Approximations and Inference of Dynamics on Networks

Alice L. Tapper

Department of Mathematics

University of Leeds

A thesis submitted for the degree of

Doctor of Philosophy

12th September 2021

Acknowledgements

My thanks first and foremost to my supervisors, Jonathan Ward and Richard Mann, for their continued guidance, advice and support throughout my PhD. A thank you also to John Paul Gosling, who also acted as my supervisor during my first year. I also acknowledge the funding and support provided by my CASE studentship partners, Jaywing Intelligence, and the help provided by my industry supervisor Peter Laflin.

I'm also incredibly grateful to my family and friends for their emotional support during this time. In particular: to my Mum and Dad, for always being there to listen; to Polly, Sally and Grace, who could understand and sympathise with the stresses; to Liam and Matti, for being my bubble and keeping me sane during lockdown; and to the Alexs, for being my ever-reliable climbing buddies.

Abstract

The study of dynamics on networks is a subject area that has wide-reaching applications in areas such as epidemic outbreaks, rumour spreading, and innovation diffusion. In this thesis I look at how to both approximate and infer these dynamics. Specifically, I first explore mean-field approximations for SIS epidemic dynamics. I outline several established approximations of varying complexity, before investigating how their accuracy depends on the network and dynamical parameters. Next, I use a method called approximate lumping to coarse-grain SIS dynamics, and I show how this method allows us to derive mean-field approximations directly from the full master equation description, rather than via ad hoc moment closures, as is common. Finally, I consider inference of network dynamic parameters on multilayer networks. I focus on a case study of SIS dynamics occurring on a two-layer network, where the dynamics on one of the layers is unobserved or "hidden". My goal is to estimate the SIS parameters, assuming I only have data about the events occurring on the visible layer. To do this I develop several simpler approximate models of the dynamics which have tractable likelihoods, and then use Markov chain Monte Carlo routines to infer the most likely parameters for these approximate dynamics.

Contents

List of Figures

Chapter 1

Introduction

1.1 Motivation

From its roots in graph theory, the science of networks is a subject that holds real-world relevance across a huge number of applied fields, simply because connected systems are everywhere. Networks appear extensively in biology, where they can be used to model both large-scale structures such as ecosystems of interacting species [7], or fine-scale structures such as amino acids and protein-protein interactions [8]. Networks appear in sociology, where they can be used to describe the social ties between people [9]. Notably, social media has led to the creation of online social networks, which can be mined to create huge network data sets and which have opened up unique opportunities for the marketing and advertising sectors [10, 11]. Network medicine is an emerging field which connects together symptoms and treatments in an attempt to identify, prevent and treat disease [12]. Networks can even describe many practical structures in society, such as transport systems [13] and energy grids [14].

Networks have particularly wide-reaching applications in the study of different dynamical systems. Many dynamical processes occur on a connected system, and incorporating information about the underlying network into any model of the process will provide a more accurate picture. Phenomena that have been modelled this way include epidemic dynamics [15], opinion dynamics [16], rumour spreading [17], the spread of memes online [18], the diffusion of innovation [19], racial segregation [20] and even the evolution of language [21]. This thesis was

largely inspired by social contagion dynamics, because this PhD project was a CASE partnership with Jaywing Intelligence, a data-led insight and marketing agency based in Leeds. A discussion of how the work in this thesis might be used in this industry context, and open questions relating to this application, feature in the conclusions in Chapter 6.

The thesis will also focus on dynamical processes largely used in the study of epidemic dynamics. This research area has proved especially topical over the last few years during the COVID-19 pandemic, where the use of models of disease spread in social and transport networks to estimate infection rates and predict the effects of interventions [22] has had literally life-or-death consequences. This focus is also relevant to the social contagion context, as some social contagion models are inspired in part by common epidemic models. I will elaborate on this point in Section 1.3.2.

In this thesis, I will explore several problems around the subject of approximating and inferring dynamics on networks. To this end, the next few sections will introduce some basic network concepts and structures. A majority of these definitions can be found in the core text by Newman [23], but I have aimed to relate these ideas to real-world networks to show their relevance. I will also summarise some of the dynamical processes most commonly modelled on networks, and the broad approaches used to solve these systems. The final section of this chapter will provide a breakdown of the thesis, and will introduce the questions I will be looking to answer in this work.

1.2 Networks

1.2.1 Notation

A graph (or network) can be described as an ordered pair $G = (V, E)$ comprising a set V of n vertices (or nodes), and a set E of m edges (or links). If $i, j \in V$ are vertices then the edge $(i, j) \in E$ if and only if i and j are connected. If i and j are connected we say that j is adjacent to i, and j is i's neighbour. If a vertex i is one of the two endpoints of an edge, we say that the vertex is incident to the

edge. I will use the words graph and network interchangeably during this thesis, and similarly for node and vertex, and edge and link.

Edges can be directed or undirected, depending on the nature of the relationship. For example, if the vertices represent two employees, where an edge between two vertices represents two employees working on a project together, then this network is undirected, since if person A works with person B then person B also works with person A. However, if the network edges instead represent one employee sending another employee an email, this network is directed: if person A sends person B an email, this is not necessarily reciprocated by person B sending person A an email. For directed networks, an edge (i, j) refers to a directed connection from node i to node j, and it is distinct from the edge (j, i).

A simple graph does not have any self-edges (edges beginning and starting at the same node) or multi-edges (multiple edges between the same pair of nodes). A multigraph is a graph where multi-edges are allowed. If we have a situation that is best modelled by defining different types of connections or relations, then we might model this as a multilayer network, where nodes can exist on different layers. A more detailed definition of multilayer networks will be provided in Chapter 5.

