaves M unchanged. For notational simplicity, let $T = \{a, b\}$. In order to show that a and b are automorphic in $\mathcal{R}_T(M) \in \mathbb{W}^{2\times 2}$, we must show that $\mathcal{R}_T(G)$ is equal to $\mathcal{R}_T(G)$ after interchanging its rows and columns. We can write this in terms of matrices as

$$\mathcal{R}_T(M) = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \mathcal{R}_T(M) \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}.$$

Now we note that because a and b are automorphic in M then

$$M_{TT} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} M_{TT} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \qquad M_{T\bar{T}} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} M_{T\bar{T}}, \qquad M_{\bar{T}T} = M_{\bar{T}T} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}.$$

Now we use the definition of the *Isospectral Matrix Reduction* given in Definition 2.12, to show

$$\begin{aligned}
\mathcal{R}_T(M) =& M_{TT} - M_{T\bar{T}}(M_{\bar{T}\bar{T}} - \lambda I)^{-1} M_{\bar{T}T} \\
=& \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} M_{TT} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} - \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} M_{T\bar{T}}(M_{\bar{T}\bar{T}} - \lambda I)^{-1} M_{\bar{T}T} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \\
=& \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \left(M_{TT} - M_{T\bar{T}}(M_{\bar{T}\bar{T}} - \lambda I)^{-1} M_{\bar{T}T} \right) \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \\
=& \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \mathcal{R}_T(M) \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}.
\end{aligned}$$

Thus a and b are automorphic in $\mathcal{R}_T(G)$.

Now we assume that a and b are latently automorphic in G. By definition, two vertices are latently automorphic if a and b are automorphic in $\mathcal{R}_S(G)$ for some $S \subset V$. We can apply the above result to the matrix $\mathcal{R}_S(M)$. Suppose a and b are automorphic in $\mathcal{R}_S(M)$, then they are also automorphic in $\mathcal{R}_T(\mathcal{R}_S(M)) = \mathcal{R}_T(M)$ (this last equality is Theorem 1.3 in [7]). Therefore we get the result for the associated graphs for these matrices.

Proof of (2)

Suppose there exists an automorphism between a and b in the reduced graph $\mathcal{R}_{\{a,b\}}(G)$. Then we know $\mathcal{R}_{\{a,b\}}(G)$ has the form in Figure 4.11 and the adjacency matrix has the form

$$\begin{pmatrix} h(\lambda) & f(\lambda) \\ f(\lambda) & h(\lambda) \end{pmatrix}.$$

Now we calculate $SF(a)$ using Theorem 2.1 in [7]. This calculation is the same for both vertices,

$$SF(a) = \frac{f^2(\lambda)}{\lambda - h(\lambda)} + h(\lambda) = SF(b).$$

Thus we have shown that the smash functions for both vertices are equal.

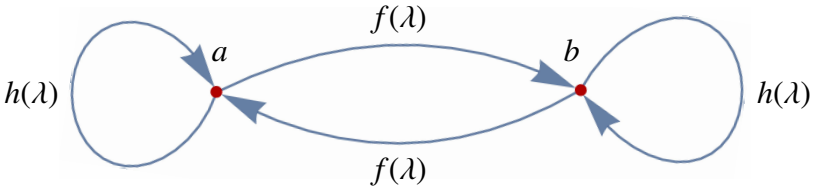


Figure 4.11: The isospectral reduction of a graph over two vertices which are latently automorphic

Suppose that there is not a latent automorphism between a and b in the undirected graph G. When we reduce to just these two vertices, then $\mathcal{R}_{\{a,b\}}(G)$ must have the form in Figure 4.12, where $g(\lambda), h(\lambda), f(\lambda) \in \mathbb{W}$. Because a and b are not latently automorphic, we know that $h \neq g$, otherwise this reduced graph would have a standard automorphism. Now using Theorem 2.1 in [7] page 24, suppressing λ dependence for notational simplicity, we calculate that the resulting smash functions are

$$SF(a) = \frac{f^2 + h(\lambda - g)}{\lambda - g}$$

$$SF(b) = \frac{f^2 + g(\lambda - h)}{\lambda - h}.$$

Now suppose by way of contradiction that both vertices have the same smash function. After we set the two expressions equal to each other we simplify to

