The Influence Model: A Tractable Representation for the Dynamics of Networked Markov Chains

by

Chalee Asavathiratham

S.B., Massachusetts Institute of Technology (1996)
M.Eng., Massachusetts Institute of Technology (1996)

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

at the

Massachusetts Institute of Technology

October 2000

Author_____

Department of Electrical Engineering and Computer Science
October 24, 2000

Certified by_____

George C. Verghese
Professor of Electrical Engineering
Thesis Supervisor

Accepted by_____

Arthur C. Smith
Chairman, Departmental Committee on Graduate Students

The Influence Model: A Tractable Representation for the Dynamics of Networked Markov Chains

by

Chalee Asavathiratham

Submitted to the Department of Electrical Engineering and Computer Science
on October 24, 2000, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Abstract

In this thesis we introduce and analyze the *influence model*, a particular but tractable mathematical representation of random, dynamical interactions on networks. Specifically, an influence model consists of a network of nodes, each with a status that evolves over time. The evolution of the status at a node is according to an internal Markov chain, but with transition probabilities that depend not only on the current status of that node, but also on the statuses of the neighboring nodes. Thus, interactions among the nodes occur probabilistically, starting when a change of status at one node alters the transition probabilities of its neighbors, which then alter those of their neighbors, and so on.

More technically, the influence model is a discrete-time Markov process whose state space is the tensor product of the statuses of all the local Markov chains. We show that certain aspects of the dynamics of the influence model can be studied through the influence matrix, a reduced-order matrix whose dimension is the sum rather than the product of the local chain dimensions. We explore the eigenstructure of the influence matrix and explicitly describe how it is related to that of the full-order transition matrix. From the influence matrix, we also obtain the influence graph, which allows the recurrent states of the influence model to be found by graph-theoretic analysis on the reduced-order graph. A nested hierarchy of higher-order influence matrices, obtained from Kronecker powers of the first-order influence matrix, is exposed. Calculations on these matrices allow us to obtain progressively more elaborate statistics of the model at the expense of progressively greater computational burden.

As a particular application of the influence model, we analyze the "to link or not to link" dilemma. Suppose that a node is either in a 'healthy' or 'failed' status. Given that connecting to the network makes its status dependent on those of its neighbors, is it worthwhile for a node to connect to the network at all? If so, which nodes should it connect to in order to maximize the 'healthy' time? We formulate these questions in the framework of the influence model, and obtain answers within this framework. Finally, we outline potential areas for future research.

Thesis Supervisor: George C. Verghese
Title: Professor of Electrical Engineering

Dedication

To

My dearest lifelong influences:

Papa, Mama,

Lena and Malee.

Acknowledgements

First and foremost, I'd like to thank my most amazing thesis advisor, Prof. George Verghese. He is undoubtedly the person from whom I have learned the most at MIT, both academically and personally. I would not have been able to write half this thesis had it not been his help and encouragement, which have guided me through these years. His dazzling intelligence, his tireless devotion, and his fatherly wisdom will forever be remembered by this disciple of his.

I'm grateful to Prof. Bernard Lesieutre, my thesis reader and our research group's major "power." His helpful suggestions and encouragement have been an important part in my doctoral research from the start. I also thank Prof. John Tsitsiklis for graciously participating in my thesis committee and for making many useful comments on the research.

My gratitude extends to Dr. Paul Beckmann, my supervisor at Bose for several summers, for all his kindnesses and for the trust that he has given me. I thank Prof. Alan Oppenheim, who taught me so many things from signal processing to homework processing. It is a privilege to have been a teaching assistant for his classes several times.

I am also thankful to Prof. Robert Gallager, my graduate academic advisor, Dr. Arthur Berger, who taught me all about Markov chains, and Prof. Gilbert Strang, for his interest and helpful comments on the earlier part of my research.

My thanks to Prof. John Doyle at Caltech for his interest and suggestions, to Prof. Jean Carlson at U.C. Santa Barbara for pointers to the voter model, and to Prof. Duncan Watts at Columbia University for his interest and pointers to the threshold model. I am grateful to EPRI and DoD for support of this work under an initiative for Complex Interactive Networks and Systems (EPRI project WO-8333-06, ARP project DAAG55-98-1-3-001). A special thanks to Dr. Massoud Amin at the Electric Power Research Insitute for his support.

I also owe everyone in our research group a big debt: Sandip Roy for so many stimulating conversations and for suggesting ideas for further research, Aurelie Thiele for reading and commenting on my thesis so diligently, Ernst Scholtz for all the good laughs, Vahe Caliskan for all the little things that add up big, Babak Ayazifar, Jamie Byrum, and of course, Vivian Mizuno, the good spirit of our lab (what would we all do without you?).

Thanks to all my Thai friends who have been the constant voices of support: Poompat "Tengo", Ariya, Poonsaeng, Yot, the Thai badminton team (Pradya, Kulapant, Chayakorn, Siwaphong) and all the folks at the Thai Students at MIT. My 9 years at MIT would have been miserable without you all.

Finally, I thank Papa and Mama, who have shaped me into who I am from my very first

steps. Throughout my life, they have been giving me the best that they can offer. I thank Papa for spending all those evenings teaching me all that vocabulary, motivating me, and instilling in me the dream of coming to MIT. I thank Mama for, oh, just about everything I have or will ever have. I thank my sisters, Lena and Malee, for being all that a brother can hope for. I hope that by completing this thesis I have made them all proud and happy.

Contents

List of Figures

List of Tables

Chapter 1

Introduction and Overview

Consider the following examples, which are typical of those that have motivated this thesis. What is the common theme running through these different scenarios?

- **Power Blackout** On August 10, 1996, an electrical transmission line in the western United States accidentally caused a short circuit by sagging too close to a nearby tree due to thermal expansion [23, 10, 36]. The line was automatically removed from service by a circuit breaker, with the power flow being rerouted to nearby power lines. However, due to various circumstances including heavy loading throughout the system, additional line faults, and malfunctioning equipment, a series of power outages followed. Within seven minutes, the effect of the initial accident managed to sever the power flow through the important California-Oregon interties, resulting in power outages for nearly 7.5 million customers along the entire western United States.

- **Traffic Congestion** An accident has just occurred at a major street intersection during rush hour. The accident blocks the traffic flow through that intersection and soon causes an area of grid lock because the congestion spreads to nearby intersections. Drivers at the perimeter circumvent the blocked intersections by seeking alternate routes, and eventually the grid lock eases out.

- **Cold Spreading** John caught a cold from inadequate rest. The next morning, Mary stopped by his office and contracted the symptoms. In the evening, her son at home hugged her and got a cold from her. The next day, a couple of his friends at school caught the cold from him, and so it went.

- **Product Popularity** In 1994, the makers of *Hush Puppies* brand shoes were experiencing another gloomy year for their products — falling sales, declining popularity, and an ever decreasing number of store outlets [18]. Then out of the blue came a surge in demand. Without any advertising or promotional effort from its manufacturer, the shoes became an object of "haute couture." Starting with a handful of youngsters in New York City's East Village and Soho who wore the then-anonymous brand just to be different, the shoes caught the eyes of two fashion stylists, who then brought the shoes to the attention of other famous designers, who then ignited the Hush Puppies craze throughout the country. Hush Puppies sales went from 30,000 pairs a year in 1994, to 430,000 in 1995, and to 1,7200,000 in 1996.

In an abstract sense, each example above contains two common elements: nodes, and interactions among them. Each example involves dynamics on a network, or the dynamics of a network. Power stations dynamically interact with each other through power flows on the transmission grid. Street intersections, on the other hand, interact with each other through the traffic flows that connect

them. And individuals interact with each other through their social networks. In each of these settings, there are natural questions one might want to answer. For instance:

- A power station is vulnerable to disturbances or failures at its neighbors. On the other hand, being part of the power grid certainly has its advantages, including the chance for neighboring plants to help supply power when the local demand is greater than the capacity of the generator. So, given an existing power grid, should a new station operate alone or should it connect to the network? If it should connect, which station should it connect to?
- What is the likelihood that street intersections A, B and C would be congested simultaneously?
- Suppose we understand exactly what the vast social network is like in New York City. What is the probability that a group of Hush Puppies wearers would eventually set off a city-wide trend?

This thesis introduces and explores a new model for interaction on networks for which we call *the influence model*. The model has the potential to represent, in an abstract but tractable form, scenarios and questions such as those above, and possibly more elaborate situations involving complex interactions among different networks. Although limited by the quasi-linear interaction that it restricts the nodes to, the influence model still displays rich structure and behavior, so we are optimistic about the eventual scope of its application.

1.1 General Description

The quickest way to understand the influence model is through examples, and the one in Figure 1.1 will serve as our illustrative guide in this overview. As this thesis is mainly motivated by power systems, this particular example of the influence model is a highly simplified representation of demand and service of some power grid. As is the case for every influence model, it is a stochastic, dynamical system defined on a graph, and is described at two levels: the network level and the local level.

At the network level, each node can be treated as one active entity, and is called a *site*. For this example, a site can either be a power station (generator) or a load. Each site has a status that evolves over time. A power station may be represented as being in one of three possible statuses at any given time: normal, alert or failed. The loads, which can be cities or factories where power is actually consumed, might be in either high or low status, depending on the present level of demand. Looking inside each site, we find its local structure. If all the sites are disconnected, each local

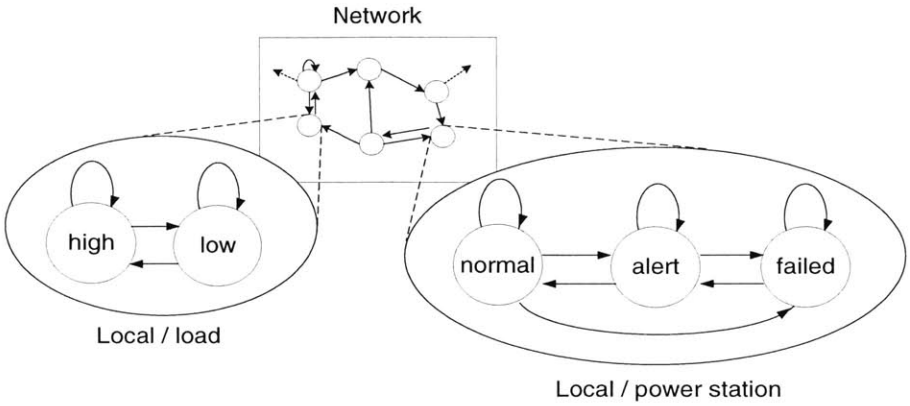


Figure 1.1: Example of an influence model

structure is a fixed Markov chain that describes the status of its site. However, with the network connections, the transition probabilities of each local chain are likely to depend not only on the current status of its site, but also on those of its neighboring sites.

The influence model is structured in a way that allows this influence of neighboring sites on the local transitions to be represented, although only in a quasi-linear form. For instance, we can construct the influence model for the above example such that a power station which is currently on alert can have a high probability of moving to failed status if it is surrounded by a combination of high loads and failed generators. On the other hand, if it is being influenced by low loads and normal generators, then it would have a high probability of reverting to normal status. The network connections would tell us not only which site can affect which, but also by *how much* one site can influence another's status. These influences effectively create a network of interacting Markov chains. The majority of this thesis is thus spent on analyzing this random system[1].

Our influence model is itself a huge Markov chain —the *master Markov chain*— in which each state corresponds to a state of the influence network. However, the order of the master chain is the *product* of the orders of the local chains, so the master chain is difficult or impossible to construct or work with. What we establish is that the particular structure of the influence model permits a full hierarchy of tractable lower-order models to be constructed, thereby permitting a very detailed study of the influence model.

[1]The actual behavior of a power system is, of course, much more complicated; we use this context only as rather abstract motivation.

1.2 Comparison to Previous Work

The concept of interactions on networks is not new, and has appeared in various forms in a variety of fields. These various models have been given an equally diverse list of names, depending on the applications for which they are intended and their specific structural features. Each model has a similar basic set-up that consists of a fixed network and some local rule by which the nodes interact. Some of these models are listed in Table 1.1. There are also references that deal with evolving rather than fixed networks, but these take us further away from influence models as dealt with in this thesis (although future work may well address evolving rather than fixed networks.)

Area	*Model*	*Key References*
Physics	stochastic Ising model	Glauber [19]
	cellular automata	Wolfram [41]
Mathematics	infinite particle system	Spitzer [39]
	voter model	Holley and Liggett [25]
	contact process	Harris [24]
Biology	invasion process	Clifford and Sudbury [4]
Sociology	threshold model	Granovetter [20]
	interactive Markov chain	Conlisk [5]
Economics	local interaction game	Ellison [15]
	strategy revision process	Blume [2]

Table 1.1: Some previous models of interactions on networks

Comparison of the results in this thesis to previous models has to be done with care, because of the technical differences in the way the models are set up. In general, even a slight change in the interaction rules can change the system behavior dramatically. Nevertheless, among the models above, many are different from one another in some fundamental ways: deterministic vs. stochastic, arbitrary vs. structured grid, etc. Thus, a technical result from one model may only provide a superficial guide into the behavior of another model. Moreover, as these models are motivated by different applications, even a qualitative insight obtained from studying one model might not have a meaningful interpretation in another. With these cautions, we outline below the differences between the previous models and the influence model.

1.2.1 Infinite Particle System

The term *infinite particle systems* [14, 21, 34, 39] or *interacting particle systems* is actually a general term that covers several stochastic models of interaction on networks. Some of the more widely recognized infinite particle systems are the voter model [25, 13], the contact process [22], and the stochastic Ising model [3, 19]. The standard set-up for each model is as follows. The network is generally an infinite d-dimensional lattice, with each site having two statuses. Each site has an "alarm clock" that strikes randomly with an exponential interarrival time. When the clock strikes, the site switches to the opposite status. The arrival rate of the clock at any given time depends on the current statuses of the site and its neighbors. The differences among the three models lie in how the rates depend on the statuses [14].

Voter Model The infinite particle system that most resembles the influence model is the *voter model*. The voter model was introduced independently in [25] and in [4]. Despite its name, the voter model seems to arise from a mathematical interest in [25] rather than from a serious motivation in sociology or political science. On the other hand, in [4], where the voter model is introduced as the *invasion process*, the model was proposed specifically as a model for spatial conflict of different species. In both versions, each site has a status '1' or '0' at any given time. The arrival rate of the alarm clock is proportional to the number of neighbors that are currently in the opposite status. Thus, the more neighbors with the opposite status a site has, the faster it will switch to the neighbor's value. If the statuses of all the neighbors agree with that of a site, then the status of that site will not change.

A special case of our influence model can be considered to be a natural discrete-time version of the voter model. This special case, referred to as the *binary influence model*, is discussed in detail in Chapter 3. The differences between the binary influence model and the voter model are not many, and the two most important ones are as follows. First, our model evolves in discrete rather than continuous time. Although this might seem like an unimportant difference, it raises the issue of periodicity in the network graph. In a continuous-time model, the process is guaranteed to reach a consensus (an all-ones or all-zeros state), whereas in discrete time, convergence to a consensus is only guaranteed when the underlying graph is aperiodic. Second, the voter model literature focuses on infinite graphs, or graphs with highly regular structure, such as the lattice [25, 34], the torus [8, 9], or infinite translation-invariant graphs [35] (graphs that look the same no matter which node we view them from). Our work, on the other hand, applies to finite but arbitrarily connected and arbitrarily weighted graphs. Although the results in [13] apply to finite graphs, the authors investigated only graphs in which the branches have uniform weight and are undirected. This assumption also brings about an implicit consequence that the graph is irreducible. Our model allows for arbitrary weights and an arbitrary number of classes in the underlying graph.

These generalizations could prove to be important degrees of freedom in particular applications.

The major departure from the traditional voter model starts in Chapter 4, where we first present the general version of our influence model. With the full generality, the influence model allows us significantly greater freedom in choosing the model parameters. For instance, the site no longer needs to be binary-valued, the internal Markov chains are (finite but otherwise) arbitrary, and the manner in which a site's status affects that of another site can also be tuned. This generality has brought forth several new features and questions. First, the recurrence structure, which reduces to the all-ones or all-zeros absorbing states in the case of the voter model, is much richer in the influence model. We explore this structure using a graph-theoretic analysis (Chapter 4). Second, because of the finiteness of our model, we are able to represent the dynamics of the system with a single matrix, the influence matrix (Chapter 5), of order equal to the *sum* of the orders of the local Markov chains. It turns out that the matrix approach also leads us to a *higher-order analysis*, the analysis of the joint-status of arbitrary collections of sites (Chapter 6). To the best of our knowledge, this type of analysis has only been done for second-order statistics and in a much more restrictive setting [34].

On the other hand, there are also results in the voter model literature that our analyses cannot achieve. These are generally results that depend on the fact that the grid is infinite, such as the rate at which the average cluster size grows (because our model has an upper limit on the cluster size), or the number of extremal invariant distributions as a function of the dimension of the lattice (because our model is not a lattice).

Other Infinite Particle Systems Apart from the voter model, other infinite particle systems are sufficiently different from the influence model that we only touch on them here. In the Ising model, if the status of a site i is $s_i \in \{-1, +1\}$, then the arrival rate for this site is

$$\exp(-\beta \sum_{j \in N_i} s_i s_j) \text{ for some } \beta \geq 0,$$

where N_i is the set of neighbors of site i [34]. That is, the rate of the alarm clock decreases exponentially with the number of neighbors with like statuses. In other words, this arrival rate favors a configuration in which the sites have uniform statuses.

In the contact process, the status of site i is $s_i \in \{0, 1\}$ and the arrival rate of the site is 1 if $s_i = 1$, and is otherwise $k\lambda$, where k is the number of neighbors currently in status 1 and λ is some fixed constant. Intuitively one can think of the contact process as the rate at which a site contracts a disease from a neighbor. When a site is sick (status '1'), the rate at which it will recover is constant (rate 1). When it is healthy (status '0'), then the rate at which it becomes sick

increases linearly with the number of sick neighbors (rate $k\lambda$).

1.2.2 Interactive Markov Chain

In [5], Conlisk introduces the *interactive Markov chain*, a deterministic, discrete-time dynamical system of the form

$$\mathbf{m}[k+1] = P(\mathbf{m}[k])\mathbf{m}[k]. \tag{1.1}$$

Here, the state vector $\mathbf{m}[k]$ is a nonnegative vector whose entries sum to 1 at each time k. Motivated by sociological applications, each entry $m_i[k]$ represents the fraction of the population with some attribute i. The matrix $P(\mathbf{m}[k])$ is a function of $\mathbf{m}[k]$ and has columns with nonnegative entries that sum to 1, i.e., $P(\cdot)$ is a transposed stochastic matrix. The dependence of $P(\cdot)$ on $\mathbf{m}[k]$, Conlisk explains, reflects the interactive nature of sociological dynamics, which takes into account the current social structure for its evolution. Then the author proceeds to give several forms of $P(\cdot)$, each with a specific application in the field. Since this evolution is deterministic, one can consider the interactive Markov chain as a particular deterministic, nonlinear, dynamical system. Subsequent papers [6, 7] focus exclusively on the mathematical part and discuss the stability of this nonlinear system for specific examples of $P(\cdot)$. In [33], Lehoczky justified the fact that the evolution is deterministic by showing that if each person's status evolves according to a Markov chain with a state transition matrix given by (the transpose of) $P(\cdot)$ above, then by the central limit theorem, each fraction of the aggregate population can be described by (1.1). Despite the similar titles, the interactive Markov chain and the influence model are vastly different. Among other things, in the evolution of the influence model, each site depends on only the status of its neighbors, as opposed to the aggregate state of the entire system. Thus, we will not pursue the interactive Markov chain any further.

1.2.3 Threshold Model

Another interesting but unrelated model from Table 1.1 is the *threshold model*. This model was first introduced by Granovetter [20] under careful sociological justifications. The model was further adopted by Morris [37] and Watts [40]. In the threshold model, each site has a status "1" if the number of 1's held by its neighbors exceeds a given threshold; otherwise, it has a status "0". In [20], the network considered is the complete graph (although this is not explicitly noted in the paper). Without the network topology being an issue, Granovetter focuses on the effect of the individual thresholds on the collective behavior, arguing that group behavior can be highly sensitive to the

exact threshold distribution. In [37], the emphasis is placed on the effect of the network structure on the global spread of a given status. In [40], most results are drawn from analysis and experiments on large, randomly generated graphs, where the effect of the graph parameters on the possibility of network-wide spread is explored.

From the author's personal experience with simulations, threshold switching causes the system to behave very differently from the influence model. In general, the threshold system has a very abrupt "all-or-nothing" spreading behavior, especially on random graphs. With a fixed threshold on a random graph, the spread of the 1-status is either limited to a small subset of the graph, or so widespread that it covers every site. Indeed, so clear is the distinction in the two cases that Watts unambiguously refers to the cluster of 1's in his paper as "local cascade" and "global cascade." This all-or-nothing observation hints that the effect of the individual threshold somehow translates into another "threshold" at the system level. Interesting as it is, the threshold model is very different from the influence model, and will not be discussed any further in the thesis.

1.2.4 Other Models

The term *cellular automata* refers to a large collection of models inspired by various applications in diverse fields, a sample of which is collected in the books [11, 41]. In economics, the paper [2] studies interaction on lattices from an economic point of view. In [15], Ellison explores the dynamics of a large population when each individual plays *coordination games* among neighbors on a circle network.

1.2.5 Summary of Contribution

While it is relatively easy to set up rules of local interaction, analyzing the system behavior resulting from a given set of rules is generally hard. The main contribution of this thesis is the proposal of a network interaction model that is satisfactorily tractable, yet contains some of the desirable features highlighted below. Another summary with more technical detail is provided in Chapter 7.

- **Arbitrary Network Structure** By allowing each site to contain an arbitrary (finite) local chain and the network to have an arbitrary (finite) graph and influence structure, the influence model gives us an important level of modeling versatility. Previous models generally impose additional restrictions to simplify the analysis, such as requiring lattice-structure networks, or allowing only binary-status sites, or needing uniform-weight edges.

- **Graph-Theoretic Analysis of Recurrent States** Since the network is allowed to be arbitrary, the behavior of the influence model is critically dependent on the underlying structure of the graph. We will use a graph-theoretic approach to determine the recurrent states of the influence model. This kind of analysis is usually not done in previous models since they generally involve only simple network structures such as the lattice grid, or the regular graphs.
- **Higher-Order Analysis** The influence model is also amenable to the analysis of joint-statuses, or the statuses of any specified group of sites. This is in contrast to previous models, which can either describe the status of an individual site, or the collective status of *all* sites.
- **"To Link or not to Link" Dilemma** One interesting question naturally arises in the analysis of the influence model. Suppose each site is either in a 'healthy' or 'failed' status at any given time. Given that connecting to the network makes the status of a site dependent on its neighbors, should a site connect to the network or should it operate in isolation in order to maximize the 'healthy' time? If it should connect to the network, which sites should it connect to and with what edge weights? It turns out that this question can be framed and answered nicely because of the way the influence model is defined.

1.3 Chapter Outline

In Chapter 2, we introduce the basic notions — primarily concerning Markov chains — that are necessary for the rest of the thesis. In Chapter 3, we present a special case of the influence model called the *binary influence model.* We derive basic results regarding their convergence on ergodic graphs, and graphically explain the dual of this process in terms of coalescing random walk. In Chapter 4, we introduce the general influence model and determine its recurrent classes by analyzing the structure of the influence graph, with the help of a 'hopping dot' picture. Several small examples are provided. A concept called product path is also introduced. In Chapter 5, we analyze the influence matrix, and relate it to the state-transition matrix of the master Markov chain. Towards the end of the chapter, we raise and answer the "to link or not to link" question. The answer to this question leads naturally to Chapter 6, which discusses the higher-order influence matrices. Finally Chapter 7 concludes the thesis and outlines directions for future research.

Chapter 2

Background Review

This chapter defines the necessary basics on directed graphs and Markov chains. These results will be used extensively in later chapters when we discuss influence models. The material presented can be found in standard textbooks such as Horn and Johnson [26], Gallager [16] and Brémaud [3]. Only the topics that are relevant to later discussions are covered here.

Notation: Throughout this thesis, vectors are either denoted as boldfaced lower-case letters ($\mathbf{a}$, $\mathbf{b}$, $\mathbf{x}_1$, $\mathbf{y}_2$), or as Greek letters (α, β). All vectors are column vectors. Matrices are written as upper-case Roman letters (A) and their entries denoted by the corresponding lower-case letters with subscripts to indicate the row and column (a_{ij}). The entries of sums and products of matrices are denoted by brackets and subscripts ($[A+B]_{ij}$ and $[AB]_{ij}$). The transpose of a matrix A is denoted A'. The symbols $\mathbf{1}_m$ and $\mathbf{0}_m$ denote the length-m all-ones and all-zeros column vectors respectively. When it is clear from the context, we will simply write $\mathbf{1}$ and $\mathbf{0}$ to reduce notational clutter. The $n \times n$ identity matrix is denoted by I_n.

2.1 Directed Graphs

In this section, we cover certain fundamental characterizations of directed graphs that will be used throughout the thesis, primarily in the analyses of Markov chains and influence graphs. Most terminology to be introduced is standard. Readers who are already familiar with Markov chains may skip this section and only return to it when the need arises in subsequent chapters.

Let $A = [a_{ij}]$ be an $n \times n$ matrix. Define the *directed graph of* A, denoted by $\Gamma(A)$, as the directed graph on nodes 1 to n, where a directed edge from i to j, denoted by (i,j), exists if and only if $a_{ij} \neq 0$. The *edge weight* is given by a_{ij}.

A *path* p is an ordered sequence of nodes $p = (j_1, j_2, \ldots, j_k)$ such that edge (j_i, j_{i+1}) exists for all $1 \leq i \leq k$. Node j_1 is called the *source* and j_k the *destination* node of path p. The *length* of path p, denoted $\ell(p)$, is the number of edges on it, in this case $k-1$. Note that an edge may be counted towards the length multiple times if it appears more than once. A path is *cyclic* if the source and destination nodes are the same. We will also refer to cyclic paths as *cycles*. Note that a *self-loop* or a cycle that contains only one edge such as (i,i) can also occur (precisely when $a_{ii} \neq 0$).

THEOREM 2.1
Let A be a square matrix with all nonnegative entries. There exists a path of exactly length k from node i to j on $\Gamma(A)$ if and only if $[A^k]_{ij} > 0$

Proof. See [26], Theorem 6.2.16. □

2.1.1 Classes

Definition Node j is *accessible* to node i if there exists a path in which i is the source and j the destination. It should be noted that accessibility is a property that depends merely on the existence of a path, not on its length or its edge weights. Two nodes i and j are said to *communicate* if both nodes are accessible to each other. It then follows that communication is an equivalence relation. Using this equivalence relation, we can partition the nodes into disjoint sets called *classes*, which have the following properties:

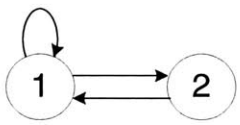
- every node in a class communicates with every other, and
- a node outside of a class does not communicate with any node inside the class.

The graph $\Gamma(A)$ is called *irreducible* if it has only one class. A matrix A is termed *irreducible* if $\Gamma(A)$ is irreducible. The partition of a graph into classes does not depend on the specific weights on the edges. So we will use a '1' to denote a nonzero entry in the following examples.

Example 1: These examples show directed graphs and their classes for the given matrices.

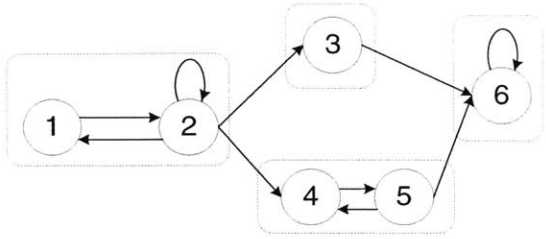
(a) In this example, $\Gamma(A)$ is irreducible.

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



(b) In this example, $\Gamma(A)$ has 4 classes: $\{1,2\}$, $\{3\}$, $\{4,5\}$, and $\{6\}$.

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.1}$$



□

For every directed graph with a finite number of nodes, the partitioning of its nodes into classes is unique. Furthermore, the interconnection among the classes must be *acyclic*. That is, any path that leaves a class cannot return to it. If we consider the macroscopic view of a graph obtained by lumping each class into a single node, then we will get a graph such as in Figure 2.1. Here multiple edges from one class to another are lumped into a single edge for easy visualization.

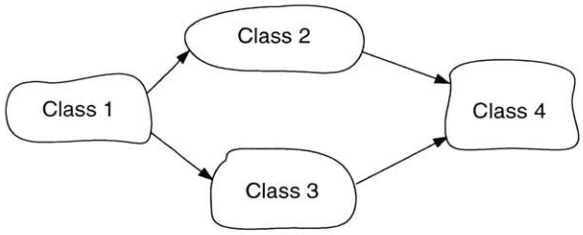


Figure 2.1: A macroscopic view of a directed graph where each class is represented as a node.

Classes that have only incoming edges are called *recurrent*. A class that is not recurrent is called *transient*. Classes that have only outgoing edges are called *autonomous*. Classes that are not autonomous are called *dependent*. In Figure 2.1, Class 4 is recurrent, while Classes 1, 2 and 3 are transient. On the other hand, Class 1 is autonomous, while Classes 2,3, and 4 are dependent. Any directed graph with a finite number of nodes must have at least one autonomous and one recurrent class. Unless there is only one class in the graph, a recurrent class is not autonomous. Intuitively, these two types of classes are the ones at the extreme ends of the graph.

2.1.2 Classes and Matrices

A *permutation matrix* P is an $n \times n$ matrix in which each entry is either a 0 or a 1, and every row and every column contains precisely a single 1. For a given $m \times n$ matrix A, multiplication by P on the right has the effect of permuting the columns of A. Specifically, if $p_{ij} = 1$, then the ith column of A is equal to the jth column of AP. In contrast, for a given $n \times m$ matrix B, multiplication by P on the left permutes the row of B; if $p_{ji} = 1$, then the ith row of B is equal to the jth row

of PB. Thus, for any given square matrix A and two permutation matrices P_1, P_2, the product P_1AP_2 simultaneously permutes the rows and the columns of A. In order to permute the rows and columns by the same reordering, P_1 and P_2 must satisfy $[P_1]_{ij} = [P_2]_{ji}$ for all i, j. That is, $P_1 = P_2'$. In general, when P is a permutation matrix, we refer to the product of the form $P'AP$ a *cogredient of* A ([1], Definition 1.2). Because a cogredient $P'AP$ is similar to A, they share the same set of eigenvalues. The set of permutation matrices is closed under multiplication. That is, a product of two permutation matrices is also a permutation matrix.

There is a natural interpretation of a cogredient in terms of directed graphs. The graph $\Gamma(P'AP)$ can be obtained from $\Gamma(A)$ by relabeling the index of each node according to the permutation P. That is, if $p_{ij} = 1$, the node with index i on $\Gamma(A)$ would be relabeled as index j on the graph $\Gamma(P'AP)$. For instance, if

$$A = \begin{bmatrix} .2 & .4 & 1 \\ .1 & & \\ & .5 & \end{bmatrix}, \quad \text{and} \quad P = \begin{bmatrix} & 1 & \\ & & 1 \\ 1 & & \end{bmatrix} \quad \text{so that} \quad P'AP = \begin{bmatrix} & & .5 \\ 1 & .2 & .4 \\ & .1 & \end{bmatrix} \tag{2.2}$$

then the graphs $\Gamma(A)$ and $\Gamma(P'AP)$ are shown in Figure 2.2. Notice how the graph topology and the edge weights are unchanged by node renumbering (although only two edge weights are shown to reduce figure clutter).

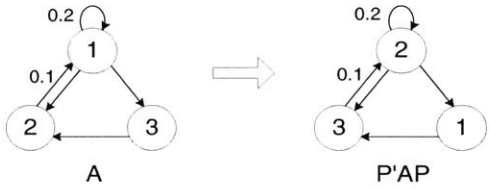


Figure 2.2: Graph $\Gamma(A)$ and its renumbered version $\Gamma(P'AP)$.

On the graph $\Gamma(A)$ where A is an $n \times n$ matrix, each node has its own unique integer index, which could be any number from 1 to n. So for any two subsets B and C of nodes on $\Gamma(A)$, define A_{BC} as the $|B| \times |C|$ submatrix of A obtained by selecting the rows and columns of A that correspond to the indices of the nodes in B and in C respectively. For instance, for the A in eq. (2.2), if $B = \{2, 3\}$ and $C = \{1, 2\}$, then $A_{BC} = \begin{bmatrix} .1 & \\ & .5 \end{bmatrix}$. A submatrix of the form A_{BB} is termed a *principal submatrix* of A and is simply denoted by A_B.

THEOREM 2.2

For a given square matrix A, let $R_1, \ldots, R_t$ be the partition of the nodes of $\Gamma(A)$ into classes. Then there exists a cogredient $P'AP$ of A such that $P'AP$ is in the block-triangular form with the

matrices A_{R_i}'s on the diagonal, i.e.,

$$P'AP = \begin{bmatrix} A_{R_1} & & \\ * & \ddots & \\ * & * & A_{R_t} \end{bmatrix}. \tag{2.3}$$

Here * represents some entries that are possibly nonzero (but are not of concern to us).

Proof. Let $R_1, \ldots, R_r$ be the recurrent classes of $\Gamma(A)$ and let R^c be the union of all its transient classes.

There must be a cogredient $P_1'AP_1$ in the (hollowed) lower block-triangular form:

$$P_1'AP_1 = \begin{bmatrix} A_{R_1} & & & \\ & \ddots & & \\ & & A_{R_r} & \\ A_{R^cR_1} & \cdots & A_{R^cR_r} & A_{R^c} \end{bmatrix}. \tag{2.4}$$

This can be done by renumbering the nodes in the recurrent classes sequentially from the first class to the rth one, and then numbering nodes in the transient classes last. In (2.4), the entries on the left and right of submatrices $A_{R_1}, \ldots, A_{R_n}$ are zero because each R_i is a recurrent class; there can be no edge that connects a node inside a recurrent class into another node outside of it. In the bottom rows in (2.4), each $A_{R^cR_i}$ represents the edges from the transient classes to class R_i, and the block A_{R^c} represents the connections within the transient classes.

Now consider the bottom-right block A_{R^c} as a matrix in its own right. If it is irreducible, then we have proved the claim above. If it is reducible, then we can renumber it into the block-triangular form as well. Suppose the recurrent classes of $\Gamma(A_{R^c})$ are $S_1, \ldots, S_s$ and the transient classes are collectively represented by S^c, then there is a cogredient $\widetilde{P}'A_{R^c}\widetilde{P}$, which is block-triangular and has the submatrices $A_{S_1}, \ldots, A_{S_s}$ and A_{S^c} on the diagonal. Define the permutation matrix $P_2 \triangleq \text{diag}(I_{|R_1|}, \ldots, I_{|R_r|}, \widetilde{P})$ (i.e., P_2 is a block-diagonal matrix with the given matrices on

the diagonal). Then we have

$$P_2'(P_1'AP_1)P_2 = \begin{bmatrix} A_{R_1} & & & \\ & \ddots & & \\ & & A_r & \\ * & \dots & * & \widetilde{P}'A_{R^c}\widetilde{P} \end{bmatrix} = \begin{bmatrix} A_{R_1} & & & & & & \\ & \ddots & & & & & \\ & & A_{R_r} & & & & \\ * & \dots & * & A_{S_1} & & & \\ * & \dots & * & & \ddots & & \\ * & \dots & * & & & A_{S_s} & \\ * & \dots & * & * & \dots & * & A_{S^c} \end{bmatrix} \tag{2.5}$$

Note that each S_i is also a transient class of $\Gamma(A)$, so to have the A_{S_i}'s on the diagonal in eq. (2.5) is still consistent with the claim in the Theorem. By renumbering the bottom-right blocks (such as A_{R^c} and A_{S^c}) recursively until an irreducible submatrix appears, we will have obtained a matrix $P'AP$ (where P is a product of permutation matrices) in lower block-triangular form that satisfies the form in (2.3). Since permutation matrices are closed under multiplication, P is a valid permutation matrix, and, therefore, $P'AP$ is a valid cogredient as claimed. □

COROLLARY 2.3

For a given square matrix A, let $\{R_i\}$ be the classes of $\Gamma(A)$, then the eigenvalues of A are those of all the A_{R_i}'s, counting multiplicities.

Proof. Let $P'AP$ be a cogredient described in Theorem 2.2. By [26], p. 62, Prob. 5, the eigenvalues of $P'AP$ are those of all the A_{R_i}'s, counting multiplicities. Because A and $P'AP$ are similar, they have the same eigenvalues. □

Consider the relation between $\Gamma(A)$ and $\Gamma(A')$. Each of these two graphs can be derived from the other by reversing the direction of every edge. Note that the communication relations between nodes are still unchanged by the reversal of the edges. Therefore, the class partition must still be the same for both graphs. The difference, however, is that through the edge reversal, an autonomous class is turned into a recurrent class, and vice versa, because all the outgoing edges become incoming edges. Thus, we have essentially arrived at the following corollary.

COROLLARY 2.4

A class R is autonomous with respect to $\Gamma(A)$ if and only if R is recurrent with respect to $\Gamma(A')$.

2.1.3 Periods

The next fundamental characteristic that is of importance to our later discussion is the period of a class. To define it, consider a node i and all the cycles in that class that pass through it. Call these cycles $\{p_1, p_2, \dots\}$. Such cycles always exists, unless the class consists of a single node with no self-loop, in which case the period is undefined for that node. Then the *period* of node i is defined as the greatest common divisor of the lengths $\{\ell(p_1), \ell(p_2), \dots\}$. A node with a period of 1 is *aperiodic*. Otherwise it is *periodic*.

For example, in Figure 2.3, node 2 has a period of 1, because the cycle $(2, 4, 2)$ has length 2, while the cycle $(2, 3, 1, 2)$ has length 3. In Figure 2.4, node 2 has a period of 2, because all cycles

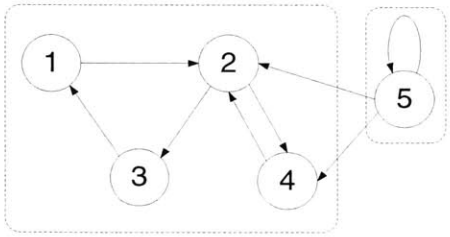


Figure 2.3: A directed graph with two classes. The dashed line shows the grouping of the nodes into two classes, both aperiodic. The right class is autonomous, while the left recurrent.

that pass through it have even length. In fact, the equivalent diagram on the right of Figure 2.4 makes clear that every node in that graph must have a period of 2.

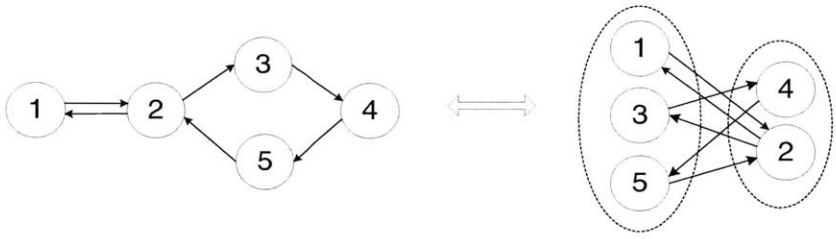


Figure 2.4: A directed graph with one single class. Rearranging the graph into the equivalent one on the right makes evident its period is 2.

In practice, nodes are almost always aperiodic, unless they reside on a very small or a large but highly structured graph. The more cycles that pass through a node, the more likely it will be aperiodic. For instance, having both odd and even cycles passing through it is enough to guarantee the aperiodicity of a node. In particular, any node that has a self-loop is aperiodic. The following theorem shows that periodicity is a class property.

THEOREM 2.5

Every node in a give class has the same period.

Proof. See [16], p. 106. □

THEOREM 2.6

Within a class of period d, one can divide the nodes into d subclasses $T_1, \ldots, T_d$ such that the subclasses are connected in a circle. That is, every edge must connect from a node in some subclass T_i to another node in T_{i+1}, or from T_d to T_1.

Proof. See [16], p. 107. □

In the following, we will refer to the sets T_i as *subclasses*. The graph in Figure 2.4 has one class and two subclasses; that in Figure 2.5 has one class and three subclasses. If we assume that

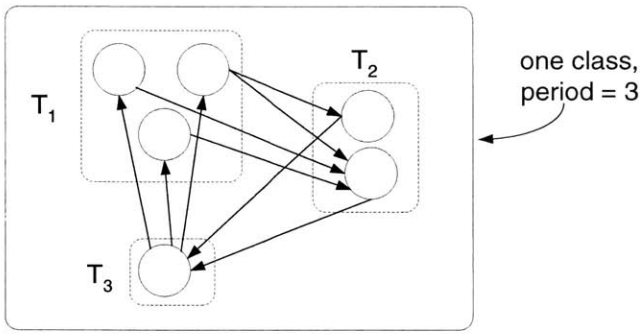


Figure 2.5: A graph that contains a single class with a period of 3, divided into its three subclasses.

in numbering the nodes, we enumerate those in T_i before those in T_{i+1}, then the matrix A that defines the graph of the corresponding class of period d can be written as

$$A = \begin{bmatrix} 0 & R_1 & & \\ & \ddots & \ddots & \\ & & 0 & R_{d-1} \\ R_d & & & 0 \end{bmatrix}. \tag{2.6}$$

It is easy to show that A^d has a block diagonal form:

$$A^d = \begin{bmatrix} Q_1 & & \\ & \ddots & \\ & & Q_r \end{bmatrix} \tag{2.7}$$

where $Q_i = R_i \cdots R_d R_1 \cdots R_{i-1}$. If we recall Theorem 2.1, then the block-diagonal form of (2.7) is not surprising, because any path of length d must terminate at a node that belongs to the subclass from which it originates.

To summarize, the nodes on a general directed graph can be partitioned into classes, and within each class, one can further partition them into subclasses. A graph that comprises just a single aperiodic class is called *ergodic.*

2.2 Perron-Frobenius Theory

The Perron-Frobenius Theorem describes some special features of the eigenstructure of nonnegative matrices. This theorem and its corollaries will be critical to our later discussion of the dynamics of Markov chains and influence matrices. For more details on nonnegative matrices, see [1] or [26], for example.

A matrix A is *positive*, denoted $A > 0$, if all of its entries are real and strictly positive. A matrix A is *nonnegative*, denoted $A \geq 0$, if all of its entries are real and nonnegative. For two matrices A and B, $A \geq B$ means $A - B \geq 0$. For any square matrix A, we refer to the set of eigenvalues of A as its *spectrum* and denote it by $\sigma(A)$. The *spectral radius of* A is the real and nonnegative scalar

$$\rho(A) \stackrel{\triangle}{=} \max_{\lambda \in \sigma(A)} |\lambda|.$$

It is worth noting that for a general square matrix A, $\rho(A)$ may or may not be one of its eigenvalues. However, in the case of nonnegative square matrices, the spectral radius $\rho(A)$ is always an eigenvalue itself, as precisely stated in the following theorem.

THEOREM 2.7 (Perron-Frobenius)
Let $A \geq 0$ be a square and irreducible matrix. Then $\rho(A) \in \sigma(A)$. There is a positive right eigenvector $\mathbf{v} > 0$ corresponding to eigenvalue $\rho \stackrel{\triangle}{=} \rho(A)$ such that the following properties hold:

(1) For any $\mathbf{x} \geq 0$, if $A\mathbf{x} \geq \rho\mathbf{x}$, then $A\mathbf{x} = \rho\mathbf{x}$.

(2) If $A\mathbf{x} = \rho\mathbf{x}$, then $\mathbf{x} = c\mathbf{v}$ for some constant c.

Proof. See [16], Theorem 5, p. 115. □

Note that Theorem 2.7 does not exclude the possibility of an irreducible matrix having more than one eigenvalue with the magnitude $\rho(A)$. For instance, if the matrix A in (2.6) is nonnegative, then it always contains d eigenvalues with magnitude equal to its spectral radius. What the theorem states is that there exists an eigenvalue that is both largest in magnitude *and* positive, and through property (2), that this eigenvalue has a geometric multiplicity of 1; we show shortly that its algebraic multiplicity is also 1. We can, therefore, unambiguously refer to it as the *dominant eigenvalue* of an irreducible nonnegative matrix A.

COROLLARY 2.8

The dominant eigenvalue ρ of an irreducible matrix $A \geq 0$ has a left eigenvector $\pi > 0$. π is the unique (within a scale factor) eigenvector of λ and is the only nonnegative, nonzero vector (within a scale factor) that satisfies $\pi' A \geq \rho\pi'$.

Proof. By applying Theorem 2.7 to A', and by recognizing that $\sigma(A) = \sigma(A')$, we have the above corollary. □

COROLLARY 2.9

Let ρ be the dominant eigenvalue of an irreducible matrix $A \geq 0$ and let the right and left eigenvectors be $\mathbf{v} > 0$ and $\pi > 0$ respectively. Then, within a scale factor, $\mathbf{v}$ is the only nonnegative right eigenvector of A, i.e., no other eigenvalue has a nonnegative eigenvector. Similarly, within a scale factor, π is the only nonnegative left eigenvector of A.

Proof. See [16], Corollary 2, p. 116. □

COROLLARY 2.10

The dominant eigenvalue of an irreducible matrix $A \geq 0$ has algebraic multiplicity 1.

Proof. If not, then there must be, from Jordan form theory ([26]), a vector $\mathbf{w}$ such that $(A-\rho I)\mathbf{w} = \mathbf{v}$, the right eigenvector of ρ. Pre-multiplying both sides by π', the left eigenvector of ρ, and noting that $\pi'(A - \rho I) = 0$, we get $\pi'\mathbf{v} = 0$. But since $\pi'\mathbf{v} > 0$, we arrive at a contradiction. □

COROLLARY 2.11

The dominant eigenvalue of an irreducible matrix $A \geq 0$ is a strictly increasing function of every entry of A.

Proof. See [16], Corollary 5, p. 116. □

COROLLARY 2.12
If $A \geq 0$, $A\mathbf{v} = \lambda\mathbf{v}$ *and* $\mathbf{v}$ *is real and positive, then* $\lambda = \rho(A)$.

Proof. See [26] Corollary 8.1.30. □

Note that Corollary 2.12 is different from the theorems that precede it in that it does not require A to be irreducible.

2.3 Markov Chains

Our review of background material culminates in this section with the basic theory of Markov chains. The exposition to follow, although self-contained, will be brief, because its purpose is not to educate a reader about the subject, but rather to set up the language and results to be used in this thesis.

A square matrix A is *stochastic* if it is nonnegative and each row sums to 1, i.e., $A\mathbf{1} = \mathbf{1}$, where $\mathbf{1}$ is the all-ones vector. A matrix A is *substochastic* if $A\mathbf{1} \leq \mathbf{1}$ with the inequality being strict in at least one row. Because each row of a stochastic matrix A sums to 1, one distinctive feature of its graph $\Gamma(A)$ is that the sum of all the edge weights leaving a node is 1.

The graph $\Gamma(A)$ of a stochastic matrix A corresponds to a *Markov chain.* We will abuse the terminology slightly by referring to $\Gamma(A)$, or sometimes even A itself as the Markov chain. Each node on $\Gamma(A)$ can be interpreted as a *status* of some system that evolves randomly over time, as described next. At any given time, the system is assumed to be in one of the n possible statuses (or at one of the n nodes of the graph), which are represented by integers 1 to n. At time k, the status of the system is captured in the *status vector*

$$\mathbf{s}[k] \stackrel{\triangle}{=} [0 \cdots 0\ 1\ 0 \cdots 0]'. \tag{2.8}$$

This length-n vector is an indicator vector whose only nonzero entry is a 1 in the position corresponding to the current status of the system.

A *Probability Mass Function vector*, or *PMF vector*, is a vector $\mathbf{p} \geq 0$ such that $\mathbf{p}'\mathbf{1} = 1$. Given a length-$n$ PMF vector $\mathbf{p}$, we denote by

$$\mathbf{s} = Realize(\mathbf{p}) \tag{2.9}$$

the random realization of the status of the chain according the PMF provided in $\mathbf{p}$. We can think of the actions performed by (2.9) as rolling an n-faced die whose probability of turning up face i is p_i, and assigning the actual face that comes up to $\mathbf{s}$ in the format given in (2.8).

We assume that the initial status $\mathbf{s}[0]$ is independently realized from some given PMF. Given a realization of $\mathbf{s}[0]$, the Markov chain generates a sequence of random vectors $\{\mathbf{s}[k]\}$ according to the following evolution equations:

$$\mathbf{p}'[k+1] = \mathbf{s}'[k]A \tag{2.10}$$

$$\mathbf{s}[k+1] = Realize(\mathbf{p}[k+1]) \tag{2.11}$$

The sequence $\{\mathbf{s}[k]\}$ is a *Markov process*, and A is referred to as the *state-transition matrix* for the process. The vector $\mathbf{p}[k+1]$ in (2.10) is a valid PMF because it is nonnegative and

$$\mathbf{p}'[k+1]\mathbf{1} = \mathbf{s}'[k]A\mathbf{1} = \mathbf{s}'[k]\mathbf{1} = 1.$$

If $\mathbf{s}[k]$ is given, then $\mathbf{p}[k+1]$ is fully determined. Otherwise, $\mathbf{p}[k+1]$ is in general random.