A path is a sequence of vertices $P = (v_1, v_2, ..., v_l)$ such that v_i is adjacent to v_{i+1} for $1 \leq i < l$. A path from v_1 to v_l is a path of length $l - 1$. A path can in general revisit a vertex, but if the path does not contain repeats of a vertex then we call it a simple path. If we are considering a directed network, then the sequence of vertices must be connected by edges that are all directed in the same direction. A vertex j is reachable from a vertex i if there exists a path from i to j. A cycle is a path wherein a vertex is reachable from itself.

A subgraph of a graph G is a graph formed from a subset of the vertices and edges of G. A connected component of an undirected graph G is a subgraph of G in which any two vertices belonging to the subgraph are connected to each other by paths, and which is connected to no additional vertices in G. If the network is directed we can define two types of components: strongly connected components, where every vertex is reachable from every other vertex, and weakly connected components, where every vertex is reachable from every other vertex only if the directed network became undirected.

Networks can be represented pictorially (see Figure 1.1 for an example), but for the sake of calculating certain metrics, and particularly to store information about a network in a computer, a numerical representation is useful. An adjacency matrix of a simple network is a matrix A with elements A_{ij} where

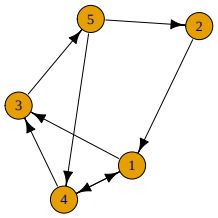
$$A_{ij} = \begin{cases} 1 \text{ if there is an edge between vertices } i \text{ and } j, \\ 0 \text{ otherwise.} \end{cases}$$

For a network with multi-edges the matrix element A_{ij} is equal to the number of edges between i and j. For directed networks the matrix element A_{ij} is equal to 1 if there is an edge going from vertex i to vertex j, so A_{ij} is not necessarily equal to A_{ji}. In later chapters I will sometimes equivalently use G and g_{ij} to denote the adjacency matrix of a network and its elements, to avoid confusion with other matrices labelled A introduced later.

Another way to represent a network is via an edge list, i.e. a list of all pairs of nodes which are connected by an edge. If the network is directed then the pair represents an edge going from the first node listed to the second node. If the network is undirected then the order of the pair doesn't matter.

Related to both of these concepts is the adjacency list. The adjacency list is in fact a set of lists, containing for each vertex a list of the other vertices it is connected to by an edge. This representation makes it very quick to identify a vertex's neighbours.

Figure 1.1 shows an example of a directed network, with the corresponding adjacency matrix, edge list and adjacency list, to illustrate these different representations. Which representation of the network is more convenient depends on the application. For example, if we have to delete an edge between node i and j in a network consisting of n nodes and m edges, this is an operation of $\mathcal{O}(1)$ time-complexity if we store the network as an adjacency matrix, while it is an operation of $\mathcal{O}\left(\frac{m}{n}\right)$ time-complexity if we store the data as an adjacency list. However, if we have an algorithm that requires us to scan through the neighbours of a node, the adjacency list is usually the better choice, since this operation has time-complexity $\mathcal{O}\left(\frac{m}{n}\right)$, which is an improvement on the $\mathcal{O}(n)$ complexity of the same operation performed using the adjacency matrix.



(a)

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

(b)

from	to
2	1
4	1
5	2
1	3
3	5
4	3
1	4
5	4

(c)

vertex	outgoing edges
1	3,4
2	1
3	5
4	1,3
5	2,4

(d)

Figure 1.1: An example of a directed network (a), along with the corresponding adjacency matrix (b), the corresponding edge list (c), and the corresponding adjacency list (d). Note that for a directed network, the adjacency list might feature lists of the outgoing edges (as in this example), or it might feature lists of the ingoing edges, or it might feature lists of each.

1.2.2 Metrics

The degree of a node is equal to the number of edges incident to it, with self-edges counted twice. In a network with no self-edges, then the degree of a node is equal to the number of adjacent nodes it has. Given an undirected graph $G = (V, E)$ where nodes have degree $\deg(i)$:

$$\sum_{i \in V} \deg(i) = 2|E|.$$

For directed networks we can distinguish between the in-degree of a node i, which is the number of nodes connected to i by an edge pointing towards i, and the out-degree of a node i, which is the number of nodes connected to i by an edge

pointing away from i. For example, in the context of social media a node's in-degree might represent the number of users who follow that individual, and the node's out-degree might represent the number of users who that user follows.

The degree distribution $p(k)$ describes the fraction of nodes with degree k. If the network is directed then one can distinguish between the in-degree distribution and the out-degree distribution. The degree distribution describes one aspect of the large-scale structure of a network, but fails to tell us how the nodes connect to each other and what kinds of patterns emerge from those connections. The degree distribution of most real-world networks is characterised by having a few nodes with very high degree, and a lot of nodes with very low degree, something I will expand upon in the next section.