$$\left(\lambda^2 + \lambda g + \lambda h + gh + f^2\right)(g - h) = 0.$$

Because $h \neq g$, this simplifies to

$$\lambda^2 + \lambda g + \lambda h + gh + f^2 = 0,$$

which implies that

$$g = \frac{f^2}{h - \lambda} + \lambda.$$

As long as the original graph only has real or complex weights (not rational functions), we claim that the rational functions g, h and f must have a larger degree in the denominator than in the

numerator (or equal degrees if the function is a constant). To prove this claim we use an equivalent formula for the isospectral graph reductions which is found in [7], Chapter 2. Edge weights of any isospectral reduction can be calculated as sums of *branch products*, where a branch product is given by

$$\mathcal{P}_\omega(\beta) = \omega(e_{12}) \prod_{i=2}^{m-1} \frac{\omega(e_{i,i+1})}{\lambda - \omega(e_{ii})}. \tag{4.3}$$

(for a a description of this formula see Equation 2.1 in [7]). From this formula we see that when a graph is reduced, factors involving λ are only added in the denominator of the branch product. Therefore, the edge weight of a graph resulting from any sequence of reductions is always a rational function whose numerator degree is less than or equal to the degree of its denominator.

Now we calculate

$$g(\lambda) = \frac{f^2}{h - \lambda} + \lambda = \frac{(a/b)^2}{c/d - \lambda} + \lambda = \frac{a^2 d}{b^2(c - d\lambda)} + \lambda = \frac{p}{q} + \lambda = \frac{p + q\lambda}{q}$$

where a, b, c, d, p, q are polynomials and $\deg(a) \leq \deg(b)$ and $\deg(c) \leq \deg(d)$. Now if we can show that $\deg(p) < \deg(q)$, then this would imply $\deg(p) \neq \deg(q+1)$ and thus

$$\deg(p + \lambda q) = \max\{\deg(p), \deg(q+1)\} > \deg(q)$$

which would contradict the fact that the degree of the polynomial in the denominator of $g(\lambda)$ must have be greater than or equal to the degree of the polynomial in its numerator. Thus all we need to show is that $\deg(p) < \deg(q)$, where $p = a^2 d$ and $q = b^2(c - d\lambda)$. We know that

$$\deg(c) \leq \deg(d) \Rightarrow \deg(c) < \deg(d\lambda) \Rightarrow \deg(c - \lambda d) = \deg(\lambda d) > \deg(d).$$

Because $\deg(a^2) \leq \deg(b^2)$ and $\deg(c - \lambda d) > \deg(d)$, we can conclude that $\deg(a^2 d) < deg\Big(b^2(c - d\lambda)\Big)$, and therefore $\deg(p) < \deg(q)$. Thus we have shown that $SF(a)$ cannot be equal to $SF(b)$.

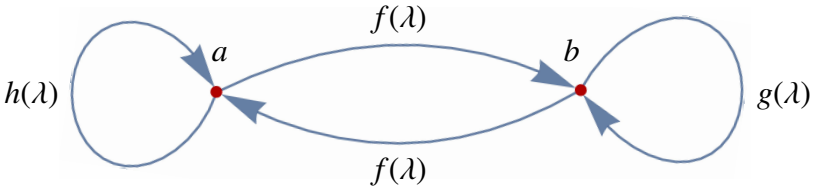


Figure 4.12: The isospectral reduction of an undirected graph over any two vertices. In the case of a latent symmetry between a and b, we have $h(\lambda) = g(\lambda)$

Proof of (3)

Equation 1.11 in [7], gives the following equality

$$\det(\mathcal{R}_S(M) - \lambda I) = \frac{\det(M - \lambda I)}{\det(M_{\bar{S}\bar{S}} - \lambda I)}.$$

Thus if we take the reduction set S to be the single vertex a, we get

$$\begin{aligned} \det(\mathcal{R}_{\{a\}}(M) - \lambda I) &= \frac{\det(M - \lambda I)}{\det(M_{\bar{a}\bar{a}} - \lambda I)} \\ \Leftrightarrow SF(a) &= \frac{\det(M - \lambda I)}{\det(M_{\bar{a}\bar{a}} - \lambda I)} + \lambda \\ \Leftrightarrow SF(a) &= \frac{p(\lambda)}{p_a(\lambda)} + \lambda \end{aligned}$$

where $p(\lambda)$ is the characteristic polynomial of M, and $p_a(\lambda)$ is the characteristic polynomial of $M_{\bar{a}\bar{a}}$, (which corresponds to the subgraph $G\backslash a$). Now suppose $SF(a) = SF(b)$ for two vertices $a, b \in G$, then

$$SF(a) = SF(b) \iff \frac{p(\lambda)}{p_a(\lambda)} + \lambda = \frac{p(\lambda)}{p_b(\lambda)} + \lambda \iff p_a(\lambda) = p_b(\lambda).$$