Example 2: Suppose a power station's status at any given time is in one of the following three statuses: normal, alert or failed. We may model its operating conditions with a 3×3 stochastic matrix A whose Markov chain is shown in Figure 2.6. Here $\Gamma(A)$ is ergodic, because all nodes

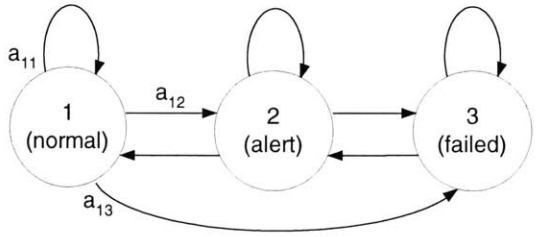


Figure 2.6: A 3-status Markov chain modeling a power station's operating conditions

communicate and are aperiodic. The outgoing edge weights sum to 1, so in particular, $a_{11} + a_{12} + a_{13} = 1$. Suppose the system starts from a normal status, i.e., $\mathbf{s}[0] = [1\ 0\ 0]'$ and evolves according to (2.10)-(2.11). We can imagine the 'life' of the system being represented by a dot (or token) that hops from one node to the next. When in node i, the dot will hop to node j with probability a_{ij}. Whichever node the dot lies on at time k is the system status at that time. This hopping-dot picture will be useful later on when discussing the influence model and its recurrent states. □

2.3.1 Status Probabilities

In this section, we address the following questions: given an arbitrary stochastic matrix A and a distribution on the initial status $\mathbf{s}[0]$, what is the probability that the system will be in a given status at a given time? What can be predicted about $\mathbf{s}[k]$ as $k \to \infty$?

The following proposition is straightforward, but fundamental to our answers to the above questions. The notation $E(\cdot)$ will be used for expectation or expected value.

PROPOSITION 2.13

$$E(s_i[k]) = Prob(\textit{system is in status } i \textit{ at time } k).$$

Proof. This follows from the fact that each $s_i[k]$ is a binary random variable, so its expected value must equal the probability of its being 1. □

Given the initial status $\mathbf{s}[0]$, the closed-form expression for the conditional expectation $E(\mathbf{s}[k] \,|\, \mathbf{s}[0])$ is given by

$$E(\mathbf{s}'[k] \,|\, \mathbf{s}[0]) = \mathbf{s}'[0]A^k, \tag{2.12}$$

We can show this relation by induction. First, we see that (2.12) holds for $k = 1$ because

$$E(\,\mathbf{s}[1] \,|\, \mathbf{s}[0]\,) = p_1[1]\begin{bmatrix}1\\0\\\vdots\\0\end{bmatrix} + p_2[1]\begin{bmatrix}0\\1\\\vdots\\0\end{bmatrix} + \cdots + p_n[1]\begin{bmatrix}0\\\vdots\\0\\1\end{bmatrix} \quad \text{where } \mathbf{p}[1] = A'\mathbf{s}[0]$$

$$= \mathbf{p}[1] \tag{2.13}$$

$$= A'\mathbf{s}[0]. \tag{2.14}$$

Now given that (2.12) holds up to $k \leq m$, we can write

$$\begin{aligned} E\Big(\mathbf{s}[m+1]\,\Big|\,\mathbf{s}[0]\Big) &= E\Big(E\Big(\mathbf{s}[m+1]\,\Big|\,\mathbf{s}[m]\Big)\,\Big|\,\mathbf{s}[0]\Big) && \text{by iterated expectation} \\ &= E\Big(A'\mathbf{s}[m]\,\Big|\,\mathbf{s}[0]\Big) && \text{by the same reasoning as (2.14)} \\ &= A'(A')^m\mathbf{s}[0] && \text{by linearity of expectation operator} \\ &= (A')^{m+1}\mathbf{s}[0] && (2.15) \end{aligned}$$

By induction, this shows (2.12) must hold for all k. Taking the expectation of (2.15) with respect to $\mathbf{s}[0]$, we then get

$$E(\mathbf{s}[k]\,) = (A')^k E(\mathbf{s}[0]), \quad \text{or} \quad E(\mathbf{s}'[k]) = E(\mathbf{s}'[0])A^k \tag{2.16}$$

In view of Proposition 2.13, eq. (2.16) thus provides a closed-form expression for the PMF governing the system at all time instants k.

The behavior of A^k as $k \to \infty$ is best explained in connection with the structure of $\Gamma(A)$. We will present the result for the most important case first, namely that of ergodic graphs or the associated (ergodic) chains, and then proceed to more general structures.

2.3.2 Case I: Ergodic Markov Chains

If $\Gamma(A)$ is ergodic, then $\lambda = 1$ is the dominant eigenvalue and all other eigenvalues have strictly smaller magnitude (see [16] p. 117). The right eigenvector of $\lambda = 1$ is clearly $\mathbf{1}$, because $A\mathbf{1} = \mathbf{1}$. The left eigenvector π, which is positive, is assumed to be normalized so that $\pi'\mathbf{1} = 1$. Then the ergodicity of A implies that

$$\lim_{k\to\infty} A^k = \mathbf{1}\pi'.$$

Therefore,

$$E(\mathbf{s}'[k] \,|\, \mathbf{s}[0]) = \mathbf{s}'[0]A^k \to \mathbf{s}'[0]\mathbf{1}\pi' = \pi'. \tag{2.17}$$

That is, regardless of $\mathbf{s}[0]$, the system will eventually be in state i with probability π_i. The PMF vector π is generally called the vector of *steady-state probabilities* of A.

In terms of the hopping-dot picture of Example 2, the ergodicity of $\Gamma(A)$ ensures the dot will forever roam about $\Gamma(A)$ without ever being trapped inside any group of nodes. Also, the fact that the same steady-state probabilities are reached regardless of the starting status means that if the dot has hopped around long enough, then it will look as though any node could have been the starting status.

2.3.3 Case II: Irreducible Periodic Chains

Now we assume that $\Gamma(A)$ consists of a single class of period $d > 1$. Recall from Theorem 2.6 that the nodes in this case can be partitioned into d subclasses. Assume that nodes within the same

subclass have consecutive indices so that eq. (2.7) applies, i.e.,

$$A^d = \begin{bmatrix} Q_1 & & \\ & \ddots & \\ & & Q_r \end{bmatrix} \tag{2.18}$$

Each Q_i corresponds to an ergodic Markov chain (see [16], Ex. 4.13) and thus has its own vector of steady state probabilities $\pi_{(i)}$. Hence,

$$\lim_{k\to\infty} A^{dk} = \begin{bmatrix} \mathbf{1}_{n_1}\pi'_{(1)} & & \\ & \ddots & \\ & & \mathbf{1}_{n_d}\pi'_{(d)} \end{bmatrix} \tag{2.19}$$

where n_i is the dimension of Q_i.

Eq. (2.19) shows that if we take snapshots of the system every d steps, then $\Gamma(A)$ can be effectively viewed as d disconnected, ergodic Markov chains. As explained in Theorem 2.6, the structure of $\Gamma(A)$ is such that the system status cycles around the loop of subclasses every d steps. Thus, whichever subclass $\mathbf{s}[0]$ starts from will be the subclass within which $\mathbf{s}[kd]$ lies for all k.

2.3.4 Case III: General Markov Chains

For general Markov chains $\Gamma(A)$, there are many possibilities for the asymptotic behavior of A^k, depending on the structure of $\Gamma(A)$. In particular, A^k might or might not converge, or only certain submatrices of A^k may converge, but not others.

To illustrate this, suppose there are r recurrent classes $\Gamma(A)$ with periods $d_1, \cdots, d_r$ respectively. Assume without loss of generality that nodes are numbered in the form in (2.4):

$$A = \begin{bmatrix} A_1 & & & \\ & \ddots & & \\ & & A_r & \\ * & \cdots & * & C \end{bmatrix}.$$

The kth power of A is then of the following form

$$A^k = \begin{bmatrix} A_1^k & & & \\ & \ddots & & \\ & & A_r^k & \\ * & \cdots & * & C^k \end{bmatrix}. \tag{2.20}$$

The block-diagonal form in the top part of (2.20) implies that if $\mathbf{s}[0]$ is such that the chain starts in a *recurrent status*, i.e. a status inside some recurrent class, then it will remain in that class for all time. In the hopping dot analogy from Example 2, this is the situation in which the dot starts and thus remains inside some recurrent class A_i permanently, because, by definition, there is no path to lead the dot out of a recurrent class. Within the recurrent class A_i, the probability of occupying each status at any given time can be analyzed in the same way as in Secs. 2.3.2 and 2.3.3, because the matrix A_i is stochastic. Thus, if its period d_i is greater than 1, then A_i^k will not converge to a constant matrix but will rather converge to a periodic loop.

If the hopping dot initially starts in one of the transient classes, then it will eventually drift into one of the recurrent classes that are accessible from its starting point. This is confirmed by the fact that C^k approaches zero ([16], Ex. 4.9 and 4.10), which means that the probability of the system being in a status inside a transient class decreases to zero over time.

Chapter 3

Binary Influence Model

We now introduce the *binary influence model*, which generates a Markov process that models propagation on a network. The binary influence model is a special case of the general influence model to be introduced in subsequent chapters. We study this simplified version first, because it is easier to understand and serves well as a motivation for the full model.

3.1 Introduction

In the binary influence model, each node of the network graph has a *status* value that varies over time as it is 'influenced' by its neighbors. The status of each node at any given time step is assumed to be 0 or 1, which may represent any two different statuses such as 'on' vs. 'off', 'healthy' vs. 'sick', or 'normal' vs. 'failed'. Interaction only occurs between neighboring nodes as each node is influenced to imitate the status of its neighbors.

Because of the general way in which it is defined, the binary influence model can potentially illuminate our understanding of the qualitative behavior of a number of systems. In power systems, this model can be used as a highly simplified paradigm for cascading blackouts. Here the network graph would represent the power grid, and each node would be a substation or a power plant whose status value is amenable to a binary label. To simulate cascading failure, we can start with a network in which every node is in 'normal' state and then initiate a node failure by turning the status at some node to 'failed'. Cascading failure occurs when a failed node causes its neighbors to fail, and those neighbors induce more failures, and so on.

This same model can also be used to model an election process with two candidates. Each node is now a person, and the network graph represents the social connections that influence a person's opinion. The candidate whom he or she favors constitutes the status value. At each time step, the voter re-evaluates his or her choice, taking into account the current opinion of the neighbors. Our model will be general enough to take into account the various degrees to which a voter believes in his or her own previous decision and the degree to which a voter is influenced by his or her neighbors. As we shall see, for certain social networks, there will eventually be a consensus among the voters and only one candidate wins, regardless of the initial state.

The influence model can also provide a highly abstracted representation of other systems in diverse applications such as territorial species invasion, product popularity in marketing or the collapse of an economy system (where each node is a financial institution).

In the following sections, we will first define the binary influence model formally. Then we proceed to characterize different structures of the associated graphs, which will determine whether

all nodes will ultimately reach a common value, or whether the state will wander indefinitely. The probability of the nodes reaching a certain consensus will be calculated using an efficient method. Finally we explore further variations and extensions of the model.

3.2 Model Description

Assume that we are given an $n \times n$ stochastic matrix D, called the *network influence matrix*, or simply the *network matrix*. Notice that if D were being used to describe a Markov chain, it would have been called the state-transition matrix of the chain, but we assign it a different name because we are using it for a different purpose. The graph $\Gamma(D')$ will be called the *network influence graph*, or simply the *network graph*. Its nodes are referred to as *sites*. Since it is defined from D' as opposed to D, the network graph is just like a Markov chain, except all the edges are reversed. Thus, instead of having the sum of outgoing edges equal to 1, the sum of edges pointing *into* a site is 1. As we will see, this feature allows us to treat each edge weight as the relative amount of influence from the source node to the destination. An example of a network matrix and its graph is shown in Example 1.

Example 1: This is an example of a 3-site network graph. Notice that the sum of the edge weights into any site is 1.

$$D' = \begin{bmatrix} .2 & .4 & 0 \\ .7 & 0 & 1 \\ .1 & .6 & 0 \end{bmatrix}$$

□

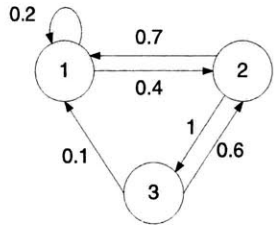


Figure 3.1: Example of a network graph $\Gamma(D')$.

At each time index k, site i has a *status*, denoted by $s_i[k]$, which can be either 0 or 1. The

vector

$$\mathbf{s}[k] \stackrel{\triangle}{=} [\, s_1[k] \ \ldots \ s_n[k] \,]'$$

is called the *state vector* of the graph at time k.

Convention on Terminology: Nodes, Sites, Statuses and States Since this thesis will introduce various kinds of graphs, we use different terms to refer to different types of graphs as well as their nodes. The term *node* is generic and applies to all types of graphs. A node on a network influence graph is called a *site*, and each site has a *status* that varies over time. A *state* refer to the collection of all statuses on a network graph at a given time. On Markov chains, a node is also called a *status*. The reason that the term status is used for both types of graphs is that in later chapters, we will allow an internal Markov chain to exist within each site. Hence, the status of a site will be the status of its Markov chain.□

The *binary influence model* refers to the following two evolution equations for calculating the vector $\mathbf{r}[k+1]$ of probabilities, and the realization of $\mathbf{s}[k+1]$:

$$\mathbf{r}[k+1] = D\,\mathbf{s}[k] \tag{3.1}$$

$$\mathbf{s}[k+1] = \mathit{Bernoulli}(\mathbf{r}[k+1]) \tag{3.2}$$

The initial state $\mathbf{s}[0]$ is independently realized from some given distribution, yielding a status of 0 or 1 for each site. The sequence of random vectors $\{\mathbf{s}[k]\}$ generated by (3.1)-(3.2) will be referred to as the *binary influence process*. In (3.1), $\mathbf{r}[k+1]$ is the length-n vector whose ith entry represents the probability that $s_i[k+1] = 1$. In (3.2), the actual status of each site (and hence the state of the network) is randomly realized. The operation $\mathit{Bernoulli}(\mathbf{r}[k+1])$ can be thought of as flipping n independent coins to realize the entries of $\mathbf{s}[k+1]$, where the probability of the ith coin turning up heads (status 1) is $r_i[k+1]$. Each $r_i[k+1]$ is a valid probability. To see this, we notice that $r_i[k+1] \geq 0$ because both D and $\mathbf{s}[k]$ are nonnegative, and $r_i[k+1] \leq 1$ because

$$r_i[k+1] = \sum_j d_{ij} s_j[k] \leq \sum_j d_{ij} = 1.$$

since each $s_j[k] \leq 1$.

The one-step dynamics of the binary influence model can be easily understood once viewed on the network graph $\Gamma(D')$. An edge (i,j) exists on this graph if the status of j can be influenced by the status of i. The weight on edge (i,j) can be interpreted as the amount of influence that i exerts on j relative to the total amount of influence that j receives. The total amount of influence

received by any site is thus equal to the sum of incoming edge weights, which is 1 because D is a stochastic matrix. The probability of a site having status 1 at the next stage is the sum of the weights on the edges from the neighbors whose statuses are currently 1. This sum, therefore, always has a value ranging between 0 and 1. The following example shows a particular run of the model.

Example 2: Figure 3.2 gives an example showing the first few steps of a particular run based on the graph in Example 1. The top row of graphs shows the realized states $\mathbf{s}[k]$. The status of each site, which is either a 0 or a 1, is written in the site. The bottom row of graphs shows the probabilities $\mathbf{r}[k]$. Inside each site is $r_i[k]$, the probability that that site has status 1 at time k. □

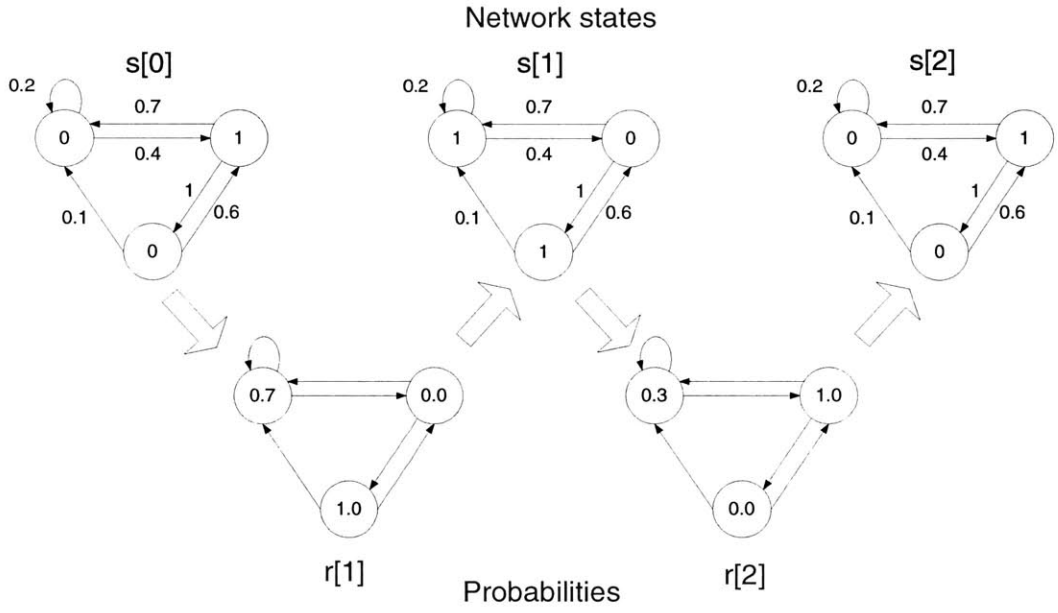


Figure 3.2: Example of a particular path of the binary influence process in its first few steps.

If all of a site's influencing neighbors have the same status, whether all 0's or all 1's, then that site will copy its neighbors' status with certainty in the next time step. If a site has a self-loop then it is one of its own influencers. This feature allows us to model situations where statuses tend to persist, because the self-loop would influence the site to repeat its previous status. In the power systems context, the self-loop can be used to model failures that are difficult to repair, for instance.

3.2.1 Comparison to Markov Processes

When compared to the evolution equations of Markov processes in (2.10)-(2.11), the binary influence model in (3.1)-(3.2) looks rather similar. However, these two models are not the same, and their

differences are as follow:[1]

- In a binary influence process, the state vectors $\mathbf{s}[k]$ can have multiple 1's among its entries, but in a Markov process, the status vector must have exactly one entry of 1.
- Intuitively, we can think of a binary influence process as having an active entity at each site. Each of these entities can be in one of two statuses (such as 'failed' or 'normal'). In a Markov chain of order m, there is only *one* active entity whose status is one of the m possibilities.
- In the binary influence model, the state vector $\mathbf{s}[k]$ multiplies D from the right, while in the Markov process, the status vector multiplies from the left:

$$\begin{aligned} \mathbf{r}[k+1] &= D\,\mathbf{s}[k] \qquad \text{(binary influence)} \\ \mathbf{p}'[k+1] &= \mathbf{s}'[k]A \qquad \text{(Markov)} \end{aligned}$$

- In a network graph $\Gamma(D')$, the sum of incoming edge weights at a site node is 1. In a Markov chain $\Gamma(A)$, the sum of outgoing edge weights at a status node is 1.
- In a network graph, edge weights are to be interpreted as influences, while in Markov chains, edge weights are the transition-probabilities.
- In the binary influence model, the vector of probabilities $\mathbf{r}[k]$ is not necessarily a PMF vector, but $\mathbf{p}[k]$ in the Markov process must be.

3.3 Model Analysis

A few fundamental questions arise regarding the influence model: What happens in the long run? Do the sites eventually converge to the same status? What structure of the network graph would permit such convergence? If they are convergent, what status would the sites take in the limit? How can the behavior be characterized when it is not convergent?

To answer these questions, it is possible in principle to study the influence model above by means of a Markov chain $\Gamma(G)$, where G is a state-transition matrix of order 2^n. Each status of $\Gamma(G)$ would correspond to one possible outcome of $\mathbf{s}[k]$. However, since the size of the state-transition matrix G grows exponentially in the number of sites, it is practically impossible to analyze G

[1] As a reminder, $\mathbf{s}[k]$ is called a *state* vector in the binary influence model, but it is called a *status* vector in the Markov process. Each entry $s_i[k]$ is called a status in both models.

directly for all but small values of n. It would be very difficult just to classify the nodes of $\Gamma(G)$ into transient and recurrent classes, let alone find the steady-state probabilities. In what follows, we will circumvent this problem by showing that certain important properties of G such as the recurrent classes as well as the probabilities of being absorbed into each of them can be analyzed with matrices of order n, as opposed to 2^n, due to the simple dynamics of the model.

By inspection, we can see that there are at least two obvious states that could be the terminal states of the binary influence model: the all-ones and the all-zeros states, or the *consensus states.* When all sites have reached either consensus, there can be no influence to switch to the opposite status, given how the model is defined. Hence, any consensus state will stabilize and remain that way forever.

Knowing that there are at least two kinds of eventuality, we can refine our questions regarding the convergence:

- Does the binary influence model always end up in one of these two consensus steady-states regardless of the initial condition? Are there any other final states beside the consensus states? If so, how does the structure of the network graph come into play?
- What are the probabilities of reaching the various recurrent classes?
- How do these results generalize to general influence models, where the number of allowable statuses is greater than two?

3.3.1 A Fundamental Proposition

The following proposition is a simple but powerful result regarding the probability governing the status of a given site. It is very similar to Proposition 2.13.

PROPOSITION 3.1

$$E(s_i[k]) = Prob(\textit{site } i \textit{ is in status 1 at time } k).$$

Proof. Again, this follows directly from the fact that $s_i[k]$ is a binary-valued random variable. Hence, its expected value is the probability of its being in status 1. □

Following similar steps as in Section 2.3.1, we arrive at the following identities

$$E(\mathbf{s}[k+1] \mid \mathbf{s}[k]) = \mathbf{r}[k+1] = D\mathbf{s}[k]. \tag{3.3}$$

It is easy to show that this leads to

$$E(\mathbf{s}[k] \mid \mathbf{s}[0]) = D^k \mathbf{s}[0]. \tag{3.4}$$

$$E(\mathbf{s}[k]) = D^k E(\mathbf{s}[0]). \tag{3.5}$$

3.3.2 Graphical Interpretation

To obtain a more physical interpretation of (3.4) in terms of the graph $\Gamma(D')$, consider the transpose of the equation,

$$E(\mathbf{s}'[k] \mid \mathbf{s}[0]) = \mathbf{s}'[0](D')^k.$$

which is equivalent to

$$E(s_j[k] \mid \mathbf{s}[0]) = \sum_{i=1}^{n} s_i[0] \, [(D')^k]_{ij}$$

In this summation, only the term with $s_i[0][(D')^k]_{ij} > 0$ would contribute to the sum. This is possible when $s_i[0] = 1$ and $[(D')^k]_{ij} > 0$. By Theorem 2.1, $[(D')^k]_{ij} > 0$ if and only if a path of exactly length k from site i to j exists on the graph $\Gamma(D')$. This means that the probability $Prob(s_j[k] = 1 \mid \mathbf{s}[0]) = E(s_j[k] \mid \mathbf{s}[0])$ comprises contributions from each site whose initial status is 1 *and* from which site j can be reached by a path of k steps. Similarly, $Prob(s_j[k] = 0 \mid \mathbf{s}[0])$ can be shown to be contributed by the sites that are k steps away and that have 0 as their initial status. To summarize, we have established the following.

COROLLARY 3.2

Given a realization of the initial state $\mathbf{s}[0]$, *a site* j *on* $\Gamma(D')$ *has a positive probability of being in status 1 (respectively 0) at time* k *if and only if there exists some path that*

(a) has length exactly k,

(b) originates at a source site whose initial status is 1 (respectively 0), and

(c) terminates at j.

3.3.3 Limitations

Although Proposition 3.1 and eq. (3.5) give a closed-form expression for the probability of each individual site being in status 1, they are still too weak to characterize the state of the whole network. The reason is that the expected value $E(\mathbf{s}[k])$ is only the *ensemble average* —the average of $\mathbf{s}[k]$ from repeated experiments at a fixed time index k. Even if $\lim_{k\to\infty} E(\mathbf{s}[k])$ converges to a constant vector $c\mathbf{1}$, there are still many possible interpretations of this limit. Does it mean that the binary influence process will always reach a consensus, and c is just the probability of reaching the all-ones consensus? Or does it mean that the process will forever flip around at random, but in such a way that the status of a node at large time index, when averaged over repeated experiments, will have a value of c? Or does the process get stuck in a deterministic, finite loop of states, where the status of a node at large time index has an ensemble average of c? Or can all these cases happen depending on the particular experiments? Proposition 3.1 alone does not exclude the possibilities of these final states.

Thus, it is necessary for us to make a more definite statement on the final states of a given binary influence process. As with the role of $\Gamma(A)$ in the analysis of its Markov chain, the structure of the graph $\Gamma(D')$ plays an important role in the convergence of $\mathbf{s}[k]$. We will start by analyzing the case of ergodic network graphs, then proceed to a more general network graph structure. The following sections will repeatedly invoke the definitions and results discussed in Chapter 2, particularly those from Section 2.1.

3.4 Ergodic Network Graphs

We now assume that $\Gamma(D')$ is an ergodic (i.e. irreducible, and aperiodic) graph. In terms of convergence behavior, ergodic network graphs are among the simplest to understand and thus, serve as our starting point. As we will show, the only recurrent states (i.e., a state on a recurrent class) of these graphs are the consensus states.

Let $\Gamma(D')$ be an ergodic network graph. Let V be the set of sites on $\Gamma(D')$, and $i \in V$ be some site. Define the *mth-level neighbor set of* i for an integer $m \geq 0$ as

$$T^m(i) \stackrel{\triangle}{=} \{v \in V \mid \text{ there exists a path of exactly } m \text{ steps from site } i \text{ to } v \}$$

By Corollary 3.2, $T(i)$ represents all the sites that could be influenced by i in one step. Thus, if site i is currently in status 1, then there is a positive probability that every site in $T(i)$

would be in status 1 in the next time step. Similarly, $T^2(i), T^3(i), \ldots$, are all the sites that could be influenced to assume status 1 by i to in 2, 3 steps and so on. To summarize, we have the following Lemma.

LEMMA 3.3
Let $i \in V$ and let $T^m(i)$ be as defined above.

$$Prob\Big(\textit{every site in } T^m(i) \textit{ has status 1 at time } k \;\Big|\; \textit{site } i \textit{ has status 1 at time } 0\Big) > 0$$

Next, we exploit the fact that D is ergodic to claim that there is a path of any desired length between any given pair of sites, provided that the path is sufficiently long.

LEMMA 3.4
If D is ergodic, $D^m > 0$ for all $m \geq n - 1$.

Proof. See [26], Corollary 6.2.20. □

COROLLARY 3.5
Given D is ergodic, for all $m \geq n - 1$, and for any site i, $T^m(i) = V$.

Proof. By Theorem 2.1, there is a path of length m from site i to j on $\Gamma(D')$ if and only $[(D')^m]_{ij} > 0$. But since $D^m = [(D')^m]' > 0$ by Lemma 3.4, there exists some path from i to any site in exactly m steps. Hence $T^m(i) = V$. □

THEOREM 3.6
The only recurrent states of a binary influence model in the case of an ergodic network graph are the all-ones and all-zeros consensus states.

Proof. If the initial state $\mathbf{s}[0]$ is the all-zeros consensus, then it will remain that way for all time. If it is not, let i be any site that has status 1 under $\mathbf{s}[0]$. By Corollary 3.5, $T^m(i) = V$ for $m \geq n - 1$, and by Lemma 3.3, there is a positive probability that at time m, all sites on $\Gamma(D')$ will be in status 1. Since the all-ones consensus is an recurrent state, the fact that it can reached from $\mathbf{s}[0]$ shows that $\mathbf{s}[0]$ is either a transient state, or the all-ones state itself. Since $\mathbf{s}[0]$ is arbitrary, there can be no other recurrent state beside the consensus states. □

3.4.1 Probability of Consensus

Having established that all ergodic autonomous classes must reach a consensus, we can apply Proposition 3.1 to determine the probability of reaching each consensus.

From Section 2.3.2, if D is ergodic, then

$$\lim_{k\to\infty} D^k = \mathbf{1}\pi' \tag{3.6}$$

where π is the left eigenvector corresponding to the eigenvalue at 1, which has been normalized so that $\pi'\mathbf{1} = 1$. If the stochastic matrix D had been used as a state-transition matrix of a Markov chain, this vector would have been the vector of steady-state probabilities. Because D is ergodic, the column vector π will be strictly positive entry-wise (see [16], p. 117). Combining (3.4) with (3.6), we conclude that

$$\begin{aligned} \lim_{k\to\infty} E(\mathbf{s}[k] \mid \mathbf{s}[0]) &= \lim_{k\to\infty} D^k \mathbf{s}[0] \\ &= \mathbf{1}\pi'\mathbf{s}[0] \end{aligned} \tag{3.7}$$

Eq. (3.7) indicates that all sites have the same probability of $\pi'\mathbf{s}[0]$ of reaching status 1. This matches what we expect since we know that the state must eventually reach a consensus. The following theorem summarizes the results of this section.

THEOREM 3.7

Let $\Gamma(D')$ be ergodic. Let π' be defined as in (3.6). Then the probability that the influence process starting from initial state $\mathbf{s}[0]$ will eventually settle in the all-ones consensus state is $\pi'\mathbf{s}[0]$. The probability of reaching the all-zeros consensus state is thus $1 - \pi'\mathbf{s}[0]$.

3.5 Periodic Irreducible Graphs

3.5.1 Introduction

Now we investigate the recurrent states of the binary influence process when $\Gamma(D')$ is irreducible and periodic. We will analyze it by reducing this to a problem of convergence on d separate ergodic graphs.

Clearly, the all-ones and all-zeros states are still two of the recurrent states of periodic irreducible graphs, as they are for all network graphs. We will show that on periodic graphs, the

consensus states are just two special cases of a class of recurrent states called limit cycles.

Recall from Section 2.3.3 that when $\Gamma(D')$ is periodic with a period d, the sites can be partitioned into d subclasses that are connected in a cycle such as in Figure 2.4 and 2.5. Consider a period-2 graph as in Figure 3.3. Assume that $\mathbf{s}[0]$ is such that all the sites in the same subclass

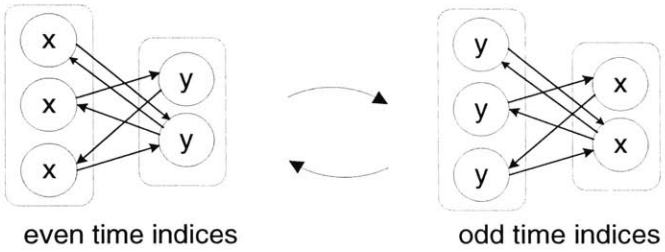


Figure 3.3: Example of a period-2 graph in a limit cycle.

have the same status, which in this figure is status x for the left subclass and y for the right. At even-numbered time indices, the left subclass is influenced only by sites in status y, and the right subclass only by sites in status x. Therefore, these statuses will deterministically switch back and forth between the two subclasses. In other words, the network graph is stuck in a loop, which we call a *limit cycle*. In what follows, we generalize the above example to arbitrary periodic irreducible graphs, and show that the limit cycles are the only possible recurrent states. A consensus state can then be regarded as a special case of limit cycle where the statuses of all subclasses are the same.

3.5.2 Analysis

Without loss of generality, all sites within the same subclass are assumed to have consecutive indices, so that we can express D as

$$D = \begin{bmatrix} 0 & & & R_d \\ R_1 & 0 & & \\ & \ddots & \ddots & \\ & & R_{d-1} & 0 \end{bmatrix}, \quad \text{or} \quad D' = \begin{bmatrix} 0 & R'_1 & & \\ & \ddots & \ddots & \\ & & 0 & R'_{d-1} \\ R'_d & & & 0 \end{bmatrix}. \tag{3.8}$$

Recall that an entry $[D']_{ij}$ represents the amount of influence that site i exerts on site j. With the configuration in (3.8), on the graph $\Gamma(D')$, subclass i influences subclass $i+1$ and so on, i.e., the influences are "forward pointing."

Let $\mathbf{u}_i[k]$ denote the state vector of the ith subclass, so that the state vector of the entire

graph can be written as

$$\mathbf{s}[k] = \begin{bmatrix} \mathbf{u}_1[k] \\ \vdots \\ \mathbf{u}_d[k] \end{bmatrix} \tag{3.9}$$

Because the subclasses are connected in a cyclic manner, the realized value of $\mathbf{u}_i[k]$ will depend only on the previous state $\mathbf{u}_{i-1}[k-1]$ of its predecessor subclass and nothing else. The portion $\mathbf{u}_{i-1}[k-1]$ will, in turn, depend on $\mathbf{u}_{i-2}[k-2]$ and so on. If we keep tracing the dependencies backward in time, we will find that $\mathbf{u}_i[k]$ depends only on the initial state $\mathbf{u}_j[0]$ of a certain subclass, and is independent of the initial condition of any other subclass $\mathbf{u}_p[0]$ for $p \neq j$. Moreover, different subclasses at time k depend on different portions of $\mathbf{s}[0]$. This observation leads to the following conclusion, which will greatly simplify our analysis by allowing us to consider the convergence of each subclass separately.

LEMMA 3.8

Assuming the initial state $\mathbf{s}[0]$ *is given, for any* $i \neq j$ *and a fixed* k, $\mathbf{u}_i[k]$ *and* $\mathbf{u}_j[k]$ *are statistically independent.*

Note that if $\mathbf{s}[0]$ has not been given, then we cannot conclude that $\mathbf{u}_i[k]$ is independent of $\mathbf{u}_j[k]$, because the portions of $\mathbf{s}[0]$ that they are dependent on might be correlated.

The dth power of D' has a block diagonal form:

$$D^d = \begin{bmatrix} D_1 & & \\ & \ddots & \\ & & D_d \end{bmatrix}, \quad \text{or} \quad (D')^d = \begin{bmatrix} D'_1 & & \\ & \ddots & \\ & & D'_d \end{bmatrix} \tag{3.10}$$

where each $D'_m = R'_m \cdots R'_d R'_1 \cdots R'_{m-1}$. Each D_m must be a valid network matrix because each row of D^d sums to unity, which means the rows of D_m sum to unity as well. The block-diagonal form of $(D')^d$ in (3.10) implies that the graph $\Gamma\big((D')^d\big)$ comprises d disconnected graphs, each one being equivalent to the graph $\Gamma(D'_m)$ for some $m = 1, \ldots, n$.

Example 3: Consider the network matrix D given by

$$D' = \begin{bmatrix} & & 0 & 0.5 \\ & & 1 & 0 \\ & & 0 & 0.5 \\ 0 & 0 & 1 & \\ 1 & 1 & 0 & \end{bmatrix}, \quad \text{so} \quad (D')^2 = \begin{bmatrix} 0.5 & 0.5 & 0 & \\ 0 & 0 & 1 & \\ 0.5 & 0.5 & 0 & \\ & & & 0 \;\; 0.5 \\ & & & 1 \;\; 0.5 \end{bmatrix} = \begin{bmatrix} D'_1 & \\ & D'_2 \end{bmatrix}.$$

This is a period-2 irreducible graph. Figure 3.4a depicts the graphs $\Gamma(D')$, while Figure 3.4b shows $\Gamma\big((D')^2\big)$. Notice how $\Gamma\big((D')^2\big)$ consists of two disconnected graphs $\Gamma(D'_1)$ and $\Gamma(D'_2)$, both of which are ergodic. The numbers $1, 2, \ldots, 5$ in the figure are the site indices, not the statuses. □

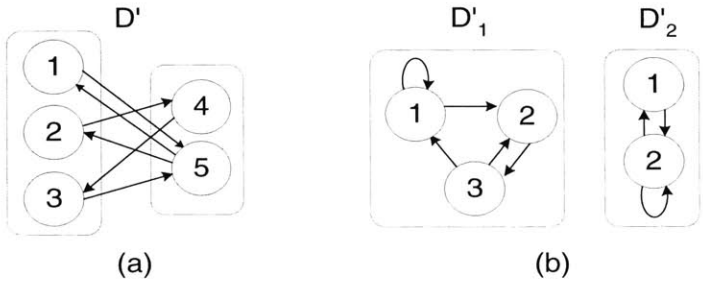


Figure 3.4: Example of a period-2 irreducible graph (a) $\Gamma(D')$, (b) $\Gamma\big((D')^2\big)$.

In general, the graphs $\Gamma(D')$ and $\Gamma\big((D')^d\big)$ are different, as Example 3 above shows, but their sites have a one-to-one correspondence with each other. Specifically, we match site 1 on $\Gamma(D')$ to site 1 on $\Gamma\big((D')^d\big)$, site 2 on $\Gamma(D')$ to site 2 on $\Gamma\big((D')^d\big)$, and so on. Since $\Gamma\big((D')^d\big)$ comprises the graphs $\{\Gamma(D'_m) \,|\, m = 1, \ldots, d\}$, we can uniquely match each site on any graph $\Gamma(D'_m)$ to a unique site on $\Gamma(D')$ as well. For instance, in Figure 3.4, sites 1, 2, 3 of $\Gamma(D'_1)$ are matched to sites 1, 2, 3 of $\Gamma(D')$ respectively, while 1,2 of $\Gamma(D'_2)$ are matched to sites 4, 5 of $\Gamma(D')$. More generally, for a site i on graph $\Gamma(D'_m)$, define $\nu(i, m)$ as the index of the matching site on $\Gamma(D')$.

LEMMA 3.9

Let i, j be any two sites on $\Gamma(D'_m)$. A path of length k from i to j on $\Gamma(D'_m)$ exists if and only if there exists a path of length kd from site $\nu(i, m)$ to site $\nu(j, m)$ on $\Gamma(D')$.

Proof. Because of (3.10), the (i, j)th entry of $(D'_m)^k$ is the entry in position $\big(\nu(i, m), \nu(j, m)\big)$ of $(D')^{kd}$. Combining this with Theorem 2.1, we have proved the lemma. □

In Figure 3.4, one can verify that the graphs conform with Lemma 3.9 by checking that a

path of length $2k$ in $\Gamma(D')$ always has at least one corresponding path of length k in either $\Gamma(D'_1)$ or $\Gamma(D'_2)$.

We will now show that eventually each subclass will reach its own consensus. First, fix a graph $\Gamma(D'_m)$ and fix a site i on it. Each $\Gamma(D'_m)$ will be an ergodic graph (see Sec. 2.3.3). Recall from Corollary 3.5 that for all sufficiently large k, the set $T^k(i)$ will eventually cover every site on $\Gamma(D'_m)$. Using Lemma 3.9, this implies that for all sufficiently large k, $T^{kd}\big(\nu(i,m)\big)$ must include every site in the mth subclass on $\Gamma(D')$. The general result of Corollary 3.2 states that the existence of a path is equivalent to the ability of the source site to influence the destination, so we conclude that regardless of the initial state $\mathbf{s}[0]$, there is always a positive probability that all sites from the mth subclass on $\Gamma(D')$ will reach a subclass consensus at some time kd. Once a subclass reaches the consensus, the state of that subclass will never change when viewed as snapshots every d steps. Since this argument must apply to each subclass m, every subclass will eventually reach its own consensus at time kd for sufficiently large k. The consensuses reached by different subclasses could be different. The statuses of these subclasses cycle around indefinitely in a loop of d states, similar to the situation shown in Figure 3.3. The following theorem summarizes this main message of this section.

THEOREM 3.10

The only recurrent states of a periodic irreducible network graph are the limit cycles.

3.5.3 Probability of Limit Cycles

A binary influence process of an irreducible periodic network graph is said to have *reached limit cycle* $\mathbf{c} \stackrel{\triangle}{=} (c_1, \ldots, c_d)$ if every site in the ith subclass has status c_i at all times kd for all k sufficiently large. For time indices that are not a multiple of d, the statuses of the subclasses can be inferred by shifting the statuses cyclically by the appropriate amount.

Given an initial state $\mathbf{s}[0]$, which limit cycles will be reached, and with what probability? Using Lemma 3.8,

$$Prob(\text{process reaches limit cycle } \mathbf{c}) = \prod_{i=1}^{d} Prob(\text{subclass } i \text{ reaches consensus } c_i).$$

Each D_i defined in (3.10) is ergodic, so

$$\lim_{k\to\infty} D_i^k = \mathbf{1}\pi_{(i)}{}'$$

where $\pi_{(i)}$ is the normalized left eigenvector of D_i corresponding to the eigenvalue at 1. Again, we can apply Corollary 3.2 to obtain the probability of the subclass consensus. If subclass i starts from initial condition $\mathbf{u}_i[0]$, then at time kd it will reach the all-ones consensus with probability

$$D_i^k \mathbf{u}_i[0] \to \mathbf{1}\pi_{(i)}{}' \mathbf{u}_i[0].$$

To summarize, we conclude the analysis of the binary influence process on a periodic irreducible graph with the following statement.

THEOREM 3.11
For a given $\mathbf{s}[0]$, *let* $\mathbf{u}_i[k]$ *and* $\pi_{(i)}$ *be defined as above. The probability of a periodic irreducible network graph reaching limit cycle* $(c_1, \ldots, c_d)$ *is* $\prod_{i=1}^{d} p_i$ *where*

$$p_i = \begin{cases} \pi'_{(i)} \mathbf{u}_i[0] & \text{if } c_i = 1 \\ 1 - \pi'_{(i)} \mathbf{u}_i[0] & \text{if } c_i = 0 \end{cases}$$

After convergence, the state of the periodic graph at any time k that is not a multiple of d can be inferred from the limit cycle by cyclically shifting the states through (k mod d) steps.

3.6 General Network Graphs

3.6.1 Autonomous and Dependent Classes

This section analyzes the binary influence process on general network graphs. If the network graph is irreducible, then it must be either ergodic or periodic, both cases of which we have already dealt with in Secs. 3.4 and 3.5. We therefore assume for the rest of the section that the network graph $\Gamma(D')$ consists of multiple classes. We assume for simplicity that $\Gamma(D')$ is a connected graph. Otherwise, the following analysis can be applied to each component separately.

Recall from Sec. 2.1.1 that a class is a set of nodes within which all nodes communicate. Any reducible graph (i.e., one having multiple classes) must have at least one autonomous class, which is one that has only outgoing edges but no incoming edges, and at least one dependent class, which is any non-autonomous class. For example, in Figure 2.3, the class on the left is dependent.

The important thing to note is that an autonomous class always settles into either a consensus state or a limit cycle, because it receives no external influence. A dependent class, on the other hand, may or may not settle down to either type of recurrent state. For example, consider the

graph in Figure 3.5. The dependent class (enclosed in a dotted line) receives external influences

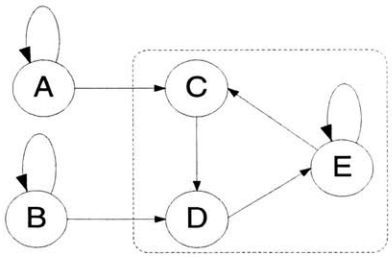


Figure 3.5: Example of a dependent class with two potentially conflicting external influences.

from sites A and B, both of which are separate autonomous. If A and B happen to have the same status, then regardless of the initial status of sites C, D, and E, the latter will eventually converge to a consensus. If A and B have different statuses, then the dependent class will never settle to a steady state. Rather, the statuses of A and B will forever circulate randomly within the dependent class.

More generally, classes on a network graph can be arbitrarily connected so long as they are not cyclical. It becomes tedious, if not impossible, to list the general necessary and sufficient conditions for a dependent class A to reach a limit cycle. This would not only depend on how A is connected to the external classes, but also on what their periods are relative to that of A, and ultimately on how those influencing classes are influenced themselves. However, we list here some of the conditions that are necessary for A to reach a limit cycle or a consensus:

- Each of the classes influencing A must itself reach a limit cycle.
- For each of the classes influencing A, the period of the limit cycles has to divide that of A.

If these conditions are not met, then the state of the dependent class A, as well as the states of all the classes downstream, will be perpetually random.

3.6.2 'Evil Rain' Model

One special form of binary influence model that we would like to pay special attention to is the *'evil rain'* model. This model can potentially serve as an abstract mathematical representation of propagating failures on a network. Unlike the previous cases of irreducible graphs, which always settle to a consensus or a limit cycle, the evil rain model is concocted with the intention of representing certain features of real networked systems. A real system is neither permanently robust,

nor completely non-functional. Take the power grid, for instance. At any time, a certain fraction of the substations could be down for a variety of reasons such as failure maintenance, human error, or repair from previous accidents. This variation of statuses across the network causes some parts of the network to be more vulnerable to small failures than others. A large cascading failure is often a result of a small initiating failure that happens to occur at a place and time where the conditions allow it to spread widely. With this motivation, we introduce the evil rain model below.

The evil rain model is a special form of the binary influence model in which the network graph consists of one dependent class and two single-site autonomous classes as shown in Figure 3.6. The two autonomous sites have a fixed status of 1 or 0 and are the sole suppliers of their

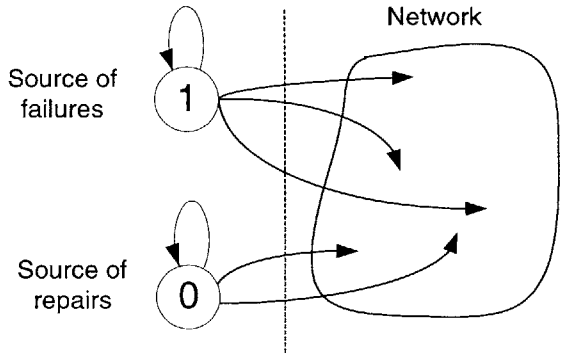


Figure 3.6: The Evil Rain Model.

respective statuses into the system. We think of the status-1 site as the 'evil rain' that sprinkles failures into the system. The status-1 then circulates and spreads around until it is overwritten by the status-0, which acts as the 'repairman' who runs around trying to fix the damage caused by the spreading evil rain.

By numbering the sites so that the evil rain and the repairman are the first two sites respectively, the influence matrix and the state vector of an evil rain model have the forms

$$D = \begin{bmatrix} 1 & & \\ & 1 & \\ \mathbf{e}_1 & \mathbf{e}_2 & F \end{bmatrix}, \qquad \mathbf{s}[k] = \begin{bmatrix} 1 \\ 0 \\ \widetilde{\mathbf{s}}[k] \end{bmatrix} \tag{3.11}$$

where each $\mathbf{e}_i = [\, e_{i1} \cdots e_{in} \,]'$ is a column vector of length n, the number of dependent sites. The entry e_{ij} is the influence from the ith autonomous site to the jth dependent site. The matrix F describes the connections within the dependent class. Since the statuses of the first two sites are fixed, we can omit them from the evolution equations (3.1)-(3.2). The reduced equations of the

evil rain model can thus be described as

$$\widetilde{\mathbf{p}}[k+1] = F\widetilde{\mathbf{s}}[k] + \mathbf{e}_1 \tag{3.12}$$

$$\widetilde{\mathbf{s}}[k+1] = Bernoulli(\widetilde{\mathbf{p}}[k+1]). \tag{3.13}$$

The state $\widetilde{\mathbf{s}}[k]$ will never reach a consensus or limit cycle, because the evil rain and the repairman sites will keep randomly injecting 1's and 0's into the system. A basic question that would be of interest for practical application is: what is the average number of sites in status 1 at a given time?

To find the average number of sites in status 1 at time k, we take the expectation of (3.13) and substitute into (3.12) to get

$$E(\widetilde{\mathbf{s}}[k+1]) = FE(\widetilde{\mathbf{s}}[k]) + \mathbf{e}_1 \tag{3.14}$$

As long as some element of e_{ij} is strictly positive, the eigenvalues F will be strictly inside the unit circle. Then (3.14) can be viewed as an linear system driven by constant input $\mathbf{e}_1$ from initial state $\widetilde{\mathbf{s}}[0]$. The steady-state solution to this equation would exist can be easily shown to be

$$\lim_{k\to\infty} E(\widetilde{\mathbf{s}}[k]) = (I-F)^{-1}\mathbf{e}_1. \tag{3.15}$$

As expected, this solution does not depend on $\widetilde{\mathbf{s}}[0]$. This is because the injection of 1's and 0's will eventually "wash out" all information of the initial state. The expected total number of status-1 sites in steady-state is simply the sum of the expected status of each site. We therefore have the following result.

THEOREM 3.12
In the evil rain model with at least some $e_{ij} > 0$, the expected number of status-1 sites in steady-state is $\mathbf{1}'(I-F)^{-1}\mathbf{e}_1$.

3.7 Dual Process: Coalescing Random Walks

As already mentioned, the binary influence model is similar to a model that has been studied under the label of *voter models*. These stochastic systems were first introduced along with their dual process called *coalescing random walks* [25]. Since then the analysis of these two processes has been significantly interlinked, as an observation in one system often provides insights into the other. In this section, we will touch on coalescing random walks as an alternative way of understanding

the binary influence model.

3.7.1 Alternative Descriptions of the Binary Influence Model

There are two different ways to describe a binary influence process. The first method, which is the method we have used so far, is to describe the evolution of the sequence of state vectors $\{\mathbf{s}[k]\}$ as a function of time. The other method, which will be the focus of this section, is the outcome matrix. This alternative description of the binary influence model not only provide us with an alternative view of the model, but also serve as the bridge between the model and its dual, the coalescing random walk processes.

Recall that in the binary influence model the probability of a site on the network graph $\Gamma(D')$ being in status c at time $k+1$ is equal to the sum of weights from the influencing neighbors who are in status c at time k. An equivalent way of describing the influence is to say that a site chooses a neighbor at random with a probability determined by the influencing weight, and then copies the status of that site. The crucial observation here is that site i chooses its neighbor with the same set of probabilities *regardless* of its current status or its neighbors'. In other words, at each time step, site i decides which site it wants to copy from by rolling an n-faced die (where n is the total number of sites) with a probability of face j showing up being equal to $[D]_{ij}$.