The global clustering coefficient describes the tendency of nodes to connect with neighbours of their neighbours: it essentially tells us to what extent the 'a friend of my friend is also my friend' principle holds in a network. This property is often also referred to as the transitivity of a network, and is defined as

$$C = \frac{\text{number of closed triplets}}{\text{number of all triplets (open and closed)}}.$$

Here a triplet is defined as three nodes which are connected, where one of the three nodes is considered the central node. If each node is connected directly to each of the other nodes with an edge (i.e. the triplet has three edges) then they form a closed triplet. If the three nodes are only connected via the central node (i.e. the triplet only has two edges) then they form an open triplet. A triangle graph (three nodes, each connected to the two other nodes) includes three closed triplets, one centred on each of the nodes. Therefore we also define the clustering coefficient as

$$C = \frac{3 \times \text{number of triangles}}{\text{number of all triplets (open and closed)}}.$$

Figure 1.2 shows how the clustering coefficient increases as we form more edges and turn open triplets into closed triplets. Newman analyses the clustering coefficient for a number of different networks [24], and finds that the higher clustering coefficients tend to appear in networks representing people and communities: for example, he finds clustering coefficients of 0.59 and 0.45 in networks representing company director connections and coauthorships in physics papers respectively,

compared to coefficients of 0.01 and 0.09 found in an electronic circuit network and a biological metabolic network.

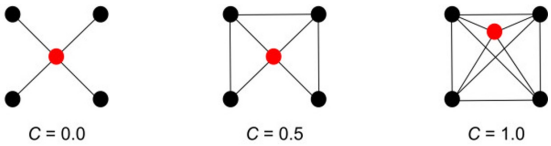


Figure 1.2: Three networks demonstrating increased clustering as more triplets are closed. Image adapted from Ravasz et al [1].

The degree assortativity coefficient r_{degree} measures to what extent nodes connect with nodes that have a similar degree. For example, are popular people generally connected to other popular people? This coefficient is given by

$$r_{\text{degree}} = \frac{\sum_{ij}(A_{ij} - k_i k_j/2m)k_i k_j}{\sum_{ij}(k_i\delta_{ij} - k_i k_j/2m)k_i k_j},$$

where m is the number of edges in the network, A_{ij} is an element of the adjacency matrix, and k_i is the degree of node i. The numerator is in fact the covariance of the degrees of neighbouring nodes calculated over all edges. We can also recognise that the first term $\sum_{ij} A_{ij}k_i k_j$ in the numerator is a measure of the degrees of neighbouring nodes. The second term $\sum_{ij} \frac{k_i k_j}{2m} k_i k_j$ is a measure of the degrees of neighbouring nodes if the nodes were placed completely randomly. Taking the difference between these two values shows whether the network displays positive or negative assortativity. The denominator acts as a normalising constant, so that $r_{\text{degree}} = 1$ for the case of a perfectly mixed network where $k_i = k_j$ for all neighbouring nodes i and j. Networks with high degree assortativity are networks where the high-degree nodes are connected to other high-degree nodes, and low-degree nodes are connected to other low-degree nodes. Networks with negative degree assortativity display the opposite behaviour, where high-degree nodes connect with low-degree nodes and vice versa. This results in the network displaying star-like branches. Examples of an assortative network and a disassortative network are shown in Figure 1.3. In his paper on assortative mixing in real-world networks [25], Newman found a general trend that social networks, such as research coauthorships and company directors, tended to be assortative,

while biological networks, such as protein interactions and the food web, tended to be disassortative.

We can consider other measures of assortativity in a network. For example, properties associated with the nodes can also be assortative or disassortative. For example, Bollen et al [26] found the happiness, or 'subjective well-being', of Twitter users was assortative across the platform, throwing up interesting questions about the impact of online social networks on our mood.

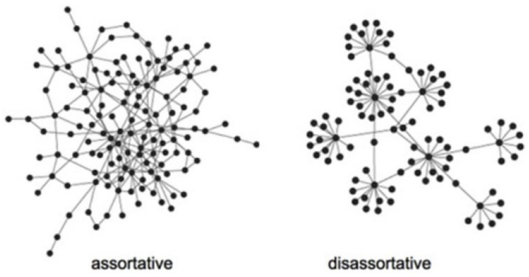


Figure 1.3: On the left we have an assortative network, and on the right we have a disassortative network, from Hao and Li [2]. Note the star-like pattern typical to disassortative networks.

A shortest path, or geodesic path, between two nodes is a path between the two nodes that visits the fewest intermediate nodes (or equivalently traverses the fewest edges). The length of a geodesic path is called the geodesic distance. This concept appears informally when we talk about 'six degrees of separation': the idea that any two people are separated by a chain of (at most) six connections [27]. If we calculate the geodesic distance between all pairs of nodes in the network, and find the average, this is called the average path length. It can be thought of as a measure of efficiency of transport on a network, or a measure of how interconnected a social network is. However, as an average it doesn't tell us about the range of geodesic distances. This is why it might be useful to also evaluate the diameter of a network, which is defined as the longest geodesic distance between any two nodes in the network.

Centrality is a measure of which nodes are the most important or influential in the network, and consequently it is a concept that is particularly useful in social network analysis. There are a number of different centrality measures used

to quantify a node's importance. Degree centrality is the most basic, and is simply the degree of a node. This is used because we would generally expect the nodes with the most neighbours to be the most influential. However, it doesn't discriminate between how important those neighbours are.

Eigenvector centrality accounts for this, giving more weight to connections with higher centrality nodes. The centrality x_i of a node i is proportional to the sum of its neighbours' centralities:

$$x_i = \kappa_1^{-1} \sum_j A_{ij} x_j,$$

where A_{ij} is an element of the network adjacency matrix, and κ_1 is the largest eigenvalue of the adjacency matrix. However, when applied to directed networks there is an issue: any node with only outgoing edges and no incoming ones will always have zero centrality. This zero-centrality node then won't contribute to the centrality of any of the nodes connected to its outgoing edges.