Because the characteristic polynomials of the subgraphs created by removing these two vertices are equal, these subgraphs have the same spectrum. Therefore we can conclude that the smash functions for two vertices are equal if and only if the vertices are *cospectral*.

This concludes the proof since we have proved all three equivalences. □

4.8 Concluding Remarks

We have proposed a novel generalization of the notion of a symmetry for a network (graph), which reveals to an extent the latent structure of both real and theoretical networks. We define these latent symmetries to be structural symmetries in an isospectrally reduced version of the original network. Though we showed some simple examples of latent symmetries in a number of simple graphs, we demonstrated they naturally occur in real networks (see Section 4.3). In addition we also define a measure of latency of a symmetry which gives a sense of scale of the symmetry or how deep the symmetry is hidden in the network. It is worth mentioning that this measure of latency can be difficult to compute as it requires searching through all possible vertex subsets of a graph. An interesting and open question is whether $\mathcal{M}(S)$ can be efficiently computed or approximated.

In the real-world networks we considered, and in the theoretical networks we numerically generated we were able to find latent symmetries that coexisted at various scales within the same network. This seems to suggest that real-world networks are not only rich with symmetries [5], but have what we might term a *hierarchical structure of symmetries* in which symmetries can be found at multiple scales within the network.

In Section 4.4, we showed that vertices in a network which are latently symmetric also have the same eigenvector centrality. Thus, in this sense latently symmetric vertices share equal importance in the network. This suggests that latent symmetries are encoding some type of network structure which is invisible to a simple symmetry search. Theorem 4.14 strengthens this result for the undirected case, showing that the Perron complement of the network over two vertices results in a symmetry if and only if those two vertices have the same eigenvector centrality in the original network.

One of the strongest cases for the importance of latent symmetries comes from the numerical study we perform in section 4.5, which shows that structural symmetries and latent symmetries are correlated in graphs generated using preferential attachment. This suggests that we should expect latent symmetries to naturally occur in real networks which form via some type of preferential attachment.

Having demonstrated the potential of latent symmetries as a concept for analyzing the structure of networks, many questions still remain. For instance, do networks utilize latent symmetries the same way they utilize standard structural symmetries? More specifically, do latently symmetric vertices often have similar functions? Or could they have complimentary functions? If a set of vertices are latently symmetric and some subset of them fail, what happens to the network?

In Section 4.6, we found latent symmetries are more likely to occur in a network built using a preferential attachment model. It is natural to ask if there are other models of network growth which also lead to the formation of latent symmetries. It was also noted that latent symmetries seem to be rare in large undirected networks. There is still work to be done to understand why this is and if there is a model which could result in latent symmetries in undirected graphs. In fact, the seeming dichotomy between directed and undirected graphs, which we find in our numerical and theoretical results, is not well understood and is a source of many open questions.

Finally, in Section 4.7 we prove the surprising fact that in undirected graphs vertices are latently symmetric if and only if they are also co-spectral. This fact opens up a new world of possible research questions. For instance, in the study of quantum walks on graphs, it has been shown that perfect state transfer can only occur between vertices which are co-spectral [79]. Perhaps this new equivalent characterization of cospectrality will be helpful in discovering other conditions for perfect state transfer.

In summary, there is much more work to do regarding latent symmetries, but from our preliminary work it seems that understanding how and why latent symmetries form could provide clues to how network structures form and are utilized.

Chapter 5. Conclusions and Future work

Throughout this dissertation, we introduce a number of new tools whose common theme is maintaining or controlling spectral properties of networks. This is significant as spectral properties of networks can be used to bridge network dynamics and structure [7],[66],[80]. Spectral properties

are also connected to centrality measures, connectivity, and other network measures. Since this dissertation focused mostly on introducing and explaining these three tools, we have not explored their potential use in other fields, especially those which are interested in dynamical networks. In future work, we plan to partner with other disciplines to use these new ideas to help solve problems in various types of networks.