The analogy with die throwing allows us to define the *outcome matrix* U as follows. Let U have n rows, and an infinite number of columns. Each row corresponds to a site, and each column corresponds to a time step. In the ith row of U, the entries are generated by repeatedly throwing an n-faced die with a PMF specified by the ith row of the network matrix D. For example, if

$$D = \begin{bmatrix} .2 & .8 & 0 \\ .4 & .3 & .3 \\ .2 & .1 & .7 \end{bmatrix} \tag{3.16}$$

then we will use the first row of D to generate the first row of U. To do that we will throw a 3-faced die with a PMF of $[.2\ .8\ 0]$. The results of the throws, say $[2\ 2\ 1\ 2\ \cdots]$, are then entered as the first row of U. Other rows of U are generated from the PMF's given by the corresponding rows of D. Suppose the realized the outcome matrix is

$$U = \begin{bmatrix} 2 & 2 & 1 & 2 & \cdots \\ 3 & 1 & 3 & 1 & \cdots \\ 3 & 3 & 2 & 1 & \cdots \end{bmatrix}. \tag{3.17}$$

The interpretation of $U = [u_{ij}]$ is that site i copies from site u_{ij} during the transition from time $j-1$ to j. To find out the status of a site at a given time, we just have to trace the history of copied sites to the left until we reach time 0. For example, if we assume the initial state of the graph is $\mathbf{s}[0] = [a\ b\ c]'$, Figure 3.7 shows that at time 4, all sites have just converged to consensus c. In the voter model literature, the matrix U is presented in a slightly different form called the

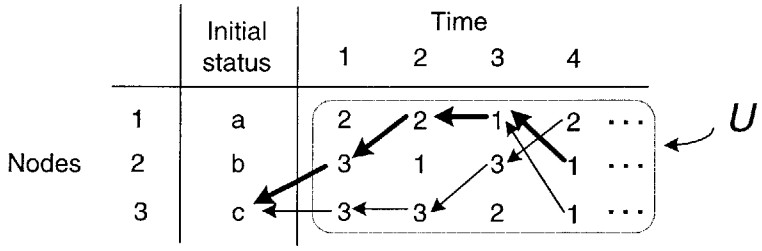


Figure 3.7: Backward tracing on the outcome matrix.

percolation substructure [14, 21, 25, 34], but the essential ideas are the same as ours.

For a given outcome matrix U, let $p_{i,k}$ be the backward path originating from site i at time k and terminating at some site at time 0. For example, in Figure 3.7, the solid line denotes path $p_{2,4}$. Let $\eta_U[k]$ be a length-n vector with the ith entry defined as

$$(\eta_U[k])_i = \text{ the site at which path } p_{i,k} \text{ terminates based on realization } U. \tag{3.18}$$

Given a sequence of $\{\eta_U[k]\}$ as a function of k, we can obtain the binary influence process by the relation:

$$s_i[k] = s_j[0] \text{ where } j = (\eta_U[k])_i\,.$$

In particular, if $\eta_U[k] = j\mathbf{1}$, then all sites at time k are holding a status that is equal to the initial status of site j; that is, the binary influence process has already reached a consensus by the time k, and the consensus is $s_j[0]$. It will also imply that $\eta_U[k+m] = j\mathbf{1}$ for all $m \geq 0$;

From Figure 3.7, a few observations can be made:

- Suppose we trace to the left all paths that originate from each site at some fixed time. As the paths move left, they tend to merge with each other so that the number of distinct paths either decreases, or remains constant.
- If all the backward paths originating from the same time k have merged into one before reaching time 0, then $\mathbf{s}[k]$ must be a consensus state. Even if they have not merged into one at the time

0, $\mathbf{s}[k]$ could still be a consensus. To see this, consider a binary influence process with initial state $\mathbf{s}[0]$. Let A be the set of sites whose initial status is 1, let A_c be the rest of the sites whose initial status is 0. As long as every site in $\eta_U[k]$ lies in A, as depicted in Figure 3.8, then

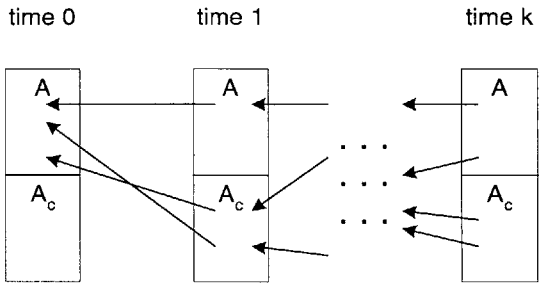


Figure 3.8: A backward tracing for the influence process that reaches a consensus. All sites in $\eta_U[k]$ is in A.

$\mathbf{s}[k]$ will have reached a consensus of all-ones.

- Although the outcome matrix U and its corresponding percolation substructure extend infinitely to the right, and do not settle to constant values, the binary influence process defined by U tends to have settled at some finite time. This is because backward paths originating from a large time index have a good chance of merging into one (if there is a nonzero probability that they could be merged) before they reach time 0.

3.7.2 Coalescing Random Walks

The coalescing random walk is another stochastic process defined on a network graph. By learning about the dual process, we can better understand the binary influence model, particularly in the study of the convergence time. Moreover, by establishing the binary influence model as the dual of a variation of the "random walk" family, we have effectively linked this model to the more mainstream literature in probability, thus allowing the possibility of applying known results about random walks to further study the binary influence model.

The *coalescing random walk* (CRW) process is defined as follows. Assume we are given a network matrix D, and its directed graph $\Gamma(D)$ (not $\Gamma(D')$). At time 0, every site holds a black dot. At every time step, the dot on site i hops to ones of its neighbors with the probability given on the branch, just like transition of a Markov chain. The difference here is that we have all dots hopping simultaneously. When two or more dots land at the same site, they coalesce into one, and continue hopping, as shown in Figure 3.9.

Another way to define the CRW process is through the outcome matrix. Given a network

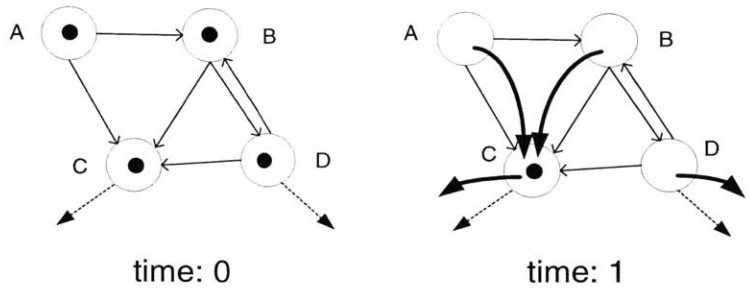


Figure 3.9: Example of a coalescing random walk. The dots from A and B coalesce when they meet in C. At the same time, the original occupants of C and D have hopped to somewhere else.

matrix D, we first generate an outcome matrix $U = [u_{ij}]$ in exactly the same way we generated (3.17). That is, we use the ith row of D as the PMF the generate the ith row of U. The entries of U designate which dots hops to where and when. Specifically, if there is a dot present at site i at time $j-1$, then at time j, it is to hop to site u_{ij}. Using (3.17) as an example, if we would like to know where the dot that started in, say, site 1 at time 0 has hopped to, then we trace the paths forward in time, as shown in Figure 3.10. Let us define $\widehat{\eta}_U[k]$ as a length-n column vector with its

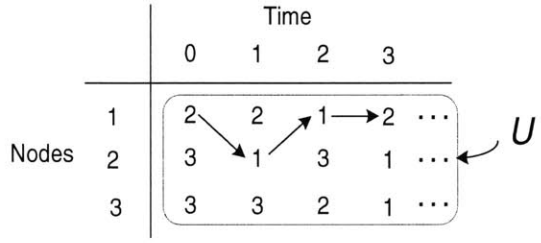


Figure 3.10: The path shows the journey of the dot that starts from site 1.

ith element given by

$(\widehat{\eta}_U[k])_i \triangleq$ the location at time k of the dot that originally started from site i, based on realization U

For Figure 3.10, $\widehat{\eta}_U[1] = [2\ 3\ 3]'$, $\widehat{\eta}_U[2] = [1\ 3\ 3]'$, $\widehat{\eta}_U[3] = [1\ 2\ 2]'$, etc.

Thus, the CRW model is essentially the binary influence model running backwards. Their common underlying machinery is the outcome matrices. Since all entries of U are independently generated (although they are not identically distributed), any statistics of forward paths should be equal to those of the backward paths. The relation between CRW and the binary influence model can be stated as:

$$Prob(U \text{ is such that } \eta_U[k] = \mathbf{v}) = Prob(U \text{ is such that } \widehat{\eta}_U[k] = \mathbf{v}) \tag{3.19}$$

for any length-n vector $\mathbf{v}$. The reason for (3.19) is as follows. Let A be the set of U such that $\eta_U[k] = v$, and let B be the set of U such that $\widehat{\eta}_U[k] = v$. For any $U_1 \in B$, consider a matrix U_2 which is equal to U_1 except that the order of the first k columns are reversed. Then $U_2 \in A$ because $\eta_{U_2}[k] = v$. Similarly, for any $U_3 \in A$, the matrix U_4 with the first k columns reversed must be in B. Hence, there is a one-to-one correspondence between the members of sets A and B, and because reversing the columns do not change the probability of each outcome matrix, the probabilities of sets A and B are equal.

This duality relation has been recognized by [25] for the voter model, which is a continuous-time, infinite-grid version of the binary influence model, with uniform influencing weights. The connection to CRW has been the key to the tractability of voter models, and has made it a favorite choice among the various interacting particle systems (see Sec. 1.2 for more details).

Naturally, various results about the binary influence model can be interpreted in terms of the CRW process. Theorem 3.6 implies that if $\Gamma(D)$ has an ergodic structure, then in the CRW model all the random walks will eventually coalesce into one. If the graph is periodic with period d, then the number of random walks will eventually be reduced to d different walks.

3.7.3 Application to Binary Influence Model

An immediate consequence of the duality between the binary influence model and the CRW model is the proposition developed in this section. Essentially, the main result here is that if we increase the number of 1's in the initial state $\mathbf{s}[0]$, then we also increase the probability that a given group of sites will all be in status 1's at all time. While this statement seems straightforward, we do not know of any obvious proof other than the following approach, whose surprising simplicity is largely due to the insight from the CRW.

Let V denote the set of sites on a network graph, and $B \subset V$ be a fixed subset of it. For any subset $A \subset V$, define $\mathbf{c}_A$ as a binary state vector in which all sites in A have a status 1. That is, $\mathbf{c}_A$ is a length-n whose ith element is a 1 if $i \in A$, and 0 otherwise. Then let

$$q_A^B[k] \stackrel{\triangle}{=} Prob(\text{ all sites in } B \text{ has status 1 at time } k \text{ given that } \mathbf{s}[0] = \mathbf{c}_A\,). \tag{3.20}$$

In all the discussions to follow, the set B will remain fixed. So we define $q_A[k] \stackrel{\triangle}{=} q_A^B[k]$ to reduce the notational clutter. We emphasize that in general

$$q_A[k] \geq Prob(\,\mathbf{s}[k] = \mathbf{c}_B \text{ given that } \mathbf{s}[0] = \mathbf{c}_A\,) \tag{3.21}$$

because in (3.20) nothing is said about the sites that are outside of the set B, but (3.21) constrains that they must be zero. A careful examination would reveal that

$$q_A[k] = Prob(\ U \text{ is such that } \forall i \in B(\eta_U[k])_i \in A\ ). \tag{3.22}$$

The best way to understand (3.22) is perhaps through the backward traces on the percolation substructure. Consider Figure 3.11, which depicts the backward traces similar to Figure 3.8. Recall that for a given U and its percolation substructure, the state of B at time k can be found by tracing the paths from all sites in B at time k to the left until time 0. Any U that counts towards $q_A[k]$ must be such that those traces will end up completely in A, which is the set of sites whose initial status is 1. With this picture, the following proposition almost becomes evident.

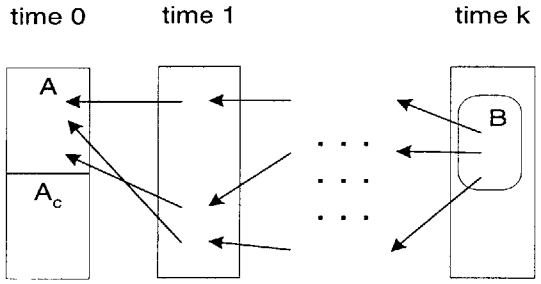


Figure 3.11: Graphical Interpretation of (3.22).

THEOREM 3.13
For any subsets $A, C \subset V$,

$$q_{A\cup C}[k] \ \geq \ q_A[k] + q_C[k] - q_{A\cap C}[k]. \tag{3.23}$$

Sketch of Proof: Consider the U's such that when we trace the paths from B at time k back to time 0, they all end up in $A \cup C$. By definition, the "measure" (or the total probability) of all such U's must be equal to $q_{A\cup C}[k]$. On the other hand, any such U must fall into one of the following four categories depending on where those backward paths terminate:

I: The paths completely terminate in $A - C$.

II: The paths completely terminate in $C - A$.

III: The paths completely terminate in $A \cap C$.

IV: The paths terminate at sites that cover two or three of the above cases.

Since these four cases are mutually exclusive, the sum of the measures of U's from all four cases must equal $q_{A\cup C}[k]$. Now because the sum of the measures from the first three cases is equal to the right-hand side of (3.23), we have achieved inequality (3.23) by dropping the measure from case IV. □

Simple as it is, Theorem 3.13 lets us obtain a statement that would have been difficult to do without the coalescing random walk argument. This following statement boils down to the fact that if the binary influence model starts off with a larger initial set of 1's, then the probability that a given set of sites will be all 1's will also be larger at all time.

COROLLARY 3.14
If $R \subset S \subset V$, then $q_R[k] \leq q_S[k]$.

Proof. Applying Theorem 3.13 with $A = R$ and $C = S - R$, we get $q_S[k] \geq q_R[k] + q_{S-R}[k] - q_{S\cap(S-R)}[k]$. But since $R \cap (S - R) = \emptyset$, $q_{R\cap(S-R)}[k] = 0$. And because $q_{S-R}[k] \geq 0$, we have the desired inequality. □

The message of Corollary 3.14 is very similar to that of eq. (3.5). Explained loosely in terms of failures, both state that a larger initial failure will result in larger failure at all time. However, neither statement is stronger than the other. Eq. (3.5) along with Proposition 3.1 provides an quantitative probability of failure for each *individual site*. Corollary 3.14 above, on the other hand, makes a statement about arbitrary *sets of sites*, but only provides bounds, not the exact probability.

3.8 Variations of Binary Influence Model

Recall that the binary influence Process consists of two evolutionary equations, which are

$$\mathbf{r}[k+1] = D\,\mathbf{s}[k] \tag{3.24}$$

$$\mathbf{s}[k+1] = Bernoulli(\mathbf{r}[k+1]). \tag{3.25}$$

In this section, we briefly explore a variation of these two equations and explain why it may provide a more realistic modeling of the cascading failure phenomena. Although the model to be proposed is not tractable as the binary influence model, it, nevertheless, seem rather promising as a starting point for further research.

We generalize the binary influence model by modifying the first equation (3.24) so that the

system is now described by

$$\mathbf{r}[k+1] = f(\mathbf{s}[k]) \tag{3.26}$$

$$\mathbf{s}[k+1] = Bernoulli(\mathbf{r}[k+1]). \tag{3.27}$$

where $f(\cdot) = [f_1(\cdot) \cdots f_n(\cdot)]'$ is a general function that maps the state vectors to the vector of probabilities. The sequence $\{\mathbf{s}[k]\}$ is still a Markov process in some 2^n-state Markov chain.

With the second equation being the same, it is still true that $E(\mathbf{s}[k]) = E(\mathbf{r}[k])$. More importantly, Proposition 3.1 which states that

$$E(s_i[k] \text{ given } \mathbf{s}[0]\ ) = Prob(\text{ site } i \text{ is in status 1 at time } k \text{ given } \mathbf{s}[0]\ ).$$

still holds. Therefore, if we are able to obtain some recursion for $E(\mathbf{s}[k])$ then we could solve for the probability of a site being in status 1 for any desired time.

3.8.1 Status-Dependent Influence Model

One interesting variation of the binary influence model is the case where the edge weights on the network graph vary as a function of a site's status. This variation could model a situation where a site is influenced differently when it is in status 0 from when it is in status 1.

As a potential application, consider the following hypothetical scenario of a power grid. Suppose the sites can be classified into two kinds — the power substations and the repair centers — as shown in Figure 3.12. In normal operation (status 0), a substation's status depends on other

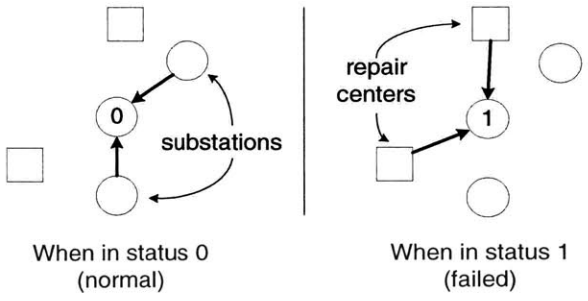


Figure 3.12: Example of status-dependent influence model. A substation receives influence from different sets of sites depending on its status.

neighboring substations, whose own statuses can vary over time. However, once the substation goes down (status 1), it requires the good influence from the repair center to bring it back up. This

means at each substation site, there are two sets of incoming edges to be applied under different statuses.

We now formulate the *status-dependent influence model* as follows. Assume that we are given two stochastic matrices $D_0 = [d_{ij}^0]$ and $D_1 = [d_{ij}^1]$. When site i is in status 0, the influence it receives from site j is d_{ij}^0. Similarly, when it is in status 1, the influence it receives from j is d_{ij}^1. The evolution equation would then be

$$\mathbf{r}[k+1] = (\mathbf{1} - \mathbf{s}[k]) \circ (D_0 \mathbf{s}[k]) + \mathbf{s}[k] \circ (D_1 \mathbf{s}[k]) \tag{3.28}$$

$$\mathbf{s}[k+1] = Bernoulli(\mathbf{r}[k+1]), \tag{3.29}$$

where $\circ$ denotes the Hadamard, or entry-wise product.

There are a variety of questions one can ask about this process. Again, the all-ones and all-zeros are clearly two of the recurrent states. As for the probability of status 1 of a site, we need a recursion for $E(\mathbf{s}[k])$. To get it, we first rewrite (3.28) to get

$$\mathbf{r}[k+1] = D_0\, \mathbf{s}[k] + \big((\mathbf{s}[k]\mathbf{s}'[k]) \circ D\big)\, \mathbf{1} \tag{3.30}$$

where $D \triangleq D_1 - D_0$. Note that when $D_1 = D_0$, the second term in (3.30) vanishes to reduce to binary influence model. After taking the expectation, we finally have

$$E(\mathbf{r}[k+1]) = E(\mathbf{s}[k+1]) = D_0\, E(\mathbf{s}[k]) + \big(E(\mathbf{s}[k]\mathbf{s}'[k]) \circ D\big)\, \mathbf{1}. \tag{3.31}$$

The appearance of the term second-order term $E(\mathbf{s}[k]\mathbf{s}'[k])$ is where the difficulty arises. If we try to obtain a recursion on the second-order term by multiplying eq. (3.30) with itself transposed, the recursion would involve terms up to the fourth order, and the recursion for the fourth order would involve terms up to the eighth order, and so on. It appears that some additional analysis is required just to numerically obtain the value of $E(\mathbf{s}[k])$ for arbitrary k.

Chapter 4

General Influence Model

4.1 Introduction

We now introduce the *General Influence Model*, which is one of the main contributions of this thesis. The general influence model has the structure of networked Markov chains, with the special property that the state-transition probabilities of each chain depend in a linear way on the statuses of its neighbors.

As with the binary case of the previous chapter, the general influence model is created to analyze the dynamics of random propagation on graphs. However, the general version includes one significant generalization of the binary case that makes it far richer: it allows each site to have an arbitrary internal Markov chain. With this flexibility, we can use the influence model to study many more types of interacting networks. Sites on the graph are no longer constrained to be of the same type. For instance, if the network is being used to model a supply chain, we can have sites that represent factories, others that represent distribution centers, and yet others that represent consumers. By analyzing the influence model, one can gain an understanding of how these nodes affect each other.

The general influence model is very amenable to theoretical analysis. In our discussion, we will define a new class of matrices called influence matrices that arise as a generalization of stochastic matrices. The influence matrix describes the first-order propagation of the influence model, capturing the dynamics of the status probabilities of the individual sites. In later chapters, we will show how to obtain higher-order descriptions of the influence model, which capture the dynamics of the joint probability of any selected group of nodes. As we will see, this higher-order analysis will also serve as the theoretical bridge between the influence matrices and the very large state-transition matrix that governs the entire network, viewed as the *master Markov chain.*

Among models developed in previous research, the one that is most similar to the influence model is perhaps the voter model. However, as explained in Section 1.2, the main results on voter models pay more attention to graphs with infinite size and regular lattice structure. Our binary influence model is already different from the voter model in certain ways. The general influence model, which is more versatile and includes much greater modeling flexibility than the binary case, represents an even greater departure from the classical voter model.

4.2 Model Definition

4.2.1 Homogeneous Influence Model

As the first step in generalizing the binary influence model, we will study the *homogeneous influence model*, where each site has an identical Markov chain (though possibly initialized differently from site to site). To motivate the model, consider the following scenario. A power grid consists of many power plants, and each one can be in one of three statuses at any given time: normal, alert, or failed. In the absence of interaction with others, a power plant's status can plausibly be modeled as evolving according to the Markov chain $\Gamma(A)$ drawn in Figure 4.1. However, due to

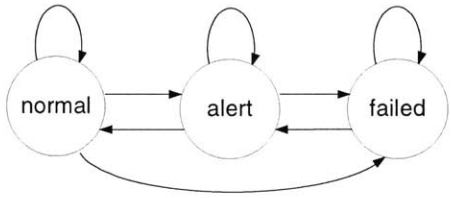


Figure 4.1: A 3-status chain modeling a power plant's operating conditions

the grid connection, power plants are linked up through a global network structure. These network interconnections cause the state-transition probabilities of each plant to depend not only on the plant's current status, but also on those of its neighbors. How might we represent the effect of the network connections on the evolution of each site's internal status? Figure 4.2 intuitively explains the approach of the homogeneous influence model.

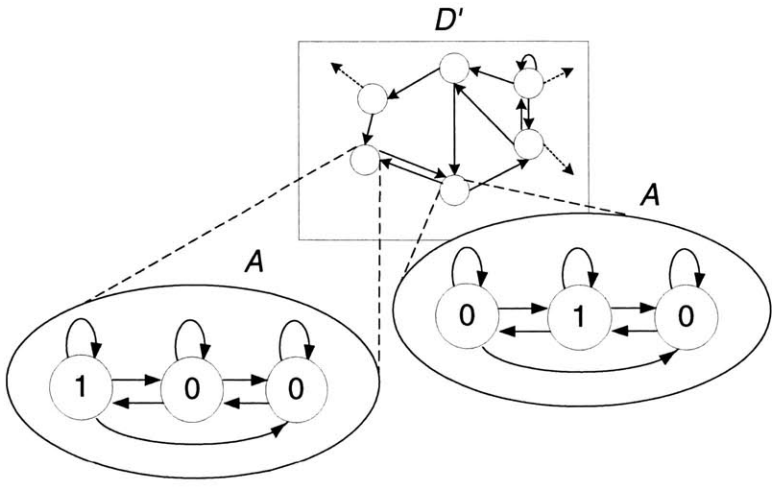


Figure 4.2: A homogeneous influence model.

In this figure, one can think of the entire network as having two levels of structure: the network level and the local level. At the network level, nodes are called *sites* and their connections are described by a *network graph* $\Gamma(D')$, where D is a stochastic matrix. Every site has an internal *local Markov chain* $\Gamma(A)$, which is the same at every site (hence a "homogeneous" influence model). At any given time, a site is in one of the statuses of $\Gamma(A)$, and its current status is indicated in our figure by a '1' in the node corresponding to that status. For the example in Figure 4.2, the current status of the site on the left is 'normal,' while the current status of the one on the right is 'alert.' The status of every site evolves with transition probabilities that depend both on its current status and the current statuses of the surrounding neighbors. Our specification of the rules that govern the evolution of the status will be described next.

Let D and A be stochastic matrices of dimensions $n \times n$ and $m \times m$ respectively. In this case, we say that there are n sites, connected by the network graph $\Gamma(D')$, and within each site is an m-status Markov chain $\Gamma(A)$. At time k, the status of site i is represented by a length-m *status vector*, an indicator vector containing a single 1 in the position corresponding to the present status, and 0 everywhere else:

$$\mathbf{s}_i'[k] = [0 \cdots 0\,1\,0 \cdots 0].$$

This interpretation is exactly the same as that used in our description of a Markov chain (see Sec. 2.3). Likewise, let $\mathbf{p}_i[k]$ denote the PMF vector governing the status of site i at time k. Let the *network state matrix*, or simply the *state matrix*, $S[k]$ be defined as a matrix that stacks up all the status vectors of the network sites as its rows. The (network) *probability matrix* $P[k]$ is similarly defined:

$$S[k] \triangleq \begin{bmatrix} \mathbf{s}_1'[k] \\ \vdots \\ \mathbf{s}_n'[k] \end{bmatrix} \qquad P[k] \triangleq \begin{bmatrix} \mathbf{p}_1'[k] \\ \vdots \\ \mathbf{p}_n'[k] \end{bmatrix}. \tag{4.1}$$

The evolution equations of the *homogeneous influence model* are now specified as:

$$P[k+1] = DS[k]A \tag{4.2}$$

$$S[k+1] = Realize(P[k+1]), \tag{4.3}$$

The operation $Realize(\cdot)$ performs a random realization for each row of $P[k+1]$ in the same manner as (3.2) in a binary influence model. The initial state $S[0]$ is assumed to be realized by some distribution that is independent of the evolution above. Each row of $P[k+1]$ is a valid PMF

because

$$P[k+1]\mathbf{1}_m = DS[k]A\mathbf{1}_m = \mathbf{1}_n,$$

where $\mathbf{1}_\ell$ is a vector of dimensional ℓ whose entries are all 1.

To get an intuitive understanding of (4.2), let us consider each row of $P[k+1]$ separately:

$$\mathbf{p}_i'[k+1] \;=\; d_{i1}\mathbf{s}_1'[k]A + \cdots + d_{in}\mathbf{s}_n'[k]A$$

If all sites were 'disconnected,' i.e., evolved independently of each other, then $D = I_n$. This means $d_{ii} = 1$ and $d_{ij} = 0$ for $j \neq i$ so that (4.4) reduces to

$$\mathbf{p}_i'[k+1] = \mathbf{s}_i'[k]A.$$

That is, $\mathbf{p}_i[k+1]$ represents the next-status PMF of site i exactly as it would in a regular Markov chain. In contrast, when the sites are connected according to $\Gamma(D')$ and D is an arbitrary stochastic matrix, eq. (4.4) reveals that site i uses a *convex combination* of these PMF's from itself and all neighboring sites to get its actual next-status PMF. The greater d_{ij} is, the greater the influence j has on the next status of site i. Therefore, one can think of d_{ij} as the amount of influence that site j exerts on the evolution of i. For this reason, D will be referred to as the *network influence matrix* and A the *local state-transition matrix.*

An alternative evolution for each $\mathbf{s}_i[k+1]$ so that the status probabilities $\mathbf{p}_i[k+1]$ remain unchanged at every step can be described as follows. At the end of time step k, site i chooses the site it will be influenced by according to PMF $\{d_{i1}, d_{i2}, \ldots, d_{in}\}$. If the chosen site is, say, j, then site i performs another realization for its actual status at time $k+1$ using the PMF $\mathbf{s}_j[k+1]A$; that is, site i assumes the status of site j while obtaining its next-status PMF. This two-step realization effectively yields the same $\mathbf{p}_i[k+1]$ as that in (4.4).

A careful examination of the homogeneous influence model shows it to be a combination of a binary influence model and Markov chains. Consider eq. (4.2), the first equation of the homogeneous influence model: $P[k+1] = DS[k]A$. Each column of $S[k]$ can contain an arbitrary number of 1's, but there must be exactly a single entry of 1 in each row. The product $DS[k]$ can be viewed as calculating the probability vectors of m separate binary influence models, one for each column of $S[k]$. On the other hand, by treating each row of $S[k]$ as a status vector of a Markov chain, the product $S[k]A$ corresponds to calculating n separate next-status PMF's. The homogeneous influence model is in this sense a most natural combination of both models.

4.2.2 A Different Form

The homogeneous influence model (4.2)-(4.3) can be rewritten so that it is in the form of matrix-vector multiplication instead of matrix-matrix multiplication. This change of form will eventually afford us not only the convenience of analyzing a single matrix as opposed to two (D and A), but also the ability to generalize the model so that each site can have an arbitrary Markov chain.

With $\mathbf{s}_i[k]$ and $\mathbf{p}_i[k]$ defined as in Sec. 4.2.1, let the *state* and *probability* vectors be defined as

$$\mathbf{s}[k] \triangleq \begin{bmatrix} \mathbf{s}_1[k] \\ \vdots \\ \mathbf{s}_n[k] \end{bmatrix} \qquad \mathbf{p}[k] \triangleq \begin{bmatrix} \mathbf{p}_1[k] \\ \vdots \\ \mathbf{p}_n[k] \end{bmatrix}. \tag{4.4}$$

Since each $\mathbf{s}_i[k]$, as well as each $\mathbf{p}_i[k]$ is a length-m column vector, both $\mathbf{s}[k]$ and $\mathbf{p}[k]$ are column vectors of length mn. These different representations are related to $S[k]$ and $P[k]$ in (4.1) by

$$\mathbf{s}[k] = \text{vec}(S'[k]) \qquad \mathbf{p}[k] = \text{vec}(P'[k]),$$

where vec$(\cdot)$ is the notation for vector stacking, as described in [27], p. 244, for example. Let the *influence matrix* be defined as the Kronecker product of D' and A:

$$H \triangleq D' \otimes A \triangleq \begin{bmatrix} d_{11}A & \cdots & d_{n1}A \\ \vdots & & \vdots \\ d_{1n}A & \cdots & d_{nn}A \end{bmatrix}. \tag{4.5}$$

Then equivalent expressions to (4.2)-(4.3) are

$$\begin{aligned} \mathbf{p}'[k+1] &= \mathbf{s}'[k]H \\ \mathbf{s}'[k+1] &= \mathit{MultiRealize}(\mathbf{p}'[k+1]). \end{aligned}$$

The operation $\mathit{MultiRealize}(\mathbf{p}'[k+1])$ treats each block of PMF's within $\mathbf{p}'[k+1]$ separately, independently realizing the new status vectors block by block.

If $D = I$, then the influence model reduces to n independent Markov chains, because all sites are disconnected from each other. If $A = I_2$, then the influence model reduces to the binary influence model.

4.3 General Influence Model

We now generalize the influence model one step further so that each site is allowed to have different Markov chains. For instance, if the network network represents the power grid, then in addition to the power plant sites, some sites may represent the loads, whose internal status characterizes the current demand level such as "high" or "low," as shown in Figure 4.3.

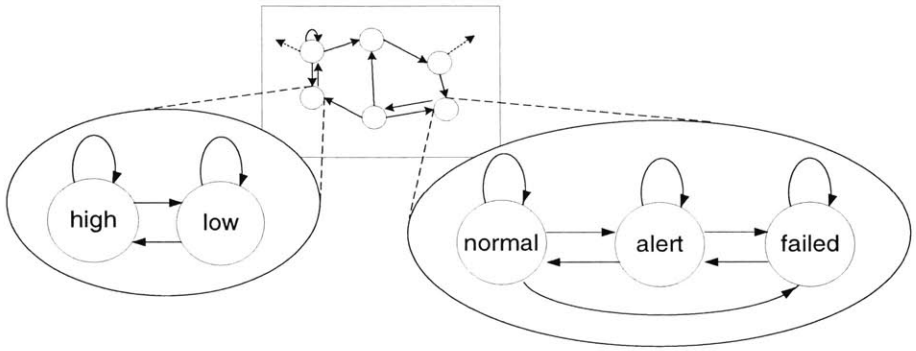


Figure 4.3: general influence model

4.3.1 Generalized Kronecker Product

Before we present the general influence model, we need to define a new notation that generalizes the Kronecker product. Let A be some $r \times t$ matrix and let B be some matrix. Recall that $A \otimes B$ denotes the regular Kronecker product:

$$A \otimes B \triangleq \begin{bmatrix} a_{11}B & \cdots & a_{1t}B \\ \vdots & & \vdots \\ a_{r1}B & \cdots & a_{rt}B \end{bmatrix}$$

For more background on regular Kronecker product, see [27], Chapter 4. We generalize this notation so that each matrix B in the (i,j)th block can be different. Specifically, we assume that a set of *dimension-compatible* matrices $\{B_{ij} \,|\, 1 \leq i \leq r,\, 1 \leq j \leq t\}$ is given, where $B_{i1}, \ldots, B_{it}$ all have the same number of rows, and $B_{1i}, \ldots, B_{ri}$ all have the same number of columns, for each i. We then define the generalized Kronecker product as

$$A \otimes \{B_{ij}\} \triangleq \begin{bmatrix} a_{11}B_{11} & \cdots & a_{1t}B_{1t} \\ \vdots & & \vdots \\ a_{r1}B_{r1} & \cdots & a_{rt}B_{rt} \end{bmatrix}$$

In the special case where $B_{ij} = B_i$ for all j, we write the above product more simply as $A \otimes \{B_i\}$. One form of usage that appears especially often later is the vector product

$$\mathbf{v} \otimes \{\mathbf{w}_i\} \triangleq \begin{bmatrix} v_1\mathbf{w}_1 \\ \vdots \\ v_n\mathbf{w}_n \end{bmatrix}$$

where $\mathbf{v} = [\, v_1 \cdots v_n \,]'$ and every $\mathbf{w}_i$ is a column vector.

4.3.2 Model Definition

Let m_i be the order of the local Markov chain at site i for $1 \leq i \leq n$. Let $\mathbf{s}_i[k]$ and $\mathbf{p}_i[k]$ be the *status vector* and the *next-status PMF vector* of site i at time k exactly as defined above, with the only difference being that these are now vectors of length m_i. Let $\mathbf{s}[k]$ and $\mathbf{p}[k]$ denote the state and probability vectors defined the same way as in (4.4); both are column vectors of length $(m_1 + \cdots + m_n)$.

For each pair of sites i and j, the *state-transition matrix* A_{ij} is an $m_i \times m_j$ nonnegative matrix whose rows sum to 1, i.e., $A_{ij}\mathbf{1}_{m_j} = \mathbf{1}_{m_i}$. The more general form of *influence matrix* is defined as

$$H \triangleq D' \otimes \{A_{ij}\} = \begin{bmatrix} d_{11}A_{11} & \cdots & d_{n1}A_{1n} \\ \vdots & & \vdots \\ d_{1n}A_{n1} & \cdots & d_{nn}A_{nn} \end{bmatrix} \tag{4.6}$$

The evolution equations of the *influence model* are defined as

$$\mathbf{p}'[k+1] \triangleq \mathbf{s}'[k]\, H \tag{4.7}$$

$$\mathbf{s}'[k+1] \triangleq \mathit{MultiRealize}(\, \mathbf{p}'[k+1] \,) \tag{4.8}$$

where $\mathit{MultiRealize}(\cdot)$ operates as before except that it now takes into account the different block lengths. Again, the initial state $\mathbf{s}[0]$ is assumed to be independently realized by some given distribution. The *influence process* is the sequence of random vectors $\{\mathbf{s}[k]\}$ generated by the influence model in (4.7)-(4.8).

By looking at the ith block of $\mathbf{p}'[k+1]$ in (4.7) closely, we find a result similar to (4.4):

$$\mathbf{p}_i'[k+1] = d_{i1}\mathbf{s}_1'[k]A_{1i} + \cdots + d_{in}\mathbf{s}_n'[k]A_{ni}. \tag{4.9}$$

That is, each term $\mathbf{s}'_j[k]A_{ji}$ on the right-hand side of eq. (4.9) produces a length-m_i valid PMF, and $\mathbf{p}'_i[k+1]$ is created from a convex combination of all such PMF's.

Again, an equivalent way of obtaining each $\mathbf{s}_i[k+1]$ is by realizing it in two steps. Site i first chooses a site to be influenced by. If the chosen site is j, then site i performs the actual status realization based on the PMF $\mathbf{s}'_j[k]A_{ij}$. The combined result is an effective PMF given in (4.9).

4.3.3 Motivating Questions

Now that we have defined the influence model in its full generality, many challenging questions arise as to how to analyze such a model. As a start, we recognize that the influence process is a discrete-time, finite-state Markov process itself. This is because the next-step probability vector $\mathbf{p}[k+1]$ depends on the current state and nothing else from the past. Let us denote the Markov chain that describes this *master* process as $\Gamma(G)$, where G is its state-transition matrix. Each possible outcome of $\mathbf{s}[k]$ is a status in the chain $\Gamma(G)$. The challenge in analyzing G arises from the fact that its dimension equals $\prod_i m_i$, the product of the dimensions of the local Markov chains, which means that the size of G grows exponentially in the number of sites. Nevertheless, despite its extremely large size for a network of moderate size (n) and complexity ($\max_i m_i$), G should bear an intimate relationship with H, since it is entirely from H that G is defined.

There are at least three crucial aspects of G for which we want to gain a thorough understanding. First, in order to determine what the states could be eventually, we need to understand the structure of $\Gamma(G)$ enough to identify its recurrent classes. This is an interesting problem not only because $\Gamma(G)$ typically contains so many states that it is impossible to draw a graph for it, but also because its interconnections will generally be dense regardless of the network structure $\Gamma(D')$. This prevents us from even listing all the entries of G, let alone analyzing it in any way. Thus, the only practical way of understanding the structure of G is through a careful study of the structure of the graph $\Gamma(H)$. Section 4.4 will show that for an irreducible $\Gamma(D')$, the recurrent classes of the master Markov chain $\Gamma(G)$ can be inferred from the structure of the graph $\Gamma(H)$.

A second set of quantities we could learn from G if we had this large matrix are the steady-state probabilities of the different states. Knowing these probabilities will enable us to quantify the likelihood of the collective status of the sites (i.e., the global state) in steady state. Again, the straightforward scheme of obtaining the dominant left eigenvector of G is impossible due to the size of the problem. We will show how H, as well as its higher-order forms, can provide information on these steady-state probabilities.

Finally the dynamics, or specifically, the eigenvalues of G help us understand the time-

domain behavior of the influence process. For instance, if we are running a Monte Carlo simulation of the influence model, then how long should the simulation be in order to achieve a steady-state behavior? As we will see, the eigenvalues of G can be found from H and its higher-order forms.

In what follows, we will study these three fundamental aspects of G through both graphical and algebraic means. In Sec. 4.4, we begin by exploring the recurrent classes of $\Gamma(G)$ through computations on the graph $\Gamma(H)$. Just as one would typically first delineate the transient and recurrent classes in a Markov chain before analyzing its state-transition matrix, we will start our analysis of the influence model through graph analysis in order to obtain an intuitive understanding of this system. We will show that the recurrent classes of the Markov chain $\Gamma(G)$ can be characterized by the structures of the influence graph $\Gamma(H)$.

Later, in Chapter 5, we embark on our detailed study of the matrix G. This will include the formal construction of G, elucidation of its explicit connection to H, and determination of their common eigenvalue and eigenvector structures. This algebraic exploration of G will give us insights into the dynamics of the master chain, which, as we will show, has an intuitively pleasing correspondence with the dynamics defined by H. Through the relation between G and H, we can derive partial information regarding the steady-state probabilities without having to construct or perform computations on G. Finally, in Chapter 6, the higher-order analysis of the influence matrix will further elucidate properties of G and provide us with a method to obtain stronger information on the steady-state probabilities, though with progressively increasing computational costs.

4.4 Determining Recurrent Classes of $\Gamma(G)$ by Analysis of $\Gamma(H)$

This section deals with one of the the most basic questions regarding the influence model: for a given influence model, what state or group of states could the influence process $\{\mathbf{s}[k]\}$ eventually settle to? As the master Markov chain $\Gamma(G)$ is one in which each node represents a possible outcome of $\mathbf{s}[k]$, we are asking the following equivalent question: what are the recurrent classes of $\Gamma(G)$? In this section, we will characterize the recurrent classes of $\Gamma(G)$ by analyzing the structure of the graph $\Gamma(H)$. This type of analysis can be very practical because it allows one to understand the recurrent classes of $\Gamma(G)$ without having to actually construct G, which is usually very large, as mentioned earlier. Indeed, as the examples below show, for moderate number of sites one can quickly describe the recurrent classes simply by inspecting $\Gamma(H)$. In the following, a *recurrent state* of an influence process is a state in which the corresponding status of the master chain $\Gamma(G)$ is a node in some recurrent class (of $\Gamma(G)$). Otherwise, it is called a *transient state.*

4.4.1 Motivating Examples

Let $H = D' \otimes \{A_{ij}\}$ be the influence matrix defined in Sec. 4.3. We shall refer to $\Gamma(H)$ as the *influence graph.* Since every node on $\Gamma(H)$ belongs to some local Markov chain, it is called a *status node.* The *network graph* is defined as $\Gamma(D')$ as usual, and its nodes are referred to as *sites.* We assume throughout that $\Gamma(D')$ is a connected graph, because if $\Gamma(D')$ has multiple components, then each disconnected component would produce an influence graph of its own, and we can then apply the analysis below to each graph separately.

Recall that even in the case of binary influence model, different configurations of the network matrix D alone can already give rise to a number of very different steady-state patterns, for instance, the consensus, limit cycles, or the 'evil rain.' Here with the freedom in choosing both D *and* all the n^2 state-transition matrices A_{ij}'s, the possibilities seem almost too many to catalog. Thus, before any formal analysis, let us first motivate the study with a few examples. In each example, we will show how the recurrent classes of $\Gamma(G)$ can be found from the structure of the influence graph $\Gamma(H)$. These examples will also reintroduce the notions 'hopping dots,' a concept that will greatly help us in visualizing later results.

Example 1: (Homogeneous Influence Model) Let $H = D' \otimes A$ be a homogeneous influence matrix whose numerical value are as shown:

$$D' = \begin{bmatrix} .5 & 1 \\ .5 & \end{bmatrix} \qquad A = \begin{bmatrix} 1 & & \\ .4 & .2 & .4 \\ & & 1 \end{bmatrix} \qquad H = \left[\begin{array}{ccc|ccc} .5 & & & 1 & & \\ .2 & .1 & .2 & .4 & .2 & .4 \\ & & .5 & & & 1 \\ \hline .5 & & & & & \\ .2 & .1 & .2 & & & \\ & & .5 & & & \end{array}\right] \tag{4.10}$$

The corresponding graphs $\Gamma(D')$, $\Gamma(A)$ and $\Gamma(H)$ are in Figure 4.4. In the graph $\Gamma(H)$, we also draw the boundary around the statuses that belong to the same site. Our convention throughout is that sites and site boundaries are drawn with curved squares or curved rectangles, while the statuses are drawn with circles.

In Figure 4.4, observe the upper site boundary of $\Gamma(H)$, which encloses statuses $\{1, 2, 3\}$. If we remove all the site-crossing edges, then we would get a graph with exactly the same structure as $\Gamma(A)$ on the left. The fact that $\Gamma(A)$ is a subgraph of $\Gamma(H)$ is entirely due to the fact that in $\Gamma(D')$ site 1 has a self-loop. In contrast, site 2 of $\Gamma(D')$ does not have a self-loop. Therefore, if we consider the lower site boundary of $\Gamma(H)$, and remove all site-crossing edges, then we would be left

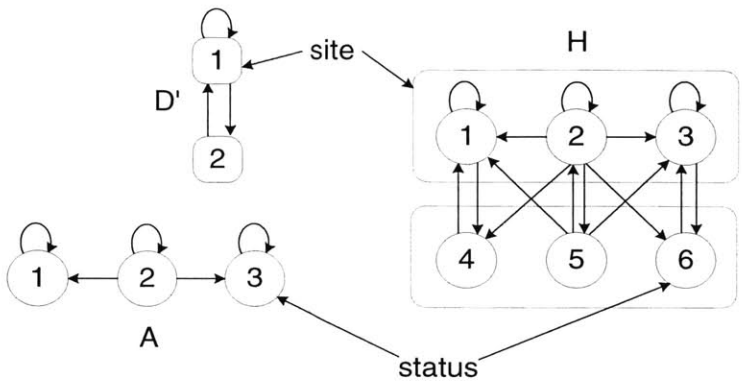


Figure 4.4: Example of a homogeneous influence graph and its constituents.

with only three disconnected statuses $\{4\}$, $\{5\}$ and $\{6\}$.

To picture the influence process graphically, it is helpful to recall the "hopping dot" interpretation introduced in Example 2 of Sec. 2.3. Unlike the Markov process, where there is only one dot in the entire system, here there is one dot within each site boundary. The current position of a dot indicates the status of the site within which it lies. The positions of the dots also indicate where the "1" entries are in the state vector $\mathbf{s}[k]$. For instance, in Figure 4.5 the positions of the dots correspond to the state $\mathbf{s}[k] = [0\ 1\ 0\ 0\ 1\ 0]'$ (the index of each status is shown in $\Gamma(H)$ in Figure 4.4).

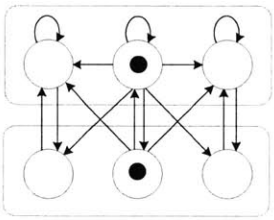


Figure 4.5: State $\mathbf{s}[k] = [0\ 1\ 0\ 0\ 1\ 0]'$

A status has a positive probability of being occupied by a dot in the next step if and only if it is currently one step away from any dot, regardless of whether the dot is within the site or outside. Wherever a dot is, it is always faced with a mixture of two kinds of outgoing edges: those that do not cross the site boundary, and those that do. Any of the non-site-crossing edges can be traversed by the dot with positive probability. The dot *cannot* traverse a site-crossing edge, but *can* "influence" the status at the other end of the edge by increasing the probability that this status becomes occupied by the dot in the other site; this is the essence of *influence* that a site exerts on its neighbors. Every dot is evidently always trapped within its site boundary.

A dot may hop to a non-adjacent status within its site if it receives external influence. For instance, consider Figure 4.6 in which the filled dots indicate the state $\mathbf{s}[k] = [0\ 0\ 1\ 1\ 0\ 0]'$ (the numbering is still as in Figure 4.4). In the following step, it is possible for the lower dot to hop to status 6 due to the influence from the upper dot. Once the dot has hopped, the system will be in

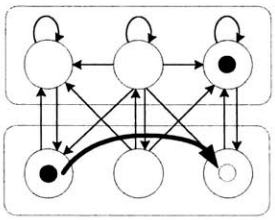


Figure 4.6: Example of a dot hopping to a non-adjacent status.

state $[0\ 0\ 1\ 0\ 0\ 1]'$ and will remain there, proving that in the master chain $\Gamma(G)$, this new state is recurrent and the original state is transient.

There is an important further point that can be made from this example. Notice that before the lower dot has hopped, the dots are in statuses 3 and 4. If there were no network connections and if each site boundary contained a local copy of $\Gamma(A)$, both statuses would have been recurrent with respect to their local chains. The fact that $[0\ 0\ 1\ 1\ 0\ 0]'$ is transient in the master chain shows that even if both dots are in statuses that would have been recurrent with respect to their local chains, the entire system might still be in a transient state of $\Gamma(G)$.

Finding the recurrent classes of the influence process is equivalent to finding all the possible configurations in which the dots can end up eventually. For this example, we can determine the recurrent classes simply by inspection, and they are as shown in Figure 4.7. Here, each recurrent class comprises a single state (or a single status of the master chain $\Gamma(G)$. □

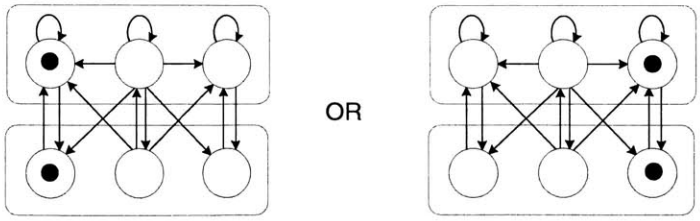


Figure 4.7: Recurrent states of Figure 4.4. The location of the solid dot is the final status of that site.