To avoid this issue we can use Katz centrality, where every node is given a non-zero base value of centrality regardless of its connections. The Katz centrality is defined as

$$x_i = \alpha \sum_j A_{ij} x_j + \beta,$$

where β is a positive constant defining how much initial centrality each node is assigned by default. The choice of α determines the balance between the eigenvector term and the constant β centrality term. Using this definition, we can also calculate the Katz centrality of node i as

$$x_i = \beta \sum_{l=0}^{\infty} \sum_{j=1}^{n} \alpha^l (A^l)_{ij}.$$

Here we can recognise that the element (i, j) of the adjacency matrix A raised to the power of l (A^l) is equal to the number of paths between nodes i and j of length l. In this way the Katz centrality of a node can also be thought of as the sum of the path lengths l to all other nodes in the network, with these distances attenuated by a factor α^l as they increase. Whereas previously the parameter β was needed to ensure non-zero centralities for certain nodes, this is no longer the case in this formulation. We can also ignore the $l = 0$ contribution in the sum,

since this is just a constant value for each node. Therefore we reach the other commonly used expression for the Katz centrality of a node:

$$x_i = \sum_{l=1}^{\infty} \sum_{j=1}^{n} \alpha^l (A^l)_{ij}.$$

We can also define measures of centrality based on shortest paths. For example, the betweenness centrality for a vertex is the number of shortest paths between all pairs of nodes in the network that pass through the vertex. We would expect nodes with high betweenness centrality to have a large amount of influence over information or objects passing between other nodes in the network. The closeness centrality of a vertex i looks at the mean of the shortest paths d_{ij} from vertex i to vertex j, averaged over all other $n-1$ vertices $j \neq i$ in the network. We then take the inverse of this value, so that nodes that are closer to others have a higher closeness centrality value.

Centrality measures have many applications due to their ability to identify influential nodes in a network. In epidemic dynamics, centrality can be used to try and flag which nodes are likely to be "superspreaders" [28]. Centrality measures have also been used to identify users in online networks for use in marketing, such as in Laflin et al [29], where the degree centrality and Katz centrality of users on a Twitter network are compared. However, we should be careful when drawing conclusions from these centrality measures. The relevance of a centrality measure depends highly on context. For example, a node with a high betweenness centrality might have a lot of control over interactions between different clusters in a network, or it might simply be on the periphery of both clusters and not important to either. It has also been shown that which nodes are the most influential is often not independent of the dynamics being studied. For example, Ferraz de Arruda et al [30] show that, for non-spatial networks, the degree centrality holds more relevance for epidemic spreading, while the closeness centrality is more relevant to rumour dynamics.

1.2.3 Models

Now that I have introduced various different network properties, it is worth considering a number of different network models that can be used to generate or

describe networks with specific properties. In particular, we are interested in what kind of network properties generally arise in the networks we actually see in real-life.

The most basic network model, the Erdős-Rényi model, generates random networks with a fixed number of vertices. Technically there are two Erdős-Rényi models. The first is the $G(n, m)$ model, where a graph is chosen uniformly at random from all possible graphs having n nodes and m edges. The second is the $G(n, p)$ model, where a graph is constructed by including each edge (and its attached nodes) with probability p independent from all other edges. Strictly these models are not defined in terms of a single randomly generated network, but as a probability distribution over all possible networks. $G(n, p)$ is the ensemble of networks with n vertices in which each graph G with m edges appears with probability

$$P(G) = p^m(1-p)^{\binom{n}{2}-m}.$$

To calculate the degree distribution consider a given vertex in the graph. The vertex is connected with independent probability p to each of the $n-1$ other vertices. The probability of having degree k is the probability of being connected to k vertices and not to the remaining $n-1-k$: $p^k(1-p)^{n-1-k}$. There are $\binom{n-1}{k}$ ways of choosing the k vertices, and so the probability of having degree k is

$$p_k = \binom{n-1}{k} p^k (1-p)^{n-1-k}.$$

So the $G(n, p)$ model has a binomial degree distribution. An example of an Erdős-Rényi can be seen in Figure 1.4(c).

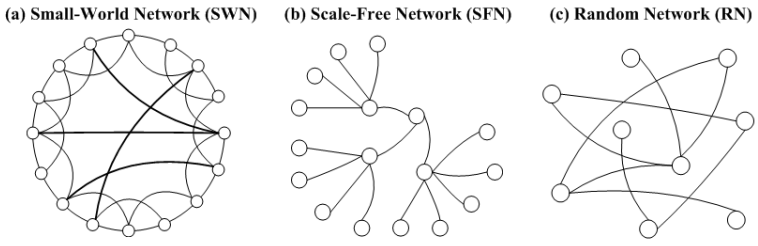


Figure 1.4: Example networks for the small-world model (a), the preferential attachment model (b), and the Erdős-Rényi model (c), from Huang et al [3].

However, this binomial degree distribution with most nodes having degree close to the mean degree is not one commonly found in real-life networks. Most real-life networks have a few nodes with very high degree, and a large number of nodes with low degree [31], with a degree distribution shaped similarly to Figure 1.5(a). If the degree distribution is such that the probability p_k of a node having degree k obeys

$$p_k = Ck^{-\alpha},$$

for some constant exponent α, then we call it a power law distribution, or a scale-free distribution [32]. The term 'scale-free' is subject to much ambiguity, and there is much debate around whether many real-life networks are truly scale-free. Broido and Clauset [33] investigate this in a paper which assesses the structure of almost 1000 networks, including social, biological, technological, transportation and information networks. They conclude that while some technological and biological networks appear to be "strongly" scale-free, social networks at at best "weakly" scale-free. However, while most real-world networks don't follow this power-law exactly, the 'heavy-tail' is still a common feature.