In Chapter 2, we introduced a class of models for network growth based on the notion of specialization. This model produces networks with topologies that are increasingly hierarchical, more sparse, and modular, three properties typically found in many real-world networks. Further we were able to find rules for choosing the base at each step to generate models that numerically showed other network properties of interest. The first question that naturally arises is "can we prove our numerical claims analytically?" It is a very difficult question to determine the expected value for any of the properties we tracked including global clustering coefficient, degree distribution, or even sparsity for each stage of any specialization. Perhaps we could make estimates for the asymptotic behavior of these properties for at least some of the rules we have introduced.

Moreover, because a network can be specialized over *any* subset of its elements, the number of rules for choosing a base to specialize are endless. Another line of future work involves inventing new rules for developing other properties of networks. These new rules could be used for developing networks with a specific application in mind. Perhaps they could be used to predict how a particular real network would evolve over time. This type of research would require working with an expert in the field which studies the specific network. Alternatively, we could design rules that would develop a theoretical network with a particular property, such as a specific degree distribution after many iterates. Further, we could find such rules to develop properties which we can prove analytically.

In Section 2.6, we found that if a network was *intrinsically stable*, then growing the network via specialization will not disrupt the network stability. We are interested in exploring if there is a way to maintain other network dynamics. For instance, we could ask, "Is there a condition for which a network which synchronizes will continue to synchronize after being specialized?"

In Chapter 3, we first extended the technique of equitable decompositions to matrices which have separable automorphisms, then to prime-powered automorphisms and finally to *all* automorphisms. In some sense we completed this theory since there are no more classes of automorphisms to which we can extend these decompositions.

Although most of this chapter is written in terms of matrices, it is equivalent to work with the associated adjacency graph, where a matrix automorphism corresponds to a symmetry in the graph. This theory has roots in the graph interpretation as one of the smaller matrices which results from the decomposition is the divisor matrix. The divisor matrix' adjacency graph is the quotient graph for the equitable partition associated with the particular automorphism. The term equitable decomposition comes from this fact.

We note that although every automorphism of a graph induces an equitable decomposition, there are equitable decompositions which are not associated with an automorphism (see [81]). One thing which we have not explored is the possibility of finding a decomposition which creates the divisor matrix for an equitable partition not associated with an automorphism. It would be interesting to develop such a technique since it would add a new class of matrices which we could equitably decompose.

We also developed a few applications for equitable decompositions including Gershgorin estimates. One avenue of future research could extend these ideas to other eigenvalue estimation techniques including Brauer regions and Brualdi regions [52].

In Chapter 4, we proposed a novel generalization of network symmetry called latent symmetries. The latent symmetries are standard network symmetries in an isospectrally reduced graph. Much of Chapter 4 provided examples of latent symmetries and argued their usefulness for analyzing network structure. This included demonstrating that latent symmetries have the same eigenvector centrality. Now that latent symmetries have been defined and found in both real and theoretical networks, we can work toward understanding what they mean and how they are used in networks. For instance, we can study how networks utilize vertices which are latently symmetric. Do vertices which are latently symmetric have similar or complementary functions? Do these vertices

add robustness to the network? There is still much to learn about latent symmetries and how they function.

Finally in Section 4.8 we made the connection between latent symmetries in undirected graphs and cospectral vertices. Cospectral vertices have recently become an important object in the study of quantum walks on graphs as vertices being cospectral is a necessary condition for perfect state transfer to occur between them [79]. Some further research we plan to pursue is determining if the graph structure revealed by latent symmetries is useful in unraveling the open questions regarding perfect state transfer.

The tools we developed in this dissertation are interesting by themselves and they create many open questions to explore. Because these tools can be useful for various applications, numerous different courses of future research are available to us. As we pursue as many of these directions as we can, we hope to continue to find solutions to interesting problems and also to inevitably find more interesting questions.

Bibliography

[1] Miklós Bóna. A Walk Through Combinatorics: An Introduction to Enumeration and Graph Theory. World Scientific Publishing Company, 2002.

[2] Duncan J Watts and Steven H Strogatz. Collective dynamics of small-world networks. Nature, 393(6684):440, 1998.