Example 2: In the previous example, the two recurrent states of $\Gamma(G)$ both involve statuses that would have been recurrent with respect to their local chains. This example will provide a contrast to the previous example by showing that a status that is transient with respect to its local chain

can be occupied by some dot in the recurrent classes of $\Gamma(G)$, provided that the influence matrix H is not homogeneous. Suppose the different component matrices are

$$D' = \frac{1}{2}\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \qquad A_{11} = \frac{1}{2}\begin{bmatrix} 1 & 1 & \\ & 1 & 1 \\ & 1 & 1 \end{bmatrix} \qquad A_{22} = \frac{1}{2}\begin{bmatrix} 1 & 1 & \\ 1 & 1 & \\ & 1 & 1 \end{bmatrix} \qquad A_{12} = A_{21} = I_3,$$

which results in the following influence matrix:

$$H = \frac{1}{4}\left[\begin{array}{ccc|ccc} 1 & 1 & & 2 & & \\ & 1 & 1 & & 2 & \\ & 1 & 1 & & & 2 \\ \hline 2 & & & 1 & 1 & \\ & 2 & & 1 & 1 & \\ & & 2 & & 1 & 1 \end{array}\right] \tag{4.11}$$

Figure 4.8 shows the structure of $\Gamma(H)$. Using the hopping dot analogy, if the dot in the upper site

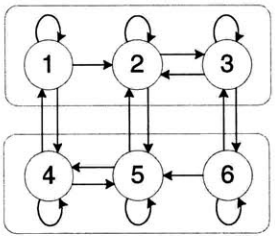


Figure 4.8: $\Gamma(H)$ for H in (4.11).

drifts to status 3 (as it does with positive probability), then it will cause the lower dot to occupy status 6 with positive probability, even though 6 is a transient status of the local chain in the lower site. Similarly, if the lower dot drifts to status 4, it causes the upper dot to be in status 1 with positive probability even though 1 is a transient status of the local chain in the upper site. Hence, the recurrent states of $\Gamma(G)$ can involve both 1 and 6. For this example, it is not hard to see that the recurrent class of $\Gamma(G)$ consists of all the 9 possible states of this system (3 possible locations for the upper dot, multiplied by 3 for the lower dot). In other words, $\Gamma(G)$ is an irreducible (and indeed, ergodic) Markov chain. □

Example 3: In both previous examples, the recurrent classes of $\Gamma(G)$ involved statuses that were recurrent with respect to their local chains. We show in this example that this not always the case; a status that is recurrent with respect to its local chain need not be involved in any recurrent

class of $\Gamma(G)$. Let there be two sites, and let the order of the local chains be 3 and 1 respectively. Suppose the matrices are given as

$$D' = \begin{bmatrix} .5 & .5 \\ .5 & .5 \end{bmatrix} \qquad H = \left[\begin{array}{ccc|c} .5 & & & .5 \\ .2 & .1 & .2 & .5 \\ & & .5 & .5 \\ \hline & & .5 & .5 \end{array}\right]$$

The matrices A_{ij} can be easily inferred from D' and H and can be verified to have each row summing to 1. The network graph $\Gamma(D')$ is irreducible, while the influence graph $\Gamma(H)$ is as in Figure 4.9a. By inspection, the only recurrent state for $\Gamma(G)$ is shown in Figure 4.9b. As we can

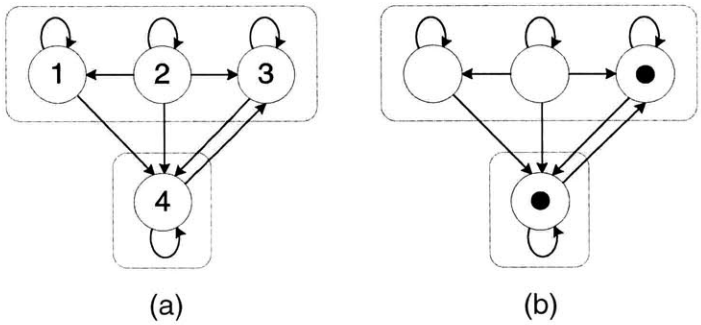


Figure 4.9: (a) Status indices (b) Recurrent state.

see, even though status 1 is recurrent within the upper site, it is not part of the recurrent state of $\Gamma(G)$. □

Example 4: (Binary Influence Model) As mentioned in Sec. 4.2.2, the binary influence model is a special case of the homogeneous influence model where the local chains have state-transition matrices $A = I_2$. In Chapter 3, we explored the binary influence model using only the network graph $\Gamma(D')$. The influence graphs of these models have the unusual property that $\Gamma(H)$ is always disconnected, regardless of whether the network graph is connected. For instance, let

$$D' = \begin{bmatrix} .2 & .5 & .5 \\ .4 & 0 & .5 \\ .4 & .5 & 0 \end{bmatrix}, \quad \text{and} \quad H = D' \otimes I_2 = \begin{bmatrix} .2 & & .5 & & .5 & \\ & .2 & & .5 & & .5 \\ .4 & & 0 & & .5 & \\ & .4 & & 0 & & .5 \\ .4 & & .5 & & 0 & \\ & .4 & & .5 & & 0 \end{bmatrix}.$$

Their graphs are shown in Figure 4.10. The structure of $\Gamma(H)$ simply consists of two identical but

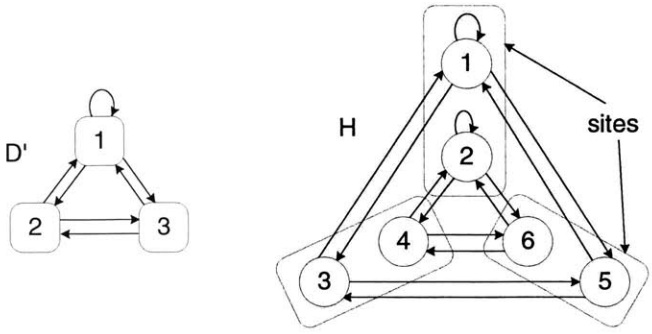


Figure 4.10: The influence graph for a binary influence model.

disconnected graphs, one for status "0" and the other for status "1", and each with the structure of $\Gamma(D')$. This structure is obtained regardless of the structure of $\Gamma(D')$. From this figure, it is not difficult to see that when all the dots reside on the same subgraph, they would always remain there, which is equivalent to the binary influence process having reached a consensus. It is less obvious, however, that the consensus states are the only possible recurrent states for *all* ergodic $\Gamma(D')$ — though we have confirmed that to be the case in Chapter 2. □

4.4.2 Basic Relations Between $\Gamma(D')$ and $\Gamma(H)$

At this point, we can begin our formal analysis of the recurrent states. As the previous examples show, the results will depend on the structures of $\Gamma(D')$ and $\Gamma(H)$ simultaneously. The first set of results, developed in this section and the next concerns the basic relations between the topologies of $\Gamma(D')$ and $\Gamma(H)$. Based on these results, Sec. 4.4.4 describes the recurrent states for the case of irreducible network graphs. Then in Sec. 4.4.7 we briefly outline the case of arbitrary network graphs.

We will now describe the relation the classes on $\Gamma(D')$ and those on $\Gamma(H)$. For any site i on the graph $\Gamma(D')$, let $\phi(i)$ denote the set of statuses on $\Gamma(H)$ that are enclosed within the boundary of site i. For instance, in Figure 4.4, $\phi(1) = \{1, 2, 3\}$. In Figure 4.10, $\phi(2) = \{3, 4\}$. The following propositions lay out some basic relations between the paths on $\Gamma(D')$ and those on $\Gamma(H)$.

LEMMA 4.1

An edge (i, j) on $\Gamma(D')$ exists if and only if there exists an edge on $\Gamma(H)$ from each $x \in \phi(i)$ to some $y \in \phi(j)$.

Proof. By definition, the fact that an edge (i, j) exists on $\Gamma(D')$ means $d_{ji} \neq 0$. Now the paths

from the statuses in $\phi(i)$ to those in $\phi(j)$ on $\Gamma(H)$ are described by the (i,j)th block of H (see eq. (4.6)) which is $d_{ji}A_{ij}$. Since the rows of A_{ij} sum to 1 by definition, there must be at least one positive entry in every row of A_{ij}. By extension, there is at least one positive entry in every row of $d_{ji}A_{ij}$ as well, because $d_{ji} \neq 0$. Therefore, for any $x \in \phi(i)$, in the row of H that corresponds to x, there is at least one positive entry to some position $y \in \phi(j)$, implying that edge (x, y) exists on $\Gamma(H)$. The converse can be shown similarly. □

COROLLARY 4.2

If on $\Gamma(H)$ there exists an edge γ from some status in $\phi(i)$ to some status in $\phi(j)$, then every status in $\phi(i)$ has an edge to some status —though not necessarily the same one— in $\phi(j)$.

Proof. If edge γ exists, then the corresponding entry of H must be nonzero, which implies that d_{ji} is nonzero. Then we can apply Lemma 4.1 to get the desired result. □

COROLLARY 4.3

A path on $\Gamma(D')$ from sites i to j exists if and only if for any status $x \in \phi(i)$, there exists a path on $\Gamma(H)$ from x to some status in $\phi(j)$.

Proof. This is proved by repeatedly applying Lemma 4.1 to each edge on the path from i to j. □

Now that we have established the basic results on paths, let us move further to the issues of classes. Recall that a class is a set of nodes that communicate. As a result of Corollary 4.3, if D is reducible (has multiple classes), then so is H. This is because if site i does not have a path to site j on $\Gamma(D')$, then no status in $\phi(i)$ can have any path to any status in $\phi(j)$. However, the converse is not true; if H is reducible, then D may or may not be reducible. For instance, consider $\Gamma(H)$ from Example 1, which is repeated in Figure 4.11. The classes of $\Gamma(H)$ are enclosed in the ovals.

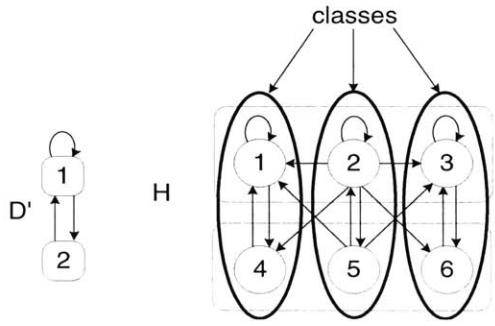


Figure 4.11: Classes of $\Gamma(H)$ from Example 1.

Even though $\Gamma(H)$ has 3 classes, $\Gamma(D')$, which is shown on the left, has only 1 class. As another example, in Figure 4.10 $\Gamma(H)$ has 2 classes, but $\Gamma(D')$ has only 1 class.

4.4.3 Paths and Probabilities

The previous section focuses on the structural properties of $\Gamma(H)$ and $\Gamma(D')$. In this section, we will relate those properties to probabilities. Although the influence model in consideration now is more general than the binary version of Chapter 3, many of the following results are still very similar. Hence, some of the results below are simply stated without the proofs. Readers can refer to Sec. 2.3 and 3.3.1 for explanations.

LEMMA 4.4

$$E(s_x[k] \mid \mathbf{s}[0]) \quad = \quad Prob(\text{ status } x \text{ will be occupied by a dot at time } k \mid \mathbf{s}[0])$$

Proof. Because $s_x[k] = 1$ when x is occupied, and 0 when it is not, its expected value must equal the probability of occupation. □

THEOREM 4.5

$$E(\mathbf{s}'[k] \mid \mathbf{s}[0]) = E(\mathbf{p}'[k] \mid \mathbf{s}[0]) = \mathbf{s}'[0]H^k \tag{4.12}$$

Note that Theorem 4.5 implies that $E(\mathbf{s}'[k]) = E(\mathbf{s}'[0])H^k$.

COROLLARY 4.6

Let x and y be any two statuses on $\Gamma(H)$. If under the initial state $\mathbf{s}[0]$, the status x is occupied by a dot, and if there is a path of length exactly k on $\Gamma(H)$ from x to y, then there is a positive probability that status y will be occupied by some dot at time k.

Proof. If x is occupied at time 0, then $s_x[0] = 1$. By Theorem 2.1, if a path of k steps from x to y exists then $[H^k]_{xy} > 0$, which means $E(s_y[k] \mid \mathbf{s}[0]) = \left[\mathbf{s}'[0]H^k\right]_y > 0$. Then by applying Lemma 4.4, we have proved the claim. □

Although this last corollary follows formally from the two preceding propositions, it deserves a careful intuitive explanation. In Corollary 4.6, the destination status y might be in a different site from x. Since no dot can cross any site boundary, the reason that y can be occupied is not because the dot at x hops along the path until it reaches y, but rather, because x exerts influence along

the path until y is occupied. For instance, consider Figure 4.12. Suppose, as shown in (a), there exists some path from status x in site A to status y in site B (all other edges are omitted). First the A-dot hops along the path until it reaches its boundary. Then, in (b), it exerts the influence on the status in site B that is adjacent to it and on the path. In the following step (c), the B-dot hops from wherever it is onto this adjacent influenced status. Finally, the B-dot continues down the path until, with positive probability, it reaches the destination y at time k.

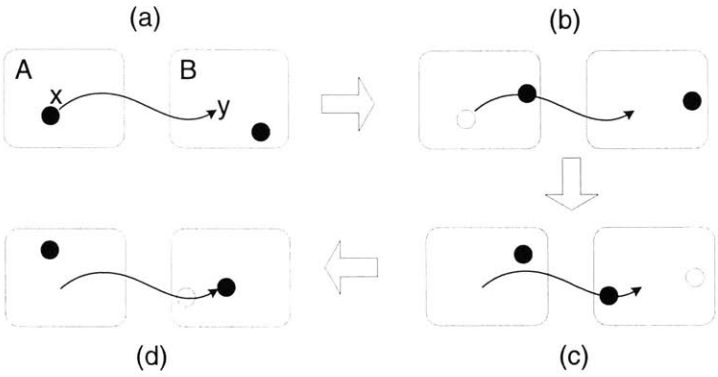


Figure 4.12: Intuitive explanation for Corollary 4.6.

4.4.4 Irreducible Network Graphs

We first analyze the recurrent states of irreducible (i.e, single-class) network graphs. From a practical point of view, the irreducible case is also the most important one because many large networks in real-life tend to be irreducible. The more edges in a network, the more likely that it will be irreducible.

The fact that $\Gamma(D')$ is irreducible does not mean that $\Gamma(H)$ will also be irreducible. When $\Gamma(D')$ is irreducible, then all its sites communicate. Translating to $\Gamma(H)$ by Corollary 4.3, this only means we can go from *any* status in *any* given site to *some* status in *any* desired site. This is still not sufficient to conclude anything about the reducibility of $\Gamma(H)$. For instance, in Figure 4.11, $\Gamma(D')$ contains 1 class, but $\Gamma(H)$ has 3; classes $\{1,4\}$ and $\{3,6\}$ are recurrent, while $\{2,5\}$ is transient. On the other hand, in Figure 4.8 of Example 2, $\Gamma(H)$ is irreducible just like its $\Gamma(D')$.

Global and Local Classes As explained earlier, finding the recurrent states of $\Gamma(G)$ is equivalent to finding all the possible configurations in which the dots can end up on its influence graph $\Gamma(H)$. We thus expect that the recurrent states of $\Gamma(G)$ can be described in terms of the structure of $\Gamma(H)$ and the hopping dots. Thus, we now turn our attention to the study of the classes on $\Gamma(H)$ and

	Transient	Recurrent
Local	$\{2\}$	$\{1\}$, $\{3\}$
Global	$\{2,5\}$	$\{1,4\}$, $\{3,6\}$

Table 4.1: Classes of the graph in Figure 4.11

their relations to the structures of the local chains $\Gamma(A_{ii})$.

A set of statuses is a *global class* if it is a class with respect to $\Gamma(H)$. A global class is *globally recurrent* if it is recurrent with respect to $\Gamma(H)$. Otherwise, it is *globally transient.* The influence graph $\Gamma(H)$ always contains at least one recurrent class, because all finite directed graphs do.

Let us for the moment disregard all the edges that cross the boundary of a particular site i and only focus on the subgraph of $\Gamma(H)$ that lies strictly inside it. Because $H = D' \otimes \{A_{ij}\}$, this subgraph is described by $\Gamma(d_{ii}A_{ii})$. When $d_{ii} > 0$, we say that the ith local chain *exists.* When $d_{ii} = 0$, this subgraph is simply a set of disconnected statuses. When the local chain exists, we call a set of statuses a *local class* if it is a class with respect to the local chain $\Gamma(d_{ii}A_{ii})$. A local class can be *locally recurrent* or *locally transient* with respect to its own chain. For example, in $\Gamma(H)$ of Figure 4.11, only the upper local chain exists, and the classes are as in Table 4.1.

Because a class must include all the statuses that communicate with each other, a global class always includes a *whole* local class. That is, if any global class —whether transient or recurrent— has a status in common with any local class (whose local chain exists), then that entire local class must be a subset of the global class. Also, by definition, in a globally recurrent class one can take any path from any status on it to any other status (even those outside its site boundary), and be guaranteed that there is a path back to the starting status. This means if a globally recurrent class contains a status x that belongs to local chain i, then regardless of whether x belongs to a locally recurrent or transient class, that globally recurrent class must contain at least one locally recurrent class from local chain i.

Classes in Homogeneous Influence Models We now take a diversion to focus on the special case of homogeneous influence models. It turns out that for these models, there is a special relation between the local and global classes, as suggested by Table 4.1. Each global class on a homogeneous influence graph is actually composed of local classes that are images of one another from different sites. For instance, in the above table, the global class $\{1,4\}$ is made up of statuses 1 and 4, both of which are the "left" recurrent status of $\Gamma(A)$. The set $\{1,6\}$, which mixes the "left" and "right" recurrent statuses, is, on the other hand, not a global class. We formalize this observation and characterize the global classes on homogeneous influence graphs as follows.

On the graph $\Gamma(H)$ of a homogeneous influence matrix $H = D' \otimes A$, the subgraph that lies

strictly inside the boundary of site i is described by $\Gamma(d_{ii}A)$. Every status in $\Gamma(d_{ii}A)$ is the unique *image* of a particular status in $\Gamma(A)$. For example, in Figure 4.4, status 1 in $\Gamma(A)$ has two images in $\Gamma(H)$, one inside each site; its image in the top site is status 1, and its image in the lower site is status 4. For a status x in $\Gamma(A)$, we denote its image in site i by $x_{(i)}$. The definition of an image extends to sets of statuses naturally. For example, in Figure 4.4, the set of statuses $\{5, 6\}$ in $\Gamma(H)$ is an image of the set of statuses $\{2, 3\}$ in $\Gamma(A)$.

Suppose we are given a path $p = (p_1 = x, p_2, \ldots, p_r = z)$ that connects statuses x to z on $\Gamma(A)$. Also, we are given another path $q = (q_1 = i, q_2, \ldots, q_r = k)$ connecting sites i to k on $\Gamma(D')$. Note that p and q have the same length of $r - 1$. We call a path $s = (s_1, \ldots, s_r)$ on $\Gamma(H)$ a *product path of p and q*, denoted $s = p \diamond q$, if for each $1 \leq t \leq r$, $s_t = (p_t)_{(q_t)}$. That is, s_t is the image in $\Gamma(d_{q_t,q_t}A)$ of p_t. For an intuitive visualization, see Figure 4.13. Shown on the left in this figure are

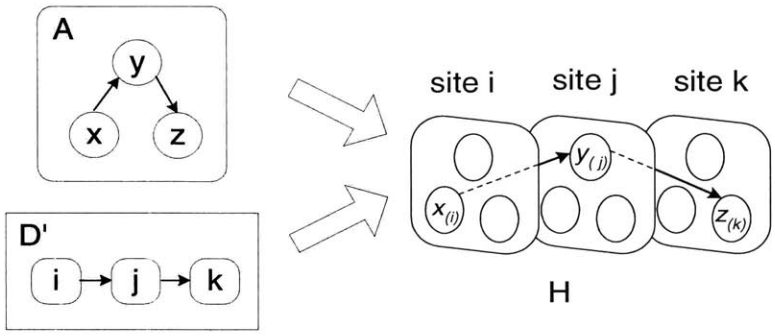


Figure 4.13: A product path.

paths $p = (x, y, z)$ and $q = (i, j, k)$ on graph $\Gamma(A)$ and $\Gamma(D')$ respectively. On $\Gamma(H)$ on the right, we show the product path $s = p \diamond q = (x_{(i)}, y_{(j)}, z_{(k)})$. Loosely speaking, in each step that we move on the product path s, we move a step along p and a step along q at the same time.

THEOREM 4.7

Let $H = D' \otimes A$ be a homogeneous influence matrix, and let $s = p \diamond q$ be a product path. Then edges on s always exist on $\Gamma(H)$. Conversely, every path s on a homogeneous $\Gamma(H)$ can be written as $s = p \diamond q$ where p and q are paths on $\Gamma(A)$ and $\Gamma(D')$ respectively.

Proof. An edge $(x_{(i)}, y_{(j)})$ on $\Gamma(H)$ exists if and only if $d_{ji}a_{xy} > 0$. This is true if and only if $a_{xy} > 0$ and $d_{ji} > 0$. Or equivalently, edges (x, y) and (i, j) must both exist on $\Gamma(A)$ and $\Gamma(D')$ respectively. Extending this argument to paths is straightforward. □

Let the statuses on $\Gamma(A)$ be partitioned into classes $R_1, \ldots, R_w$. For each class k and each site i, let $R_k^{(i)}$ be the image of class R_k on $\Gamma(d_{ii}A)$. Thus, the sets $\{R_1^{(i)}, \ldots, R_w^{(i)}\}$ is also a partition

of $\phi(i)$, the statuses inside boundary of site i. Two integers a and b are called *relatively prime* if their greatest common divisor is 1.

THEOREM 4.8

Let all the variables be as defined in the previous paragraph. If D is irreducible, and if the period of D is relatively prime to the period of every class R_k, then there are w classes in $\Gamma(H)$ with the kth class equal to

$$P_k \triangleq R_k^{(1)} \cup \cdots \cup R_k^{(n)}.$$

Moreover, P_k is a globally recurrent class if and only if R_k is a recurrent class with respect to $\Gamma(A)$.

Proof. See Appendix A. □

Further reading on product graphs can be found in [28].

Recurrent States We now remove the restriction to homogeneous influence models and return to the case of general influence models. Our main question is still: what are the recurrent states of $\Gamma(G)$ with an irreducible network graph? Using the hopping dot analogy, if all the sites were disconnected, then the dots would eventually drift into one of the recurrent classes of their respective local chains. With the inter-site connections, having all dots trapped inside locally recurrent chains is no longer sufficient to guarantee that the influence process has settled into a recurrent state of $\Gamma(G)$. Recall from Figure 4.6 of Example 1 that even if every dot is trapped in a globally recurrent class (i.e. a recurrent class in $\Gamma(H)$), but different dots reside in different classes, then some dot can still hop out of its class due to inter-site influences.

On the other hand, if the dots are all trapped inside the *same* globally recurrent class in $\Gamma(H)$, then they can never leave. This is not only because the dots do not have a path out of the globally recurrent class, but also because they will never receive any influence to hop out of it; all the influences are from within that global class, and will always stay inside it, as shown in Figure 4.14. In this diagram, it is assumed that there are three sites, and the globally recurrent class is the union of all the three locally recurrent classes shown.

At this point, we are tempted to conjecture that in the recurrent states of an influence process, all the dots will be permanently locked inside a single globally recurrent class. To confirm this conjecture we must be able to answer the following question: does a globally recurrent class always have "room" to trap the dots from all sites? In other words, is it possible that a globally recurrent class does not include a status from every site? If there exists some globally recurrent class that does not, then our conjecture cannot be correct; there would have to be at least two active

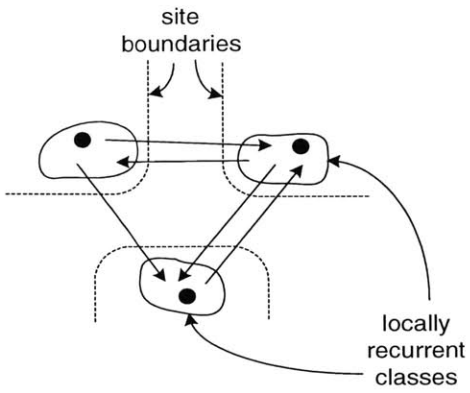


Figure 4.14: A segment of $\Gamma(H)$ showing all the dots trapped in the same globally recurrent class. The globally recurrent class is all the three locally recurrent classes combined.

globally recurrent classes involved in a recurrent state in order to hold all the dots. Consequently, it would be possible that in the recurrent states of an influence process, the dots would still shuffle in and out of globally recurrent classes. See Figure 4.15 for instance. In this 3-site influence graph,

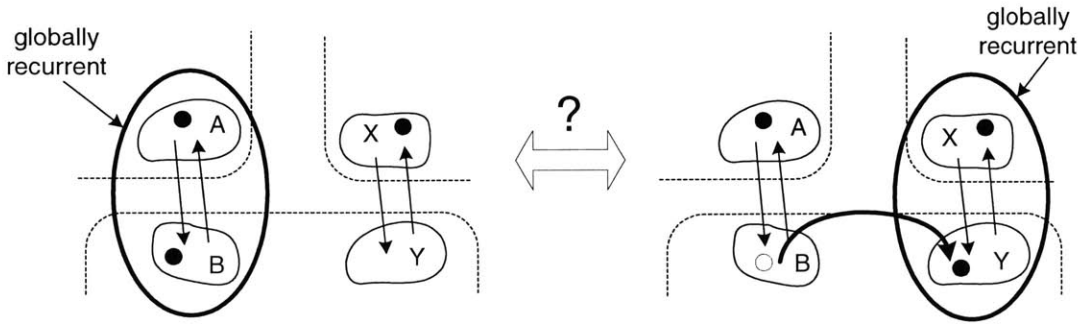


Figure 4.15: If a globally recurrent class did not include a locally recurrent class from every site, then it would not be able to permanently lock in the dots.

the locally recurrent classes A and B together constitute a globally recurrent class, while X and Y make up another. In the left figure, even if two dots have already been captured in A and B, the dot in the locally recurrent class X can exert influence on class Y. Eventually the dot at B will leap back and forth randomly between B and Y, proving that neither of the globally recurrent classes is able to lock in those dots permanently.

As it turns out, a case such as Figure 4.15 cannot happen. The reason is due to the following lemma.

LEMMA 4.9

If $\Gamma(D')$ is irreducible, then every globally recurrent class includes at least one status from every

site.

Proof. Let R be a globally recurrent class and let x be any status on it. Let i be the site whose boundary encloses x, i.e., $x \in \phi(i)$. For any other site j on $\Gamma(D')$, there must exist a path on $\Gamma(D')$ from i to j, because $\Gamma(D')$ is irreducible. By Corollary 4.3, there must also be a path from x to some status $y \in \phi(j)$. Now because R is recurrent, it must include every status that is reachable by x. Therefore, $y \in R$. Since j is arbitrary, this statement must apply to every site. □

COROLLARY 4.10

For every site i for which $d_{ii} > 0$, at least one locally recurrent class of $\phi(i)$ is entirely contained within a globally recurrent class.

Proof. Because a globally recurrent class always exists, we can apply Lemma 4.9, which states that it must have a status in common with $\phi(i)$. Then because $d_{ii} > 0$, the ith local chain exists, which means the globally recurrent class must include a whole locally recurrent class, as explained earlier. □

We can see from the examples in Sec. 4.4.1 that they all conform with Lemma 4.9. Intuitively, this means that within each globally recurrent class, there is room for the dots from every site. If all the dots are held inside the same globally recurrent class, they will remain there.

However, *will* the dots ever end up in the same globally recurrent class? Indeed, we are not even sure whether each dot will move into a globally recurrent class at all. At this point, all we know is that if the local chain exists, the dots will drift into one of their locally recurrent classes, which, unfortunately, may not belong to a globally recurrent class. For example, consider the influence graph $\Gamma(H)$ of Example 3, which is repeated in Figure 4.16. In this figure, status 1

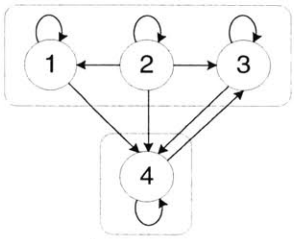


Figure 4.16: The influence graph $\Gamma(H)$ from Example 3 in Sec. 4.4.1.

by itself constitutes a locally recurrent class of the upper site, but with respect to $\Gamma(H)$, it is a globally transient class. The lower dot is obviously stuck at status 4, its only status. Suppose the upper dot starts in status 2. Then it has a positive probability of drifting into status 1. If there

had been no connection, the upper dot would have remained there for all time, and would have proved that a dot need not end up in a globally recurrent class. However, due to the influence from the lower site, the upper dot will eventually hop to status 3, which is its final destination. In this example, the edge (4,3) (the one from status 4 to 3) is the critical path of influence that enables the upper dot to eventually hop out of status 1. Our question now is: in general, does a critical edge such as (4,3) always exist? In other words, is it possible that a dot will drift into a locally recurrent class that happens to be a part of some globally transient class, and remain there forever, because there is no external influence to attract it out? The answer to this question is negative and is answered by the following theorem. This theorem shows that from any status on an influence graph, there is always a path from that status back to some status in the same site that belongs to a globally recurrent class. Translated to the example above, there is guaranteed to be a path from status 1 back to another status (status 3) in the upper site that belongs to a globally recurrent class. Thus, just by being in status 1, the upper dot is effectively *influencing itself* to move into a globally recurrent class.

THEOREM 4.11

If the network graph $\Gamma(D')$ is irreducible, then on $\Gamma(H)$, there exists a path from any status x to another status y such that both the following conditions hold:

(a) y belongs to some globally recurrent class.

(b) y is in the same site as x, i.e., if $x \in \phi(i)$, then $y \in \phi(i)$.

Proof. If x is part of a globally recurrent class, then the claim is immediately satisfied because there must be a path from x to itself. If x is part of a globally transient class T, then let i be the site within which x lies, so $x \in \phi(i)$. Since T is transient, there must be a path from x to some status b outside of T. Whichever site b belongs to, by the fact that D is irreducible and by Corollary 4.3, there must exist a path from b back to some status $c \in \phi(i)$. We know that $c \notin T$, because otherwise we would have found a path from x to b, then from b to c, and from c back to x again (because c and x would be in the same class T), which means x and b would communicate. However, this is not possible because, by choice, $b \notin T$. If c is part of a globally recurrent class, then the claim is proved. If it is not, we can apply the argument over again to find paths from c to d, then d to e, and so on, where these statuses $c, d, e, \ldots$ are all distinct statuses in $\phi(i)$. Because $\phi(i)$ is finite, and because $\phi(i)$ contains statuses that belong to a globally recurrent class (see Lemma 4.9), eventually the path from x must run into one of those statuses. □

By combining Theorem 4.11 with Corollary 4.6, we can conclude that every dot will eventually move into some globally recurrent class, although not necessarily the same one. Since there

could be several globally recurrent classes, different dots may end up moving into different globally recurrent classes. This means even after they all enter globally recurrent classes, if they are not all in the same class, then they can still hop out of their respective globally recurrent classes. Will they ever hop into the same class and thereby get trapped in that class? The answer depends on the structure of $\Gamma(D')$.

4.4.5 Case I: Ergodic $\Gamma(D')$

Eventually every dot must move into some globally recurrent class. Take one such dot from any globally recurrent class. Suppose it is currently occupying status $x \in \phi(i)$. Recall from Corollary 4.6 that if one can find a path from x to any status y in m steps, then m steps later, y has a positive probability of being occupied by some dot. As an implication of this, suppose we can find two paths of the *same length* m, both originating at x, but with one path terminating at some status in, for example, $\phi(1)$ and the other path terminating in $\phi(2)$. Since x is in a globally recurrent class, the destinations of these two paths are still in the same class. The existence of these paths implies that m steps later, there is a positive probability that the dots of sites 1 and 2 would be occupying the destinations of these paths *simultaneously*. This would mean that m steps later, there is a positive probability that the dots of sites 1 and 2 would be inside the same globally recurrent class as x. However, even if both dots actually move inside that global class, they might not remain there, because the dots from the rest of the sites can attract them out of that global class; only when all the dots are in the same global class can we guarantee a permanent capture of the dots.

Thus, to show that every dot can end up the same global class, we must find n different paths of the same length m that lead from x to to all the n different sites on $\Gamma(H)$. Then m steps later, there is a positive probability that all the dots would be trapped in the same globally recurrent class.

How do we find such paths? Let V be the set of sites on $\Gamma(D')$. For a site $i \in V$, recall the definition

$$T^m(i) \triangleq \{v \in V \mid \text{ there exists a path of exactly } m \text{ steps on } \Gamma(D') \text{ from site } i \text{ to } v \},$$

which was first introduced in Sec. 3.4. By Corollary 3.5, if $\Gamma(D')$ is ergodic, then for sufficiently large m, we achieve $T^m(i) = V$, which means that on $\Gamma(D')$ one can go from i to any site in m steps. Using Corollary 4.3, this translates to the ability to go on $\Gamma(H)$ from any status $x \in \phi(i)$ to some status y in any desired site in exactly m steps. So if we pick x to be from the globally recurrent class mentioned in the previous paragraph, then in m steps we can reach some status

within every site. Thus, we have proved the following main result.

THEOREM 4.12

If the network graph $\Gamma(D')$ is ergodic, then there is a one-to-one correspondence between each globally recurrent class of the influence graph $\Gamma(H)$ and a recurrent class of the master chain $\Gamma(G)$. Specifically, every state in any particular recurrent class of $\Gamma(G)$ has all its corresponding dots trapped within a particular globally recurrent class of $\Gamma(H)$.

4.4.6 Case II: Periodic Irreducible $\Gamma(D')$

If $\Gamma(D')$ is periodic and irreducible then it must have the form of subclasses being connected in a circle, as in Figure 2.5. This will also cause H to have the following cyclic form (with appropriate numbering of the sites):

$$H = \begin{bmatrix} 0 & H_1 & & \\ & \ddots & \ddots & \\ & & 0 & H_{d-1} \\ H_d & & & 0 \end{bmatrix}.$$

Each H_k consists of all the block matrices $\{d_{ji}A_{ij}\}$ with i being a site from subclass k and j from subclass $k+1$. For example, Figure 4.17 is an example of a $\Gamma(H)$ whose $\Gamma(D')$ has a period of 3. Inside each site boundary, none of the local chains exists, because $d_{ii} = 0$ for all i.

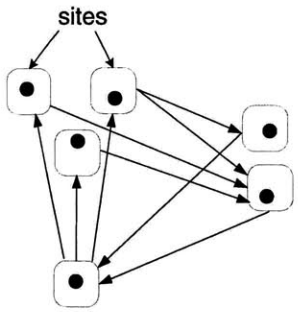


Figure 4.17: An influence graph $\Gamma(H)$ whose $\Gamma(D')$ is periodic.

At some point, all the dots will move into some globally recurrent classes. As before, these could be different classes. If that is the case, then the dots will move around until the number of occupied globally recurrent classes reduces to just one per subclass. Below we offer an intuitive explanation for this.

Suppose that on $\Gamma(H)$ there are exactly 3 global classes: R_1 and R_2, which are both recurrent, and T, which is transient. In Figure 4.18 we show a portion of a typical $\Gamma(H)$, displaying only the connection from sites in subclass i to those in subclass $i+1$. Since a globally recurrent class must

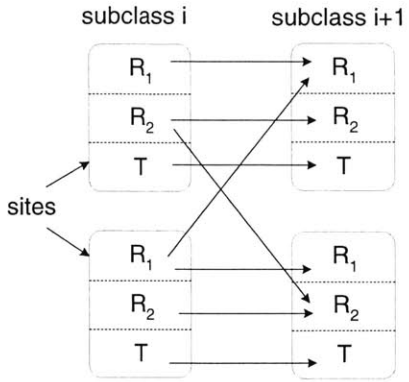


Figure 4.18: A portion of $\Gamma(H)$ between subclass i and $i+1$.

include a status from every site, each site in this figure must have some status that belongs to R_1 and some that belongs to R_2. Hence, within each site, we partition the statuses into 3 groups as shown in this figure. Notice that the edges never mix the recurrent classes; R_1 only connects to R_1, while R_2 only connects to R_2.

Continuing on this example, we now place a dot into each site and observe the one-step transition from time k to $k+1$ in Figure 4.19. These dots are called s, t, u and v, as shown in the figure. At time k, we would like to focus only on s and t. Therefore, they are drawn as solid dots, while u and v are drawn hollow because they are irrelevant to our point. At this time, both s and t have already been inside the globally recurrent classes R_1 and R_2 respectively, and are now exerting their influences on u and v. The edges from s are causing a positive probability that in the following step, u and v would both jump into R_1. On the other hand, t is influencing u and v to move into R_2.

Hence, the state of subclass of $i+1$ at time $k+1$ could have 4 possible outcomes:

(a) u and v both in R_1;

(b) u and v both in R_2;

(c) u in R_1, but v in R_2;

(d) u in R_2, but v in R_1.;

In the figure, we only show cases (a) and (b).

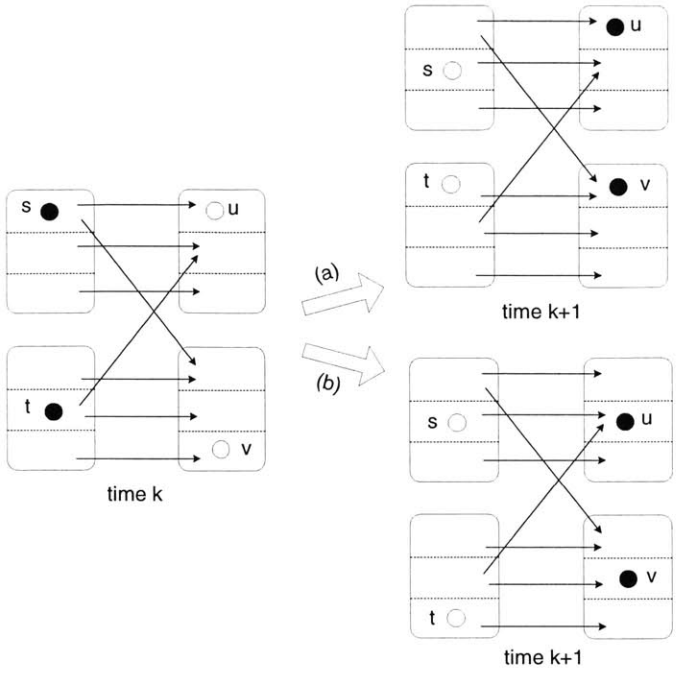


Figure 4.19: Example of two possible transitions for an influence graph with a periodic irreducible network graph.

Suppose (a) actually happens. Then both u and v would be in R_1. Then at time $k+2$, the dots in subclass $i+2$ (not drawn here) must also be in R_1, because they have no other influence beside those coming from R_1. Again in time $k+3$, the dots in subclass $i+3$ would also be in R_1, and so on. Similarly, if (b) happens, then the same argument would apply except that the dots will be in R_2 instead.

If (c) or (d) happens, then both R_1 and R_2 would be occupied by some dots in subclass $i+1$. Our argument above can be applied again and again every time step. Eventually all the dots would hop on to one of the classes, leaving the other one empty. This is consistent with our assertion earlier that dots in the same subclass will eventually end up in the same globally recurrent class.

We can generalize this idea to the case of multiple globally recurrent classes, and the same logic would still apply; within each subclass, only one globally recurrent class will be occupied eventually. Note that different subclasses may occupy different globally recurrent classes. Suppose there are d subclasses in $\Gamma(D')$. Then there could be up to d different occupied globally recurrent classes in a recurrent state of the master chain $\Gamma(G)$. Suppose at some large time index kd, which is a multiple of d, the system has already reached a recurrent state of $\Gamma(G)$. If all the dots in the ith subclass are trapped in the c_ith globally recurrent class, then we say that this influence process

has reached the *limit cycle* $\mathbf{c} = (c_1, \ldots, c_d)$. From this we can infer the occupied globally recurrent classes at time $(kd+1)$ by shifting the entries in $\mathbf{c}$ to the right in a cyclic manner; at time $(kd+1)$, the dots the 1st subclass will all be in the c_dth globally recurrent class, the 2nd subclass will be in the c_1th globally recurrent class, the 3rd in the c_2th and so on. Note that this definition of limit cycles is a generalization of the the one first introduced in Sec. 3.5. The following theorem summarizes our discussion in this section.

THEOREM 4.13
If $\Gamma(D')$ *is irreducible and periodic, and if* $\mathbf{s}[k]$ *is a state in an recurrent class of* $\Gamma(G)$*, then under* $\mathbf{s}[k]$ *all the hopping dots from the same subclass are trapped within the same globally recurrent class of* $\Gamma(H)$.

4.4.7 General Network Graphs

When we allow an influence graph to have an arbitrary $\Gamma(D')$, there end up being many more possibilities for the recurrent states of $\Gamma(G)$. We will settle for only the most basic statement that applies to all influence graphs.

The analysis in the Sec. 4.4.4 can be applied towards influence models with arbitrary network graphs $\Gamma(D')$, but it can only make statements about the sites in the autonomous classes of $\Gamma(D')$. The hopping dots of these sites will eventually be trapped in the *autonomously recurrent* classes defined below, which are essentially the globally recurrent classes with respect to the autonomous subgraph of $\Gamma(H)$.

Autonomously Recurrent Classes For any directed graph $\Gamma(C)$, let S be a set of nodes on $\Gamma(C)$. Recall from Sec. 2.1.1 the definition of a principal submatrix C_S, which is defined as the square submatrix of C that is obtained by selecting the rows and columns that correspond to the nodes in S. The graph $\Gamma(C_S)$ can be obtained from $\Gamma(C)$ by removing all the nodes that are not included in S and all the edges connected to them. Thus, the graph $\Gamma(C_S)$ is always a subgraph of $\Gamma(C)$.

From the previous sections, recall the definition of $\phi(i)$, which represents the set of statuses of $\Gamma(H)$ that are within site i. We now generalize this definition so that it applies to sets. For any subset Q of sites on $\Gamma(D')$, define $\phi(Q) \triangleq \{\phi(q) \mid q \in Q\}$.

Let R be an autonomous class of $\Gamma(D')$. A set S of statuses on $\Gamma(H)$ is an *autonomously recurrent class* of $\Gamma(H)$ if S is a recurrent class of of $\Gamma(H_{\phi(R)})$. The following example will clarify the definition.

Example 5: Let the matrices be given as follows.

$$D' = \begin{bmatrix} .5 & 1 & .4 \\ .5 & & .4 \\ & & .2 \end{bmatrix} \qquad A = \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix} \qquad H = D' \otimes A = \left[\begin{array}{cc|cc|cc} 0 & .5 & 0 & 1 & 0 & .4 \\ 0 & .5 & 0 & 1 & 0 & .4 \\ \hline 0 & .5 & & & 0 & .4 \\ 0 & .5 & & & 0 & .4 \\ \hline & & & & 0 & .2 \\ & & & & 0 & .2 \end{array}\right]$$

The associated graphs $\Gamma(D')$, $\Gamma(A)$, and $\Gamma(H)$ are shown in Figure 4.20.

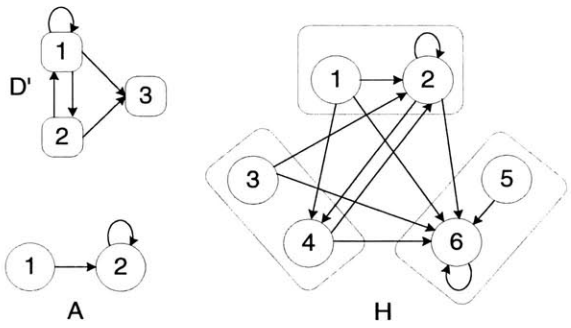


Figure 4.20: Graphs for Example 5.

The graph $\Gamma(D')$ has two classes: $\{1, 2\}$, which is autonomous, and $\{3\}$, which is dependent. With $R = \{1, 2\}$, we get $\phi(R) = \{1, 2, 3, 4\}$. The submatrix $H_{\phi(R)}$ will then be

$$H_{\phi(R)} = \left[\begin{array}{cc|cc} 0 & .5 & 0 & 1 \\ 0 & .5 & 0 & 1 \\ \hline 0 & .5 & & \\ 0 & .5 & & \end{array}\right] \tag{4.13}$$

and $\Gamma(H_{\phi(R)})$ is given in Figure 4.21. Since $\{2, 4\}$ is the recurrent class of $\Gamma(H_{\phi(R)})$, as enclosed in the oval in the diagram, the set of statuses $\{2, 4\}$ of $\Gamma(H)$ is an autonomously recurrent class. □

In Examples 1 through 4 of Sec. 4.4.1, D' is irreducible. In those cases, $\Gamma(D')$ is itself an autonomous class, so an autonomously recurrent class reduces to just a globally recurrent class. However, when D is reducible, an autonomously recurrent class is globally transient with respect to $\Gamma(H)$.

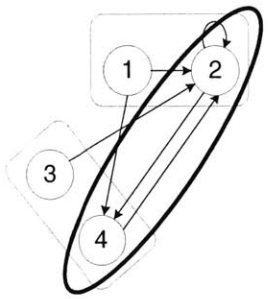


Figure 4.21: The subgraph $\Gamma(H_{\phi(R)})$. The oval show the statuses that are topologically equivalent to the autonomously recurrent class of $\Gamma(H)$.

Intuitively and referring to Figure 2.1, one can think of an autonomously recurrent class as the class containing the "rightmost statuses of the leftmost sites." To explain this, see Figure 4.22. Suppose on $\Gamma(D')$ we arrange the classes so that they point from left to right. Then the autonomous class comprises a set of "leftmost sites." From the subgraph of $\Gamma(H)$ comprising the statuses of those sites, we extract recurrent classes, which are then the "rightmost statuses." These are the autonomously recurrent classes. In the following chapter, we will show that autonomously

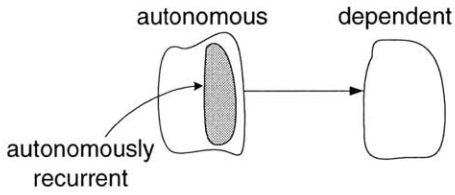


Figure 4.22: "The rightmost statuses of the leftmost sites." An intuitive visualization of an autonomously recurrent class.

recurrent classes are crucial elements in the analysis of the influence matrix H.

Returning to the main point of this section, our analysis in Sec. 4.4.4 can only predict the recurrent states of the sites in an autonomous class R. It says that the dots of the sites in R will eventually be trapped in one of the autonomously recurrent classes. This is because, as a subgraph of $\Gamma(H)$, no status in $\Gamma(H_{\phi(R)})$ receives any influence from a status whose site is not in R. This makes the graph $\Gamma(H_{\phi(R)})$ look as though it is defined from an irreducible D. Thus, the statuses in $\Gamma(H_{\phi(R)})$ will behave as if there is no status other than those in $\phi(R)$. The analysis of the rest of the $\Gamma(H)$ will depend on how these autonomous sites converged. We leave this as a direction for future research.

Chapter 5

Influence Matrix Analysis

This chapter continues our investigation of the general influence model and its dynamics. In the previous chapter, we have characterized the recurrent states of an influence process mostly through the study of the influence graph structure. Our analysis has relied on the picture of the hopping dots in order to aid the visualization and explanation. Although we have identified the classes of the influence graph within which the dots can be "trapped," there still remain important questions. For instance, what are the probabilities that the dots would be trapped in a given class? How do such probabilities evolve as a function of time? As mentioned earlier, the straightforward approach to such questions is to resort to analyzing the master Markov chain $\Gamma(G)$, the master Markov chain in which each status corresponds to a possible outcome of the state vector. If we could afford the computation of G^n for all n, then the above questions would be easy to answer. However, this is practically impossible due to the large size of the problem, as the dimension of G is exponential in the number of sites. Thus, any calculation involving G that can be manipulated so that it only involves the influence matrix H would be of great practical benefit, because the size of H is only linear in the number of sites, and thus much smaller than that of G in general. This is, therefore, our motivation in this chapter for analyzing the matrix H rigorously, and establishing its precise relation with G.

While the approach to studying the influence model in the previous chapter has been graphical, the analysis in this chapter will largely involve linear algebra. We first analyze the eigenstructure of the matrix H. We introduce a class of matrices called the *event* matrices. This class of matrices will be the important element linking H and G. Then we analyze G and some of its eigenstructure. Using the relation between G and H, we will obtain certain projected probabilities of the influence process. These projected probabilities will be shown to be meaningful quantities in their own right. Finally, we will explain how the eigenvalues of H are related to those of G, and how this relation can be interpreted intuitively.

Before reading the following sections, it might be helpful to review the notation on generalized Kronecker product in Sec. 4.3.1, as this will used throughout the chapter.

5.1 The Influence Matrix H

In this section, some basic properties of H will be shown. The main message is that the autonomously recurrent classes defined at the end of Chapter 4 are to influence matrices what recurrent classes are to stochastic matrices; they correspond to principal submatrices of H that contain the eigenvalue at 1.

Recall that an *influence matrix* is defined as

$$H \triangleq D' \otimes \{A_{ij}\}$$

where D is an $n \times n$ stochastic matrix, and each A_{ij} is an $m_i \times m_j$ nonnegative matrix. In addition, each row of A_{ij} must sum to 1, i.e., A_{ij} must satisfy $A_{ij}\mathbf{1}_{m_j} = \mathbf{1}_{m_i}$. As in the previous chapter, it will be assumed throughout this section that $\Gamma(D')$ is a connected graph. In general, the influence matrix H is not stochastic, because neither its row sum nor its column sum is the all-ones vector. Still, H is in many ways like a stochastic matrix: it is nonnegative, and as we will show, has 1 as its dominant eigenvalue. In a way, this is not too surprising, since H is derived from a convex combination of stochastic matrices.

In the following discussion, we will show that the dominant eigenvalue of every influence matrix is 1, and in the process of doing so, we will demonstrate how the multiplicity of this eigenvalue is tied to the structure of the underlying influence graph.

First, some eigenvalues in $\sigma(H)$, the set of eigenvalues (or the spectrum) of H, are identified by the following theorem.

THEOREM 5.1

Let $H = D' \otimes \{A_{ij}\}$ be any influence matrix. If $\mathbf{w}$ is a left eigenvector of D with a corresponding eigenvalue λ, then $(\mathbf{w} \otimes \{\mathbf{1}_{m_i}\})$ is a right eigenvector of H with the same eigenvalue. Conversely, if any right eigenvector $\mathbf{v}$ of H with a corresponding eigenvalue of τ can be expressed as $\mathbf{a} \otimes \{\mathbf{1}_{m_i}\}$, then $\mathbf{a}$ is a left eigenvector of D with eigenvalue τ.