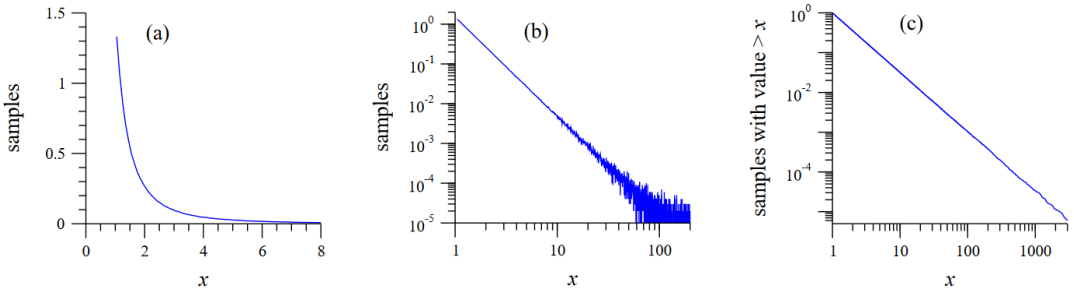


Figure 1.5: Plots showing the different ways of displaying a power law distribution, to help identify the distribution and determine the value of the power-law exponent. Panel (a) shows a histogram of 1 million random numbers, generated from a power-law probability distribution $p(x) = Cx^{-\alpha}$, where $\alpha = -2.5$. The log-log plot (b) of the same data reveals a roughly straight line, indicative of a power law distribution. The complementary cumulative distribution (c) reduces the amount of noise in the tail of the distribution, and also follows a power law. These plots are from Newman's paper on power laws [4].

The question of how to best identify a power law distribution, and how to calculate the correct value of the exponent α, is covered in a paper by Newman [4]. If we plot the histogram a power law distribution on a log-log scale, then the resulting histogram will be a straight line obeying

$$\ln p(x) = -\alpha \ln p(x) + c,$$

as in Figure 1.5(b). However, the tail towards the right hand side will inevitably exhibit a lot of noise, because the number of samples in the histogram bins are small and statistical fluctuations are therefore more noticeable. A solution to this is to plot the complementary cumulative distribution of the same data, which is the probability $P(x)$ that x has a value greater than or equal to x, as in Figure 5(c). This distribution also follows a power law, but with exponent $\alpha - 1$. To find the value of α we could naively do a least-square fit to the slope of this cumulative distribution, but a more reliable method is to perform maximum-likelihood analysis.

The preferential attachment model of Barabási and Albert [34] attempts to explain how these power law degree distributions might have formed. It is an example of a growth model, where the network grows node by node, and each new node forms connections due to a set procedure. Each new node forms exactly c connections, and these connections are made to existing nodes with probability precisely proportional to their current degree. This ultimately results in a degree distribution where the probability p_k of having degree k is approximately of the form

$$p_k = Ck^{-3},$$

i.e. here the power law exponent α is equal to 3. The model of Barabási and Albert is a model of an undirected network, but similar growth models based on the same principle of preferential attachment exist for directed networks, such as Price's model. It can be seen in Figure 1.6 that preferential attachment models accurately generate fat-tailed degree distributions. The phrase "fat-tailed" is used instead of scale-free, as the scale-free condition is more stringent. Another example of a scale-free network is shown in Figure 1.4.

However, one flaw of the network models described so far is that the resulting networks do not tend to exhibit the notable clustering that is present in many

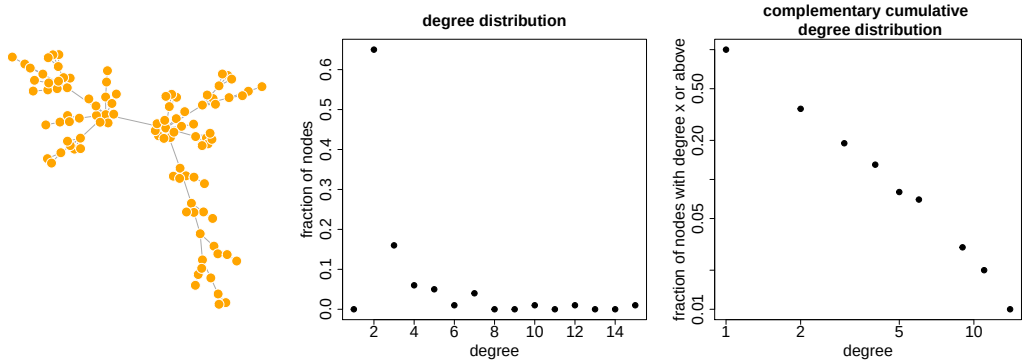


Figure 1.6: Example of a Barabási Albert network formed of 100 nodes, and the corresponding degree distribution and complementary cumulative degree distribution. The complementary cumulative degree distribution is plotted on a log-log scale to show the resulting approximate straight line.

real-life networks. The small-world model of Watts and Strogatz [5] was designed to mimic this clustering. The phrase 'small-world' is also used because the model can be used to generate networks with short average path lengths relative to the number of nodes, another property common to real-life networks [35]. To understand the Watts and Strogatz model it is easer to first understand a simpler 'circle model' that generates clustering. An example network generated by this model is shown in Figure 1.7, labelled 'Regular'. In the circle model, we can arrange vertices equally spaced on a circle. Each vertex is then connected to the $\frac{c}{2}$ vertices either side (c must be an even number). In Figure 1.7 $c = 4$, and so each vertex is connected to the two vertices either side of it.