[3] Miroslav Fiedler. Algebraic connectivity of graphs. Czechoslovak mathematical journal, 23(2):298–305, 1973.

[4] Wayne Barrett, Amanda Francis, and Benjamin Webb. Equitable decompositions of graphs with symmetries. Linear Algebra and its Applications, 513:409–434, 2017.

[5] Ben D MacArthur, Rubén J Sánchez-García, and James W Anderson. Symmetry in complex networks. Discrete Applied Mathematics, 156(18):3525–3531, 2008.

[6] Mark Newman. Networks: An Introduction. Oxford University Press, Inc., New York, NY, USA, 2010.

[7] Leonid Bunimovich and Benjamin Webb. Isospectral transformations. Springer Monographs in Mathematics, Springer, New York, 2014.

[8] G. Karlebach and R. Shamir. Modelling and analysis of gene regulatory networks. Nature Reviews Molecular Cell Biology, 9:770–780, 2008.

[9] Albert-Laszlo Barabasi and Zoltan N Oltvai. Network biology: Understanding the cell's functional organization. Nature reviews genetics, 5(2):101, 2004.

[10] Ed Bullmore and Olaf Sporns. Complex brain networks: Graph theoretical analysis of structural and functional systems. Nature Reviews Neuroscience, 10(3):186, 2009.

[11] Rebecca M Clark and Jennifer H Fewell. Transitioning from unstable to stable colony growth in the desert leafcutter ant acromyrmex versicolor. Behavioral ecology and sociobiology, 68(1):163–171, 2014.

[12] CT Holbrook, TH Eriksson, RP Overson, J Gadau, and JH Fewell. Colony-size effects on task organization in the harvester ant pogonomyrmex californicus. Insectes sociaux, 60(2):191–201, 2013.

[13] Jennifer H Fewell, Dieter Armbruster, John Ingraham, Alexander Petersen, and James S Waters. Basketball teams as strategic networks. PloS one, 7(11):e47445, 2012.

[14] Thilo Gross and Hiroki Sayama. Adaptive networks. In Adaptive Networks, pages 1–8. Springer, 2009.

[15] Olaf Sporns. Structure and function of complex brain networks. Dialogues in Clinical Neuroscience, 15(3):247, 2013.

[16] Carlos Espinosa-Soto and Andreas Wagner. Specialization can drive the evolution of modularity. PLoS computational biology, 6(3):e1000719, 2010.

[17] Ron Milo, Shai Shen-Orr, Shalev Itzkovitz, Nadav Kashtan, Dmitri Chklovskii, and Uri Alon. Network motifs: Simple building blocks of complex networks. Science, 298(5594):824–827, 2002.

[18] Mark EJ Newman. Modularity and community structure in networks. Proceedings of the national academy of sciences, 103(23):8577–8582, 2006.

[19] A. Clauset, C. Moore, and Mark E J Newman. Hierarchical structure and the prediction of missing links in networks. Nature, 453(7191):98–101, May 2008.

[20] Jure Leskovec, Kevin J Lang, Anirban Dasgupta, and Michael W Mahoney. Statistical properties of community structure in large social and information networks. In Proceedings of the 17th international conference on World Wide Web, pages 695–704. ACM, 2008.

[21] Mark EJ Newman. The structure and function of complex networks. SIAM review, 45(2):167–256, 2003.

[22] Mark D Humphries and Kevin Gurney. Network small-world-ness: A quantitative method for determining canonical network equivalence. PloS one, 3(4):e0002051, 2008.

[23] LA Bunimovich and BZ Webb. Isospectral graph transformations, spectral equivalence, and global stability of dynamical networks. Nonlinearity, 25(1):211, 2011.

[24] Jinde Cao. Global asymptotic stability of delayed bi-directional associative memory neural networks. Applied Mathematics and Computation, 142(2-3):333–339, 2003.

[25] Chang-Yuan Cheng, Kuang-Hui Lin, and Chih-Wen Shih. Multistability in recurrent neural networks. SIAM Journal on Applied Mathematics, 66(4):1301–1320, 2006.

[26] Shengshuang Chen, Weirui Zhao, and Yong Xu. New criteria for globally exponential stability of delayed cohen–grossberg neural network. Mathematics and Computers in Simulation, 79(5):1527–1543, 2009.