Proof. Using the fact that $A_{ij}\mathbf{1}_{m_j} = \mathbf{1}_{m_i}$,

$$\begin{aligned} H(\mathbf{w} \otimes \{\mathbf{1}_{m_i}\}) &= \begin{bmatrix} d_{11}A_{11} & \cdots & d_{n1}A_{1n} \\ \vdots & & \vdots \\ d_{1n}A_{n1} & \cdots & d_{nn}A_{nn} \end{bmatrix} \begin{bmatrix} w_1\mathbf{1}_{m_1} \\ \vdots \\ w_n\mathbf{1}_{m_n} \end{bmatrix} \\ &= \begin{bmatrix} (d_{11}w_1 + \cdots + d_{n1}w_n)\mathbf{1}_{m_1} \\ \vdots \\ (d_{1n}w_1 + \cdots + d_{nn}w_n)\mathbf{1}_{m_n} \end{bmatrix} \\ &= \lambda(\mathbf{w} \otimes \{\mathbf{1}_{m_i}\}). \end{aligned}$$

The converse of the theorem also follows in a straightforward manner. □

Theorem 5.1 states that $\sigma(D') \subset \sigma(H)$, or equivalently, $\sigma(D) \subset \sigma(H)$. This provides us with

n eigenvalues of H. Since the dimension of H is $\sum_i m_i$, there are another $(\sum_i m_i - n)$ eigenvalues of H to be found. In general there is no simple way to describe them, except for the case of homogeneous influence matrices, where $H = D' \otimes A$. In that case, being a standard Kronecker product, the eigenvalues will be all possible combinations of the eigenvalues of D' and A, i.e.,

$$\sigma(H) = \{\lambda_i \widetilde{\lambda}_j \,|\, \lambda_i \in \sigma(D'),\ \widetilde{\lambda}_j \in \sigma(A)\}$$

and their eigenvectors will be the Kronecker products of the corresponding eigenvectors from D' and A. See [27], Theorem 4.2.12 for proof.

Because $\sigma(D') \subset \sigma(H)$, one of the eigenvalues in $\sigma(H)$ must be 1. Unfortunately, we still cannot apply the Perron-Frobenius Theorem to conclude that 1 is the dominant eigenvalue of H, because, among other things, H might be reducible. Our approach, therefore, is to identify certain submatrices of H that can shown to be irreducible, then make statements about their dominant eigenvalues, and then relate them to the dominant eigenvalue of H.

Kronecker Factorization For any subset S of statuses on $\Gamma(H)$, the (principal) submatrix H_S can be obtained by selecting the rows and columns of H that correspond to the statuses in S, or equivalently, by deleting the rows and columns that do not correspond to the statuses in S. After the deletion, let R be the set of sites that still has at least one status remaining in H_S. Now define D_R to be the (principal) submatrix of D corresponding to the sites in R. Then submatrix H_S can still be factorized as

$$H_S = D'_R \otimes \{\widetilde{A}_{ij}\}. \tag{5.1}$$

where $\widetilde{A}_{ij}$ is a submatrix of some $A_{i'j'}$. We write D'_R in (5.1) to emphasize the fact that it is a principal submatrix of D'. For example, suppose

$$D' = \begin{bmatrix} .6 & 1 & .5 \\ .4 & & .4 \\ & & .1 \end{bmatrix}, \qquad \text{and} \qquad H = D' \otimes \{A_{ij}\} = \left[\begin{array}{cc|c|cc} .6 & & 1 & .5 & \\ & .6 & & .1 & .4 \\ \hline .2 & .2 & & .4 & \\ .2 & .2 & & & .4 \\ \hline & & & .1 & \\ & & & & .1 \end{array}\right]$$

Although the 9 matrices $\{A_{ij} \mid 1 \le i, j \le 3\}$ are not shown explicitly, then can be inferred from H

above. For instance, we can infer that

$$A_{11} = \begin{bmatrix} 1 & \\ & 1 \end{bmatrix}, \text{ and } \qquad A_{13} = \begin{bmatrix} 1 & \\ .2 & .8 \end{bmatrix} \tag{5.2}$$

because the top-left block of H is $d_{11}A_{11} = (0.6)\,[\begin{smallmatrix} 1 & \\ & 1 \end{smallmatrix}]$, while the top-right block of H is $d_{31}A_{13} = (0.5)\,[\begin{smallmatrix} 1 & \\ .2 & .8 \end{smallmatrix}]$.

If $S = \{1, 5, 6\}$, then to obtain H_S we have to eliminate rows $\{2, 3, 4\}$ and columns $\{2, 3, 4\}$ from H. After the elimination, the sites that still have at least one status remaining in H_S are sites $R = \{1, 3\}$. The resulting D'_R and H_S are

$$D'_R = \begin{bmatrix} .6 & .5 \\ & .1 \end{bmatrix}, \qquad \text{and} \qquad H_S = \left[\begin{array}{c|cc} .6 & .5 & \\ \hline & .1 & \\ & & .1 \end{array}\right].$$

The first crucial point to be reiterated here is that D'_R is a submatrix of D', obtained by deleting row 2 and column 2 from D'. Note that D_R is no longer stochastic. We can write $H_S = D'_R \otimes \{\widetilde{A}_{ij}\}$. Again, the 4 matrices $\{\widetilde{A}_{ij} \mid 1 \leq i \leq 2\}$ can be inferred from H_S. For example,

$$\widetilde{A}_{11} = 1, \text{ and } \qquad \widetilde{A}_{12} = [1 \quad]. \tag{5.3}$$

The second crucial point here is that each $\widetilde{A}_{ij}$ is a submatrix of some $A_{i'j'}$. For instance, by comparing (5.3) to (5.2), we can see that $\widetilde{A}_{11}$ is a submatrix of A_{11} and $\widetilde{A}_{12}$ is a submatrix of A_{13}.

Using the procedure of rows and columns deletion as described, we are always left with a unique factorization of H_S in the form of eq. (5.1). Therefore, for any subset S of statuses on $\Gamma(H)$, we will refer such expression of H_S as the *Kronecker factorization.*

Recall the following definition of an autonomously recurrent class from Sec. 4.4.7. We repeat it here for convenience of reference.

Definition 1 Let R_0 be an autonomous class of $\Gamma(D')$. A set S of statuses on $\Gamma(H)$ is an *autonomously recurrent class* of $\Gamma(H)$ if S is a recurrent class of $\Gamma(H_{\phi(R_0)})$.

An autonomously recurrent class always exists in any influence graph. This is because an autonomous class R_0 always exists in $\Gamma(D')$, and in $\Gamma(H_{\phi(R_0)})$ there must always be a globally recurrent class.

The following short lemma will be used in the proof of Theorem 5.4

LEMMA 5.2
For a given class R of $\Gamma(D')$, D_R is stochastic if and only if R is autonomous.

Proof. Given that R is a class, D_R is stochastic if and only if R is recurrent in $\Gamma(D)$. But by Corollary 2.4, R is recurrent in $\Gamma(D)$ if and only if it is autonomous in $\Gamma(D')$. Combining the two statements, we have proved the above assertion. □

So far, the term *stochastic matrix* has only been applied to square matrices. We now generalize it so that it applies to a non-square matrix. A $p \times q$ nonnegative matrix A is *stochastic* if $A\mathbf{1}_q = \mathbf{1}_p$. It is *substochastic* if $A\mathbf{1}_q \leq \mathbf{1}_p$ with the inequality being strict in at least one position.

LEMMA 5.3
Let S be a class of $\Gamma(H)$, and let $H_S = D'_R \otimes \{\widetilde{A}_{ij}\}$ be the Kronecker factorization of H_S. Then S is a recurrent class of $\Gamma(H_{\phi(R)})$ if and only if every $\widetilde{A}_{ij}$ is stochastic.

Proof. Fix an $\widetilde{A}_{ij}$. Let $A_{i'j'}$ be the matrix from which the submatrix of $\widetilde{A}_{ij}$ is extracted. Thus, $i', j' \in R$ and consequently, $\phi(i') \subset \phi(R)$ and $\phi(j') \subset \phi(R)$. Because $A_{i'j'}$ is stochastic, the only way $\widetilde{A}_{ij}$ can be substochastic is that a nonzero column of $A_{i'j'}$ has been dropped in order to obtain $\widetilde{A}_{ij}$. This is possible if and only if there is an edge from some status $x \in \phi(i') \cap S$ to another status $y \in (\phi(j') - S)$.

If S is a recurrent class of $\Gamma(H_{\phi(R)})$, then such an edge (x, y) cannot exist, because S must include every status in $\phi(j') \subset R$ that can be accessed from x. Therefore, by the reason in the above paragraph, $\widetilde{A}_{ij}$ is stochastic. Since i and j are arbitrary sites, we have proved the forward part of the claim.

Conversely, if $\widetilde{A}_{ij}$ is stochastic, then such an edge (x, y) does not exist. Then there is no outgoing edge from $\phi(i') \cap S$ to any status in $(\phi(j') - S)$. Since i and j are arbitrary, we can extend our conclusion to say that there must be no outgoing edge from S to any status in $\phi(R)$. In other words, S is a recurrent class of $\Gamma(H_{\phi(R)})$. This proves the reverse part of the claim. □

THEOREM 5.4
Let S be a class of $\Gamma(H)$, and let $H_S = D'_R \otimes \{\widetilde{A}_{ij}\}$ be the Kronecker factorization of H_S. Then S is autonomously recurrent if and only if D_R and every $\widetilde{A}_{ij}$ are stochastic.

Proof. First assume that S is an autonomously recurrent class. Let R_0 be the autonomous class of $\Gamma(D')$ from which S is defined in the sense of Definition 1. Since S is a globally recurrent class of $\Gamma(H_{\phi(R_0)})$ whose network graph $\Gamma(D_{R_0})$ is irreducible, S must still retain at least one status of

every site in R_0, Lemma 4.9. This means $R = R_0$. Therefore, $D_R = D_{R_0}$, which is stochastic by Lemma 5.2. To show that the matrices $\{\widetilde{A}_{ij}\}$ are stochastic, we invoke Lemma 5.3 by recognizing that S is a recurrent class of $\Gamma(H_{\phi(R_0)})$. Therefore, every $\widetilde{A}_{ij}$ must be stochastic. Hence, we have proved the forward part of the claim.

Conversely, assume D_R and every $\widetilde{A}_{ij}$ are stochastic, and S is known to be a class. Then D_R must be irreducible. Otherwise, there must be two sites $p, q \in R$ such that p and q do not communicate on $\Gamma(D'_R)$. On the other hand, the fact that $p, q \in R$ means that there exist two statuses $u, v \in S$ such that $u \in \phi(p) \cap S$ and $v \in \phi(q) \cap S$. Since S is a class, u and v communicate. By Corollary 4.3, p and q must communicate as well. Hence, we have a contradiction. So D_R must be irreducible and R must be a class. Because D_R is also stochastic, by Lemma 5.2, R is an autonomous class. Then by Lemma 5.3, S must be recurrent with respect to $\Gamma(H_{\phi(R)})$, which, by definition, means that S is autonomously recurrent. □

Note that Theorem 5.4 only applies when S is known to be a class, which immediately implies that D_R is irreducible. If S is only some set of statuses whose D_R and $\{\widetilde{A}_{ij}\}$ from the Kronecker factorization are stochastic, then it might not be autonomously recurrent. For instance, if we let S be the set of all statuses on $\Gamma(H)$, then clearly $H_S = H$, which yields $H = D' \otimes \{A_{ij}\}$ as its Kronecker factorization. In this case, even though D and $\{A_{ij}\}$ are certainly stochastic, S will not be autonomously recurrent —or even a class at all— if D is, for instance, reducible.

The main consequence of Theorem 5.4 is that it qualifies H_S for the Perron-Frobenius Theorem, which is the key for establishing the following statement.

THEOREM 5.5

For any autonomously recurrent class S on $\Gamma(H)$, H_S has a unique dominant eigenvalue at 1.

Proof. By Theorem 5.4, we can express H_S as $H_S = D'_R \otimes \{\widetilde{A}_{ij}\}$, where D_R and every $\widetilde{A}_{ij}$ are stochastic. Because D_R is irreducible and stochastic, by Corollary 2.8 D_R must have a positive left eigenvector π. Because H_S is an influence matrix, we can apply Theorem 5.1, to conclude that it must have an eigenvalue at 1 with a corresponding eigenvector of the form $\pi \otimes \{\mathbf{1}_{p_i}\}$. Because S is irreducible, by the Perron-Frobenius Theorem (Theorem 2.7), it must have a unique dominant eigenvalue. But since $\pi \otimes \{\mathbf{1}_{p_i}\}$ is positive, by Corollary 2.9 it must be a right eigenvector of the dominant eigenvalue of H_S. Thus, $\lambda = 1$ must the unique dominant eigenvalue of H_S. □

We have seen that the dominant eigenvalues of autonomously recurrent classes are 1. Could there be any other types of classes of $\Gamma(H)$ whose submatrix also has a dominant eigenvalue at 1?

Also, how can we be sure that the spectral radius of an influence matrix has to be 1? The following two theorems answer these questions.

For a class S of $\Gamma(H)$, whether or not S is autonomously recurrent, H_S is irreducible. Therefore, by the Perron-Frobenius Theorem, H_S has a unique dominant eigenvalue that is real and positive.

THEOREM 5.6
Let S be a class of $\Gamma(H)$ that is not autonomously recurrent, then the real dominant eigenvalue of H_S is strictly less than 1.

Proof. Let $H_S = D'_R \otimes \{\widetilde{A}_{ij}\}$ be the Kronecker factorization of H_S. Now by Theorem 5.4, the fact that S is not autonomously recurrent implies that some of the matrices D_R and $\widetilde{A}_{ij}$ must be substochastic. Thus, we can add some nonnegative matrix to D_R in order to turn it into a stochastic matrix $\overline{D}$. Similarly, we can add to each $\widetilde{A}_{ij}$ some nonnegative matrix to obtain the stochastic $\overline{A}_{ij}$. One of these matrices being added must be nonzero. Let the new Kronecker product be $\overline{H} \triangleq \overline{D}' \otimes \{\overline{A}_{ij}\}$. The graph $\Gamma(\overline{H})$ is irreducible like $\Gamma(H_S)$, because by adding to D_S and $\{\widetilde{A}_{ij}\}$, we can only create more edges. Therefore, the entire set of nodes on $\Gamma(\overline{H})$ is an autonomously recurrent class of $\Gamma(\overline{H})$ itself. Hence, by Theorem 5.4, $\overline{H}$ has a dominant eigenvalue at 1. Because H_S is irreducible, by Corollary 2.11, its dominant eigenvalue must be less than that of $\overline{H}$. This completes the proof. □

THEOREM 5.7
The dominant eigenvalue of H is 1. Moreover, the (algebraic) multiplicity of this eigenvalue is equal to the number of autonomously recurrent classes on $\Gamma(H)$.

Proof. Let $\{S_1, \ldots, S_s\}$ be a partition of the statuses of $\Gamma(H)$ into classes. By Corollary 2.3, the eigenvalues of H are those of the H_{S_i}'s, counting multiplicities. Then the assertion follows from Theorems 5.5 and 5.6. □

5.2 The Event Matrix B

The previous section describes a basic feature of the eigenstructure of an influence matrix by showing that its dominant eigenvalue is always at 1. In this section and the next, we will shift our focus to an important matrix called the event matrix, which is denoted by B. It is a concept that needs to be introduced before we proceed to the topic of master Markov chains in the following

section.

Recall that the state of an influence process at time k is captured in the state vector $\mathbf{s}[k]$. In this section, we systematically list each possible outcome of $\mathbf{s}[k]$, and stack these outcomes row by row into one large matrix called the *event matrix* B. Thus, the rows of B can be regarded as the sample space of the influence process. The importance of B is twofold. First, by listing the outcomes as rows, B provides a reference for the way we order those events. Second, B will turn out to play a crucial role in understanding the overall dynamics of the influence process.

5.2.1 Definition

Given a set of integers $m_1, \cdots, m_n$ we construct a sequence of matrices $\{B_{(i)}\}$ from $i = 1$ to $i = n$ through a recursive procedure as follows:

$$\begin{aligned} B_{(1)} &\triangleq I_{m_1} \\ B_{(i)} &\triangleq [\, B_{(i-1)} \otimes \mathbf{1}_{m_i} \mid \mathbf{1}_{\mu_{i-1}} \otimes I_{m_i} \,] \end{aligned} \tag{5.4}$$

where $\mu_i \triangleq \prod_{j=1}^{i} m_j$. The matrix $B_{(i)}$ is defined as the *event matrix for* $m_1, \cdots, m_i$. In particular, $B_{(n)}$ is referred to as simply the *the event matrix* and, will be denoted by

$$B \triangleq B_{(n)}.$$

For example, if $(m_1, m_2) = (2, 3)$, then

$$\begin{aligned} B_{(1)} &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \text{ and} \\ B_{(2)} &= B \\ &= [B_{(1)} \otimes \mathbf{1}_3 \mid \mathbf{1}_2 \otimes I_3] \\ &= \left[\begin{array}{cc|ccc} 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{array}\right]. \end{aligned} \tag{5.5}$$

The dimension of B is $(\prod_i m_i) \times (\sum_i m_i)$. So in general, it is a very tall and narrow matrix.

In terms of the influence model, if the integer m_i represents the order of the local chain A_{ii}, then as described earlier, each row of B is one possible outcome for the state vector

$$\mathbf{s}'[k] = \begin{bmatrix} \mathbf{s}_1'[k] & \cdots & \mathbf{s}_n'[k] \end{bmatrix}.$$

Each row of B is thus called an *event*. To match the dimension of each vector $\mathbf{s}_i'[k]$, notice how within an event entries can be partitioned into contiguous blocks of sizes $\{m_i\}$. For example, each row of B in (5.5) has two blocks of sizes 2 and 3 respectively. Within each block is a single nonzero entry of 1 indicating the status the corresponding site. Therefore, if we let

$$\mathbf{t}_i \triangleq \begin{bmatrix} \mathbf{0}_{(m_1+\cdots+m_{i-1})} \\ \mathbf{1}_{m_i} \\ \mathbf{0}_{(m_{i+1}+\cdots+m_n)} \end{bmatrix} \tag{5.6}$$

then for all $1 \leq i \leq n$,

$$B\mathbf{t}_i = \mathbf{1}. \tag{5.7}$$

THEOREM 5.8
For $1 \leq i \leq n$,

$$rank\Big(B_{(i)}\Big) = \Big(\sum_{k=1}^{i} m_k\Big) - i + 1$$

Proof. See Appendix B. □

5.2.2 Event Addresses

It is more convenient to refer to the events by a multiple-integer index where the integers indicate the statuses of the corresponding sites. That is, we can assign to each row of B an *address*, which is an n-tuple string of the form $\mathbf{j} = (j_1, \cdots, j_n)$ where $1 \leq j_i \leq m_i$. An event of address $\mathbf{j}$ is defined as row $\mathbf{j_m}$ of B where

$$\mathbf{j_m} \triangleq (j_1 - 1)(m_2 \cdots m_n) \; + \; (j_2 - 1)(m_3 \cdots m_n) \; + \; \cdots \; + \; j_n.$$

For example, the addresses for each event in the B in (5.5) are

$$B = \left[\begin{array}{cc|ccc} 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{array}\right] \begin{array}{l} \rightarrow (1,1) \\ \rightarrow (1,2) \\ \rightarrow (1,3) \\ \rightarrow (2,1) \\ \rightarrow (2,2) \\ \rightarrow (2,3) \end{array}$$

The event that corresponds to address $\mathbf{j}$ will be denoted by $\mathbf{b_j}$. Notice that $\mathbf{j}$ and $\mathbf{b_j}$ can be derived from one another as long as we know the sizes $\{m_i\}$ of the local chains. In this sense, the address of an event *is* the description of that event; they are merely two different representations of the same thing. This reference-by-address scheme will be used in the rest of the chapter.

5.2.3 The Null Space of B

The null space $\mathcal{N}(B)$ of B arises as a connection between H and G. We need it as a tool to classify the eigenvectors of H. Let A be an $(m_1 + \cdots + m_n) \times (n-1)$ matrix whose entries are as given:

$$A \triangleq \begin{bmatrix} \mathbf{1}_{m_1} & & & & \\ -\mathbf{1}_{m_2} & \mathbf{1}_{m_2} & & & \\ & -\mathbf{1}_{m_3} & & & \\ & & \ddots & & \\ & & & \mathbf{1}_{m_{n-1}} & \\ & & & -\mathbf{1}_{m_n} & \end{bmatrix}.$$

THEOREM 5.9

The columns of A form a basis for $\mathcal{N}(B)$.

Proof. By recalling the definition in eq. (5.6), each column of A can be written as $(\mathbf{t}_i - \mathbf{t}_{i+1})$. Therefore, using (5.7),

$$B(\mathbf{t}_i - \mathbf{t}_{i+1}) = \mathbf{1} - \mathbf{1} = 0.$$

So $BA = 0$. Next because A is in echelon form, all the columns are linearly independent. Furthermore, the number of columns of A matches $\dim(\mathcal{N}(B)) = n - 1$ as stated in Theorem 5.8. Therefore, the columns of A form a basis for $\mathcal{N}(B)$. □

COROLLARY 5.10

$$\mathcal{N}(B) = \left\{ \mathbf{v} \;\middle|\; \mathbf{v} = \mathbf{a} \otimes \{\mathbf{1}_{m_i}\} \text{ and } \mathbf{a}'\mathbf{1} = 0 \right\} \tag{5.8}$$

Proof. From Theorem 5.9, any $\mathbf{v} \in \mathcal{N}(B)$ can be written as $\mathbf{v} = A\gamma$ for some $\gamma \in \mathbb{R}^{n-1}$. Since

$$A\gamma = \begin{bmatrix} \gamma_1 \mathbf{1}_{m_1} \\ (\gamma_2 - \gamma_1)\mathbf{1}_{m_2} \\ \vdots \\ -\gamma_{n-1}\mathbf{1}_{m_n} \end{bmatrix},$$

the coefficients of the $\mathbf{1}_{m_i}$'s sum to zero as claimed. Conversely, if we are given any $\mathbf{v}$ in the form in $\mathbf{v} = \mathbf{a} \otimes \{\mathbf{1}_{m_i}\}$ where $\mathbf{a}'\mathbf{1} = 0$, then it is straightforward solve for γ so that $\mathbf{v} = A\gamma$. □

5.3 The Master Markov Chain G

We now formally construct the state-transition matrix G of the *master Markov chain* $\Gamma(G)$. As mentioned earlier, the Markov chain $\Gamma(G)$ will be one in which each status represents a possible snapshot or state of the influence process. Since B lists all the possible events of the process, it is no surprise that there should be some sort of relation among G, B and H, and we will indeed see such a relation shortly.

5.3.1 Definition

Given an influence matrix H, its *master Markov chain* $\Gamma(G)$ is defined as the Markov chain whose every status represents a possible state of the influence process. Since there are $\mu_n = m_1 \cdots m_n$ possible outcomes for the state $\mathbf{s}[k]$ of the influence process, the dimension of G is $\mu_n \times \mu_n$. We will order these states in the same way that they are ordered in the event matrix B. Each row of G, as well as each column, can then be referenced by an n-tuple address $\mathbf{j} = (j_1, \ldots, j_n)$ as introduced Sec. 5.2.2. Consequently, we can reference an entry of G using a pair of addresses such as $(\mathbf{j}, \mathbf{k})$ where $\mathbf{j}$ is the address of the row and $\mathbf{k}$ the address of the column. For any two valid addresses $\mathbf{j}$ and $\mathbf{k}$, the corresponding entry of G is defined as

$$g_{\mathbf{j},\mathbf{k}} = Prob\Big(\mathbf{s}[k+1] = \mathbf{b}_\mathbf{k} \;\Big|\; \mathbf{s}[k] = \mathbf{b}_\mathbf{j} \Big) \tag{5.9}$$

From (4.7), given that $\mathbf{s}[k] = \mathbf{b_j}$, the next-status PMF is

$$\mathbf{p}'[k+1] = \mathbf{b}_\mathbf{j}' H. \tag{5.10}$$

Let $\{p_{i,\ell}\}$ denote the entries of $\mathbf{p}[k+1]$, where $p_{i,\ell}$ represents the probability that site i will be in status ℓ at time $k+1$, given $\mathbf{s}[k]$. Then for any address $\mathbf{k} = (k_1, \cdots, k_n)$,

$$g_{\mathbf{j},\mathbf{k}} = \prod_{i=1}^{n} p_{i,k_i}. \tag{5.11}$$

By varying over all $\mathbf{j}$ and $\mathbf{k}$ in the order they appear in B, eqs. (5.10) and (5.11) uniquely define all entries of G.

Beside its large size, the matrix G tends to be dense, even if the influence matrix H is sparse. For instance, in the case of a homogeneous influence model where $H = D' \otimes A$, the density of H (the number of nonzero entries relative to the total number of entries) is always no more than that of D. Hence, a sparse D would imply a sparse H. However, if $A > 0$, then no matter how sparse D is, all $(\mu_n)^2$ entries of G would be filled. The reason is that $A > 0$ would mean that every site has a positive probability of being in any status in every time step, which causes $g_{\mathbf{j},\mathbf{k}} > 0$ for all $\mathbf{j}$, $\mathbf{k}$. This shows that G is, in general, a very big and very dense matrix. It is for these very reasons that we have to study the connection between G and H in the hope of relegating certain computations on G to a reduced-order computation involving only H.

5.3.2 G as a Markov Chain

To emphasize the meaning of G, we define a Markov process $\{\mathbf{f}[k]\}$ using G as the state-transition matrix. Thus, the evolution equations of this chain are:

$$\mathbf{q}'[k+1] = \mathbf{f}'[k]G \tag{5.12}$$

$$\mathbf{f}[k+1] = \mathit{Realize}(\mathbf{q}[k+1]) \tag{5.13}$$

On the other hand, let $\{\mathbf{s}[k]\}$ denote the influence process that evolves according to (4.7)-(4.8) with H as its influence matrix. If the initial conditions of these two processes satisfy $\mathbf{f}'[0]B = \mathbf{s}'[0]$, then for any valid address $\mathbf{i}$,

$$\mathit{Prob}\Big((\mathbf{f}[k])_\mathbf{i} = 1 \Big) = \mathit{Prob}\Big(\mathbf{s}[k] = \mathbf{b_i} \Big). \tag{5.14}$$

Eq. (5.14) implies that if one observes the sequence $\{\widetilde{\mathbf{s}}[k]\}$ defined as

$$\widetilde{\mathbf{s}}'[k] = \mathbf{f}'[k]B, \tag{5.15}$$

then there would be no statistical difference between the true influence process $\{\mathbf{s}[k]\}$ and $\{\widetilde{\mathbf{s}}[k]\}$. This is because, by (5.14), $\mathbf{f}[k]$ always selects row $\mathbf{i}$ from B with the same probability that $\mathbf{s}[k]$ is equal to that row. In other words, the Markov process $\{\mathbf{f}[k]\}$ defined above is merely a different representation of the influence process $\{\mathbf{s}[k]\}$.

Since $\widetilde{\mathbf{s}}[k]$ and $\mathbf{s}[k]$ are statistically the same,

$$E(\widetilde{\mathbf{s}}[k]) = E(\mathbf{s}[k]). \tag{5.16}$$

Taking the expectation in (5.15) and substituting (5.16) into it, we have

$$E(\mathbf{s}'[k]) = E(\mathbf{f}'[k])B. \tag{5.17}$$

Note that $E(\mathbf{f}[k])$ is a PMF vector because $E(\mathbf{f}'[k])\mathbf{1} = E(\mathbf{f}'[k]\mathbf{1}) = 1$.

PMF Marginalization Eq. (5.17) is intriguing. It shows that apart from being an event-listing matrix, B can also be thought of as an operator that calculates the marginal probabilities from the PMF vector $E(\mathbf{f}[k])$. To see this, let us partition $\mathbf{s}[k]$ into

$$\mathbf{s}'[k] = \begin{bmatrix} \mathbf{s}'_1[k] & \cdots & \mathbf{s}'_n[k] \end{bmatrix}.$$

Each $E(\mathbf{s}_i[k])$ from (5.17) is a valid PMF vector of length m_i, because

$$E(\mathbf{s}'[k])\mathbf{t}_i = E(\mathbf{f}'[k])B\mathbf{t}_i = E(\mathbf{f}'[k])'\mathbf{1} = 1,$$

where $\mathbf{t}_i$ is defined in (5.6), and we have evoked eq. (5.7) for the second equality. Let $s_{j\ell}[k]$ denote the ℓth entry of $\mathbf{s}_j[k]$. Because each $s_{j\ell}[k]$ is a binary variable, its expected value equals the probability of its being a "1" at that time. Thus, $E(s_{j\ell}[k])$ would tell us the probability of site j being in status ℓ at time k. On the other hand, for any valid address $\mathbf{i}$, $E\left((\mathbf{f}[k])_{\mathbf{i}}\right)$ represents the probability of the influence process being in state $\mathbf{b_i}$ at time k (see eq. (5.14)); each entry of $E(\mathbf{f}[k])$ describes the status of every node simultaneously. So, through the multiplication of B in eq. (5.17), we are effectively converting the *joint PMF* $E(\mathbf{f}[k])$ governing the joint status (or the state) of *all sites* into n different *marginal PMF's* $E(\mathbf{s}_1[k]), \ldots, E(\mathbf{s}_n[k])$ governing the status of each *individual site*. Thus, in the process of marginalizing $E(\mathbf{f}[k])$ to obtain $E(\mathbf{s}[k])$, B "removes" any information regarding the correlation among sites.

5.3.3 Relation Between G and H

THEOREM 5.11
For all $k \geq 1$,

$$G^k B = BH^k \tag{5.18}$$

Proof. From Sec. 2.3, the closed-form expression for $E(\mathbf{f}[k])$ in terms of the initial state vector is

$$E(\mathbf{f}'[k]) = E(\mathbf{f}'[0])G^k. \tag{5.19}$$

After multiplying this by B from the right, and substituting in (5.17), we obtain

$$E(\mathbf{s}'[k]) = E(\mathbf{f}'[k])B = E(\mathbf{f}'[0])G^k B \tag{5.20}$$

On the other hand, from Theorem 4.5 and from the fact that $E(\mathbf{f}'[0])B = E(\mathbf{s}'[0])$,

$$E(\mathbf{s}'[k]) = E(\mathbf{s}'[0])H^k = E(\mathbf{f}'[0])BH^k. \tag{5.21}$$

Equating (5.20) to (5.21), we have

$$E(\mathbf{f}'[0])G^k B = E(\mathbf{f}'[0])BH^k \tag{5.22}$$

Because (5.22) must hold for all initial status vectors $E(\mathbf{f}[0])$, we have arrived at the relation above, as claimed. □

Theorem 5.11 above represents the simple but fundamental relation between G and H. It will be the foundation on which the connection between the spectrum of G and and H is based for the rest of the chapter. Indeed, the simplicity of this theorem is the key to the tractability of the influence model.

5.3.4 Evolution of PMF's

Theorem 5.11 and its proof show that instead of tracking the evolution of $E(\mathbf{f}[k])$, the PMF of the joint status involving every site, one can track the evolution of $E(\mathbf{s}[k])$, the collection of PMF's of the individual status of each site. Although the latter vector provides no information about the correlation among sites, it requires much less computational burden to obtain. The computation of H^k involves matrices of order $(m_1 + \cdots + m_n)$, as compared to G^k, which is of order $\mu_n = m_1 \cdots m_n$.

In order to understand the evolution of $E(\mathbf{s}[k])$, one needs to understand the eigenstructure of H. Specifically, the eigenstructure of H will dictate how fast $E(\mathbf{s}[k])$ will approach the steady state. To clarify this statement, suppose for simplicity that the eigenvalues of H are $\lambda_0, \ldots, \lambda_{\nu_n-1}$, where $\nu_n = m_1 + \cdots + m_n$. Suppose for simplicity that these eigenvalues are distinct so that their eigenvectors are linearly independent. Assume also that these are ordered so that

$$1 = \lambda_0 > |\lambda_1| \geq \cdots \geq |\lambda_{\nu_n-1}|.$$

Let $\mathbf{w}_i$ and $\mathbf{v}_i$ be the left and right eigenvectors corresponding to λ_i respectively, and let these vectors be normalized so that $\mathbf{w}_i'\mathbf{v}_i = 1$. Then by spectral decomposition,

$$H^k = \mathbf{v}_0\mathbf{w}_0' + \sum_{i=1}^{\nu_n-1} \lambda_i^k \mathbf{v}_i\mathbf{w}_i'. \tag{5.23}$$

Since the eigenvalues are distinct, all the eigenvalues in the second-term in (5.23) are smaller than 1 in magnitude. The summation in the second-term would thus eventually vanish as $k \to \infty$, leaving only the steady state term $\mathbf{v}_0\mathbf{w}_0'$. To relate this to the evolution of the status-PMF, since $E(\mathbf{s}'[k]) = E(\mathbf{s}'[0])H^k$,

$$E(\mathbf{s}'[k]) = E(\mathbf{s}'[0])\mathbf{v}_0\mathbf{w}_0' + \sum_{i=1}^{\nu_n-1} \lambda_i^k E(\mathbf{s}'[0])\mathbf{v}_i\mathbf{w}_i'. \tag{5.24}$$

We say that $E(\mathbf{s}[0])$ *has a component along the ith eigenvector* if $E(\mathbf{s}[0])'\mathbf{v}_i \neq 0$. If $E(\mathbf{s}[0])$ has a component along an eigenvector $\mathbf{v}_i$, then the term involving λ_i^k will remain in the summation in (5.24) for all finite k, because λ_i^k only decays asymptotically to zero.

Recall from Theorem 5.1 that we can express $\mathbf{v}_0$, the right eigenvector corresponding to the eigenvalue at 1, as

$$\mathbf{v}_0 = \rho \otimes \{\mathbf{1}_{m_i}\},$$

where ρ is the left eigenvector of D corresponding to the eigenvalue at 1, and $\rho'\mathbf{1} = 1$. The first term in (5.24) therefore always reduces to

$$E(\mathbf{s}'[0])\mathbf{v}_0\mathbf{w}_0' = \mathbf{w}_0,$$

because

$$\begin{aligned} E(\mathbf{s}'[0])\mathbf{v}_0 &= [\, E(\mathbf{s}'_1[0]) \;\cdots\; E(\mathbf{s}'_n[0]) \,] \begin{bmatrix} \rho_1 \mathbf{1}_{m_1} \\ \vdots \\ \rho_n \mathbf{1}_{m_n} \end{bmatrix} \\ &= \rho_1 + \cdots + \rho_n \\ &= 1. \end{aligned}$$

Therefore, the steady-state value of (5.24) is

$$E(\mathbf{s}'[k]) \rightarrow E(\mathbf{s}'[0])\mathbf{v}_0\mathbf{w}'_0 = \mathbf{w}'_0 \tag{5.25}$$

If $E(\mathbf{s}[0])$ has a component along any eigenvector other than the one with an eigenvalue at 1, then that component reflects the departure of $E(\mathbf{s}[0])$ from the steady-state value of $E(\mathbf{s}[k])$. The rate at which each of these terms decays to zero will be governed by how fast λ_i^k reaches zero as a function of k. This is our motivation for understanding the spectrum of H. By knowing its eigenstructure, we can quantify how much deviation an initial state $E(\mathbf{s}[0])$ has from the steady state, and how fast $E(\mathbf{s}[k])$ will approach the steady state. Moreover, by relating back to the eigenstructure of G, we can determine the conditions on $E(\mathbf{f}[0])$ that will cause $E(\mathbf{s}[0])$ to have particular deviations from the steady state.

5.3.5 Eigenstructure of G and H

For the rest of the chapter, it will be assumed that $H = D' \otimes \{A_{ij}\}$ has distinct eigenvalues. Because by Theorem 5.5, H always has a dominant eigenvalue at 1, and its algebraic multiplicity is equal to the number of autonomously recurrent classes on $\Gamma(H)$, our assumption implies that $\Gamma(H)$ must have only one autonomously recurrent class. It also implies that D has distinct eigenvalues, because otherwise by Theorem 5.1, H would have had the same repeated eigenvalues. Having distinct eigenvalues also means that $\Gamma(D')$ has only one autonomous class.

COROLLARY 5.12

Let $\mathbf{w}$, $\mathbf{v}$ *be a left and right eigenvector of* H *with a corresponding eigenvalue of* λ. *If* $B\mathbf{v} \neq 0$, *then* $B\mathbf{v}$ *is a right eigenvector of* G *with* λ *as its corresponding eigenvalue. Moreover, if* λ *is a non-repeated eigenvalue of* G, *then* $\mathbf{w}' = \widetilde{\mathbf{w}}'B$, *where* $\widetilde{\mathbf{w}}$ *is a left eigenvector of* G *corresponding to* λ.

Proof. Letting $k = 1$ in (5.18) and multiplying by $\mathbf{v}$ from the right, we have $GB\mathbf{v} = BH\mathbf{v} = \lambda B\mathbf{v}$. So if $B\mathbf{v} \neq 0$, it must be a right eigenvector of G with eigenvalue λ. Since λ is an eigenvalue of G, it must have a corresponding left eigenvector $\widetilde{\mathbf{w}}$ such that $\widetilde{\mathbf{w}}'G = \lambda\widetilde{\mathbf{w}}'$. Multiplying (5.18) by $\widetilde{\mathbf{w}}'$ from the left, we get $\widetilde{\mathbf{w}}'GB = \lambda\widetilde{\mathbf{w}}'B = \widetilde{\mathbf{w}}'BH$. This shows that if $\widetilde{\mathbf{w}}'B \neq 0$, then it must be a (transposed) left eigenvector of H with eigenvalue λ. If λ is a non-repeated eigenvalue of G, then as left and right eigenvectors of G with the same eigenvalue, $\widetilde{\mathbf{w}}'(B\mathbf{v}) \neq 0$, which implies that $\widetilde{\mathbf{w}}'B \neq 0$. Thus, $\widetilde{\mathbf{w}}'B$ is a left eigenvector of H_r with eigenvalue λ.

On the other hand, since $\mathbf{w}$ is the eigenvector of H corresponding to λ, and since H has distinct eigenvalues, it must be that $\mathbf{w}' = c\widetilde{\mathbf{w}}'B$, where c is some nonzero constant. Without loss of generality, we can lump c into $c\widetilde{\mathbf{w}}$, and the claim is proved. □

COROLLARY 5.13

Let $\widetilde{\mathbf{v}}$, $\widetilde{\mathbf{w}}$ be a left and right eigenvector of G with a corresponding eigenvalue of λ. If $\widetilde{\mathbf{w}}'B \neq 0$ then $\widetilde{\mathbf{w}}'B$ is a left eigenvector of H with λ as its eigenvalue. Moreover, if λ is a non-repeated eigenvalue of G, then $\widetilde{\mathbf{v}} = B\mathbf{v}$, where $\mathbf{v}$ is a right eigenvector of H corresponding to λ.

Proof. The proof is similar to that of Corollary 5.12. □

Corollary 5.12 implies that certain eigenvalues and eigenvectors of G can be obtained form H. We define the set of *relevant eigenvalues of H* as

$$\kappa \triangleq \{\text{ The eigenvalues of } H \text{ whose corresponding right eigenvectors } \mathbf{v} \text{ satisfy } B\mathbf{v} \neq 0\} \quad (5.26)$$

It then follows from Corollary 5.12 that $\kappa \subset \sigma(G) \cap \sigma(H)$. Note that this definition leaves no ambiguity as to which eigenvalue is relevant, because the eigenvalues of H are distinct by our assumption. (If H contained repeated eigenvalues, then there could be an eigenvalue whose set of corresponding eigenvectors contains those that satisfy $B\mathbf{v} = 0$ as well as those that do not, thus making the definition of κ above ambiguous.)

Another point to note is that the eigenvalue $\lambda = 1$ is always relevant. From Theorem 5.1, a right eigenvector of this eigenvalue is $\mathbf{v} = \rho \otimes \{\mathbf{1}_{m_i}\}$ where ρ is the left eigenvector of D corresponding to $\lambda = 1$. From Corollary 5.10, we know that this $\mathbf{v} \notin \mathcal{N}(B)$ because $\rho'\mathbf{1} \neq 0$.

An eigenvector of H that corresponds to a relevant eigenvalue is called a *relevant eigenvector*. The rest of eigenvalues in $\sigma(H)$ are called *irrelevant eigenvalues*, and are denoted by

$$\kappa_c \triangleq \sigma(H) - \kappa$$

The corresponding eigenvectors of κ_c are called *irrelevant eigenvectors.*

The label "irrelevant" for κ_c is justified by the following observation: regardless of the value of $E(\mathbf{f}[0])$, no expected status vector $E(\mathbf{s}'[0]) = E(\mathbf{f}'[0])B$ will ever have any component along the irrelevant eigenvectors. This is due to the fact that for any irrelevant (right) eigenvector $\mathbf{v}$,

$$E(\mathbf{s}'[0])\mathbf{v} = E(\mathbf{f}'[0])B\mathbf{v} = 0.$$

This means that in the analysis of eq. (5.24) we can altogether disregard those terms that involve irrelevant eigenvalues.

The identity of irrelevant eigenvalues is revealed in the theorem that follows the next lemma.

LEMMA 5.14

The number of relevant eigenvalues is given by:

$$|\kappa| = \sum_{i=1}^{n} m_i - n + 1 = |\sigma(H)| - (n-1).$$

Proof. Let K_c be a matrix of $|\kappa_c|$ columns, with each one being an irrelevant eigenvector corresponding to a different $\lambda \in \kappa_c$. By definition, $BK_c = 0$. Since H contains distinct eigenvalues, the columns of K_c must be linearly independent. From Theorem 5.9, $\dim(\mathcal{N}(B)) = n - 1$. So there can be no more than $n-1$ columns in K_c, which means $|\kappa_c| \leq n - 1$. Because H contains $\sum_i m_i$ distinct eigenvalues, $|\kappa| + |\kappa_c| = \sum_i m_i$. Hence,

$$|\kappa| \geq (\sum_i m_i) - n + 1. \tag{5.27}$$

Now let K be a matrix of $|\kappa|$ columns, each one being a relevant eigenvector of a different $\lambda \in \kappa$. From Corollary 5.12, the columns of BK are the right eigenvectors of G corresponding to eigenvalues in κ. The columns of BK must be linearly independent, because they correspond to distinct eigenvalues. The rank of BK, namely $|\kappa|$, cannot exceed the rank of B, which is $(\sum_i m_i) - n + 1$ by Theorem 5.8. Combining this with (5.27), we conclude that $|\kappa| = (\sum_i m_i) - n + 1$. □

THEOREM 5.15

$$\kappa_c = \left\{ \lambda \;\middle|\; \lambda \in \sigma(D) \text{ and } \lambda \neq 1 \right\} \tag{5.28}$$

Proof. Given that λ is an eigenvalue of D with a corresponding left eigenvector $\mathbf{a}$, by Theorem 5.1

we know

$$H\mathbf{v} = \lambda\mathbf{v} \qquad \text{where } \mathbf{v} = \mathbf{a} \otimes \{\mathbf{1}_{m_i}\}. \tag{5.29}$$

Also,

$$\begin{aligned}
&\lambda \neq 1 \rightarrow \mathbf{a}'\mathbf{1} = 0 && \text{because } \mathbf{a} \text{ and } \mathbf{1} \text{ are left eigenvectors} \\
& && \text{of different eigenvalues (Thm. 1.4.7 in [26])} && (5.30) \\
&\rightarrow B\mathbf{v} = 0 \text{ where } \mathbf{v} = \mathbf{a} \otimes \{\mathbf{1}_{m_i}\} && \text{by Corollary 5.10 and (5.30)} && (5.31) \\
&\rightarrow \lambda \in \kappa_c && \text{by (5.29), (5.31) and (5.26)} && (5.32)
\end{aligned}$$

This proves κ_c contains at least all of $\sigma(D)$ except for the eigenvalue at 1. Because H contains distinct eigenvalues, so must D (see reasoning in the first paragraph of Sec. 5.3.5). This means $|\sigma(D)| = n$, and as a result, $|\kappa_c| \geq n - 1$. But by Lemma 5.14, $|\kappa_c| = n - 1$. So κ_c cannot include any other eigenvalue other than $\sigma(D) - \{1\}$. □

Theorem 5.15 in combination with Theorem 5.1 thus completely describe the irrelevant eigenvalues and eigenvectors for the case of distinct eigenvalues in H.

As for the relevant eigenvalues and eigenvectors, in general, there is no simple way to describe them except for the eigenvalue at 1, whose right eigenvector has been described in Theorem 5.1, or for the case of homogeneous influence matrices $H = D' \otimes A$. In the latter case, any left or right eigenvector of $H = D' \otimes A$ can be factorized into

$$\text{left: } \mathbf{w}_H = \mathbf{v}_D \otimes \mathbf{w}_A \qquad \text{right: } \mathbf{v}_H = \mathbf{w}_D \otimes \mathbf{v}_A \tag{5.33}$$

where in (5.33) the notations $\mathbf{w}_X$ and $\mathbf{v}_X$ denote a left and a right eigenvector of matrix X respectively (see [27] about eigenvectors of Kronecker products). In particular, let ρ and α be the left eigenvectors corresponding to the eigenvalue at 1 of D and A respectively. Then the eigenvectors of H corresponding to the eigenvalue at 1 are

$$\text{left: } \mathbf{1}_n \otimes \alpha \qquad \text{right: } \rho \otimes \mathbf{1}_m \tag{5.34}$$

where n and m are the orders of D and A respectively.

5.3.6 Intuitive Interpretation of κ

The eigenvalues in κ, when considered as a part of $\sigma(G)$, can be intuitively thought of as the eigenvalues whose duties are to "correct" the marginal distribution of $E(\mathbf{f}[k])$ to its proper steady-state value. To understand this, let us return to the discussion in Sec. 5.3.4. As before, for $i = 0, \ldots, \nu_n - 1$, let λ_i denotes the ith eigenvalue of H (where the eigenvalues are sorted by their magnitudes), and let $\mathbf{w}_i$ and $\mathbf{v}_i$ denote corresponding left and right eigenvectors respectively. Likewise, assume that G also has distinct eigenvalues $\widetilde{\lambda}_0 = 1, \ldots, \widetilde{\lambda}_{\mu_n - 1}$, which are similarly sorted by magnitude, and let their corresponding left and right eigenvectors be denoted by $\widetilde{\mathbf{w}}_i$ and $\widetilde{\mathbf{v}}_i$ respectively. By combining Corollaries 5.12 and 5.13 and the assumption that G has distinct eigenvalues, we can see that definition (5.26) is equivalent to

$$\kappa \triangleq \{ \text{The eigenvalues of } G \text{ whose corresponding left eigenvectors } \widetilde{\mathbf{w}} \text{ satisfy } \widetilde{\mathbf{w}}' B \neq 0 \}. \quad (5.35)$$

Assuming that $\widetilde{\mathbf{w}}_i' \widetilde{\mathbf{v}}_i = 1$, the evolution of $E(\mathbf{f}[k])$ can be expressed as

$$E(\mathbf{f}'[k]) = E(\mathbf{f}'[0])G^k = \widetilde{\mathbf{w}}_0' + \sum_{i=1}^{\mu_n - 1} \widetilde{\lambda}_i^k E(\mathbf{f}'[0]) \widetilde{\mathbf{v}}_i \widetilde{\mathbf{w}}_i'. \quad (5.36)$$

where we have assumed that the zeroth right eigenvector has been normalized to $\widetilde{\mathbf{v}}_0 = \mathbf{1}$, so $E(\mathbf{f}'[0])\widetilde{\mathbf{v}}_0 = 1$ because $E(\mathbf{f}[0])$ is a PMF. Note that some of the $\widetilde{\lambda}_i$ in the second term in (5.36) will be from $\kappa \subset \sigma(G) \cap \sigma(H)$. Recall that $E(\mathbf{f}[0])$ is said to have *a component along the ith eigenvector* if $E(\mathbf{f}'[0])\widetilde{\mathbf{v}}_i \neq 0$.

We say that the PMF $E(\mathbf{f}[0])$ has the *correct steady-state marginals*, or simply *correct marginals* if the corresponding $E(\mathbf{s}[0])$ is equal to the steady-state eigenvector of H,

$$E(\mathbf{s}'[0]) = E(\mathbf{f}'[0])B = \mathbf{w}_0'. \quad (5.37)$$

In other words, if $E(\mathbf{f}[k])$ has the correct marginals, then $E(\mathbf{s}[k])$ in (5.24) would simply not evolve, because (5.37) shows that it is already starting from the steady state.

If $E(\mathbf{f}[0]) = \widetilde{\mathbf{w}}_0$, the steady state of G, then it certainly must have the correct marginals, because

$$\widetilde{\mathbf{w}}_0' B = \mathbf{w}_0' \quad (5.38)$$

by Corollary 5.13, but what are the other possibilities? In general, $E(\mathbf{f}[k])$ can have the correct marginals if and only if it is any PMF vector of the form $\widetilde{\mathbf{w}}_0 + \mathbf{x}$ where $\mathbf{x} \in \mathcal{N}(B')$, the left null

space of B, because $(\widetilde{\mathbf{w}}_0 + \mathbf{x})'B = \widetilde{\mathbf{w}}_0'B$.

THEOREM 5.16

$E(\mathbf{f}[0])$ has a component along a relevant eigenvector if and only if $E(\mathbf{f}[0])$ does not have the correct (steady-state) marginals.