We now compute the clustering coefficient for a network generated using the circle model. To traverse a triangle on such a network we must take two steps forward around the circle, and then one step back to the original vertex. The last step can be at most $\frac{c}{2}$ spacings, and the number of ways to choose the two steps forward is equal to the number of ways of choosing the target vertices for those steps from the $\frac{c}{2}$ possibilities, which is $\binom{c/2}{2} = \frac{1}{4}c(\frac{1}{2}c - 1)$. Therefore the number of triangles in a circle with n nodes is $\frac{1}{4}nc(\frac{1}{2}c - 1)$. Since the number of connected triples is $\binom{c}{2} = \frac{1}{2}c(c-1)$, the clustering coefficient is

$$C = \frac{\frac{1}{4}nc(\frac{1}{2}c - 1) \times 3}{\frac{1}{2}nc(c-1)} = \frac{3(c-2)}{4(c-1)}.$$

The model of Watts and Strogatz starts with the circle model, and then rewires some edges at random. Specifically, with probability p we remove an edge and replace it with one that joins two vertices chosen uniformly at random. The randomly placed edges are often referred to as shortcuts, due to the fact that they create shortcuts from one part of the circle to another. When $p = 0$ we have the circle model, which has high clustering but no small-world short average path length. This is shown in Figure 1.7, labelled 'Regular'. When $p = 1$ we have a random graph, with a short average path length but low clustering. This is also shown in Figure 1.7, labelled 'Random'. However, as p is increased from 0 the clustering is maintained up to quite large values, while the small-world behaviour appears for relatively low values, showing that the two effects are indeed compatible. This is a small-world network, labelled 'Small-world' in Figure 1.7.

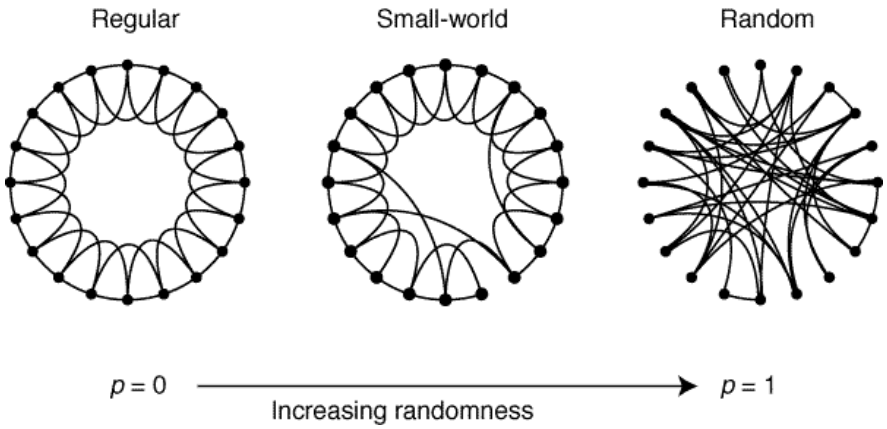


Figure 1.7: Example of the transition from the circle model into a small-world network into a random network, as we increase the probability of shortcuts forming. The regular circle model has high clustering but large average path length. The random network has short average path length but low clustering. The intermediate state, the small-world network, is the sweet spot where both short average path length and high clustering occur. This image is from Watts and Strogatz [5].

In this way the Watts and Strogatz model manages to capture both the clustering and small-world effect commonly found in real-life networks. However, the

degree distribution does not mimic well the fat-tailed distributions of most real-life networks. By considering a slightly different version of the Watts and Strogatz model, where shortcuts are added but none of the circle model connections are removed, it can be shown that the degree distribution follows

$$p_k = e^{-cp}\frac{(cp)^{k-c}}{(k-c)!},$$

i.e. the degree distribution has a peaked shape, and so does not obey a strict power law, but does have a fat-tail.

The final network model I will describe is the exponential random graph model (ERGM). The exponential random graph is an ensemble model, which means, as with random graphs, it is a set of possible networks plus a probability distribution over them. Rather than defining merely the probability of an edge forming, as in the Erdős-Rényi model, we in effect define the probabilities of any number of a range of "configurations" occurring. The ensemble generated is then the set of possible networks that have a set number of nodes, with their probabilities of occurring determined by how many of these configurations they feature. A list of common configurations is shown in Figure 1.8. The density configuration effectively corresponds to the number of edges, and the reciprocity configuration corresponds to the number of reciprocated edges in a directed network.

Consider $\mathcal{G}$ to be the set of all graphs with n vertices, and define an ensemble by assigning each graph G in the set $\mathcal{G}$ a probability $P(G)$ such that

$$\sum_{G\in\mathcal{G}} P(G) = 1.$$

The mean of any network measure $\langle x_i\rangle$ within this ensemble will be given by

$$\langle x_i\rangle = \sum_{G\in\mathcal{G}} P(G)x_i(G).$$

If we then fix the mean value of each of our measures within the ensemble, this becomes a constraint on the graph probability distribution. The configuration probabilities that we choose to define determine which measure mean values $\langle x_i\rangle$ we set. However, the number of these measures will typically be far less than the possible number of graphs in the ensemble. As a result they do not specify the probability distribution $P(G)$ precisely.