[27] Michael A Cohen and Stephen Grossberg. Absolute stability of global pattern formation and parallel memory storage by competitive neural networks. IEEE transactions on systems, man, and cybernetics, (5):815–826, 1983.

[28] Tao Li, Ting Wang, and Shumin Fei. Stability analysis on discrete-time cohen-grossberg neural networks with bounded distributed delay. In Control Conference (CCC), 2011 30th Chinese, pages 1081–1086. IEEE, 2011.

[29] Lin Wang and Guan-Zhong Dai. Global stability of virus spreading in complex heterogeneous networks. SIAM Journal on Applied Mathematics, 68(5):1495–1502, 2008.

[30] Tansu Alpcan and Tamer Basar. A globally stable adaptive congestion control scheme for internet-style networks with delay. IEEE/ACM Transactions on Networking (TON), 13(6):1261–1274, 2005.

[31] LA Bunimovich and BZ Webb. Restrictions and stability of time-delayed dynamical networks. Nonlinearity, 26(8):2131, 2013.

[32] Disambiguation. Wikipedia, the free encyclopedia. [Online; accessed 1-August-2017].

[33] Mercury. Wikipedia, the free encyclopedia. [Online; accessed 1-August-2017].

[34] Uri Alon. Network motifs: Theory and experimental approaches. Nature Reviews Genetics, 8(6):450, 2007.

[35] B. MacArthur, R. Sánchez-García, and J. Anderson. Symmetry in complex networks. Discrete Applied Mathematics, 156(18):3525 – 3531, 2008.

[36] Giulio Tononi, Olaf Sporns, and Gerald M Edelman. Measures of degeneracy and redundancy in biological networks. Proceedings of the National Academy of Sciences, 96(6):3257–3262, 1999.

[37] Albert-László Barabási. Network Science. Cambridge university press, 2016.

[38] Aaron Clauset, Cosma Rohilla Shalizi, and Mark EJ Newman. Power-law distributions in empirical data. SIAM review, 51(4):661–703, 2009.

[39] Takamitsu Watanabe and Naoki Masuda. Enhancing the spectral gap of networks by node removal. Physical Review E, 82(4):046102, 2010.

[40] Roger A Horn, Roger A Horn, and Charles R Johnson. Matrix Analysis. Cambridge university press, 1990.

[41] Yurong Liu, Zidong Wang, Alan Serrano, and Xiaohui Liu. Discrete-time recurrent neural networks with time-varying delays: Exponential stability analysis. Physics Letters A, 362(5-6):480–488, 2007.

[42] C. Godsil and G.F. Royle. Algebraic Graph Theory. Graduate Texts in Mathematics. Springer New York, 2001.

[43] D. Cvetković, P. Rowlinson, and S. Simić. An Introduction to the Theory of Graph Spectra. London Mathematical Society Student Texts. Cambridge University Press, 2009.

[44] D. Aldous and B. Pittel. On a random graph with immigrating vertices: Emergence of the giant component. Random Struct. Algorithms, 17(2):79–102, 2000.

[45] S. Strogatz. Sync : the Emerging Science of Spontaneous Order. Theia, New York, 2003.

[46] D. Watts. Small worlds: The dynamics of networks between order and randomness. Princeton University Press, Princeton, NJ, 1999.

[47] S. Gerschgorin. Uber die abgrenzung der eigenwerte einer matrix. Izvestija Akademii Nauk SSSR, Serija Matematika, 7(3):749–754, 1931.

[48] B. Hasselblatt and A. Katok. A First Course in Dynamics: with a Panorama of Recent Developments. Cambridge University Press, 2003.

[49] A. Brauer. Limits for the characteristic roots of a matrix. iv: Applications to stochastic matrices. Duke Math. J., 19(1):75–91, 03 1952.

[50] R. Brualdi. Matrices, eigenvalues, and directed graphs. Linear and Multilinear Algebra, 11(2):143–165, 1982.

[51] Krautstengel A. Varga, R. On gershgorin-type problems and ovals of cassini. ETNA. Electronic Transactions on Numerical Analysis [electronic only], 8:15–20, 1999.

[52] RS Varga. Gershgorin and his circles in springer series in computational mathematics, 36, 2004.

[53] K. Healy. Using metadata to find paul revere. https://kieranhealy.org/blog/archives/2013/06/09/using-metadata-to-find-paul-revere/, 2013. [Accessed: 2017-1-29].