Proof. By multiplying B on the right of (5.36), and substituting (5.38), we get

$$\begin{aligned} E(\mathbf{f}'[k])B &= \widetilde{\mathbf{w}}_0'B + \sum_{i=1}^{\mu_n - 1} \widetilde{\lambda}_i^k E(\mathbf{f}'[0])\widetilde{\mathbf{v}}_i\widetilde{\mathbf{w}}_i'B \\ &= \mathbf{w}_0' + \sum_{\widetilde{\lambda}_i \in \kappa} \widetilde{\lambda}_i^k E(\mathbf{f}'[0])\widetilde{\mathbf{v}}_i\widetilde{\mathbf{w}}_i'B. \end{aligned} \tag{5.39}$$

where the second term in the summation only involves the eigenvalues in κ, because of (5.35). If $E(\mathbf{f}[k])$ does not have a component along the relevant eigenvector, then the second term in (5.39) must vanish, leaving only the term $\mathbf{w}_0'$ on the right-hand side. Conversely, if $E(\mathbf{f}'[k])B = \mathbf{w}_0'$, then

$$\sum_{\widetilde{\lambda}_i \in \kappa} \widetilde{\lambda}_i^k E(\mathbf{f}'[0])\widetilde{\mathbf{v}}_i\widetilde{\mathbf{w}}_i'B = 0$$

for all k, which can only be satisfied if $E(\mathbf{f}'[0])\widetilde{\mathbf{v}}_i\widetilde{\mathbf{w}}_i'B = 0$ for all i. Now since $\widetilde{\mathbf{w}}_i'B \neq 0$, it must be that $E(\mathbf{f}'[0])\widetilde{\mathbf{v}}_i = 0$. That is, $E(\mathbf{f}[k])$ does not have a component along any relevant eigenvector. □

This theorem states that if $E(\mathbf{f}[0])$ does not start off with the correct marginals, it will have some component along the relevant eigenvectors. These components represent the deviations from the steady state of $E(\mathbf{f}[k])$, and they will decay according to the eigenvalues in κ, until $E(\mathbf{f}[k])$ have the "corrected" marginals.

Interestingly, if $E(\mathbf{f}[0])$ *does* start off with the correct marginals, but $E(\mathbf{f}[0]) \neq \widetilde{\mathbf{w}}_0$, then the evolution of $E(\mathbf{f}[k])$ would still have contributions from the terms corresponding to the eigenvalues of G that are not in κ. Intuitively, these other eigenvalues are performing the duties of "correcting the correlations" among the sites. This suggests that the spectrum of G can be partitioned into those that correct the marginals, or the averages of each individual site, and those that correct the correlations, or the joint-averages of groups of sites. Indeed, as we shall see in the following chapter, among the correlation-correcting eigenvalues of G, one can make even further partitions: some of them specifically correct the second-order statistics, some the third-order statistics, and so on. For now, the point to make is that each eigenvalue of G has a clearly defined purpose, which is to correct the statistics of $E(\mathbf{f}[0])$.

5.4 To Link or Not To Link

We now tackle an important question that has been raised in Sec. 4.2.1 as a motivation to study the influence model: what are the effects of network connection on the status of each individual site? The answer to this question will be discussed below after some simplifying assumptions. We will then explain why this answer naturally motivates us to study the higher-order influence descriptions in the following chapter.

Again, using the power system scenario, suppose each power plant can be either "normal" or "failed. The operating conditions of the power plant may be (coarsely!) modeled by some Markov chain $\Gamma(A)$ in the form shown in Figure 5.1. Without the connection to the network, a power plant

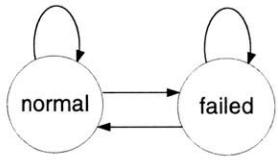


Figure 5.1: A Markov chain modeling a power plant's operating conditions

will operate alone and will achieve a steady-state failure probability that depends only on A. If it chooses to connect to the network, and thereby to influence and to be influenced by it, then the status of this site will depend not only on its own current status, but also on the current statuses of its neighbors. From the point of view of that power plant, this dependency represents a two-edged sword. The benefit is that when it is failed, the network has the potential to bring it back up. On the other hand, when that site is normal, a failed set of neighbors can influence it to fail. The main questions are: should the site connect to the network, or should it operate in isolation? If it should connect to the network, which sites should it connect to, and what are the optimal branch weights?

We can see that this "to link or not to link" question is effectively an optimization problem, which, in order to be answered quantitatively, requires one to define the precise objective function and the constraints. However, with the assumptions we are about to make, the answer to this problem will be simple enough that we can skip those formal definitions altogether. First, we assume that the the influence graph is homogeneous, i.e., $H = D' \otimes A$. Second, assume the local chain A and the network matrix D after the addition of the new site are ergodic, and consequently so is H. Being ergodic, both D and A have steady-state left eigenvectors ρ and α respectively, both of which are assumed to be normalized so their entries sum to 1. In other words, when the new site operates alone, it has a steady-state PMF of α.

With these assumptions, the answer is surprisingly simple: it doesn't matter how the new

site is connected to the network, as long as it is only concerned about its own steady-state probability, not the joint probability between its status and that of some other site. The reason is that $E(\mathbf{s}[k] \mid \mathbf{s}[0]) = \mathbf{s}'[0]H^k$, and by (5.34) and the fact that H is ergodic,

$$\begin{aligned} E(\mathbf{s}[k] \mid \mathbf{s}[0]) &= \mathbf{s}'[0]H^k \\ &\to \mathbf{s}'[0](\rho \otimes \mathbf{1}_m)(\mathbf{1}_n' \otimes \alpha') \\ &= [\alpha' \ \cdots \ \alpha'] \end{aligned} \tag{5.40}$$

where in (5.40), we used the fact that $\mathbf{s}'[0](\rho \otimes \mathbf{1}_m) = 1$. Eq. (5.40) shows that every site, both new and old, will have the same steady-state PMF, regardless of how D has changed, provided that D is still ergodic.

5.4.1 Experiments

To confirm the answer to the "to link or not to link" question as well as to gain more insight into the system, we performed the following simulations of two homogeneous influence models.

To highlight the effect of network connection, we generated two 30-site homogeneous influence models with the same local chain A but different network connections:

$$\begin{aligned} H_a &= I \otimes A \\ H_b &= D' \otimes A. \end{aligned}$$

In H_a, all the sites are disconnected. In H_b, D is a randomly generated network matrix such that every site has a self-loop and that the incoming edges (including the self-loop) at every site all have equal weights; for a site with k neighbors, the weight of each incoming edges is equal to $1/(k+1)$. This means in H_b every site receives equal influence from the neighbors and itself. The local chains are identical at every site and are given by

$$A = \begin{bmatrix} .99 & .01 \\ .08 & .92 \end{bmatrix}.$$

We will refer to the statuses as "normal" and "failed" respectively. The steady state PMF vector of A (the normalized left eigenvector of the eigenvalue at 1) is [.889 .111].

Typical snapshots of the simulation for 10 consecutive time steps are shown in Figures 5.3 and 5.4. In each figure, the left column corresponds to the results from H_a, and the right column to the results from H_b. In order to make these plots, we had to assign an (x, y) coordinate to each

site. Then, while creating the matrix D for H_b, the edges were created in such a way that each site tended to connect to a near site rather than a far one. The coordinate assignments, as well as the bias in favor of near sites were implemented merely to avoid clutters of tangled edges in the plots. They are inconsequential to the simulation dynamics. Note that in the plots for H_a, edges were drawn between sites to make it easy to compare with the plots for H_b, although in the actual simulations for H_a these edges were ignored —they do not imply any influence. Sites in normal status are denoted by thin circles, and sites in failed status are drawn in thick circles.

From observing the simulations in motion, a few qualitative differences can be seen. First, there are significantly more state changes in H_b, as can be seen from the frequent flickering of the thick circles. Second, failures in H_b that occur in clusters are significantly more persistent than those that occur in isolation. This is because failed sites sustain each other with like influences in a cluster. In contrast, an isolated failed site would be surrounded by normal statuses, and therefore has a tendency to quickly revert to normal status.

The simulation was carried out for a total duration of 2×10^5 time steps for both models. At the end of the simulation, a histogram of the number of failed sites at each time step was plotted for each system and compared in Figure 5.2[1] The mean number of failed sites per time

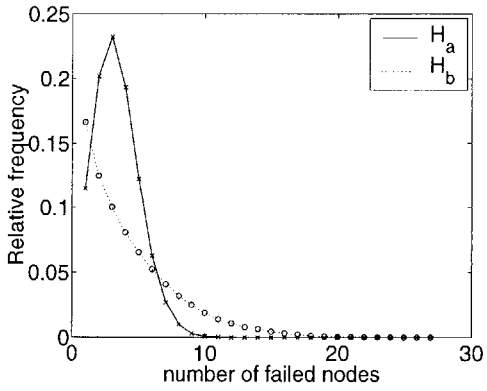


Figure 5.2: Histogram of number of failed sites at each given time steps.

step are 3.29 and 3.34 for H_a and H_b respectively. Both are sufficiently close to our expectation of $0.11 \times 30 = 3.33$ (probability of failure in steady state times the number of sites). This is also in agreement with our answer to the "to link or not to link" question — it doesn't matter what D is, because the individual steady-state PMF would remain unchanged.

[1]By the ergodic theorem ([3], Thm. 1.10.2), both histograms are guaranteed to converge. Moreover, even though they are obtained from a single run of the experiment, at convergence each histogram would be the correct steady-state distribution of the respective system.

The difference between the two histograms in Figure 5.2, however, is that the histogram for H_b has a visibly greater standard deviation than that for H_a (3.6 vs. 1.8); it has higher probabilities for small and large failures, but lower probabilities for the average-sized failures. This is consistent with the earlier description of qualitative differences.

The above observations are consistent with our intuition; through the connections among the sites, the correlation of the statuses are increased among neighbors. Since the correlation among neighbors are not reflected in the individual PMF's, the following question surfaces: what analysis of the influence model can one make to rigorously show the effect of network connection? Our answer is to extend the analysis of individual statuses to analysis of *joint-statuses*, a collection of statuses that simultaneously describes a group of sites. This is the motivation for the higher-order analysis developed in the following chapter.

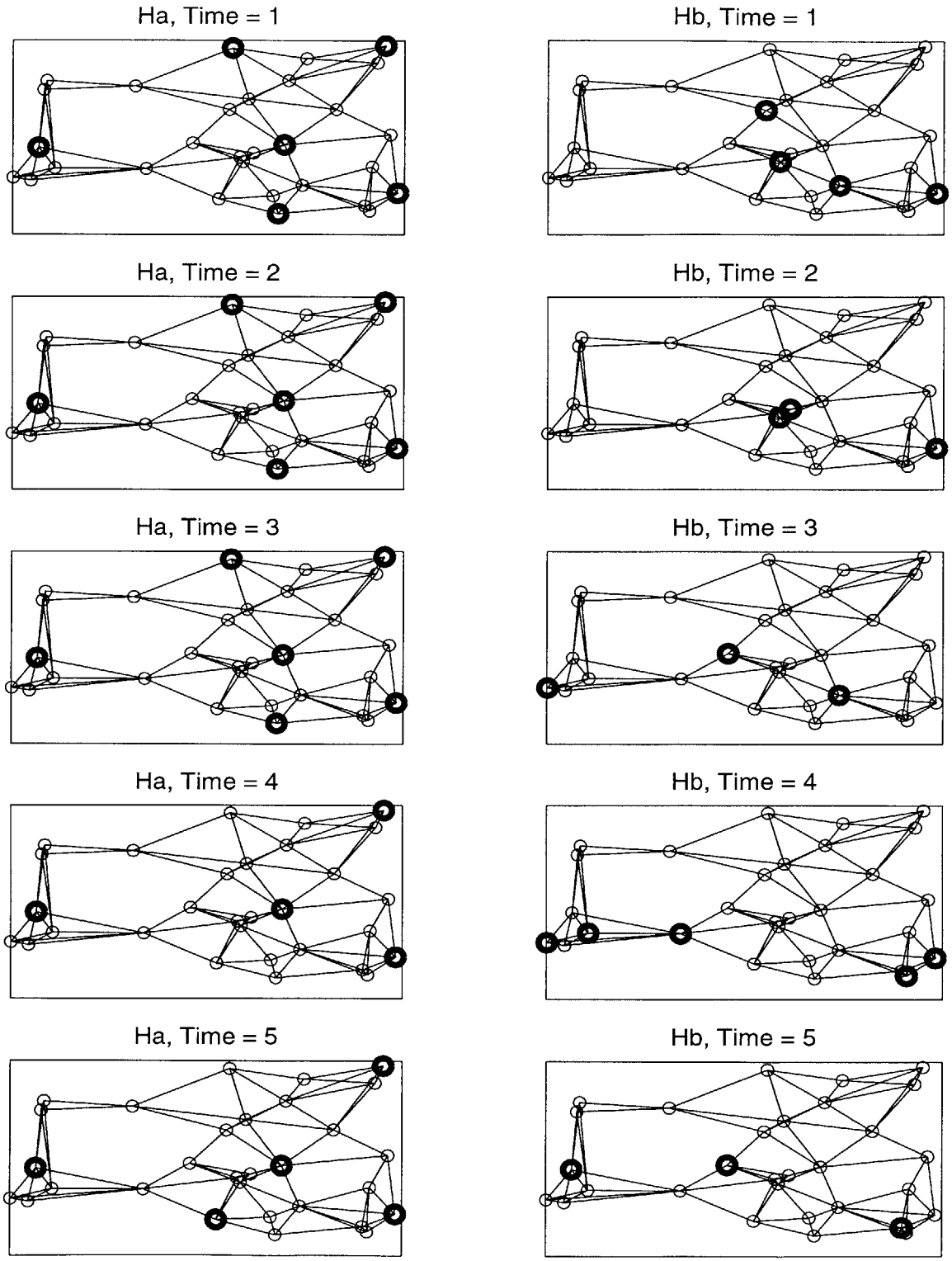


Figure 5.3: Sample run from step 1 to 5. Thick circles denote failed sites while thin circles denote normal sites. Notice how isolated failures in H_b revert to normal much more quickly than in H_a.

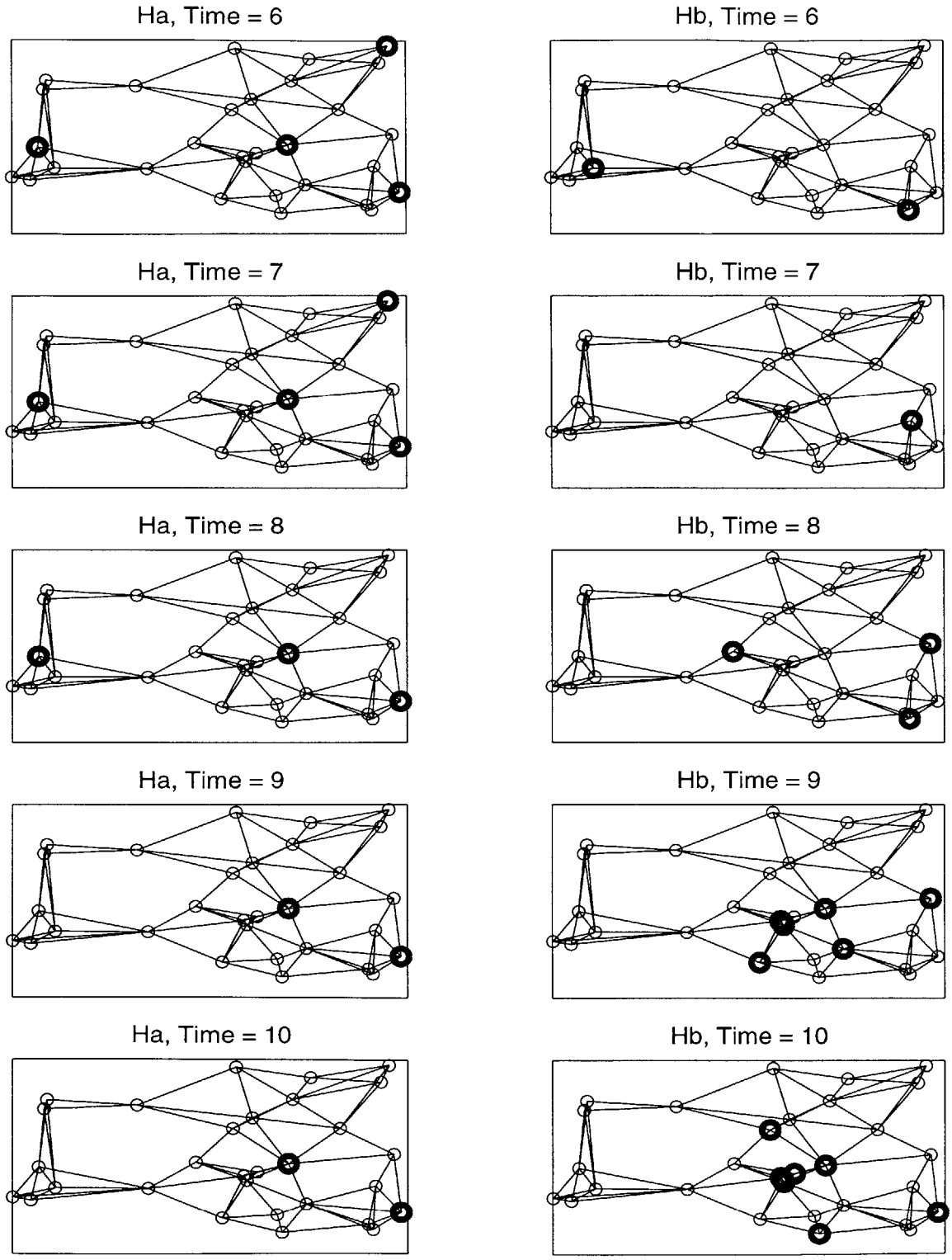


Figure 5.4: Continued sample run from step 6 to 10. Notice how new failures in H_b tend to be caused by neighbors that have failed in the previous time steps.

Chapter 6

Higher-Order Analysis

6.1 Introduction

Apart from being able to predict the steady-state probability of the individual sites, one is often interested in the *joint status* of two or more sites. For example, in the case of power systems, a planner might inquire about the likelihood that a certain group of generators would be down simultaneously. Since our analysis of the influence model so far only applies to individual sites, an extension of the analysis is needed to determine such collective behavior of sites.

In this chapter, we present *higher-order* analysis of the influence model. An rth-order joint status is a collection of statuses of a group of r sites. The higher the order, the larger the size of the problem becomes. This graceful growth is desirable to us from both theoretical and practical points of view. Theoretically, the higher-order model serves as the link between the dynamics of the first-order H and that of the master Markov chain G. Practically, higher-order analysis makes computations on influence models scalable. It allows us the flexibility to obtain progressively more elaborate statistics at the expense of progressively greater computation, instead of jumping directly from the first-order to the master chain G.

Our approach in this chapter is as follows. First, we introduce the joint-status vector, its meaning, and the issues regarding redundant joint statuses. Then, we will derive the higher-order influence matrices, and expose the relations between them and the master Markov chain G. Finally, we expose the "telescoping" relations of the relevant eigenvalues, and explain how the relevant eigenvalues of higher-order influence matrices correct higher-order statistics of sites.

6.2 Joint-Statuses

Ultimately, the purpose of higher-order influence matrices — or for that matter, the purpose of this entire chapter — is to be able to answer a question of the following sort: in a given influence model, what is the probability that sites a, b, c are simultaneously in statuses x, y, z respectively? In what follows, we will refine this question by setting it up mathematically so that we can answer it. As the first step, in Sec. 6.2.1, we will show what joint-statuses are, and how they are different from the status vectors used thus far. Then in Sec. 6.2.2 we restate the main problem of this chapter, and finally from Sec. 6.2.3 onward derive the answer to this problem in detail.

6.2.1 Definition

As before, let $H = D' \otimes \{A_{ij}\}$ be an influence matrix and let $m_1, \ldots, m_n$ be the orders of the local chains, i.e. the sizes of $A_{11}, \ldots, A_{nn}$ respectively. In this chapter, we will refer to H as a *first-order influence matrix* instead of just an influence matrix, to distinguish it from higher-order influence matrices soon to be introduced. Recall that a *status vector* of site i at time k is denoted by $\mathbf{s}_i[k]$, a length-m_i binary vector with a single entry of 1:

$$\mathbf{s}_i[k] = [0 \cdots 1 \cdots 0]'.$$

We say that site i is in status x at time k if the position at which the entry '1' appears in $\mathbf{s}_i[k]$ is x.

Now let us extend this notation to multiple sites. For a given ordered set of r sites $\mathbf{i} = (i_1, \ldots, i_r)$, we say that *the sites in* $\mathbf{i}$ *are in joint-status* $\ell = (\ell_1, \ldots, \ell_r)$ *at time* k if, for each $1 \leq t \leq r$, the site i_t is in status ℓ_t. Note that the sites in $\mathbf{i}$ need not be distinct. A convenient way to represent the joint-status for $\mathbf{i}$ is by the *joint-status vector*, which is defined as the following Kronecker product:

$$\mathbf{s}_{\mathbf{i}}[k] \stackrel{\triangle}{=} \mathbf{s}_{i_1}[k] \otimes \cdots \otimes \mathbf{s}_{i_r}[k].$$

Note that the subscript of $\mathbf{s}_{\mathbf{i}}[k]$ is a vector, not a scalar. The vector $\mathbf{s}_{\mathbf{i}}[k]$ is a binary vector of length $\mu_{\mathbf{i}} \stackrel{\triangle}{=} m_{i_1} \cdots m_{i_r}$.

Example 1: Suppose

$$\mathbf{s}_1[k] = [1 \;\; 0]' \qquad \mathbf{s}_2[k] = [0 \;\; 0 \;\; 1]' \qquad \mathbf{s}_3[k] = [0 \;\; 1]'. \tag{6.1}$$

Then a few examples of the joint-status vectors are

$$\mathbf{s}_{(1,2)}[k] = \mathbf{s}_1[k] \otimes \mathbf{s}_2[k] = [0 \;\; 0 \;\; 1 \;\; 0 \;\; 0 \;\; 0]' \tag{6.2}$$

$$\mathbf{s}_{(2,1)}[k] = \mathbf{s}_2[k] \otimes \mathbf{s}_1[k] = [0 \;\; 0 \;\; 0 \;\; 0 \;\; 1 \;\; 0]' \tag{6.3}$$

$$\mathbf{s}_{(3,3)}[k] = \mathbf{s}_3[k] \otimes \mathbf{s}_3[k] = [0 \;\; 0 \;\; 0 \;\; 1]' \tag{6.4}$$

□

Like the individual-status vector $\mathbf{s}_i[k]$, there is only a single nonzero entry of 1 in the joint-status vector $\mathbf{s}_{\mathbf{i}}[k]$. The position of this entry '1' is a function of both the joint-status ℓ and

$\ell = (\ell_1, \ldots, \ell_r)$	$f(\ell, \mathbf{m_i})$
$(1, \ldots, 1, 1)$	1
$(1, \ldots, 1, 2)$	2
$\vdots$	$\vdots$
$(1, \ldots, 1, m_{i_r})$	m_{i_r}
$(1, \ldots, 2, 1)$	$m_{i_r} + 1$
$\vdots$	$\vdots$
$(m_{i_1}, \ldots, m_{i_{r-1}}, m_{i_r} - 1)$	$\mu_{\mathbf{i}} - 1$
$(m_{i_1}, \ldots, m_{i_{r-1}}, m_{i_r})$	$\mu_{\mathbf{i}}$

Table 6.1: Table of ℓ and $f(\ell, \mathbf{m_i})$.

$\mathbf{m_i} = (m_{i_1}, \ldots, m_{i_r})$:

$$f(\ell, \mathbf{m_i}) \stackrel{\triangle}{=} (\ell_1 - 1)(m_{i_2} \cdots m_{i_r}) + (\ell_2 - 1)(m_{i_3} \cdots m_{i_r}) + \cdots + (\ell_{r-1} - 1)m_{i_r} + \ell_r. \tag{6.5}$$

Despite the cumbersome appearance of (6.5), this function can actually be easily understood when viewed in a table as shown in Table 6.1. Reading down the table, one notices that in the right column the $f(\cdot, \cdot)$ simply sequences from 1 to $\mu_{\mathbf{i}}$, while ℓ sequences "last-digit first" from $(1, \ldots, 1)$ to $(m_{i_1}, \ldots, m_{i_r})$. Since the left column list all possibilities of ℓ, and since $\mu_{\mathbf{i}}$ is the length of vector $\mathbf{s_i}[k]$, this table shows that each position in $\mathbf{s_i}[k]$ uniquely corresponds to a joint-status ℓ. Therefore, we will use the multiple-integer ℓ both as a joint-status and as an addressing scheme. We say that *the sites in* $\mathbf{i}$ *are in joint-status* ℓ *at time* k if the entry '1' of $\mathbf{s_i}[k]$ is at position $f(\ell, \mathbf{m_i})$; and when we refer to the *entry at address* ℓ *of* $\mathbf{s_i}[k]$, we are referring to the entry at position $f(\ell, \mathbf{m_i})$ of $\mathbf{s_i}[k]$.

6.2.2 A Motivating Proposition

We now present a basic proposition that has surfaced repeatedly in this thesis in various contexts. For a group of sites $\mathbf{i}$ and one of its addresses ℓ, we refer to $(\mathbf{s_i}[k])_\ell$ as the entry in position $f(\ell, \mathbf{m}_i)$ of vector $\mathbf{s_i}[k]$.

PROPOSITION 6.1
For any group of sites $\mathbf{i}$,

$$Prob(\text{ the sites in } \mathbf{i} \text{ are in joint status } \ell \text{ at time } k\ ) = E\left[\ (\mathbf{s_i}[k])_\ell\ \right] \tag{6.6}$$

Proof. Because each $(\mathbf{s_i}[k])_\ell$ is a binary random variable, its expected value is equal to the probability of its being a 1. □

Proposition 6.1 is the key. It says that if we know $E(\mathbf{s_i}[k])$ for all $\mathbf{i}$ and all k, then we know the probability of each joint-status for all time. Our main question for the rest of this chapter is then: how can we determine $E(\mathbf{s_i}[k])$ for a given influence model without resorting to Monte Carlo simulation? Are there evolution equations that will provide us with $E(\mathbf{s_i}[k])$ for all k? The answers to both of these questions are affirmative, and we devote the rest of this chapter to answering them.

6.2.3 Primary and Secondary Groupings

A *grouping* is defined as an ordered set of r sites $\mathbf{i} = (i_1, \ldots, i_r)$. Throughout this section we assume that $r \leq n$, where n is the number of the sites in the influence model. We also assume that r is fixed so that a grouping always contains the same number of elements. A grouping $\mathbf{i} = (i_1, \ldots, i_r)$ is called *primary* if it contains only distinct sites and if the indices of these sites are sorted in a strictly increasing order, i.e., $i_1 < \cdots < i_r$. A grouping that is not primary is called *secondary*. If $\mathbf{i}$ is primary, then its joint-status vector $\mathbf{s_i}[k]$ is called a *primary joint-status*. Similarly, if $\mathbf{i}$ is secondary, then $\mathbf{s_i}[k]$ is called a *secondary joint-status*.

Given that there are n sites, there are n^r possible groupings. Among those, only $\binom{n}{r}$ are primary, because there are that many ways to choose r distinct sites from n available ones. The remaining $n^r - \binom{n}{r}$ groupings are secondary.

Example 2: Suppose there are 3 sites in an influence model and we fix the order r to be 2. Then the listing of all the groupings and their classifications are given in Table 6.2. As expected, there are 3^2 groupings, of which only $\binom{3}{2} = 3$ are primary. □

Grouping	Primary	Secondary
(1, 1)		•
(1, 2)	•	
(1, 3)	•	
(2, 1)		•
(2, 2)		•
(2, 3)	•	
(3, 1)		•
(3, 2)		•
(3, 3)		•

Table 6.2: List of all groupings and their classifications.

The primary groupings are given this name for two important reasons. First, by knowing

the joint-statuses of every primary grouping, we know the joint-statuses of every possible grouping; every secondary joint-status can be derived from some primary grouping, as we will show in Sec. 6.2.4. Second, given the state vector $\mathbf{s}[k]$, the statuses in the following time step of the sites in a primary grouping are independent, thus simplifying the analysis on their joint-PMF's. We will discuss this second property in more detail in Sec. 6.3.4.

6.2.4 Marginalization Matrices

As just claimed, every secondary joint-status can be derived from some primary joint-status. In this section, we identify the primary joint-statuses from which a given secondary joint-status can be derived. The two joint-statuses will be related through a matrix called a marginalization matrix. We will explain how the marginalization matrices can be constructed through several examples.

For two groupings $\mathbf{i} = (i_1, \ldots, i_r)$ and $\mathbf{j} = (j_1, \ldots, j_r)$, we write $\mathbf{i} \subset \mathbf{j}$ if the unordered version of $\mathbf{i}$ is a subset of the unordered version of $\mathbf{j}$, i.e., $\{i_1, \ldots, i_r\} \subset \{j_1, \ldots, j_r\}$. For example, $(1,1) \subset (1,2)$, and $(3,2,2) \subset (2,3,4)$.

THEOREM 6.2

Let $\mathbf{i}$ *be a grouping. There exists a primary grouping* $\mathbf{j}$ *such that* $\mathbf{i} \subset \mathbf{j}$. *Furthermore, there exists a matrix* $M[\mathbf{j}, \mathbf{i}]$ *such that*

$$\mathbf{s}'_{\mathbf{i}}[k] = \mathbf{s}'_{\mathbf{j}}[k] M[\mathbf{j}, \mathbf{i}]$$

Although conceptually Theorem 6.2 is not difficult to understand, the constructive proof that we are going to provide requires a certain amount of bookkeeping of details. Thus, rather than proving it formally, we shall only explain why the claim is true in an informal manner, and along the way give several examples to familiarize the reader with the procedures to convert a primary joint-status to a secondary joint-status.

The first claim of Theorem 6.2, regarding the existence of such a primary grouping, is explained first. In the cases (a) and (b) below, we explain this claim while simultaneously constructing such a primary grouping.

(a) If a given grouping $\mathbf{i}$ contains only distinct sites, we can sort the indices of the sites in it to obtain a sorted grouping $\mathbf{j}$. Since $\mathbf{j}$ contains only distinct sites which are already sorted, $\mathbf{j}$ is a primary grouping with the property that $\mathbf{i} \subset \mathbf{j}$ as claimed.

(b) On the other hand, if $\mathbf{i}$ has some repeated sites, then the number of distinct sites in $\mathbf{i}$ must

be strictly less than r. Let t be the number of distinct sites in $\mathbf{i}$. Then we can find $(r-t)$ additional distinct sites to create a new grouping $\mathbf{k}$, which is made up of exactly r distinct sites. That is, $\mathbf{k}$ is a grouping that combines the t distinct sites in $\mathbf{i}$ and the additional $(r-t)$ distinct sites that we add. The $(r-t)$ new sites can always be found, because we still have available to us $(n-t)$ distinct sites that are not in $\mathbf{i}$, and $r-t \leq n-t$. To make the procedure well-defined, we assume that the sites with the smallest indices will be chosen among those available ones. Once we have $\mathbf{k}$, we can sort it as in case (a) to obtain a primary grouping $\mathbf{j}$, which again satisfies $\mathbf{i} \subset \mathbf{j}$.

As a result of the cases (a) and (b) above, whether or not the grouping $\mathbf{i}$ contains repeated sites, we are left with a unique primary grouping $\mathbf{j}$ such that $\mathbf{i} \subset \mathbf{j}$, which we call the *designated primary grouping of* $\mathbf{i}$ and denote by $\omega(\mathbf{i}) \stackrel{\triangle}{=} \mathbf{j}$.

A few examples of groupings of order 3 and their designated primary counterparts are given in Table 6.3. One can verify that in each case, $\omega(\mathbf{i})$ is primary (i.e., sorted and having distinct sites), and that $\mathbf{i} \subset \omega(\mathbf{i})$. This completes our explanation for the first part of the claim in Theorem 6.2.

$\mathbf{i}$	$\omega(\mathbf{i})$
$(3,1,2)$	$(1,2,3)$
$(8,8,8)$	$(1,2,8)$
$(2,1,1)$	$(1,2,3)$

Table 6.3: Examples of some groupings and their designated primary counterparts.

The second claim of Theorem 6.2 involves the construction of the matrix $M[\mathbf{j},\mathbf{i}]$ for some primary $\mathbf{j}$ such that $\mathbf{i} \subset \mathbf{j}$. This part of the claim is a bit more complicated than the first, so before we present the construction procedure for a general grouping $\mathbf{i}$, we provide a few examples as motivation first. For the rest of the chapter, for any grouping $\mathbf{i}$, define $M_{\mathbf{i}} \stackrel{\triangle}{=} M[\omega(\mathbf{i}),\mathbf{i}]$ as the *marginalization matrix of* $\mathbf{i}$. When $\mathbf{i}$ is primary, $\omega(\mathbf{i}) = \mathbf{i}$. In that case, $M_{\mathbf{i}} = I$.

Example 3: In this example, we will simply present a few marginalization matrices and verify briefly that they have the desired properties, without yet describing how they are constructed. The purpose is to accustom the reader with the general idea of what these matrices look like.

Let us revisit the 3-site example used in Examples 1 and 2. Given that the orders of the local chains are 2, 3 and 2 respectively (from Example 1), the two examples of marginalization

matrices that we focus on now are as shown:

$$M_{(1,1)} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad M_{(2,1)} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \tag{6.7}$$

Focusing on $M_{(1,1)}$ first, since $\omega[(1,1)] = (1,2)$, the first marginalization matrix $M_{(1,1)}$ is a 6×4 matrix, which is equal to the order of $\mathbf{s}_{(1,2)}[k] \times$ the order of $\mathbf{s}_{(1,1)}[k]$. For the statuses defined in (6.1), $\mathbf{s}_{(1,1)}[k] = [1\ 0\ 0\ 0]$ and $\mathbf{s}_{(1,2)}[k]$ is as given in (6.2). We can see that $\mathbf{s}'_{(1,1)}[k] = \mathbf{s}'_{(1,2)}[k]M_{(1,1)}$ as claimed. Notice that columns 2 and 3 of $M_{(1,1)}$ are zero. This is expected because the second and third entries of $\mathbf{s}_{(1,1)}[k]$ are always zero. The reason is that when site 1 is in status 1, $\mathbf{s}_{(1,1)}[k]$ is in joint-status (1,1), which corresponds to having an entry '1' in the first position; when site 1 is in status 2, then $\mathbf{s}_{(1,1)}[k]$ would be in joint-status (2,2), which corresponds having a '1' in the fourth position. The second and third position will never have an entry '1' in them because they represent invalid joint-statuses for $\mathbf{s}_{(1,1)}[k]$.

Now, let us turn our attention to $M_{(2,1)}$. Since $\omega[(2,1)] = (1,2)$, $M_{(2,1)}$ is a 6×6 matrix. Again, we can verify from (6.2) and (6.3) that $\mathbf{s}'_{(2,1)}[k] = \mathbf{s}'_{(1,2)}[k]M_{(2,1)}$. Notice that $M_{(1,1)}$ and $M_{(2,1)}$ both have a single nonzero entry of 1 in each row. As we shall see, this is true for all marginalization matrices. In particular, $M_{(2,1)}$ also has a single '1' in every column, thus qualifying it a permutation matrix (see Sec. 2.1.2). We shall see too that this is always the case when $\mathbf{i}$ and $\omega(\mathbf{i})$ are permutations of each other, or equivalently, when $\mathbf{i}$ has distinct sites. □

We now describe the procedure to construct the matrix $M[\mathbf{j},\mathbf{i}]$ for the general case. Let $\mathbf{i} = (i_1, \ldots, i_r)$ be a grouping and let $\mathbf{j} = (j_1, \ldots, j_r)$ be a primary grouping such that $\mathbf{i} \subset \mathbf{j}$. Since $\mathbf{i} \subset \mathbf{j}$, for all $1 \leq t \leq r$, the site i_t of $\mathbf{i}$ must also be a member of the ordered set $\mathbf{j}$. Since $\mathbf{j}$ only contains distinct sites, the site i_t must uniquely correspond to some site in $\mathbf{j}$. Thus, for each $1 \leq t \leq r$, we define $\theta(t)$ as the re-indexing function such that $i_t = j_{\theta(t)}$. In other words, $\theta(\cdot)$ is defined such that

$$(i_1, \ldots, i_r) = (j_{\theta(1)}, \ldots, j_{\theta(r)}) \triangleq \Theta(\mathbf{j})$$

We call such function $\theta(\cdot)$ and $\Theta(\cdot)$ the *scalar and vector re-indexing functions for* $\mathbf{i}$ *and* $\mathbf{j}$ respectively.

Example 4: From Table 6.3, $(3,1,2) \subset (1,2,3)$. So we let $\mathbf{i} = (3,1,2)$ and $\mathbf{j} = (1,2,3)$. Then we

set $\theta(1) = 3$, $\theta(2) = 1$ and $\theta(3) = 2$, so that

$$\Theta(\mathbf{j}) = (j_3, j_1, j_2) = (3, 1, 2) = (i_1, i_2, i_3)$$

as desired. Notice that here $\Theta(\cdot)$ is essentially a permutation function, which is a one-to-one function, since $\mathbf{i}$ and $\mathbf{j}$ are permutations of each other.

The function $\theta(\cdot)$ needs not be a one-to-one function, as this next example shows. Again, from Table 6.3, let $\mathbf{i} = (8, 8, 8)$ and $\mathbf{j} = (1, 2, 8)$. In this case, we must set $\theta(1) = \theta(2) = \theta(3) = 3$, so that

$$\Theta(\mathbf{j}) = (j_3, j_3, j_3) = (8, 8, 8) = (i_1, i_2, i_3).$$

□

Recall that for each grouping $\mathbf{i}$, the joint-status vector $\mathbf{s}_\mathbf{i}[k]$ simultaneously indicates the statuses of sites $i_1, \ldots, i_r$ at time k. In particular, there is a '1' at address ℓ of $\mathbf{s}_\mathbf{i}[k]$ if and only if the sites in $\mathbf{i}$ are in joint-status ℓ at that time.

Since $\mathbf{i} \subset \mathbf{j}$, if we know the joint-status of the sites in $\mathbf{j}$ at time k, we must know the joint-status of the sites in $\mathbf{i}$ at that same time. Equivalently, if we know the position of the '1' entry in $\mathbf{s}_\mathbf{j}[k]$, then we must know the position of the '1' entry in $\mathbf{s}_\mathbf{i}[k]$. More specifically, if the '1' entry is at address $\rho \stackrel{\triangle}{=} (\rho_1, \ldots, \rho_r)$ of $\mathbf{s}_\mathbf{j}[k]$, then $\mathbf{s}_\mathbf{i}[k]$ would have a '1' in position $\Theta(\rho) \stackrel{\triangle}{=} (\rho_{\theta(1)}, \ldots, \rho_{\theta(r)})$, where $\Theta(\cdot)$ is the vector re-indexing function of $\mathbf{i}$ and $\mathbf{j}$. The reason for position $\Theta(\rho)$ is simple: since the site indices $j_{\theta(t)}$ and i_t are equal, when $j_{\theta(t)}$ is in status $\rho_{\theta(t)}$, so is i_t.

The marginalization matrix $M[\mathbf{j}, \mathbf{i}]$ is a matrix that satisfies $\mathbf{s}'_\mathbf{i}[k] = \mathbf{s}'_\mathbf{j}[k] M[\mathbf{j}, \mathbf{i}]$. This shows that $M[\mathbf{j}, \mathbf{i}]$ has as many rows as the order of $\mathbf{s}_\mathbf{j}[k]$, and as many columns as that of $\mathbf{s}_\mathbf{i}[k]$. Therefore, we can refer to each column of $M[\mathbf{j}, \mathbf{i}]$ by a unique address $\ell = (\ell_1, \ldots, \ell_r)$ in which $1 \leq \ell_t \leq m_{i_t}$ for all t. That is, we use address ℓ to refer to a column of $M[\mathbf{j}, \mathbf{i}]$ in the same manner we use it to refer to a position in $\mathbf{s}_\mathbf{i}[k]$. Similarly, each row of $M[\mathbf{j}, \mathbf{i}]$ can also be referred to by an address ρ, the same way each entry of $\mathbf{s}_\mathbf{j}[k]$ is referred to. Combining these observations, we come to an important fact about $M[\mathbf{j}, \mathbf{i}]$: for each pair of row and column ρ and ℓ,

$$\Big(M[\mathbf{j}, \mathbf{i}]\Big)_{\rho, \ell} = \begin{cases} 1 & \text{if } \ell = \Theta(\rho) \\ 0 & \text{otherwise} \end{cases} \tag{6.8}$$

In words, eq. (6.8) means entry (ρ, ℓ) of $M[\mathbf{j}, \mathbf{i}]$ is a 1 if and only if the fact that $\mathbf{s}_\mathbf{j}[k]$ is in joint-status ρ implies that $\mathbf{s}_\mathbf{i}[k]$ is in joint-status ℓ. This definition implies that a nonzero entry in

$M[\mathbf{j},\mathbf{i}]$ always occurs in a position of the form $(\rho, \Theta(\rho))$, where ρ is the address of its row. As a consequence, each row of $M[\mathbf{j},\mathbf{i}]$ must have exactly a single nonzero entry of '1', as seen in $M_{(1,1)}$ and $M_{(2,1)}$ in Example 3.

To familiarize ourselves with the matrix $M[\mathbf{j},\mathbf{i}]$ even further, we make two more characterizations of $M[\mathbf{j},\mathbf{i}]$ in interesting special cases.

Observation 1: *The matrix* $M[\mathbf{j},\mathbf{i}]$ *is a permutation matrix if and only if* $\mathbf{i}$ *and* $\mathbf{j}$ *are permutations of each other.*

Note that this statement is not so obvious because $M[\mathbf{j},\mathbf{i}]$ permutes the entries of $\mathbf{s_j}[k]$, not those of $\mathbf{j}$. We have seen an example of this case from $M_{(2,1)}$ in Example 3. The explanation for this observation is as follows. The groupings $\mathbf{i}$ and $\mathbf{j}$ are permutations of each other if and only if for each $1 \leq t \leq r$, the scalar re-indexing function $\theta(t)$ maps t to a unique integer in the set $\{1,\ldots,r\}$, which means that $\theta(\cdot)$ is a one-to-one function.

When the scalar re-indexing function $\theta(\cdot)$ is a one-to-one function on the set of integers $\{1,\ldots,r\}$, the vector re-indexing function $\Theta(\cdot)$ is a one-to-one function on the set of addresses $\Xi \stackrel{\Delta}{=} \{\,(\rho_1,\ldots,\rho_r) \mid 1 \leq \rho_t \leq m_{j_t}\,\}$ of $\mathbf{s_j}[k]$ as well. The reason is that its inverse exists; given an address ℓ of $\mathbf{s_i}[k]$,

$$\rho = \Theta^{-1}(\ell) \stackrel{\Delta}{=} (\ell_{\theta^{-1}(1)},\ldots,\ell_{\theta^{-1}(r)}).$$

Since a one-to-one function on the set Ξ implies that $\Theta(\cdot)$ is a permutation function, we have completed our explanation for Observation 1.

Observation 2: *If* $\mathbf{i}$ *has any repeated sites, and the order of every local chain satisfies* $m_{i_t} \geq 2$, *then* $M[\mathbf{j},\mathbf{i})$ *must have some zero columns.*

We have seen this case in $M_{(1,1)}$ in (6.7). The reason is that there are certain positions of $\mathbf{s_i}[k]$ that can never have a '1' in them. These are the entries corresponding to joint-statuses in which a single site have different statuses. For example, if $i_1 = i_2$, then an entry of address ℓ in $\mathbf{s_i}[k]$ is guaranteed to be zero if $\ell_1 \neq \ell_2$; the site i_1 must have the same status as site i_2, since they are the same site.

Example 5: Let us now revisit the marginalization matrices in Example 3 and see how they were constructed.

In the first one, $\mathbf{i} = (1,1)$ and $\mathbf{j} = (1,2)$. So the scalar re-indexing function used was $\theta(1) = \theta(2) = 1$. For every row address ρ of $M_{(1,1)}$, the pairs of $(\rho, \Theta(\rho))$ are listed in Table 6.4a.

There we also show which position of $\mathbf{s_i}[k]$ the address $\Theta(\rho)$ really corresponds to by evaluating the function $f(\Theta(\rho),(2,2))$ defined in (6.5), where $(2,2) = \mathbf{m_i}$, the orders of the sites in $\mathbf{i}$. Using the first and the last columns of this table, we have determined every nonzero position of $M_{(1,1)}$.

In the second matrix $M_{(2,1)}$, $\mathbf{i} = (2,1)$ and $\mathbf{j} = (1,2)$. So $\theta(1) = 2$ and $\theta(2) = 1$. The rest of the procedures are the same. □

Rows		Columns	
Index	ρ	$\Theta(\rho)$	$f(\Theta(\rho),(2,2))$
1	(1,1)	(1,1)	1
2	(1,2)	(1,1)	1
3	(1,3)	(1,1)	1
4	(2,1)	(2,2)	4
5	(2,2)	(2,2)	4
6	(2,3)	(2,2)	4

(a)

Rows		Columns	
Index	ρ	$\Theta(\rho)$	$f(\Theta(\rho),(3,2))$
1	(1,1)	(1,1)	1
2	(1,2)	(2,1)	3
3	(1,3)	(3,1)	5
4	(2,1)	(1,2)	2
5	(2,2)	(2,2)	4
6	(2,3)	(3,2)	6

(b)

Table 6.4: Listing of $(\rho, \Theta(\rho))$ for **(a)** $M_{(1,1)}$ **(b)** $M_{(2,1)}$.

6.3 Higher-Order Influence Matrices

Recall that in Sec. 6.2.2, we stated a proposition which motivates us to find the evolution equations for $E(\mathbf{s_i}[k])$. Knowing the expected value of this vector is equivalent to knowing the probability of each joint-status of the sites in $\mathbf{i}$. We subsequently classified the joint-statuses into primary and secondary joint-statuses, and explained in detail on how the secondary joint-statuses can be derived from the primary ones. In this section, we return to the message from Sec. 6.2.2 and explain how the analysis so far can be applied.

By knowing the expected joint-statuses for all primary groupings, we know the expected joint-statuses for every grouping, primary or secondary. This is due to the fact that for any grouping $\mathbf{i}$,

$$\mathbf{s'_i}[k] = \mathbf{s'}_{\omega(\mathbf{i})} M_{\mathbf{i}}.$$

Hence, by the linearity of expectation operators,

$$E(\mathbf{s'_i}[k]) = E(\mathbf{s'}_{\omega(\mathbf{i})}[k]) M_{\mathbf{i}}. \tag{6.9}$$

Since $\omega(\mathbf{i})$ is primary, if we have the expected joint-status of every primary grouping, we must have the expected joint-status of $\omega(\mathbf{i})$, and consequently, of $\mathbf{i}$. Our problem can then be reduced to finding the evolution for expected joint-statuses of every *primary grouping*. This will be our goal for the rest of Sec. 6.3.

6.3.1 Lexicographical Ordering

Before deriving the evolution equations for primary groupings, there is a straightforward but necessary concept that needs to be introduced, namely the concept of lexicographical ordering. As mentioned earlier, there are n^r possible groupings. These groupings are listed in Table 6.5 below. The *lexicographical ordering* is defined as the order in which groupings appear in this table. An

Grouping
$(1,\ldots,1,1)$
$(1,\ldots,1,2)$
$\vdots$
$(1,\ldots,1,n)$
$(1,\ldots,n,1)$
$(1,\ldots,n,2)$
$\vdots$
$(n,\ldots,n,n-1)$
$(n,\ldots,n,n)$

Table 6.5: The lexicographical ordering.

ordered collection of groupings $\mathbf{i}_1, \mathbf{i}_2, \ldots, \mathbf{i}_k$ is said to be in *lexicographical order* if the order in which these groupings appear is the same as they do in Table 6.5.