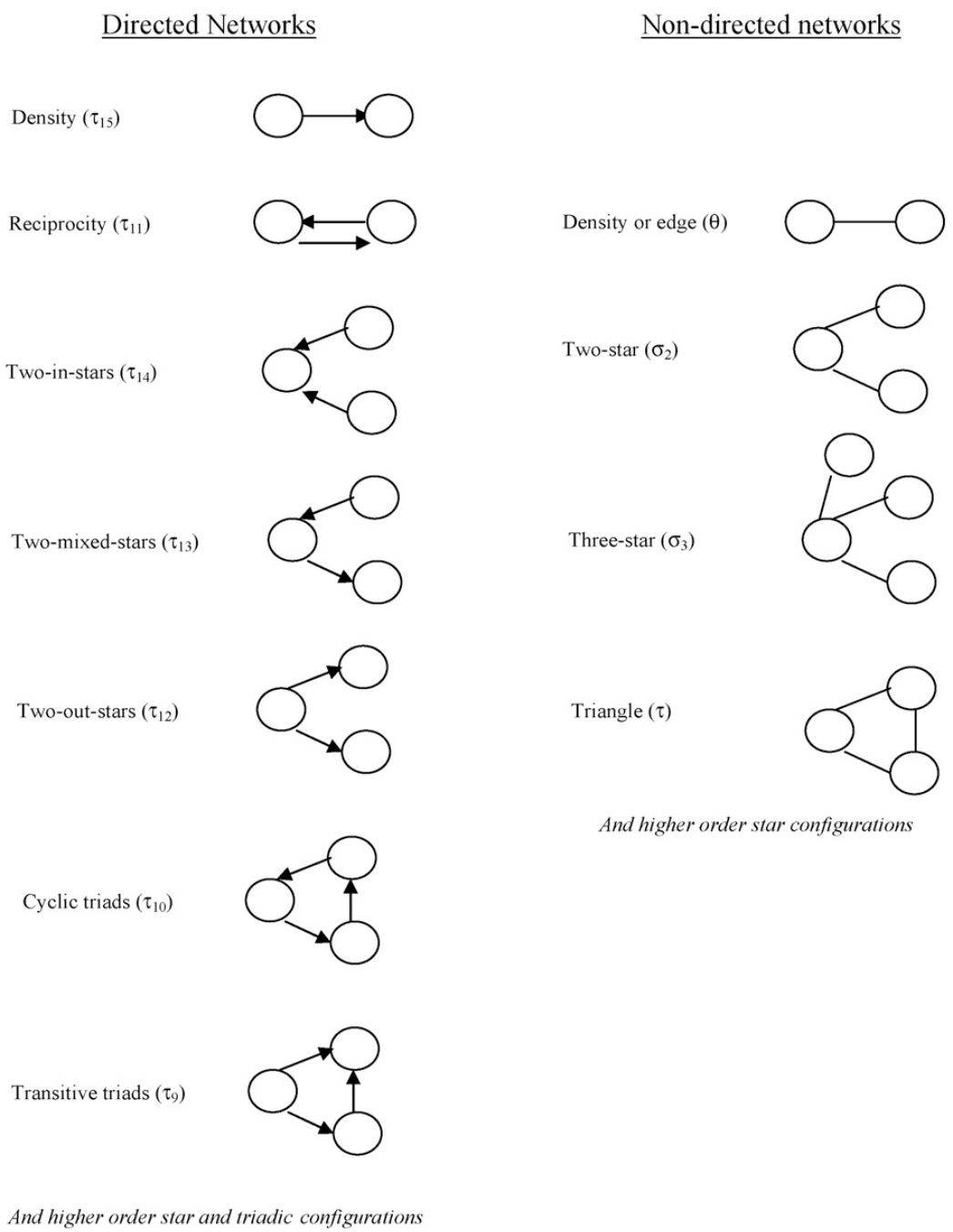


Figure 1.8: Common configurations, from Robins et al [6]. These are just some of the potential configurations one can model with an ERGM. The two-star model referenced in the text uses the Density and Two-star configurations.

We want to make the best choice for the probability distribution given only a small number of constraints. It turns out that the best choice [23] is one that maximises the Gibbs entropy of the system

$$S = -\sum_{G\in\mathcal{G}} P(G)\ln P(G),$$

subject to the known constraints. To maximise this entropy subject to constraints we use the method of Lagrange multipliers. Introducing Lagrange multipliers α and β_i as the Lagrange multipliers for the constraints $\sum_{G\in\mathcal{G}} P(G) = 1$ and $\sum_{G\in\mathcal{G}} P(G)x_i(G) = \langle x_i\rangle$ respectively, we want to maximise

$$-\sum_{G\in\mathcal{G}} P(G)\ln p(G) - \alpha\left[1 - \sum_{G\in\mathcal{G}} P(G)\right] - \sum_i \beta_i\left[\langle x_i\rangle - \sum_{G\in\mathcal{G}} P(G)x_i(G)\right].$$

If we differentiate this with respect to $P(G)$ and set the result to zero to find the maximum, we find that

$$-\ln P(G) - 1 + \alpha + \sum_i \beta_i x_i(G) = 0,$$

which can be solved to find

$$P(G) = \exp\left[\alpha - 1 + \sum_i \beta_i x_i(G)\right],$$

or

$$P(G) = \frac{e^{H(G)}}{Z},$$

where $Z = e^{1-\alpha}$ is the partition function and

$$H(G) = \sum_i \beta_i x_i(G)$$

is the graph Hamiltonian. The partition function Z can be thought of as the normalising constant for the probability distribution, and can be found by requiring that

$$\sum_{G\in\mathcal{G}} P(G) = \frac{1}{Z}\sum_{G\in\mathcal{G}} e^{H(G)} = 1.$$

This fixes the value of Z as

$$Z = \sum_{G \in \mathcal{G}} e^{H(G)},$$

which in turn also fixes the value for the Lagrange multiplier α. However, no equivalent general formulas for the values of β_i exist. They can be found by substituting the ERGM definition $P(G) = \frac{e^{H(G)}}{Z}$ into the constraint $\sum_{G \in \mathcal{G}} P(G) x_i(G) = \langle x_i \rangle$, which results in a solvable set of non-linear simultaneous equations that are dependent on the form of the Hamiltonian.