[54] Carl P Dettmann and Georgie Knight. Symmetric motifs in random geometric graphs. Journal of Complex Networks, 2017.

[55] Diego Garlaschelli, Franco Ruzzenenti, and Riccardo Basosi. Complex networks and symmetry i: A review. Symmetry, 2(3):1683–1709, 2010.

[56] Ben D MacArthur and Rubén J Sánchez-García. Spectral characteristics of network redundancy. Physical Review E, 80(2):026117, 2009.

[57] Fernando Antoneli and Ian Stewart. Symmetry and synchrony in coupled cell networks 1: Fixed-point spaces. International Journal of Bifurcation and Chaos, 16(03):559–577, 2006.

[58] Petter Holme. Detecting degree symmetries in networks. Physical Review E, 74(3):036107, 2006.

[59] Ian Stewart. Networking opportunity. Nature, 427(6975):601–604, 2004.

[60] Peter J Olver. The symmetry groupoid and weighted signature of a geometric object. J. Lie Theory, 26:235–267, 2015.

[61] Ian Stewart, Martin Golubitsky, and Marcus Pivato. Symmetry groupoids and patterns of synchrony in coupled cell networks. SIAM Journal on Applied Dynamical Systems, 2(4):609–646, 2003.

[62] Yanghua Xiao, Ben D MacArthur, Hui Wang, Momiao Xiong, and Wei Wang. Network quotients: Structural skeletons of complex systems. Physical Review E, 78(4):046102, 2008.

[63] Andries E Brouwer and Willem H Haemers. Spectra of Graphs. Springer Science & Business Media, 2011.

[64] Gena Hahn and Gert Sabidussi. Graph Symmetry: Algebraic Methods and Applications, volume 497. Springer Science & Business Media, 2013.

[65] Fan RK Chung. Spectral Graph Theory. Number 92. American Mathematical Soc., 1997.

[66] Juan A Almendral and Albert Díaz-Guilera. Dynamical and spectral properties of complex networks. New Journal of Physics, 9(6):187, 2007.

[67] Vespignani Alessandro and Caldarelli Guido. Large Scale Structure and Dynamics of Complex Networks: From Information Technology to Finance and Natural Science, volume 2. World Scientific, 2007.

[68] Derek de Solla Price. A general theory of bibliometric and other cumulative advantage processes. Journal of the Association for Information Science and Technology, 27(5):292–306, 1976.

[69] Yanghua Xiao, Momiao Xiong, Wei Wang, and Hui Wang. Emergence of symmetry in complex networks. Physical Review E, 77(6):066108, 2008.

[70] Category:Logic Puzzles. Wikipedia, the free encyclopedia. [Online; accessed 1-August-2017].

[71] Cellular Networks. University of Notre Dame, Network Resources. [Online; accessed 15-October-2016].

[72] Dirk Koschützki, Katharina Anna Lehmann, Leon Peeters, Stefan Richter, Dagmar Tenfelde-Podehl, and Oliver Zlotowski. Centrality indices. In Network analysis, pages 16–61. Springer, 2005.

[73] Carl D Meyer. Uncoupling the perron eigenvector problem. Linear Algebra and its applications, 114:69–94, 1989.

[74] Albert-László Barabási and Réka Albert. Emergence of scaling in random networks. science, 286(5439):509–512, 1999.

[75] Réka Albert and Albert-László Barabási. Topology of evolving networks: Local events and universality. Physical review letters, 85(24):5234, 2000.

[76] Sergey N Dorogovtsev and José Fernando F Mendes. Scaling behaviour of developing and decaying networks. EPL (Europhysics Letters), 52(1):33, 2000.

[77] PL Krapivsky, GJ Rodgers, and S Redner. Degree distributions of growing networks. Physical Review Letters, 86(23):5401, 2001.

[78] Paul Erdös and Alfréd Rényi. On random graphs I. Publicationes Mathematicae (Debrecen), 6:290–297, 1959.

[79] Chris Godsil. State transfer on graphs. Discrete Mathematics, 312(1):129–147, 2012.

[80] Mark Newman, Albert-Laszlo Barabasi, and Duncan J Watts. The Structure and Dynamics of Networks. Princeton University Press, 2011.

[81] Ada Chan and C Godsil. Graph symmetry: Algebraic methods and applications, ch. 4, 1997.