6.3.2 Expansion and Truncation Matrices

Let $\mathbf{i}_1, \ldots, \mathbf{i}_\nu$ be the list of all $\nu \triangleq n^r$ groupings arranged in lexicographical order. Define the *rth-order state vector at time k* as

$$\mathbf{s}_{(r)}[k] \triangleq \left[\mathbf{s}'_{\mathbf{i}_1}[k] \quad \ldots \quad \mathbf{s}'_{\mathbf{i}_\nu}[k]\right]'. \tag{6.10}$$

It is not difficult to show that the length of vector $\mathbf{s}_{(r)}[k]$ is $(\sum_{i=1}^{n} m_i)^r$,

Among these ν groupings, a subset of order $\eta \triangleq \binom{n}{r}$ of them are the primary ones. Thus, we define $\xi(\cdot)$ as the indexing function such that $\mathbf{i}_{\xi(j)}$ is the jth primary grouping, when they arranged in lexicographical order. The primary grouping listed in lexicographical order are then $\mathbf{i}_{\xi(1)}, \mathbf{i}_{\xi(2)}, \ldots, \mathbf{i}_{\xi(\eta)}$. Define the *rth-order primary vector at time k* as

$$\mathbf{y}_{(r)}[k] \triangleq \left[\mathbf{s}'_{\mathbf{i}_{\xi(1)}}[k] \quad \ldots \quad \mathbf{s}'_{\mathbf{i}_{\xi(\eta)}}[k]\right]'. \tag{6.11}$$

Since $\mathbf{s}_{(r)}[k]$ includes all joint-statuses, we can obtain $\mathbf{y}_{(r)}[k]$ from $\mathbf{s}_{(r)}[k]$ by deleting entries that belong to secondary joint-statuses from $\mathbf{s}_{(r)}[k]$. That is, we can write

$$\mathbf{y}'_{(r)}[k] = \mathbf{s}'_{(r)}[k]T_r \tag{6.12}$$

where

$$T_r = \begin{bmatrix} R_{11} & \cdots & R_{1\eta} \\ \vdots & & \vdots \\ R_{\nu 1} & \cdots & R_{\nu\eta} \end{bmatrix} \tag{6.13}$$

and each matrix R_{pq} is matrix of dimension $\mu_{\mathbf{i}_p} \times \mu_{\mathbf{i}_{\xi(q)}}$ defined as

$$R_{pq} = \begin{cases} I & \text{if } p = \xi(q) \\ 0 & \text{otherwise} \end{cases} \tag{6.14}$$

That is, T_r can be thought as a tall matrix obtained by starting with a big identity matrix I, then removing some of the columns. It generally has the following form:

$$T_r = \begin{bmatrix} I & & & \\ & I & & \\ & & & \\ & & I & \\ & & & \ddots \end{bmatrix}. \tag{6.15}$$

We call T_r *the rth-order truncation matrix*, or when there is no ambiguity, simply *the truncation matrix.*

On the other hand, from Sec. 6.2.4, we have shown that every joint-status is derivable from a primary joint-status via the marginalization matrix. Thus, we define the *rth-order expansion*

matrix as the matrix M_r such that

$$\mathbf{s}'_{(r)}[k] = \mathbf{y}'_{(r)}[k]M_r. \tag{6.16}$$

To visualize the structure of M_t, one can think of it as being defined by

$$M_r = \begin{bmatrix} L_1 & \cdots & L_\nu \end{bmatrix} \tag{6.17}$$

where L_t is the matrix that contains the marginalization matrix $M_{\mathbf{i}_t}$ for $\mathbf{i}_t$ with appropriate zero-padding:

$$L_t = \begin{bmatrix} 0 \\ M_{\mathbf{i}_t} \\ 0 \end{bmatrix} \tag{6.18}$$

The zero-padding is so that in the multiplication in (6.16), the matrix $M_{\mathbf{i}_t}$ will be multiplied by $\mathbf{s}_{\omega(\mathbf{i}_t)}$, the designated primary joint-status for $\mathbf{i}_t$.

The more formal definition of M_r (which is not helpful for visualization, but essential for a later proof) is as follows:

$$M_r = \begin{bmatrix} N_{11} & \cdots & N_{1\nu} \\ \vdots & & \vdots \\ N_{\eta 1} & \cdots & N_{\eta\nu} \end{bmatrix} \tag{6.19}$$

where N_{pq} is a $\mu_{\mathbf{i}_{\xi(p)}} \times \mu_{\mathbf{i}_q}$ matrix defined by

$$N_{pq} = \begin{cases} M_{\mathbf{i}_q} & \text{if } \omega(\mathbf{i}_q) = \mathbf{i}_{\xi(p)} \\ 0 & \text{otherwise} \end{cases}.$$

This means N_{pq} is defined as the $M_{\mathbf{i}_q}$, the marginalization matrix for $\mathbf{i}_q$, if the pth primary grouping happens to be the designated one for grouping $\mathbf{i}_q$.

Since a primary grouping is its own designated one, some of the N_{pq}'s must be the identity matrix. Specifically,

$$N_{pq} = I, \text{ if } \omega(\mathbf{i}_q) = \mathbf{i}_q = \mathbf{i}_{\xi(p)},$$

or equivalently,

$$N_{pq} = I, \text{ if } q = \xi(p). \tag{6.20}$$

Example 6: Let us continue our running example which has been the focus of Examples 1-3, and 5. Here the 3 sites have local orders of 2, 3 and 2 respectively. Since $r = 2$, the 2nd-order state vector for this system is the concatenation of all joint-statuses in the same order as they are listed in Table 6.2:

$$\mathbf{s}_{(2)}[k] = \left[\mathbf{s}'_{(1,1)}[k] \quad \mathbf{s}'_{(1,2)}[k] \quad \cdots \quad \mathbf{s}_{(3,3)}[k]\right]'.$$

The total length of $\mathbf{s}_{(2)}[k]$ is $(2+3+2)^2 = 49$. The 2nd-order primary vector is

$$\mathbf{y}_{(2)}[k] = \left[\mathbf{s}'_{(1,2)}[k] \quad \mathbf{s}'_{(1,3)}[k] \quad \mathbf{s}'_{(2,3)}[k]\right]'.$$

which is a vector of length $4+6+6 = 16$. The expansion matrix M_2 and the truncation matrix T_2 are of dimensions 16×49 and 49×16 respectively. All their entries are either a 0 or a 1, and the patterns of their 1-entries are shown in Figure 6.1. Notice that in columns 5 to 14, and 30 to 35,

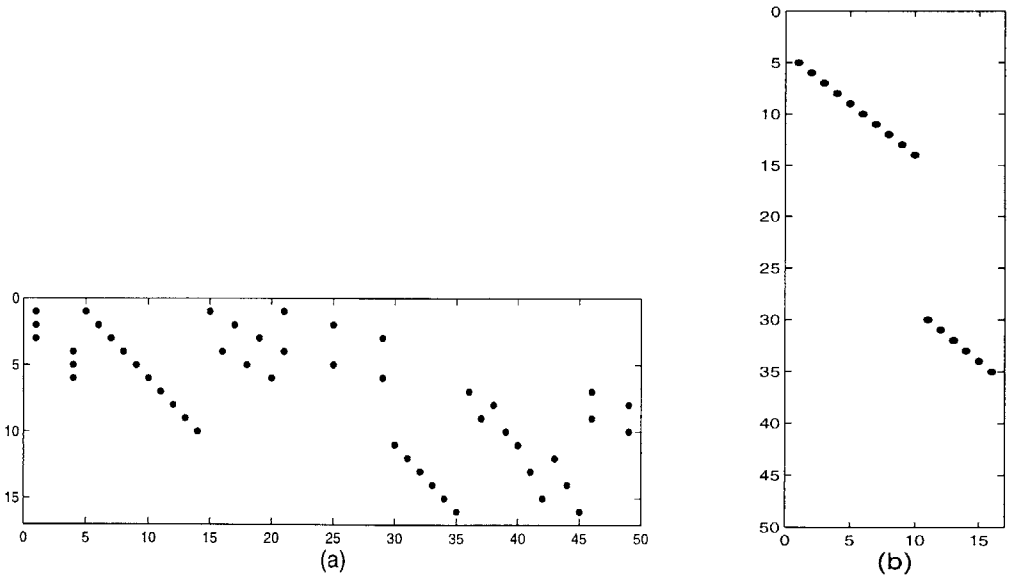


Figure 6.1: Pattern of the 1-entries in **(a)** M_2 and **(b)** T_2.

M_2 has the pattern of an identity matrix as explained. Also, see how the pattern of T_2 matches that in (6.15). □

THEOREM 6.3

$$M_r T_r = I$$

Theorem 6.3 conforms with our intuition, since we already know that by reducing the rth-order state vector $\mathbf{s}_{(r)}[k]$ to the primary vector $\mathbf{y}_{(r)}[k]$, we never lose any information. To see this, substitute (6.16) into (6.12) to get

$$\mathbf{y}'_{(r)}[k] = \mathbf{y}'_{(r)}[k] M_r T_r. \tag{6.21}$$

However, even though (6.21) must hold for every possible outcome of $\mathbf{y}_{(r)}[k]$, it alone does not allow us to conclude Theorem 6.3. The reason is that the entries in $\mathbf{y}_{(r)}[k]$ cannot be freely chosen. In particular, we generally cannot set $\mathbf{y}_{(r)}[k] = [0 \cdots 1 \cdots 0]'$, because it has to have at least one '1' entry per each primary joint-status. So we need the following more formal proof.

Proof. From (6.13) and (6.14), $R_{pq} = 0$ unless $p = \xi(q)$. Combining this fact with (6.19), we have

$$M_r T_r = \begin{bmatrix} N_{1\xi(1)} & \cdots & N_{1\xi(\eta)} \\ \vdots & & \vdots \\ N_{\eta\xi(1)} & \cdots & N_{\eta\xi(\eta)} \end{bmatrix} \tag{6.22}$$

$$= \begin{bmatrix} L_{\xi(1)} & \cdots & L_{\xi(\eta)} \end{bmatrix} \tag{6.23}$$

where the second equality follows from the definition of L_t in (6.17) and (6.18). By (6.20), all the matrices on the diagonal of (6.22) must be identity matrices, because they are of the form $N_{p\xi(p)}$. Consequently, the off-diagonal matrices in (6.22) must zero. The reason is that from (6.18), each L_t can have only one nonzero block, which we have already shown to be the identity matrix. This completes the proof. □

We can visually verify Theorem 6.3 by multiplying M_2 and T_2 in Figure 6.1 to see that the product is a 16×16 identity matrix.

6.3.3 Joint-State Vectors

In (6.10), we defined the rth-order state vector $\mathbf{s}_{(r)}[k]$ as a concatenation of all the joint-status vectors. Here we show it can be expressed as a permuted Kronecker product of the first-order state vectors. This is a rather technical point, but it will be important for our derivation of the

higher-order influence matrix later on.

For any matrix A, the *rth Kronecker-power* $A^{\otimes r}$ is defined inductively for all positive integers r by $A^{\otimes 1} \triangleq A$, and for $r \geq 2$,

$$A^{\otimes r} \triangleq A^{\otimes(r-1)} \otimes A.$$

THEOREM 6.4

There exists a unique permutation matrix P_r such that

$$\mathbf{s}'_{(r)}[k] = \mathbf{s}'[k]^{\otimes r} P_r.$$

where $\mathbf{s}[k]$ represents the first-order state vector of the influence model.

Proof. See Appendix C. □

Example 7: Using the system from Example 6, the permutation matrix P_2 such that $\mathbf{s}'_{(2)}[k] = \mathbf{s}'[k]^{\otimes 2} P_2$ is a 49×49 matrix and has the nonzero pattern as shown in Figure 6.2. □

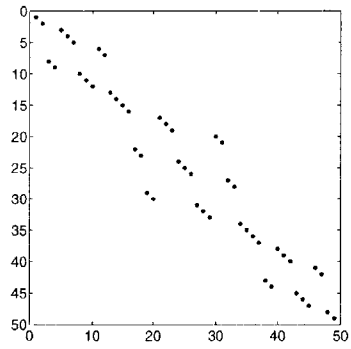


Figure 6.2: The permutation matrix for Example 7.

6.3.4 Derivation of Higher-Order Influence Matrix

At last, we have arrived at this critical section which will combine all the definitions and results in earlier sections and produce the feature of this chapter, namely the rth-order influence model.

At the beginning of Sec. 6.3 we explained our goal: to find evolution equations for $E(\mathbf{s}_\mathbf{i}[k])$ for every grouping $\mathbf{i}$ in order to know the probability of its joint-statuses. Then we explained that the evolution of the primary joint-statuses sufficed to infer all other joint-statuses. In this section,

we will use another important property of primary joint-statuses to derive the evolution of its expected value.

Recall that the next-state PMF vector of an n-site influence model is

$$\mathbf{p}'[k] = \begin{bmatrix} \mathbf{p}'_1[k] & \cdots & \mathbf{p}'_n[k] \end{bmatrix}.$$

Also, recall that the one-step evolution of the influence model conditioned on $\mathbf{s}[k]$ is given by

$$\mathbf{p}'[k+1] \triangleq \mathbf{s}'[k]\, H \tag{6.24}$$

$$\mathbf{s}'[k+1] \triangleq MultiRealize(\,\mathbf{p}'[k+1]\,) \tag{6.25}$$

In (6.25), different sites are realized independently.

For a grouping $\mathbf{i} = (i_1, \ldots, i_r)$, define

$$\mathbf{p}_\mathbf{i}[k] \triangleq \mathbf{p}_{i_1}[k] \otimes \cdots \otimes \mathbf{p}_{i_r}[k]$$

LEMMA 6.5

If $\mathbf{i} = (i_1, \ldots, i_r)$ *is a primary grouping, then*

$$E(\mathbf{s}_\mathbf{i}[k]) = E(\mathbf{p}_\mathbf{i}[k]). \tag{6.26}$$

Proof. Because all the sites in $\mathbf{i}$ are distinct, their status vectors are realized independently. Thus,

$$\begin{aligned} E\Big(\mathbf{s}_\mathbf{i}[k] \;\Big|\; \mathbf{p}[k]\Big) &= E\Big(\mathbf{s}_{i_1}[k] \;\Big|\; \mathbf{p}[k]\Big) \otimes \cdots \otimes E\Big(\mathbf{s}_{i_r}[k] \;\Big|\; \mathbf{p}[k]\Big) \\ &= \mathbf{p}_{i_1}[k] \otimes \cdots \otimes \mathbf{p}_{i_r}[k] \\ &= \mathbf{p}_\mathbf{i}[k]. \end{aligned} \tag{6.27}$$

Taking the expectation of (6.27) over $\mathbf{p}_\mathbf{i}[k]$, we have proved the claimed above. □

Using the notation in Sec. 6.3.2, let $\mathbf{i}_1, \ldots, \mathbf{i}_\nu$ be the list of all groupings arranged in lexicographical order. Let $\mathbf{i}_{\xi(1)}, \ldots, \mathbf{i}_{\xi(\eta)}$ be the list of the primary groupings, and define

$$\mathbf{p}_{(r)}[k] \triangleq \begin{bmatrix} \mathbf{p}_{\mathbf{i}_1}[k] & \cdots & \mathbf{p}_{\mathbf{i}_\nu}[k] \end{bmatrix} \tag{6.28}$$

$$\mathbf{z}_{(r)}[k] \triangleq \begin{bmatrix} \mathbf{p}_{\mathbf{i}_{\xi(1)}}[k] & \cdots & \mathbf{p}_{\mathbf{i}_{\xi(\eta)}}[k] \end{bmatrix}. \tag{6.29}$$

By comparing (6.28)-(6.29) to (6.10)-(6.11), we see that $\mathbf{p}_{(r)}[k]$ is analogous to $\mathbf{s}_{(r)}[k]$, and $\mathbf{z}_{(r)}[k]$

to $\mathbf{y}_{(r)}[k]$. This is confirmed by Lemma 6.5, which implies that

$$E\left(\mathbf{z}_{(r)}[k]\right) = E\left(\mathbf{y}_{(r)}[k]\right) \tag{6.30}$$

However,

$$E(\mathbf{p}_{(r)}[k]) \neq E(\mathbf{s}_{(r)}[k]) \tag{6.31}$$

in general. The difference between (6.31) and (6.30) is that eq. (6.30) only applies to primary groupings. The reason that the joint-PMF's of primary groupings work out so nicely is entirely because the sites in primary groupings are conditionally independent. On the other hand, secondary groupings have repeated sites, and therefore their status vectors are correlated.

A relation that is analogous to Theorem 6.4 is

$$\mathbf{p}'_{(r)}[k] = \mathbf{p}'[k]^{\otimes r} P_r. \tag{6.32}$$

Eq. (6.32) is valid because it only has to do with rearranging of the entries; it has nothing to do at all with independence. Similarly, we have

$$\mathbf{z}'_{(r)}[k] = \mathbf{p}'_{(r)}[k] T_r = \mathbf{p}'[k]^{\otimes r} P_r T_r \tag{6.33}$$

because T_r simply selects the primary groupings from $\mathbf{p}_{(r)}[k]$. Eq. (6.33) will be useful in the derivation of the higher-order influence matrix later on.

Finally, we have the desired evolution of the primary joint-statuses.

THEOREM 6.6

Using all the definitions from Sec. 6.3.2 and 6.3.3,

$$E\Big(\mathbf{y}'_{(r)}[k+1]\Big) = E\Big(\mathbf{y}'_{(r)}[k]\Big)\, H_r$$

where

$$H_r = M_r P_r' H^{\otimes r} P_r T_r.$$

Proof. For primary $\mathbf{i}$, we can write

$$\mathbf{z}'_{(r)}[k+1] = \mathbf{p}'[k+1]^{\otimes r} P_r T_r \qquad \text{by (6.33)} \tag{6.34}$$

$$= (\mathbf{s}'[k]H)^{\otimes r} P_r T_r \qquad \text{by (6.24)} \tag{6.35}$$

$$= \mathbf{s}'[k]^{\otimes r} H^{\otimes r} P_r T_r \qquad \text{by mixed-product property} \tag{6.36}$$

But

$$\mathbf{s}'[k]^{\otimes r} = \mathbf{s}'_{(r)} P'_r \qquad \text{by Theorem 6.4 and by } P_r^{-1} = P'_r \tag{6.37}$$

$$= \mathbf{y}'_{(r)}[k] M_r P'_r. \qquad \text{by (6.16)} \tag{6.38}$$

Therefore,

$$\mathbf{z}'_{(r)}[k+1] = \mathbf{y}'_{(r)}[k] M_r P'_r H^{\otimes r} P_r T_r \qquad \text{by substituting (6.38) into (6.36).} \tag{6.39}$$

Then, by taking expectation of (6.39) and substituting in (6.30), we have completed the proof. □

6.3.5 Significance of H_r

The matrix H_r defined in Theorem 6.6 is called the *rth-order influence matrix.* Exposing it in full detail is one of the major accomplishments of this thesis. This matrix enables us to obtain the probability of the rth-order joint status for any group of sites and for any time k. Naturally, this benefit comes at the expense of greater computational burden. Specifically, as a consequence of Theorem 6.6,

$$E\left(\mathbf{y}'_{(r)}[k]\right) = E\left(\mathbf{y}'_{(r)}[0]\right) H_r^k.$$

Thus, the task of computing the joint-probability depends on computing powers of H_r. The size of H_r grows with order r. As an example, suppose H is an n-site homogeneous influence matrix, with each site having a local chain of order m. Then the order of H_r is

$$\binom{n}{r} m^r.$$

To get an idea of how H_r grows, consider Table 6.6, which lists the order of H_r as a function of r. In this table, we assume that the influence model is homogeneous and that $n = 10$ and $m = 2$. At $r = 1$, we have $H_1 = H$, the first-order influence model. At $r = 10$, $H_r = G$, the state-

r	Order of H_r
1	20
2	180
3	960
4	3,360
5	8,064
6	13,440
7	15,360
8	11,520
9	5120
10	1024

Table 6.6: Order of H_r vs. r for a 10-site homogeneous influence model, with 2 statuses in each site.

transition matrix of the master Markov chain introduced in Chapter 5. For $r = 2$ to $r = 9$, we have the influence matrices of "intermediate" order. As we shall show later, the rth-order joint-PMF's would allow us to readily infer any joint-PMF's of order smaller than r. Therefore, if we have the 10th-order statistic in the above system, we would have the joint-PMF of all the orders.

Interestingly, we see that for $r = 4$ to $r = 9$, the order of H_r is actually larger than that of $H_{10} = G$. Nevertheless, we emphasize that this phenomenon is only due to the finite size of n. As n gets sufficiently large, the size of H_r for $r < n$ will be less than that of $G = H_n$, assuming that m and r are fixed. To see this for the case of homogeneous models, recall that the order of H_r is $\binom{n}{r}m^r$, while the order of $H_n = G$ is m^n. For sufficiently large n,

$$\binom{n}{r}m^r < n^r m^r < m^n \tag{6.40}$$

The first inequality of (6.40) holds because $\binom{n}{r}$, the number of ways to selecting r *distinct* objects from n available ones, is less than n^r, the number of ways to choose r objects out of n *without the restriction that the objects be distinct.* The second inequality of (6.40) follows because m^n grows exponentially in n, while n^r only grows polynomially.

Thus, for large n, the more detailed a joint-probability we want, the more computational power we have to devote for it. In Sec. 6.4, we will deal with the relation between H_r and G, the master Markov chain defined in Chapter 5.

6.3.6 Example 1: Calculating Variance with H_2

In this subsection, we demonstrate a particular use of H_r to help us further analyze the "to link or not to link" problem, which was first introduced in Sec. 5.4. Specifically, we will show that the dominant left eigenvector of H_2 can be used to analytically calculate the variance of the number of failures per time step. Without H_2, we would have to resort to Monte Carlo simulation, which is generally more time-consuming.

In Figure 5.2, we have obtained the histograms of the number of failures for two different (first-order) 30-site influence matrices, H_a and H_b, whose definitions are given in that section. Both influence matrices are homogeneous with the same 2-status local chain, representing"normal" and "failed" statuses. Their only difference is that in H_a all the sites are decoupled, whereas in H_b, each site is equally influenced by its neighbors and itself. One important result we obtained from simulating the two models was that the histogram for H_b had a greater variance than that of the uncoupled H_a. We justified this observation by describing qualitatively how the increased variance is caused by the coupling among the sites.

In this section, we will confirm this argument by calculating the variance for different influence matrices, each with a different coupling strength. Specifically, for each $0 \leq c \leq 1$, we define the network matrix $D(c)$ and its influence matrix $H(c)$ as

$$\begin{aligned} D(c) &\triangleq (1-c)I + cD \\ H(c) &\triangleq D(c)' \otimes A \end{aligned}$$

where D and A are the same ones from Sec. 5.4. It is not difficult to see that $H(c)$ is also a similar convex combination of H_a and H_b:

$$H(c) = (1-c)H_a + cH_b. \tag{6.41}$$

By varying c from 0 to 1, we can control the strength of the coupling among the sites; the higher c is, the stronger the coupling. We refer to c as the *coupling coefficient.*

We are interested in how c affects the variance of the number of failures per time step in steady-state. However, we will calculate this variance for each $H(c)$ by analyzing the dominant left eigenvector of $H_2(c)$, the second-order influence matrix of $H(c)$, rather than by Monte Carlo simulation. To do so, first consider $\mathbf{s}_i[k] = [s_{i1}[k] \;\; s_{i2}[k]]'$, the status vector of site i at time k. The scalar variable $s_{i2}[k]$ is a binary indicator of whether site i is in status failed at time k. Now, let

$\mathbf{t}[k]$ be the length-n vector containing such failure indicators of every site

$$\mathbf{t}[k] = \begin{bmatrix} s_{12}[k] & s_{22}[k] & \cdots & s_{n2}[k] \end{bmatrix}.$$

Thus,

$$\mathbf{t}'[k]\mathbf{1} = \text{ the number of failed sites at time } k.$$

The quantity we are interested in is the variance of $\mathbf{t}'[k]\mathbf{1}$. Because for all c, the influence processes defined by $H(c)$ is ergodic[1], by the ergodic theorem, the distribution of $\mathbf{t}[k]$ converges to a constant steady-state distribution for large k. Assuming that the steady state has been reached, then

$$\begin{aligned} \text{var}(\mathbf{t}'\mathbf{1}) &= E\left[(\mathbf{t}'\mathbf{1})^2\right] - \left[E(\mathbf{t}'\mathbf{1})\right]^2 \\ &= \mathbf{1}'E(\mathbf{t}\mathbf{t}')\mathbf{1} - E(\mathbf{t}')\mathbf{1}. \end{aligned} \tag{6.42}$$

In (6.42), we have dropped the time argument $[k]$ to avoid the clutter, and to reflect the fact that the distribution of $\mathbf{t}[k]$ is time-independent by the steady-state assumption above. The second term in (6.42) is equal to the expected number of failures at any given time, which we have already found in Sec. 5.4. The first term in (6.42) involves the matrix $E(\mathbf{t}\mathbf{t}')$, whose (i,j)th entry is equal to $E(s_{i2}s_{j2})$. Thus, our problem is to find this last quantity, the cross-correlation of failure indicators, for all possible pairs of sites i and j.

The cross correlations of failure indicators can be extracted from the dominant left eigenvector of $H_2(c)$ in a few steps. First we see that the dominant left eigenvector of $H_2(c)$ gives us (after appropriate scaling) the steady-state value of the primary vector $E(\mathbf{y}_{(2)}[k])$, from which we derive the steady-state second-order state vector via the relation

$$E(\mathbf{s}'_{(2)}[k]) = E(\mathbf{y}'_{(2)}[k])M_2. \tag{6.43}$$

The vector $E(\mathbf{s}'_{(2)}[k])$ gives us the joint-status all the $900 = 30^2$ pairs of sites in lexicographical ordering:

$$E(\mathbf{s}'_{(2)}[k]) = \begin{bmatrix} E(\mathbf{s}'_1 \otimes \mathbf{s}'_1) & E(\mathbf{s}'_1 \otimes \mathbf{s}'_2) & \cdots & E(\mathbf{s}'_{30} \otimes \mathbf{s}'_{30}) \end{bmatrix}',$$

where we have dropped the time argument $[k]$ from the right-hand side. Looking inside the expected

[1]Because $A > 0$, the master Markov chain G for this influence process satisfies $G > 0$ by the reasons explained in the last paragraph of Sec. 5.3.1. This therefore implies that $\Gamma(G)$ is ergodic.

joint-status for each pair (i, j), we have a length-4 vector

$$E(\mathbf{s}'_i \otimes \mathbf{s}'_j) = \begin{bmatrix} E(s_{i1}s_{j1}) & E(s_{i1}s_{j2}) & E(s_{i2}s_{j1}) & E(s_{i2}s_{j2}) \end{bmatrix}.$$

The last element is the cross-correlation of failure indicators that we are looking for.

To summarize, the procedure for finding the variance of the number of failures per step is as follows:

(a) For each $0 \leq c \leq 1$, construct $H(c)$ as in (6.41).

(b) Construct the expansion and truncation matrix M_2 and T_2. Both matrices should be the same for all c, because they are only a function of n, the number of sites, and m, the order of local chains.

(c) Construct $H_2(c) = M_2(H(c) \otimes H(c))T_2$.

(d) Find the dominant left eigenvector $\mathbf{v}$ of $H_2(c)$.

(e) Normalize $\mathbf{v}$ so that the first first four entries sum to 1. This gives us $E(\mathbf{y}_{(2)}[k])$.

(f) Obtain the $E(\mathbf{s}_{(2)}[k])$ from (6.43).

(g) Extract every fourth entry from $E(\mathbf{s}_{(2)}[k])$. These are the cross-correlations of the failure indicators.

(h) Sum the cross correlations to obtain the first term in (6.42). Since the second term is already known, by formula (6.42) we have obtained the variance of the number of failures at each time.

The result of this calculation for each $H_2(c)$ is shown in Figure 6.3. Note that we plot the *standard deviation* rather than the variance as a function of the coupling coefficient. This plot confirms our earlier intuition that as the coupling strength among the sites increases, the standard deviation of failure sizes at any given time increases.

6.3.7 Example 2: Determining Spatial Correlations from H_2

We show another application of H_2 by using it to calculate the conditional probability of failure given that a specific site has failed. We set up a 20-site homogeneous influence model with the structure of the network graph as shown in Figure 6.4. The network matrix D is defined the same

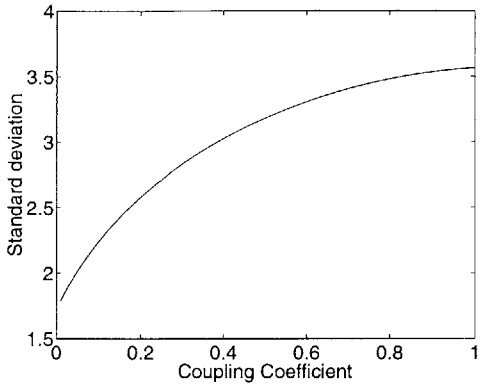


Figure 6.3: Standard deviation of the number of failures per step as a function of the coupling coefficient c.

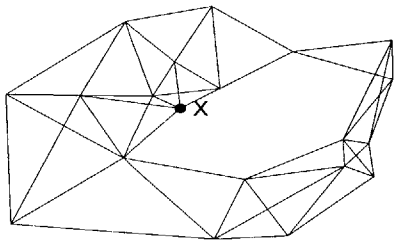


Figure 6.4: The Random Graph for Example 2.

way as in Sec. 5.4. That is, a site of k neighbors has $k+1$ incoming edges (after including the self-loop) and all edges have weight $1/(k+1)$. The local chain A is given by

$$A = \begin{bmatrix} .9 & .1 \\ .5 & .5 \end{bmatrix}.$$

Thus, it still has the same structure as that used in the previous example (the one in Fig. 5.1). The steady-state vector of A is [.833 .167]. From the influence matrix $H = D' \otimes A$, we obtained the second-order influence matrix H_2, calculated its dominant left eigenvector, normalized that vector so that the first four entries sum to 1, and converted from the steady-state primary vector $E(\mathbf{y}_{(2)}[k])$ to 2nd-order state vector $E(\mathbf{s}_{(2)}[k])$. As explained in Sec. 6.3.6, every fourth entry of $E(\mathbf{s}_{(2)}[k])$ is the cross-correlation of the failure indicator variables between every pair of sites. Each of these entries can also be interpreted as the joint-probability of the two sites failing simultaneously.

To show the effect of the spatial correlation on failures, we fix a site, which is marked X in Figure 6.4, and solved for the failure probability of each other site *given* that X has failed. This

conditional probability can be found by dividing by 0.167 the joint probability of each site failing simultaneously with site X. This constant is the individual probability of failure (i.e., the second entry from the dominant left eigenvector of matrix A). Normalizing the joint-probability by this constant thus gives us the conditional probability. These conditional probabilities are shown in Figure 6.5. From this figure, we can see that the conditional probabilities of failure are higher

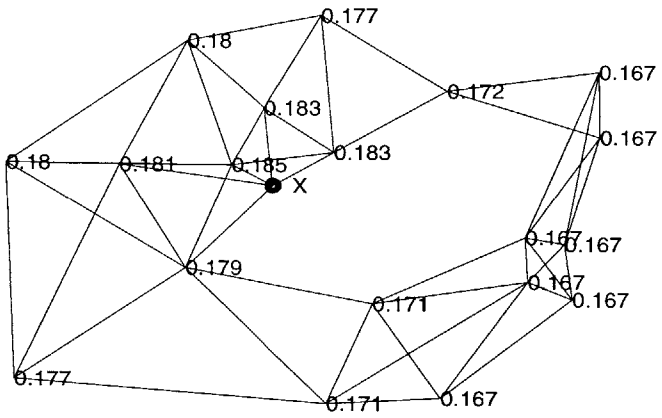


Figure 6.5: The probability of each site failing given that the center site has failed.

among the sites near site X, and gradually fall off as one moves away from it. The sites that are sufficiently far away from X are so independent from X that their conditional failure probabilities are almost equal to the unconditional one, which is 0.167. Notice also that the conditional failure probability only increases due to network connection; all labeled conditional probabilities are 0.167 or greater.

At first glance, the conditional probabilities in Figure 6.5 seem somewhat lower than one might expect. The reason is that these are the conditional probabilities of a site failing *in the same time index* given that X is currently failed. However, in the actual dynamics of the influence model, a failure at X would propagate to its neighbor *in the following time step*, assuming the neighbor is one step away. Thus, if we had calculated the conditional probability that a one-step neighbor of X fails in the following step given that X fails now, then that conditional probability would be significantly higher.

6.4 Relation between H_r and G

In the rest of this chapter, we will explore the relations between H_r and G in terms of their eigenstructure. This type of analysis is similar to what was done in Chapter 5, except that here the higher-order analysis will further illuminate the structure of eigenstructure of G. Again, throughout Sec. 6.4 we assume that r is fixed.

6.4.1 Higher-Order Event Matrix

In this section, we will introduce the higher-order event matrix B_r, which will perform much like the event matrix B did in Chapter 5. That is, B_r will serve as an event-listing matrix, a projector of PMF's (in the sense to be clarified), and finally as the matrix linking G and H_r.

Assume that we are given an influence model of n sites with orders of local chains as $m_1, \ldots, m_n$. Recall that the event matrix B is a $(\prod_i m_i) \times (\sum_i m_i)$ matrix. It contains rows that represents the outcomes of $\mathbf{s}[k]$. Each row of B is called an event, and is denoted by $\mathbf{b}'_{\mathbf{j}}$, where $\mathbf{j} = (j_1, \ldots, j_n)$, the address of that row. The addresses sequence in lexicographical order from $(1, 1, \ldots, 1)$ to $(m_1, m_2, \ldots, m_n)$.

Let P_r be the permutation matrix in Theorem 6.4, and T_r be as defined in (6.13). Define the *rth-order event matrix* B_r as

$$B_r \triangleq \begin{bmatrix} (\mathbf{b}'_{(1,1,\ldots,1)})^{\otimes r} \\ \vdots \\ (\mathbf{b}'_{(m_1,m_2,\ldots,m_n)})^{\otimes r} \end{bmatrix} P_r T_r. \tag{6.44}$$

Since B_r has the same number of rows as B, we can refer to each row of B_r using the n-tuple address as well. The meaning of B_r is straightforward when interpreted row by row: row $\mathbf{j}$ of B_r represents the rth-order primary state vector $\mathbf{y}_{(r)}[k]$ when $\mathbf{s}[k] = \mathbf{b}_{\mathbf{j}}$. To see this, recall that when $\mathbf{s}[k] = \mathbf{b}_{\mathbf{j}}$,

$$(\mathbf{b}'_{\mathbf{j}})^{\otimes r} P_r T_r = \mathbf{s}'[k]^{\otimes r} P_r T_r = \mathbf{s}'_{(r)}[k] T_r = \mathbf{y}'_{(r)}[k],$$

by Theorem 6.4, and by the definition of $\mathbf{y}_{(r)}[k]$. In other words, B_r is the matrix that lists all the possible outcomes of $\mathbf{y}_{(r)}[k]$ as its rows. We denote the row of address $\mathbf{j}$ in B_r as $\mathbf{b}^{(r)}_{\mathbf{j}}$.

As a result, each row of B_r must have $\binom{n}{r}$ 1's in it, because there are that many primary groupings of order r. In particular, when $r = n$, $B = I$, because there is only one possible grouping,

namely the grouping that contains all the sites. The rth primary vector $\mathbf{y}_{(r)}[k]$ in that case can be regarded as a status vector of the master Markov chain G, because each entry of $\mathbf{y}_{(r)}[k]$ would tell us the status of every site simultaneously.

Example 8: For a 3-site influence model whose local chains are of orders 2,3 and 2 respectively, the event matrices B, B_2 and B_3 are as shown in Figure 6.6. Their dimensions are 12×7, 12×16, and 12×12 respectively. As expected, these matrices contain $\binom{3}{1} = 3$, $\binom{3}{2} = 3$ and $\binom{3}{3} = 1$ entries in each row respectively. □

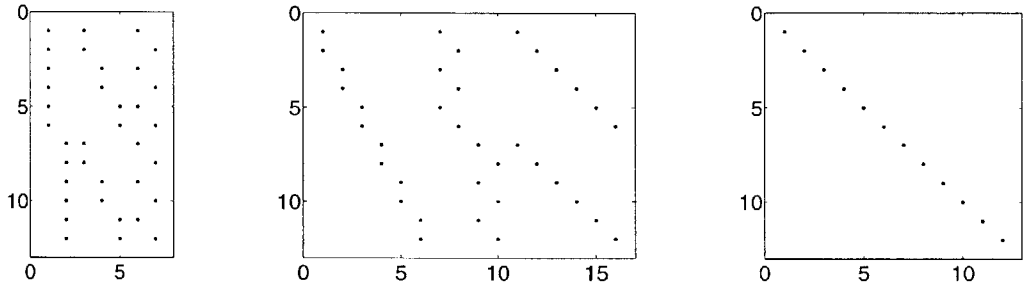


Figure 6.6: Examples of event-matrices B, B_2, and $B_3 = I$ respectively.

6.4.2 Conversion of Orders

THEOREM 6.7

There exists a matrix K_r such that

$$B_r = B_{r+1} K_r$$

Intuitively, that B_r is derivable from B_{r+1} according to Theorem 6.7 is not surprising, because each row of B_{r+1} contains the $(r+1)$st-order joint-statuses, which is more detailed information than the rth-order ones.

There are usually several possible matrices K_r that satisfy Theorem 6.7. Here we will settle on just one particular construction described in Appendix D. The resulting matrix K_r looks and feels similar to a marginalization matrix $M[\mathbf{j}, \mathbf{i}]$ introduced in Sec. 6.2.4.

Example 9: For the event matrices in Example 8, K_1 are shown in Figure 6.7. The matrix K_2 is not shown because it would be equal to B_2 itself, since $B_3 = I$. □

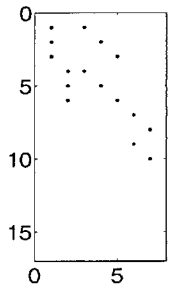


Figure 6.7: Example of K_1 for the event-matrices in Example 8.

6.4.3 Eigenstructure Relations

This section generalizes some of the results in Sec. 5.3.3 to higher-order influence matrices. In particular, it will expose the relation between the eigenstructure of G and that of each H_r. This relation leads to the telescoping relation of the relevant eigenvalues.

Let $\{\mathbf{f}[k]\}$ be the Markov process generated by the master Markov chain $\Gamma(G)$. For an event address $\mathbf{j}$, recall eq. (5.14) which says

$$Prob(\,(\mathbf{f}[k])_{\mathbf{j}} = 1\,) = Prob\,(\,\mathbf{s}[k] = \mathbf{b}_{\mathbf{j}}\,)\,. \tag{6.45}$$

But as we have explained in Sec. 6.4.1, the event $\{\mathbf{s}[k] = \mathbf{b}_{\mathbf{j}}\}$ is equivalent to the event $\{\mathbf{y}_{(r)}[k] = \mathbf{b}_{\mathbf{j}}^{(r)}\}$. Therefore, eq. (6.45) implies that

$$Prob(\,(\mathbf{f}[k])_{\mathbf{j}} = 1\,) = Prob\left(\,\mathbf{y}_{(r)}[k] = \mathbf{b}_{\mathbf{j}}^{(r)}\,\right).$$

Then by an argument similar to Sec. 5.3.2, we can show that

$$E(\mathbf{y}'_{(r)}[k]) = E(\mathbf{f}'[k])B_r \tag{6.46}$$

The following results are more general theorems that those found in Sec. 5.3.3. We keep the proof brief because of the similarity to the previous ones.

THEOREM 6.8

$$G^k B_r = B_r H_r^k$$

Proof. Because $E(\mathbf{f}'[0])G^k = E(\mathbf{f}'[k])$, by multiplying B_r from the right, we get

$$\begin{aligned} E(\mathbf{f}'[0])G^k B_r &= E(\mathbf{f}'[k])B_r \\ &= E(\mathbf{y}'_{(r)}[k]) && \text{by (6.46)} \\ &= E(\mathbf{y}'_{(r)}[0])H_r^k && \text{by Theorem 6.6} \\ &= E(\mathbf{f}'[0])B_r H_r^k && \text{by (6.46)} \end{aligned}$$

Hence, $E(\mathbf{f}'[0])G^k B_r = E(\mathbf{f}'[0])B_r H_r^k$. Since this must hold for all $E(\mathbf{f}[0])$, we have the above claim. □

As before, in the following we assume that H has distinct eigenvalues.

COROLLARY 6.9

Let $\mathbf{w}$, $\mathbf{v}$ *be a left and right eigenvector of* H_r *with a corresponding eigenvalue of* λ. *If* $B_r\mathbf{v} \neq 0$, *then* $B_r\mathbf{v}$ *is a right eigenvector of* G *with* λ *as its corresponding eigenvalue. Moreover, if* λ *is a non-repeated eigenvalue of* G, *then* $\mathbf{w}' = \widetilde{\mathbf{w}}'B_r$, *where* $\widetilde{\mathbf{w}}$ *is a left eigenvector of* G *corresponding to* λ.

COROLLARY 6.10

Let $\widetilde{\mathbf{v}}$, $\widetilde{\mathbf{w}}$ *be a left and right eigenvector of* G *with a corresponding eigenvalue of* λ. *If* $\widetilde{\mathbf{w}}'B_r \neq 0$ *then* $\widetilde{\mathbf{w}}'B_r$ *is a left eigenvector of* H_r *with* λ *as its eigenvalue. Moreover, if* λ *is a non-repeated eigenvalue of* G, $\widetilde{\mathbf{v}} = B_r\mathbf{v}$, *where* $\mathbf{v}$ *is a right eigenvector of* H_r *corresponding to* λ.

Proof. By the exact same reasoning as the proofs of Corollaries 5.12 and 5.12, except that here, we rely on Theorem 6.8 rather than Theorem 5.11. □

Define the set of *rth-order relevant eigenvalues* as

$$\kappa_r \triangleq \{\text{ The eigenvalues of } H_r \text{ whose corresponding right eigenvectors } \mathbf{v} \text{ satisfy } B_r\mathbf{v} \neq 0\} \quad (6.47)$$

For the rest of the chapter, we assume for simplicity that both G and H_r have distinct eigenvalues. Combining this assumption with Corollaries 6.9 and 6.10, we have an equivalent definition for the rth-order relevant eigenvalues

$$\begin{aligned} \kappa_r &= \{\text{ The eigenvalues of } G \text{ whose corresponding} \\ &\quad \text{left eigenvectors } \widetilde{\mathbf{w}} \text{ satisfy } \widetilde{\mathbf{w}}'B_r \neq 0\} \\ &= \sigma(G) \cap \sigma(H_r) \end{aligned} \quad (6.48)$$

THEOREM 6.11 (Telescoping)
If G has distinct eigenvalues, then

$$\kappa_r \subset \kappa_{r+1}$$

Proof. Let $\lambda \in \kappa_r$. Let $\mathbf{w}'$ be a corresponding left eigenvector of H_r, and $\widetilde{\mathbf{w}}'$ a corresponding left eigenvector of G. Then we can scale $\widetilde{\mathbf{w}}$ so that

$$\begin{aligned} \mathbf{w}' &= \widetilde{\mathbf{w}}' B_r && \text{by Corollary 6.9 and assumption} \\ & && \text{that } G \text{ has distinct eigenvalues} \\ &= \widetilde{\mathbf{w}}' B_{r+1} K_r && \text{by Theorem 6.7} \end{aligned}$$

Now, because $\mathbf{w}$ is an eigenvector,

$$\begin{aligned} \mathbf{w} \neq 0 &\rightarrow \widetilde{\mathbf{w}}' B_{r+1} K_r \neq 0 \\ &\rightarrow \widetilde{\mathbf{w}}' B_{r+1} \neq 0 \\ &\rightarrow \lambda \in \kappa_{r+1} && \text{by (6.48)} \end{aligned}$$

□

Since $H_n = G$, the telescoping relation gives us a deeper understanding of the spectrum of G; assuming that G has distinct eigenvalues, each eigenvalue of G is a relevant eigenvalue of some higher-order influence matrices.

6.4.4 Intuitive Interpretation of κ_r

In Sec. 5.3.6, we gave an intuitive explanation for $\kappa \stackrel{\triangle}{=} \kappa_1$ by explaining that these first-order relevant eigenvalues are the ones that specifically "correct" the deviation of $E(\mathbf{f}[0])$ from its steady-state. Now by a very similar argument, we will show that the rth-order relevant eigenvalues will correct the corresponding order of statistic in $E(\mathbf{f}[0])$.

Let us fix an order r. From Corollaries 6.9 and 6.10 and assuming that G has distinct eigenvalues, we can extend (5.35) to

$$\kappa_r \stackrel{\triangle}{=} \{ \text{The eigenvalues of } G \text{ whose corresponding left eigenvectors } \widetilde{\mathbf{w}} \text{ satisfy } \widetilde{\mathbf{w}}' B_r \neq 0 \}. \quad (6.49)$$

Assuming that H_r and G both have distinct eigenvalues, we can write the evolution of $E(\mathbf{f}[k])$ as

$$E(\mathbf{f}'[k]) = E(\mathbf{f}'[0])G^k = \widetilde{\mathbf{w}}_0' + \sum_{i=1}^{\mu_n - 1} \widetilde{\lambda}_i^k E(\mathbf{f}'[0])\widetilde{\mathbf{v}}_i \widetilde{\mathbf{w}}_i'. \tag{6.50}$$

Any $\widetilde{\lambda}_i \in \kappa_r$ is called an rth-order relevant eigenvalue, and its eigenvector an *rth-order relevant eigenvector*. We say that $E(\mathbf{f}[0])$ has a component along an rth-order relevant eigenvector if $E(\mathbf{f}'[0])\widetilde{\mathbf{v}}_i \neq 0$ for some $\widetilde{\lambda}_i \in \kappa_r$.

We say that $E(\mathbf{f}[0])$ has the *correct rth-order marginals* if

$$E(\mathbf{f}'[0])B_r = \mathbf{w}_0' \tag{6.51}$$

where $\mathbf{w}_0$ is a left eigenvector of H_r. The reason we regard as the "correct" rth-order statistic is that with condition (6.51), $E(\mathbf{y}_{(r)}'[0]) = E(\mathbf{f}'[0])B_r$ would simply not evolve, because it has already reached the steady-state.

THEOREM 6.12

$E(\mathbf{f}[0])$ has a component along an rth-order relevant eigenvector if and only if $E(\mathbf{f}[0])$ has the correct rth-order marginals.

We skip the proof, because it is nearly identical to that of Theorem 5.16. This shows that the eigenvalues in κ_r are the ones that correct the rth-order statistic of $E(\mathbf{f}[0])$.

Chapter 7

Conclusion

7.1 Summary

In this thesis we have presented and analyzed the influence model, a particular mathematical representation of random interactions on networks. The influence model comprises sites connected by a network structure. Each site has a status that evolves according to a Markov chain whose transition probabilities depend not only on the site's current status, but also on the statuses of its neighbors.

The results developed in this thesis form a foundation for the study of the influence model. After the thesis introduction in Chapter 1, we first introduced the background material in Chapter 2. This chapter covered the basics of general directed graphs and Markov chains. Then the special case of the binary influence model was studied in Chapter 3. If the network graph $\Gamma(D')$ is ergodic, the influence model must reach either the all-ones or all-zeros consensus. Moreover, the probability of reaching the all-ones consensus is contributed linearly by each site whose initial status is '1'. We then introduced the concept of the coalescing random walk as alternative ways to understand the binary influence model. In addition, we also used the coalescing random walk to derive some inequalities on probabilities that apply to the binary influence model.

In Chapter 4, we proceeded to define the general influence model and its influence matrix. The influence matrix $H = D' \otimes \{A_{ij}\}$ is defined as a generalized Kronecker product, where the network matrix D holds the information on how much the sites influence one other, and the matrices $\{A_{ij}\}$ describe the influences at the status level. We introduced the master Markov chain $\Gamma(G)$ as the chain in which each status represents one possible outcome of the influence process. As the order of G is much larger than that of H, we explained the practical benefits of extracting as much information about G as possible through the study of H.

The first reduced-order analysis in Chapter 4 involved determining the recurrent classes of $\Gamma(G)$ from the influence graph $\Gamma(H)$. We introduced the "hopping dots" picture to help us visualize and explain the results. The location of each dot in the influence graph defines the status of that site at any given time. These dots influence one another as they hop about. Our mission was to find all the configurations in which all the dots could be permanently trapped; this is equivalent to finding all the recurrent classes of $\Gamma(G)$. To do this, we first analyzed the structure of the influence graph for the case of an ergodic network graph. A major result of this chapter stated that when the network graph is ergodic, then the globally recurrent class (a recurrent class on the influence graph) will be the region within which all the dots in the system are trapped. Another interesting concept that emerged from the structure of homogeneous network graphs, where $H = D' \otimes A$, was the product path, a combination of a path on the network graph with another one on the local chain. Using the idea of product paths, classes on a homogeneous influence model were easily described in

terms of the local classes, provided that they satisfied a certain condition on the period, and that the network graph was irreducible.

In Chapter 5, we continued developing the relations between the influence matrix H and the state-transition matrix G of the master Markov chain. As opposed to Chapter 4, where the influence model was analyzed graphically, the analysis in this chapter was mostly through linear algebra. We revisited the definition of an autonomously recurrent class, which was introduced at the end of Chapter 4. The first important message of the chapter was that the influence matrix has a dominant eigenvalue at 1, and its multiplicity is equal to the number of autonomously recurrent classes in the influence graph; in short, an autonomously recurrent class is to an influence matrix what a recurrent class is to a Markov chain. The next major result of the chapter was the key relation $GB = BH$, where B is the event matrix. From this relation, we characterized the eigenvalues of H and partitioned its spectrum into the relevant and irrelevant eigenvalues. We explained in what sense the relevant eigenvalues are the ones whose duties are to "correct" the marginal distribution of a PMF. Finally, the chapter ended with an example of the "to link or not to link" dilemma. The conclusion of the section was that there was no difference on how a site chooses to connect to a network, assuming a homogeneous influence model with an ergodic network graph. The fraction of time that a site will spend in the 'failed' status is unaffected. Only the correlation between the status of a site and those of its neighbors increases.

In Chapter 6, we deepened our study of the influence model by venturing into the higher-order analysis. The idea was to derive a set of recursions that would provide us with not only the status probabilities of every site, but also the joint-status of any given collection of sites. The most important result of the chapter was that such recursions exist, and involve the higher-order influence matrix H_r. The matrix H_r is derived from the rth Kronecker power of H, the (first-order) influence matrix. It turned out that each higher-order influence matrix could also be related to the master Markov chain G by $GB_r = B_r H_r$, where B_r is the rth-order event matrix. We were also able to partition the spectrum of each higher-order matrix into the relevant and irrelevant eigenvalues. An important result of the chapter was the telescoping relation for the relevant eigenvalues: a relevant eigenvalue of the rth-order influence matrix is also a relevant eigenvalue of the $(r+1)$st-order influence matrix. Moreover, the nth-order influence matrix turned out to be the master Markov chain itself. The higher-order analysis gave us a deeper understanding of the spectrum of G; assuming that G has distinct eigenvalues, each eigenvalue of G is a relevant eigenvalue of some higher-order influence matrices.

Overall, the important features of the influence model can be summarized as follows:

- **General Structure** The freedom in choosing the network matrix D and the local chains $\{A_{ij}\}$

allows one to create a network with a fairly general structure, at least when compared to those in the traditional literature on voter models, where lattice grids are predominant.

- **Scalable Computation** The size of the higher-order influence matrix grows as a function of the order of the joint-status. This allows us the flexibility of obtaining intermediate-order statistics without having to compute the full-order master Markov chain matrix G.

- **Overall Tractability** As this thesis has shown, the influence model can be analyzed at various levels. It is the author's belief that the influence model is rich enough to yield much additional fruit, and interesting enough to warrant further research. It is the author's hope that these studies will lead to genuine insight, and perhaps application, in the setting of complex interactive networks.

7.2 Potential Areas for Future Research

The following is a list of some potential area for future research related to the influence models.

- **Fiedler Eigenvalue Conjecture** We conjecture that the eigenvalue with the second largest magnitude of κ_r is the same for every r. In all our numerical experiments, this has always been the case. Indeed, if one considers the form of H_r, which is

$$H_r = M_r P_r' H^{\otimes r} P_r T_r, \tag{7.1}$$

then one sees that H_r is almost a similaritry transformation of $H^{\otimes r}$. This is because $(M_r P_r')(P_r T_r) = I$, and thus, the right-hand side of (7.1) is almost in the form $C^{-1} H^{\otimes r} C$, but it is not; $M_r P_r'$ is not a square matrix, and thus, is only a left-inverse, as opposed to an inverse, of $P_r T_r$. If H_r were similar to $H^{\otimes r}$, then the conjecture would have been true, because every eigenvalue of H_r would be a product of some r (possibly repeated) eigenvalues of H. Since every eigenvalue of H has a magnitude that is less than or equal to 1, the eigenvalue with the second largest magnitude of H_r must be the same as that of H. This observation is a hint that the conjecture might be true. Since the eigenvalue of the second largest magnitude is sometimes referred to as the *Fiedler eigenvalue*, we call this the *Fiedler Eigenvalue Conjecture*. The importance of this eigenvalue is that it determines the rate of convergence to steady state, a topic often discussed under the heading of 'mixing properties' for a Markov chain [38].

- **Distinct Eigenvalue Conjecture** We conjecture that when H has distinct relevant eigenvalues, so does G. If this is the case, then we can make more compact statements on the relation

between the eigenstructure of G and H_r (which, of course, includes the first-order H). This conjecture would allow us to combine Corollaries 5.12 with 5.13 and Corollaries 6.9 with 6.10. Specifically, the left and right eigenvectors $\mathbf{v}$ and $\mathbf{w}$ of H_r would be related to the left and right eigenvectors $\widetilde{\mathbf{v}}$ and $\widetilde{\mathbf{w}}$ of G of the same relevant eigenvalue by the relations

$$\widetilde{\mathbf{v}} = B_r\mathbf{v} \qquad \text{and} \qquad \mathbf{w}' = \widetilde{\mathbf{w}}'B_r.$$

- **Reversible Graphs** Reversible Markov chains represent a special case of ergodic Markov chains in which the steady-state distribution satisfies a more restrictive set of conditions. One attractive feature of reversible chains is that the eigenvalues of their state-transition matrices are real-valued, which makes the analysis of such chains more tractable than the general case [3, 30, 31]. In particular, one can derive bounds on the Fiedler eigenvalue of a reversible state-transition matrix in terms of structural properties of the underlying graph [38]. The spectrum of a reversible Markov chain can also be related to that of an undirected graph. This connection gives the potential for one to apply known results on matrices of undirected graphs (such as Laplacian matrices) to the study of reversible Markov chains. Included as Appendix D is an explanation of the connection of the dynamics these two classes of matrices.

  By using reversible Markov chains in the influence model, we may be able to derive stronger analytical results. For instance, if a homogeneous influence matrix $H = D' \otimes A$ is defined from reversible D and A, then H also has a real-valued spectrum. Does this also mean that G is reversible, or at least, has real-valued spectrum? If G is also reversible, then it immediately qualifies for applications of many known results on reversible chains. We may be able to apply more elaborate analysis from [30, 31] such as the convergence time to the study of influence models.

- **Application to Traffic Modeling** The influence model may find potential applications in modeling of traffic networks. As briefly mentioned in Chapter 1, traffic congestion seems to have certain qualitative similarities to influence models: a random congestion occurs with a positive probability, a congestion tends to spread spatially, and the probability of a congestion at an intersection increases with the number of congested neighboring intersections. Another reason that suggests a potential fit is that traffic flows seem to be an aggregate result of various users of the street network, making it possible for an average-behavior type of analysis similar to the influence model. As a contrast, it does not seem likely that the influence model will be useful to study a system in which an idiosyncratic behavior or an individual's agenda can have a major impact on the system state.

- **Application to Queueing Networks** Another possible area of application for the influence model is the modeling and analysis of queueing networks. A queueing network can be roughly

described as follows. We have a network in which each site represents a queue. At each queue, new customers can arrive, wait for service, and leave after having been served. A customer who leaves a queue may enter another neighboring queue, or may leave the system altogether. See [16], Sec. 6.6 for more details.

One idea that seems plausible at first on how to apply the influence model to this type of system is as follows. We let each site be a finite birth-death chain and let the influences among the sites be the probability at which a customer who leaves one site enters another. Thus, the status at each site would be the number of customers currently in the queue. Having neighboring sites with long queues would increase the likelihood that a site would also have a long queue.

Unfortunately, the above approach leads to unrealistic dynamics. The ability to imitate a neighbor's queue length can can cause the number of waiting customers at a site to jump up or down abruptly if its neighbor happens to have very different number of customers in the previous time step. This would effectively result in a gross violation of internal conservation of customers.

Another possible approach is to let the influence model only represent the *change* in the queue length[1]. For example, we can let the status at each site be: -1, 0, and $+1$. These statuses are interpreted as the change in the number of customers in the previous time step. Thus, a separate state variable is needed at each queue to keep track of the actual number of customers in the queue. If the current status is $+1$, for instance, we add 1 to the queue length; if it is -1, we subtract 1 from the queue length, provided that the queue length does not fall below zero. The influence among the sites can then be specified. For instance, if we want to model two queues in tandem, then we may set status -1 of queue A to have an influence on status $+1$ of queue B.

The idea suggested in the above paragraph might lead to some interesting results. Although it is unlikely to conserve the number of customers inside a queueing network, it should be somewhat more realistic than the first version.

- **Extension of "To Link or Not To Link" Dilemma** Another interesting direction to pursue is to extend the result on the "To Link or Not To Link" dilemma. Currently, we have focused on only the case of homogeneous influence graphs. An obvious extension is to consider the case of general influence graphs.

- **Nonlinear Combination of Transition Probabilities** The fact that the next-state PMF $\mathbf{p}[k+1]$ is expressed as a linear function of the state $\mathbf{s}[k]$ has its pros and cons. While it

[1]This idea was first suggested by Sandip Roy.

makes the model analyzable, it is a rather restrictive way of specifying the next-state PMF. We believe that certain nonlinear functions of the states could yield an evolution that is more general and thus more applicable to real systems. The difficulty is that the proposed nonlinear evolution must not only produce a valid next-state PMF, but also has to be simple enough to be analyzable.

- **Application to Estimation Problems** The influence model could have some application in certain estimation and detection problems, for instance, in medical imaging. It has been proposed by [12] that Markov random field may be used in functional Magnetic Resonance Imaging. This idea is carried out further in [32] and has shown promising results after incorporating the steady-state distribution of certain interaction models into the decision process. In [32], the model in use was the Ising model, which suits the task particularly well because of the form of its steady-state distribution; its exponential form has been an important key to the simplification of the decision criterion. Thus, it is interesting to see whether the influence model can be used for such applications as well.

- **Variations on Network Graphs** Another possible research direction of the influence model is to let the network graph comprises two or more different type of graphs, such as $D = D_0 + D_1$, $D = (1-\epsilon)D_0 + \epsilon D_1$ etc. The two component graphs may represent physical systems with different time scales, different behavior (as in a power generator vs. a load), or they may be graphs with different structural properties. Another possibility is to let nodes of one kind in an influence network represent nodes of some underlying network, and nodes of another kind represent *edges* of the same underlying network. We may also let the network matrix D itself evolve over time.

Appendix A

Proof of Theorem 4.8

Assuming the definitions in the paragraph immediately preceding Theorem 4.8, our approach to proving it is as follows. Fix an integer k between 1 and w. For any two sites i and j, pick a status from $R_k^{(i)}$ and another from $R_k^{(j)}$. Denote the two selected statuses as $x_{(i)}$ and $z_{(j)}$ respectively. We will show that there exists a path from $x_{(i)}$ to $z_{(j)}$ by concatenating two paths β and η. Path β exists by Lemma A.1 and it will connect $x_{(i)}$ to some status $y_{(j)} \in R_k^{(j)}$. Path η exists by Lemma A.3 and will connect $y_{(j)}$ to $z_{(j)}$. Since $x_{(i)}$ and $z_{(j)}$ can be switched, this will mean that $x_{(i)}$ and $z_{(j)}$ communicate. If we take any status $y \notin P_k$, then y and $x_{(i)}$ do not communicate by Lemma A.4. This then proves that P_k is a class. The recurrence of P_k is shown in Corollary A.6. This would then complete the proof as desired.

Recall that a status $x_{(i)}$ in site i is an image of status x in $\Gamma(A)$. For this appendix, we will denote the latter status as $x_{(0)}$, instead of just x, to emphasize the fact that it is in $\Gamma(A)$. It then follows that $x_{(i)} \in R_k^{(i)}$ if and only if $x_{(0)} \in R_k$.

LEMMA A.1
There exists a path from $x_{(i)}$ to some status in $R_k^{(j)}$.

Proof. Since D is irreducible, a path q that connects i to j on $\Gamma(D')$ must exist. Let p be any path on $\Gamma(A)$ that originates at $x_{(0)}$, has the same length as q, and terminates at any status in R_k. Then the path $s = p \diamond q$ would have the property claimed. □

LEMMA A.2
Let $\{a_1, \ldots, a_r\}$ and $\{d_1, \ldots, d_t\}$ be two sets of positive integers whose greatest common divisors (GCD) are a and d respectively. If a and d are relatively prime, there exists a set of positive integers $\{c_k\}$ and $\{h_k\}$ such that

$$c_1 a_1 + \cdots + c_r a_r + 1 = h_1 d_1 + \cdots + h_t d_t$$

Proof. Because a and d are relatively prime, there are two positive integers b_1 and b_2 such that

$$b_1 a + 1 = b_2 d. \tag{A.1}$$

Next, by the property of GCD, there must also be integers $\{u_i\}$ and $\{v_i\}$, which are not necessarily all positive, such that

$$\begin{aligned} a &= u_1 a_1 + \cdots + u_r a_r \\ d &= v_1 d_1 + \cdots + v_t d_t \end{aligned}$$

Substituting these two equations into eq. (A.1), we have

$$(b_1 u_1) a_1 + \cdots + (b_1 u_r) a_r + 1 = (b_2 v_1) d_1 + \cdots + (b_2 v_t) d_t. \tag{A.2}$$

If, for example, the product $b_1 u_i$ is negative, we can increase it by adding the equation $d_1 a_i = a_i d_1$ to eq. (A.2), so that it becomes

$$(b_1 u_1) a_1 + \cdots + (b_1 u_i + d_1) a_i + \cdots + (b_1 u_r) a_r + 1 = (b_2 v_1 + a_i) d_1 + \cdots + (b_2 v_t) d_t.$$

Since all a_i's and d_i's are positive, we can keep doing this until all their coefficients are positive as desired. □

LEMMA A.3
The statuses in $R_k^{(j)}$ communicate.

Proof. If $d_{jj} > 0$, then the jth local chain exists and is described by $\Gamma(d_{jj} A)$. Then the statuses in $R_k^{(j)}$ must communicate by definition.

If $d_{jj} = 0$, let $y_{(j)}$ and $z_{(j)}$ be any two statuses on $R_k^{(j)}$. We will construct a path on $\Gamma(H)$ that connects $y_{(j)}$ to $z_{(j)}$. Let $\gamma = (\gamma_1 = y_{(0)}, \ldots, \gamma_e = z_{(0)})$ be a path on $\Gamma(A)$ that connects $y_{(0)}$ to $z_{(0)}$. This path exists because $y_{(0)}$ and $z_{(0)}$ are in the same class R_k. Let the period of class R_k be a. Then there must be a set of cycles $\{p_1, \ldots, p_r\}$ passing through $y_{(0)}$ such that the greatest common divisor (GCD) of the cycle lengths $\{\ell(p_1), \ldots, \ell(p_r)\}$ is equal to a.

On the other hand, on $\Gamma(D')$, there must be a set of cycles $\{q_1, \ldots, q_t\}$ passing through site j whose lengths have a GCD of d. Because d and a are relatively prime, by Lemma A.2, we can find positive integers $\{h_k\}$ and $\{c_k\}$ such that

$$h_1 \ell(p_1) + \cdots + h_r \ell(p_r) + 1 = c_1 \ell(q_1) + \cdots + c_t \ell(q_t). \tag{A.3}$$

Construct two paths α and δ by concatenating paths as follows

$$\begin{aligned} \alpha &\triangleq (p_1^{h_1}, \ldots, p_r^{h_r}) \\ \delta &\triangleq (q_1^{c_1}, \ldots, q_t^{c_t}) \end{aligned}$$

where the notation $p_i^{h_i}$ means looping h_i times in path q_i. By this construction, α and δ are cycles whose first nodes are $y_{(0)}$, and j respectively. From (A.3), $\ell(\alpha) + 1 = \ell(\delta)$. To equate the two lengths, we define $\widetilde{\delta}$ as another path on $\Gamma(D')$ that is the same as δ except with the last edge removed. So instead of terminating at j, $\widetilde{\delta}$ begins at j but terminates at some site u. This makes $\ell(\alpha) = \ell(\widetilde{\delta})$.

Consider the product path $s = \alpha \diamond \widetilde{\delta}$. This path originates from $y_{(j)}$, but it terminates at $y_{(u)}$, which is a status on a site that immediately precedes site j. We can then take advantage of that last step not only to get back to $R_k^{(j)}$, but also to move to $(\gamma_2)_{(j)}$, which is a step closer to our ultimate destination, the status $z_{(j)}$. Once we are back on $R_k^{(j)}$, the whole argument can be applied over again to find a path to $(\gamma_3)_{(j)}$, then to $(\gamma_4)_{(j)}$ etc. Eventually, we must reach $z_{(j)}$. Therefore, we have found a path from $y_{(j)}$ to $z_{(j)}$. Since these two statuses can be switched, we have thus proved that they communicate. □

LEMMA A.4

A status $y \notin P_k$ does not communicate with any status in P_k.

Proof. Suppose y communicates with some $x_{(i)} \in P_k$. Then there must be a path p on $\Gamma(H)$ from $x_{(i)}$ to y and another path q from y back to $x_{(i)}$. By Theorem 4.4.4, both p and q can be written as

$$\begin{aligned} p &= s \diamond t \\ q &= u \diamond v, \end{aligned}$$

where s, u are paths on $\Gamma(A)$, and t, v are paths on $\Gamma(D')$. Moreover, s must be a path from $x_{(0)}$ to $y_{(0)}$, while u must be a path from $y_{(0)}$ to $x_{(0)}$. Therefore, $x_{(0)}$ and $y_{(0)}$ are in the same class. But since $x_{(0)}$ is in R_k, so must $y_{(0)}$. It would then follow that $y \in P_k$. However, this contradicts the fact that $y \notin P_k$. Therefore, y does not communicate with any status in P_k. □

For any two nodes x and y on a directed graph, we say that y is *irreversibly accessible from* x if there is a path from x to y, but no path from y to x.

LEMMA A.5

A status $y_{(j)}$ is irreversibly accessible from $x_{(i)}$ on $\Gamma(H)$ if and only if $y_{(0)}$ is irreversibly accessible

from $x_{(0)}$ *on* $\Gamma(A)$.

Proof. Assume $y_{(j)}$ is irreversibly accessible from $x_{(i)}$. Let p be a path from $x_{(i)}$ to $y_{(j)}$. Because p can be factorized into $p = s \diamond t$, the path s must connect $x_{(0)}$ to $y_{(0)}$ on $\Gamma(A)$, and path t must connect site i to site j on $\Gamma(D')$. Hence, $y_{(0)}$ is accessible from $x_{(0)}$. If $x_{(0)}$ is accessible from $y_{(0)}$ as well, then let r be a path from $y_{(0)}$ to $x_{(0)}$. The path $q \stackrel{\triangle}{=} r \diamond t$ would then connect $y_{(i)}$ to $x_{(j)}$. But since $y_{(i)}$ and $y_{(j)}$ are in the same class by Lemma A.3, this would mean we have found a path from $y_{(j)}$ to $x_{(j)}$, via $y_{(i)}$. Again, by Lemma A.3, $x_{(j)}$ and $x_{(i)}$ communicate because they are in the same class. This means there is a path from $y_{(j)}$ to $x_{(i)}$, via $x_{(j)}$. This contradicts the earlier assumption that $y_{(j)}$ has no path to $x_{(i)}$. The converse can be shown similarly. □

COROLLARY A.6

P_k is a recurrent class of $\Gamma(H)$ if and only if R_k is a recurrent class of $\Gamma(A)$.

Proof. P_k is transient if and only if there exists a status $y_{(i)}$ outside of P_k such that $y_{(i)}$ is irreversibly accessible from every status in P_k. By Lemma A.5, this is the case if and only it $y_{(0)}$ is irreversibly accessible from every status in R_k, which is exactly the case when R_k is transient. □

Appendix B

Proof of Theorem 5.8

For $2 \leq i \leq n$, define the matrix

$$\widetilde{B}_{(i)} \stackrel{\triangle}{=} [\, B_{(i-1)} \otimes \mathbf{1}_{m_i} \mid \mathbf{1}_{\mu_{i-1}} \otimes \widetilde{I}\,] \tag{B.1}$$

where $\widetilde{I}$ is equal to I_{m_i} but with the last column removed. Comparing (B.1) to (5.4), we see that $\widetilde{B}_{(i)}$ is equal to $B_{(i)}$ with the last column removed.

LEMMA B.1
For $2 \leq i \leq n$,

$$rank\Big(\widetilde{B}_{(i)}\Big) = rank\Big(B_{(i-1)}\Big) + m_i - 1.$$

Proof. Suppose there exists some vectors a of length $(m_1 + \cdots + m_{i-1})$ and b of length $(m_i - 1)$ such that

$$\widetilde{B}_{(i)} \begin{bmatrix} a \\ b \end{bmatrix} = 0. \tag{B.2}$$

Substituting the definition (B.1) into (B.2),

$$\begin{aligned} 0 &= (B_{(i-1)} \otimes \mathbf{1}_{m_i})a + (\mathbf{1}_{\mu_{i-1}} \otimes \widetilde{I})b && \text{(B.3)} \\ &= (B_{(i-1)} \otimes \mathbf{1}_{m_i})(a \otimes 1) + (\mathbf{1}_{\mu_{i-1}} \otimes \widetilde{I})(1 \otimes b) \\ &= (B_{(i-1)}a \otimes \mathbf{1}_{m_i}) + (\mathbf{1}_{\mu_{i-1}} \otimes \widetilde{I}b) && \text{(B.4)} \\ &= \begin{bmatrix} (B_{(i-1)}a)_1 \mathbf{1}_{m_1} + \widetilde{I}b \\ \vdots \\ (B_{(i-1)}a)_n \mathbf{1}_{m_n} + \widetilde{I}b \end{bmatrix} && \text{(B.5)} \end{aligned}$$

where (B.4) follows from mixed-product property of Kronecker product, $(A \otimes B)(C \otimes D) = (A \otimes C)(B \otimes D)$. Notice how eq. (B.5) is vertically sectioned into n different portions. Focusing on the

kth portion of (B.5), we have

$$(B_{(i-1)}a)_k \mathbf{1}_{m_k} + \widetilde{I}b = 0. \tag{B.6}$$

Because the last entry of $\widetilde{I}b$ is always zero, and because the first term of (B.6) is a constant vector, this equation can only be satisfied if $(B_{(i-1)}a)_k = 0$. This in turn forces $\widetilde{I}b$, and hence b, to be zero. Since k is arbitrary, this means that the entire vector $B_{(i-1)}a$ must be zero. In conclusion, eq. (B.2) implies that $b = 0$ and that a is in $\mathcal{N}(B_{(i-1)})$, the null space of $B_{(i-1)}$. This means

$$\dim\Big(\mathcal{N}(\widetilde{B}_{(i)})\Big) = \dim\Big(\mathcal{N}(B_{(i-1)})\Big)$$

Finally, because $\widetilde{B}_i$ is a $\mu_i \times (\sum_{k=1}^{i} m_k - 1)$ matrix,

$$\begin{aligned}
\operatorname{rank}\Big(\widetilde{B}_{(i)}\Big) &= \min\left(\mu_i\,,\,\Big(\sum_{k=1}^{i} m_k\Big) - 1\right) - \dim\Big(\mathcal{N}(\widetilde{B}_{(i)})\Big) \\
&= \Big(\sum_{k=1}^{i} m_k\Big) - 1 - \dim\Big(\mathcal{N}(B_{(i-1)})\Big) \\
&= \Big(\sum_{k=1}^{i-1} m_k\Big) - \dim\Big(\mathcal{N}(B_{(i-1)})\Big) + m_i - 1 \\
&= \operatorname{rank}\Big(B_{(i-1)}\Big) + m_i - 1
\end{aligned}$$

as desired. □

LEMMA B.2
For $2 \le i \le n$,

$$\mathit{rank}\Big(\widetilde{B}_{(i)}\Big) = \mathit{rank}\Big(B_{(i)}\Big)$$

Proof. Since $B_{(i)} = [\widetilde{B}_{(i)}\ \mathbf{c}]$, we need the fact that the last column $\mathbf{c}$ of $B_{(i)}$ is linearly dependent on the other columns of $\widetilde{B}_{(i)}$. This is because with the last column,

$$\begin{aligned}
B_{(i)}\begin{bmatrix} \mathbf{1}_{m_1} \\ \mathbf{0}_{m_2+\cdots+m_{i-1}} \\ -\mathbf{1}_{m_i} \end{bmatrix} &= B_{(i)}\begin{bmatrix} \mathbf{1}_{m_1} \\ \mathbf{0}_{m_2+\cdots+m_{i-1}} \\ \mathbf{0}_{m_i} \end{bmatrix} - B_{(i)}\begin{bmatrix} \mathbf{0}_{m_1} \\ 0_{m_2+\cdots+m_{i-1}} \\ \mathbf{1}_{m_i} \end{bmatrix} \\
&= \mathbf{1} - \mathbf{1} \qquad \text{(B.7)} \\
&= 0. \qquad \text{(B.8)}
\end{aligned}$$

where in (B.7) we have used the eq. (5.7). □

Theorem 5.8 can now be proved as follows. Since $B_{(1)} = I_{m_1}$, it has full rank. The assertion is thus true for $i = 1$. Now assume the claim is true for all $B_{(k)}$ up to $k \leq i-1$ for $i \geq 2$. Combining Lemma B.1 and B.2, we finally conclude that

$$\begin{aligned} \operatorname{rank}\Big(B_{(i)}\Big) &= \operatorname{rank}\Big(B_{(i-1)}\Big) + m_i - 1 \\ &= \Big(\sum_{k=1}^{i} m_k\Big) + i - 1. \end{aligned}$$

By induction, this must be true for all i.

Appendix C

Proof of Theorem 6.4

LEMMA C.1
Let $\mathbf{a}, \mathbf{b}_1, \ldots, \mathbf{b}_n$ *be some vectors. Then*

$$\begin{bmatrix} \mathbf{b}'_1 \otimes \mathbf{a}' & \cdots & \mathbf{b}'_n \otimes \mathbf{a}' \end{bmatrix} = \begin{bmatrix} \mathbf{b}'_1 & \cdots & \mathbf{b}'_n \end{bmatrix} \otimes \mathbf{a}'$$

Proof. This follows from the definition of a Kronecker product. □

Recall that a permutation matrix is a square matrix that contains only 0's and 1's as its entries, and has exactly a single 1 in each row and in each column.

LEMMA C.2
Let $\mathbf{a}, \mathbf{b}_1, \ldots, \mathbf{b}_n$ *be some vectors. There exists a unique permutation matrix* P *such that*

$$\begin{bmatrix} \mathbf{a}' \otimes \mathbf{b}'_1 & \cdots & \mathbf{a}' \otimes \mathbf{b}'_n \end{bmatrix} = \left(\mathbf{a}' \otimes \begin{bmatrix} \mathbf{b}'_1 & \cdots & \mathbf{b}'_n \end{bmatrix} \right) P$$

Proof. See Corollary 4.3.10 in [27]. □

THEOREM C.3
There exists a unique permutation matrix P_r *such that*

$$\mathbf{s}'_{(r)}[k] = \mathbf{s}'[k]^{\otimes r} P_r.$$

In the following proof, we drop the time argument $[k]$ from every vector to reduce the notational clutter.

Proof. We will prove this induction. For $r = 1$, $\mathbf{s}_{(r)} = \mathbf{s}$. So the claim is trivially true. Now assume that the claim is true for up to $r = t - 1$, then we can write

$$\mathbf{s}'_{(t-1)} = (\mathbf{s}')^{\otimes(t-1)} P_{t-1} \tag{C.1}$$

for some permutation matrix P_{t-1}. At the same time, let all the $(t-1)$th-order groupings be $\mathbf{i}_1, \ldots, \mathbf{i}_\zeta$, where $\zeta = \binom{n}{t-1}$. Then we can also express $\mathbf{s}_{(t-1)}$ as

$$\mathbf{s}'_{(t-1)} = \begin{bmatrix} \mathbf{s}'_{\mathbf{i}_1} & \cdots & \mathbf{s}'_{\mathbf{i}_\zeta} \end{bmatrix} \tag{C.2}$$

Since the first-order state vectors is $\mathbf{s}' = [\mathbf{s}'_1 \ \ldots \ \mathbf{s}'_n]$, we have

$$\mathbf{s}'_{(t)} = \begin{bmatrix} \mathbf{s}'_{\mathbf{i}_1} \otimes \mathbf{s}'_1 & \cdots & \mathbf{s}'_{\mathbf{i}_1} \otimes \mathbf{s}'_n & \mathbf{s}'_{\mathbf{i}_2} \otimes \mathbf{s}'_1 & \cdots & \mathbf{s}'_{\mathbf{i}_2} \otimes \mathbf{s}'_n & \cdots & \mathbf{s}'_{\mathbf{i}_\zeta} \otimes \mathbf{s}'_1 & \cdots & \mathbf{s}'_{\mathbf{i}_\zeta} \otimes \mathbf{s}'_n \end{bmatrix} \tag{C.3}$$

which follows from the definition of $\mathbf{s}_{(t)}$ and from the definition of lexicographical ordering. Then by Lemma C.2, for each j we have

$$\begin{bmatrix} \mathbf{s}'_{\mathbf{i}_j} \otimes \mathbf{s}'_1 & \cdots & \mathbf{s}'_{\mathbf{i}_j} \otimes \mathbf{s}'_n \end{bmatrix} = \left(\mathbf{s}'_{\mathbf{i}_j} \otimes \begin{bmatrix} \mathbf{s}'_1 & \cdots & \mathbf{s}'_n \end{bmatrix} \right) P_j \tag{C.4}$$

where P_j is a unique permutation matrix. Applying (C.4) to (C.3) we get,

$$\begin{aligned}
\mathbf{s}'_{(t)} &= \begin{bmatrix} (\mathbf{s}'_{\mathbf{i}_1} \otimes \mathbf{s}')P_1 & \cdots & (\mathbf{s}'_{\mathbf{i}_\zeta} \otimes \mathbf{s}')P_\zeta \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{s}'_{\mathbf{i}_1} \otimes \mathbf{s}' & \cdots & \mathbf{s}'_{\mathbf{i}_\zeta} \otimes \mathbf{s}' \end{bmatrix} \underbrace{\begin{bmatrix} P_1 & & \\ & \ddots & \\ & & P_\zeta \end{bmatrix}}_{\widetilde{P}} \\
&= \left(\begin{bmatrix} \mathbf{s}'_{\mathbf{i}_1} & \cdots & \mathbf{s}'_{\mathbf{i}_\zeta} \end{bmatrix} \otimes \mathbf{s}' \right) \widetilde{P} && \text{by Lemma C.1} \\
&= (\mathbf{s}'_{(t-1)} \otimes \mathbf{s}')\widetilde{P} && \text{by (C.2)} \\
&= \left(\begin{bmatrix} (\mathbf{s}')^{\otimes(t-1)} P_{t-1} \end{bmatrix} \otimes \mathbf{s}' I \right) \widetilde{P} && \text{by (C.1)} \\
&= \begin{bmatrix} (\mathbf{s}')^{\otimes(t-1)} \otimes \mathbf{s}' \end{bmatrix} \underbrace{(P_{t-1} \otimes I)\widetilde{P}}_{P_t} && \text{by mixed-product property} \\
&= (\mathbf{s}')^{\otimes t} P_t.
\end{aligned}$$

Since $(P_{t-1} \otimes I)$ still has a single '1' in every row and every column, it is a permutation matrix. Also, being a product of two permutation matrices, P_t is a permutation matrix. So the claim still holds for $r = t$. By induction, this must be true for all r. □

Appendix D

Construction of K_r

A given rth-order grouping $\mathbf{i} = (i_1, \ldots, i_r)$ is *derivable* from an $(r+1)$st-order grouping $\mathbf{j} = (j_1, \ldots, j_{r+1})$ if $\{i_1, \ldots, i_r\} \subset \{j_1, \ldots, j_{r+1}\}$. Again, since there is usually more than one $(r+1)$st-order grouping from which an rth-order grouping can be derived, we have to fix a rule in picking one. So for a given rth-order grouping $\mathbf{i}$, define the *designated* $(r+1)$st-order grouping as the grouping that has all the sites in $\mathbf{i}$ and another site outside of $\mathbf{i}$ with the smallest integer. After sorting this new grouping, we are left with a unique $(r+1)$st-order grouping which is denoted as $\psi(\mathbf{i})$. Since $\mathbf{i} \subset \psi(\mathbf{i})$, it is not difficult to see that there exists a matrix $Y_{\mathbf{i}}$ such that

$$\mathbf{s}'_{\mathbf{i}}[k] = \mathbf{s}'_{\psi(\mathbf{i})}[k] Y_{\mathbf{i}}$$

We skip the explicit construction since it is similar to the construction of the marginalization matrix $M_{\mathbf{i}}$.

Let the primary groupings of order r be denoted by $\mathbf{i}_1, \ldots, \mathbf{i}_\alpha$, where $\alpha \triangleq \binom{n}{r}$. Also, let the primary groupings of order $(r+1)$ be denoted by $\mathbf{j}_1, \ldots, \mathbf{j}_\beta$, where $\beta \triangleq \binom{n}{r+1}$.

Then we recognize that the columns of B_{r+1} can be partitioned into β blocks (of unequal widths), with block p corresponding to $\mathbf{j}_p$, the pth primary joint-status of order $(r+1)$. Similarly, the columns of B_r can also be partitioned into α blocks for all the rth-order primary joint-statuses. Thus, the matrix K_r in Theorem 6.7 can be partitioned as

$$K_r = \begin{bmatrix} U_{11} & \cdots & U_{1\alpha} \\ \vdots & & \vdots \\ U_{\beta 1} & \cdots & U_{\beta\alpha} \end{bmatrix}$$

where

$$U_{pq} = \begin{cases} Y_{\mathbf{i}_q} & \text{if } \mathbf{j}_p = \psi(\mathbf{i}_q) \\ 0 & \text{otherwise.} \end{cases}$$

That is, U_{pq} is the conversion matrix $Y_{\mathbf{i}_q}$ if the pth primary grouping of order $(r+1)$ is the

designated one for the qth primary grouping of order r. This then produces a unique K_r which satisfies Theorem 6.7.

Appendix E

Reversible Markov Chains and Laplacian Matrices

This appendix explains how to convert between the matrix that describes an arbitrary undirected, edge- and node-weighted graph into the state transition matrix of a reversible Markov chain so that their eigenvalues are related. This type of conversion is generally well-known for the case of graphs with uniform node weights. However, this section considers a slightly more general case by letting the each node have an arbitrary weight.

Let $G = (V, E)$ be an undirected graph of n nodes. Assume that G is a connected graph. Let w_{ij} be weight of the branch that connects between node i and j. The *Laplacian matrix* $L(G)$ is defined as

$$[L]_{ij} \triangleq \begin{cases} -w_{ij} & \text{for } i \neq j \\ \sum_{j \neq i} w_{ij} & \text{for } i = j \end{cases}$$

For convenience, define the *line weight* of node i as

$$w_i \triangleq \sum_{j \neq i} w_{ij}$$

Define *mass matrix* as $M \triangleq \text{diag}(m_1, \ldots, m_n)$, where m_i is a strictly positive quantity representing the mass of the ith node.

Given W and M, we define the *weight-mass ratio* as

$$\alpha \triangleq \max_i \frac{w_i}{m_i}.$$

Define the *augmented weight matrix* $\widetilde{W}$ as

$$[\widetilde{W}]_{ij} \triangleq \widetilde{w}_{ij} = \begin{cases} w_{ij} & \text{for } i \neq j \\ \alpha\, m_i - w_i & \text{for } i = j \end{cases}$$

Note the crucial step of how the diagonal elements of $\widetilde{W}$ are defined. The purpose of such definition is so that the matrix $\widetilde{W}$ is nonnegative, and so that the sum of the ith row is proportional to the its mass. For convenience, let us define the sum of the ith row as *modified line weight*

$$\widetilde{w}_i \triangleq \sum_j \widetilde{w}_{ij} = \alpha\, m_i > 0.$$

Finally, define the *state-transition matrix* P as

$$[P]_{ij} \triangleq \frac{\widetilde{w}_{ij}}{\widetilde{w}_i}$$

PROPOSITION E.1

P is a valid state-transition matrix of a Markov chain.

This can seen by the fact that $\widetilde{W}$ is a nonnegative matrix, and that each row of P is defined so that it sums to unity.

For the sake of convenience, we will be a bit sloppy by using P to denote both a state-transition matrix, and the markov chain that it defines.

PROPOSITION E.2

If G is not bipartite, or if the ratios $\frac{w_i}{m_i}$ are not identical for all i, then P represents an ergodic, reversible Markov chain.

Proof: Because $\widetilde{W}$ is symmetric, every directed edge in P is matched by another edge in the opposite direction. So every node in P lies on some loop with two edges. Moreover, since G is assumed to be connected, all nodes in P communicate, i.e., they form an irreducible chain.

- If G is not bipartite, then somewhere in it must lie a loop with an odd number of edges. For any node i on that odd loop, the greatest common divisor of all loops that pass through i must be 1 (because gcd(2, an odd number) = 1), meaning i is aperiodic. Since P is irreducible, the whole chain must be aperiodic, hence ergodic.
- If the ratios w_i/m_i are not identical, then from the ways α and $\widetilde{W}$ are defined, some diagonal elements of $\widetilde{W}$ must be strictly positive. This translates to the existence of a self-loop in the Markov chain P, making it aperiodic and consequently, ergodic.

If P is ergodic, then it must have a steady-state distribution, and it will be

$$\pi_i \triangleq \frac{m_i}{\mu}$$

where μ is the total mass of the system $\mu \stackrel{\triangle}{=} \sum_i m_i$. To check that this is the correct steady-state distribution, one can verify that

$$\begin{aligned}
\pi_j &= \sum_i \pi_i P_{ij} \\
&= \sum_i \frac{m_i}{\mu} \frac{\widetilde{w}_{ij}}{\alpha m_i} \\
&= \frac{1}{\mu \alpha} \left(\alpha m_j - w_j + \sum_{i, i \neq j} w_{ij} \right) \\
&= \frac{m_j}{\mu}
\end{aligned}$$

To prove that P is reversible, we can verify that for all off-diagonal positions (i, j),

$$\pi_i P_{ij} = \frac{m_i}{\mu} \frac{\widetilde{w}_{ij}}{\alpha m_i} = \frac{m_j}{\mu} \frac{\widetilde{w}_{ji}}{\alpha m_j} = \pi_j P_{ji}.$$

Or equivalently, P is reversible because the matrix $\frac{1}{\mu} MP$ is symmetric. □

A note on reversible Markov chains In general, a markov chain P is reversible if it can made symmetric by multiplying the matrix $D = diag(\pi_1, \ldots, \pi_n)$ from the left. Since this multiplication is equivalent to scaling the ith row of P by π_i, one can easily test whether a given chain is reversible (i.e., satisfies $\pi_i P_{ij} = \pi_j P_{ji}$) while simultaneously finding its steady-state distribution by the following algorithm:

1. Set $i = 1$ and $c_1 = 1$.
2. Pick any neighbor j of node i whose c_j has not been defined. If no such j could be found go to step 5. Otherwise, continue to step 3.
3. Set $c_j \stackrel{\triangle}{=} \frac{c_i P_{ij}}{P_{ji}}$.
4. If there exists any neighbor k of j such that c_k has already been defined, and $c_j P_{jk} \neq c_k P_{kj}$, then stop; the graph is not reversible. Otherwise, set $i = j$, and go to step 2.
5. Find a node m whose c_m has already been defined but is connected to some neighbor j whose c_j is still undefined. If m can be found, set $i = m$ and go to step 2. Otherwise, continue to step 6.
6. The graph is reversible, and $\pi_i = c_i / (\sum_j c_j)$.

The algorithm above can also be applied towards continuous-time Markov chains, whose transition rates are given in matrix Q. We simply have to replace P with Q and repeat everything. However, in the final step, the resulting π_i has be interpreted as the time-average fraction of time the process spends in state i.

We now continue our main discussion on P.

PROPOSITION E.3

$$M(I-P) = \frac{1}{\alpha} L \tag{E.1}$$

Proof: By direct computation,

$$\begin{aligned} [M(I-P)]_{ij} &\triangleq \begin{cases} m_i \left(-\frac{\tilde{w}_{ij}}{\alpha m_i}\right) = -\frac{1}{\alpha} w_{ij} & \text{for } i \neq j \\ m_i \left(1 - \frac{\alpha m_i - w_i}{\alpha m_i}\right) = \frac{1}{\alpha} \sum_{k,k\neq i} w_{ik} & \text{for } i = j \end{cases} \\ &= \frac{1}{\alpha}[L]_{ij} \end{aligned}$$

If A is a matrix whose eigenvalues are real, let $\lambda_i(A)$ denotes its ith eigenvalue, sorted so that $\lambda_0(A) \leq \cdots \leq \lambda_{n-1}(A)$.

PROPOSITION E.4

The eigenvalues of P are real-valued and

$$1 - \lambda_i(P) = \frac{1}{\alpha} \lambda_{n-1-i}(M^{-\frac{1}{2}} L M^{-\frac{1}{2}})$$

Proof: By pre- and post-multiplying eq. (E.1) with $M^{-\frac{1}{2}}$, we get

$$M^{\frac{1}{2}}(I-P)M^{-\frac{1}{2}} = M^{-\frac{1}{2}} L M^{-\frac{1}{2}} \tag{E.2}$$

By similarity transformation, the eigenvalues of the product matrix on the left-hand side of eq. (E.2) are equal to those of $I - P$. But since we know that the right-hand side has real eigenvalues (because L is positive semi-definite and M is diagonal), so does the left-hand side. The negative sign in front of P then reverses the order of the eigenvalues. □

By Perron-Frobenius theorem, $-1 \leq \lambda_i(P) \leq 1$. This implies that

$$0 \leq \lambda_i(M^{-\frac{1}{2}} L M^{-\frac{1}{2}}) \leq 2\alpha.$$

In other words, through the conversion of the matrix, we have obtained an upper bound on the maximum eigenvalue of the Laplacian matrix. This bound, however, is not the tightest possible bound. Some improvement can be made by slightly adapting the proof of Theorem 2 in [29]. Before proceeding, let us recall a few facts from Chapter XIII from Gantmacher [17]. Denote by A^+ the matrix obtained by replacing each entry of A by its absolute value. If A is irreducible, then $|\lambda_i(A)| \leq \lambda_{n-1}(A^+)$, with equality if and only if $A = e^{i\phi} D A^+ D^{-1}$ where $D^+ = I$. For an irreducible non-negative matrix A, $\lambda_{n-1}(A) \leq$ the maximum row sum with equality if and only if all rows are equal.

PROPOSITION E.5

$$\lambda_{n-1}(M^{-\frac{1}{2}} L M^{-\frac{1}{2}}) \leq \max_{(i,j)\in E} \left(\frac{w_i}{m_i} + \frac{w_j}{m_j}\right)$$

Proof: The Laplacian matrix can be written in the following form

$$L = FWF' = \left(FW^{\frac{1}{2}}\right)\left(W^{\frac{1}{2}}F'\right)$$

where F is the node-edge incidence matrix, and $W = diag(w_1, \ldots, w_m)$ contains the weight of all the edges. Now let us define

$$N \stackrel{\triangle}{=} W^{\frac{1}{2}} F' M^{-1} F W^{\frac{1}{2}}$$

This matrix N and the node-weighted Laplacian matrix $M^{-\frac{1}{2}} L M^{-\frac{1}{2}}$ share the same nonzero eigenvalues. If i is an edge that points from node k to j, we define the *source* $s(i)$ and *destination* $d(i)$ of edge i as

$$s(i) \stackrel{\triangle}{=} k, \text{ and } d(i) \stackrel{\triangle}{=} j.$$

Now by similarity transformation

$$\lambda_i(N) = \lambda_i(W^{-\frac{1}{2}} N W^{\frac{1}{2}})$$

But $W^{-\frac{1}{2}} N W^{\frac{1}{2}} = F' M^{-1} F W$, and

$$[F' M^{-1} F W]_{ij} = \begin{cases} w_i \left(\frac{1}{m_{s(i)}} + \frac{1}{m_{d(i)}}\right) & \text{for } i = j \\ \frac{w_j}{m_{d(j)}} & \text{for } i \neq j \text{ and either } d(i) = d(j) \text{ or } s(i) = s(j) \\ -\frac{w_j}{m_{d(j)}} & \text{for } i \neq j \text{ and either } d(i) = s(j) \text{ or } s(i) = d(j) \end{cases}.$$

So the ith row sum of $(F'M^{-1}FW)^+$ can be interpreted as $\frac{w_{s(i)}}{m_{s(i)}} + \frac{w_{d(i)}}{m_{d(i)}}$. Moreover, the pattern of nonzero entries in $(F'M^{-1}FW)^+$ is symmetric, making it irreducible. Applying the theorem from Gantmacher as described above, the maximum eigenvalue of N can be no larger than $\max_i \left(\frac{w_{s(i)}}{m_{s(i)}} + \frac{w_{d(i)}}{m_{d(i)}}\right)$.□

The conversion between Markov and Laplacian matrices allow us to apply some result from one set of literature to apply to the other. As an example, we have the following bound.

Let S be a subset of nodes and let δS be the cut-set induced by it. The *capacity of* S is defined as

$$C_S \stackrel{\triangle}{=} \sum_{i \in S} m_i$$

and the *cut ratio induced by* S is

$$\phi_S \stackrel{\triangle}{=} \frac{\sum_{(i,j) \in \delta S} w_{ij}}{C_S}$$

Let the *cut ratio* as

$$\phi \stackrel{\triangle}{=} \min_{0<|S|<n, |C_S| \leq 1/2} \phi_S.$$

Then by adapting the bound from [38] using the conversion described in this paper, we have the following bound:

$$\lambda_1(M^{-\frac{1}{2}} L M^{-\frac{1}{2}}) \geq \frac{\phi^2}{2\alpha}.$$

Bibliography

[1] A. Berman and R. J. Plemmons. *Nonnegative Matrices in the Mathematical Sciences.* SIAM, 1994.

[2] L. E. Blume. The statistical mechanics of strategic interaction. *Games and Economic Behavior*, 5:387–424, 1993.

[3] P. Brémaud. *Markov Chains: Gibbs Fields, Monte Carlo Simulation, and Queues.* Springer-Verlag, 1999.

[4] P. Clifford and A. Sudbury. A model for spatial conflict. *Biometrika*, 60(3):581–588, 1973.

[5] J. Conlisk. Interactive Markov chains. *J. Mathematical Sociology*, 4:157–185, 1976.

[6] J. Conlisk. A stability theorem for an interactive Markov chain. *J. Mathematical Sociology*, 6:163–168, 1978.

[7] J. Conlisk. Stability and monotonicity for interactive Markov chains. *J. Mathematical Sociology*, 17(2–3):127–143, 1992.

[8] J. T. Cox. Coalescing random walks and voter model consensus times on the torus in z^d. *Annals of Probability*, 17(4):1333–1366, 1989.

[9] J. T. Cox and A. Greven. On the long term behavior of some finite particle systems. *Probability Theory and Related Fields*, 85:195–237, 1990.

[10] W. A. Mittelstadt D. N. Kosterev, C. W. Taylor. Model validation for the august 10, 1996 WSCC system outage. *IEEE Trans. on Power Systems*, 14(3):967–979, Aug. 1999.

[11] J. Demongeot, E. Golès, and M. Tchuente (Eds.). *Dynamical Systems and Cellular Automata.* Academic Press, 1985.

[12] X. Descombes, F. Kruggel, and D. Y. von Cramon. Spatio-temporal fMRI analysis using Markov random fields. *IEEE Trans. on Medical Imaging*, 17:1028–1029, Dec. 1998.

[13] P. Donnelly and D. Welsh. Finite particle systems and infection models. *Math. Proc. of Camb. Phil. Soc.*, 94:167–182, 1983.

[14] R. Durrett. An introduction to infinite particle systems. *Stochastic Processes and Their Applications*, 109–150, 1981.

[15] G. Ellison. Learning, local interaction, and coordination. *Econometrica*, 61:1047–1072, 1993.

[16] R. G. Gallager. *Discrete Stochastic Processes.* Kluwer Academic Publishers, 1996.

[17] F. R. Gantmacher. *The Theory of Matrices Volume II.* Chelsea, 1959.

[18] M. Gladwell. *The Tipping Point: How Little Things Can Make a Big Difference.* Little, Brown & Company, 2000.

[19] R. J. Glauber. Time-dependent statistics of the Ising model. *J. Mathematical Physics*, 4:294–307, 1963.

[20] M. Granovetter. Threshold models of collective behavior. *American J. Sociology*, 83(6):1420–1443, 1978.

[21] D. Griffeath. *Additive and Cancellative Interacting Particle Systems.* Springer-Verlag, 1979.

[22] D. Griffeath. The basic contact processes. *Stochastic Processes and their Applications*, 11:151–185, 1981.

[23] N. Grudinin and I. Roytelman. Heading off emergencies in large electric grids. *IEEE Spectrum*, 34(4):42–47, Apr. 1997.

[24] T. Harris. Contact interactions on a lattice. *Annals of Probability*, 2:969–988, 1974.

[25] R. A. Holley and T. M. Liggett. Ergodic theorems for weakly interacting infinite systems and the voter models. *Annals of Probability*, 3:643–663, 1975.

[26] R. A. Horn and C. R. Johnson. *Matrix Analysis.* Cambridge University Press, 1985.

[27] R. A. Horn and C. R. Johnson. *Topics in Matrix Analysis.* Cambridge University Press, 1991.

[28] W. Imrich and S. Klavžar. *Product Graphs: Structure and Recognition.* Wiley, 2000.

[29] W. N. Anderson Jr. and T. D. Morley. Eigenvalues of the Laplacian of a graph. *Linear and Multilinear Algebra*, 18:141–145, 1985.

[30] J. Keilson. *Markov Chain Models – Rarity and Exponentiality.* Springer-Verlag, 1979.

[31] F. P. Kelly. *Reversibility and Stochastic Networks.* Wiley, 1979.

[32] J. Kim. Spatio-temporal fmri signal analysis using information theory. Master's thesis, EECS Department, Massachusetts Institute of Technology, Aug. 2000.

[33] J. P. Lehoczky. Approximations for interactive Markov chains in discrete and continuous time. *J. Mathematical Sociology*, 7:139–157, 1980.

[34] T. M. Liggett. *Interacting Particle Systems.* Springer-Verlag, 1985.

[35] V. A. Malyshev. Breakdown of conservation laws in stochastic cellular automata. *Problemy Peredachi Informatsii*, 27(2):3–8, Apr.–Jun. 1991.

[36] Western Systems Coordinating Council (WSCC). Disturbance report for the power system outage that occurred on the western interconnection on august 10th, 1996 at 1548 PAST. *www.WSCC.com*, Oct. 1996.

[37] S. Morris. Contagion. *Review of Evonomic Studies*, 67:57–78, 2000.

[38] A. Sinclair and M. Jerrum. Approximate counting, uniform generation and rapidly mixing Markov chains. *Information and Computation*, 82:93–133, Jul. 1989.

[39] F. Spitzer. Interaction of Markov processes. *Advances in Mathematics*, 5:246–290, 1970.

[40] D. J. Watts. A simple model of fads and cascading failures. *Santa Fe Institute Working Paper*, 2000.

[41] S. Wolfram. *Cellular Automata and Complexity: Collected Papers*. Addison-Wesley, 1994.