Once we have an expression for the probability distribution $P(G)$ we can substitute it in to find the expectation values of other quantities of interest within the ensemble. For example, if we are interested in a quantity y then we can find its expectation value by using the formula

$$\langle y \rangle = \sum_{G \in \mathcal{G}} P(G) y(G) = \frac{1}{Z} \sum_{G \in \mathcal{G}} e^{H(G)} y(G).$$

In this way the exponential graph framework can be used to answer the question of: "If I know about the likelihood of certain configurations occuring in my network, what is the best estimate of some other property of the network?"

The ability of ERGMs to model numerous different configurations means they can be adapted to various different situations. They are especially useful in social network analysis [36], since they allow researchers to specify configurations based on node attributes (which might correspond to gender, race, expertise in a field, etc.). However, there are several problems with the exponential graph framework. One such problem occurs in the two-star model, which specifies the expected number of edges and the expected number of two-stars in the network (see Figure 1.8). Sometimes the networks generated from this model have high density, and sometimes they have low density, and this can't be predicted in advance. There are also certain combinations of α and β which cannot occur.

1.3 Dynamical processes

A dynamical system is a system whose state changes over time. The changes in time are governed by some given rules or equations, and the system can evolve

in discrete or continuous time. If the dynamics are running on a network, the network topology will affect the process. For example, if we model a diffusion process, we can impose that whatever is diffusing can only travel to adjacent nodes. In this section I will outline several different kinds of dynamical model that are commonly studied, before outlining several methods used to formulate and study these dynamics on networks in Section 1.4.

1.3.1 Epidemic models

Epidemic models simulate the spread of disease in a population. The inclusion of network topology provides information about who is in contact with whom, allowing for the epidemic spread to be more accurately modelled.

The most basic epidemic model is the SI model. In this model individuals can be in one of two states: suspectible (S) to the infection, or infected (I). Susceptible individuals can become infected through contact with an infected individual, governed by a spreading rate β, which describes how fast the disease spreads when individuals interact. This is an incredibly simplified view, but it allows for straightforward modelling using the reaction equation

$$S + I \xrightarrow{\beta} 2I.$$

The number of suspectible individuals decreases at the same rate that the number of infected individuals increases. Given a non-zero initial number of infected individuals, this model will always tend towards the absorbing state of an entirely infected population, although stochastic realisations of the model will reach this state at different times. This inevitable saturation of the population limits how applicable this model is.

One extension of the SI model to make it more realistic is by adding in the potential for recovery. In real life individuals often successfully fight off an infection. In the SIS model, an individual that recovers from the infection moves back into the susceptible state, and so there is no infection immunity. There is also a new model parameter, the recovery rate γ, governing how quickly infected individuals recover and return to the susceptible state. The reaction equations become

$$S + I \xrightarrow{\beta} 2I,$$

which governs infection as for the SI model, and

$$I \xrightarrow{\gamma} S,$$

which governs recovery. The endemic state is when the rates at which individuals are infected and recover are exactly balanced, with the number of infected individuals $I > 0$ [37]. The disease-free state in which all individuals are susceptible is an absorbing state, but for large systems this state may be extremely unlikely to occur. There also exists an epidemic threshold that separates the evolution going towards either a state which will tend to this endemic state, or towards the disease-free state. Due to the lack of infection immunity, this model can be used to describe situations such as certain sexually transmitted infections, where reinfection is common [38].

However, many diseases do confer immunity, particularly those caused by viral agents such as mumps, measles and chickenpox [39]. Therefore an obvious alternative extension to the SI model is to include long-term recovery, where recovered individuals are then immune to reinfection. Infection recovery and subsequent immunity is captured in the SIR model. There are now three states: susceptible (S), infected (I) and recovered (R). The two possible processes are $S + I \rightarrow 2I$, infection of an individual, and $I \rightarrow R$, recovery of an individual. The reaction equations become

$$S + I \xrightarrow{\beta} 2I,$$

and

$$I \xrightarrow{\gamma} R.$$

Approximations of this process on a network also generally find an epidemic threshold [37].

There are numerous other variants on these models in epidemic research, such as the SEIR model. This model includes an exposed (E) state, which can be used to model the incubation period of an illness [40]. This model in particular has been used in research modelling the COVID-19 pandemic [41], where the delay between an individual being exposed to the illness and becoming contagious can be significant.

1.3.2 Social contagion

The contagion metaphor can also apply in a social context, for example, the epidemic-like outbreak of a meme going viral online. Sometimes, models such as the SI model and SIS model can be lifted and applied directly, as in the work by Kandhway and Kuri [42]. However, in general a few adjustments have to be made in order to account for specific features of social contagion, and a good summary is provided in Pastor-Satorras et al [15]. For example, the transmission of information generally involves an intentional act by the sender and receiver, and people won't 'recover' from the infection in the same sense. This calls for either the spreading process or the recovery process of the epidemic model to be changed.

Some popular rumour spread models are an adaptation of the SIR model, with the recovery process altered. In these models the recovered state corresponds to an individual who has heard the rumour but is no longer interested in spreading it, and the infected state corresponds to an individual interested in spreading the rumour. Daley and Kendall [43] first formalised the process, including three different interactions

$$S + I \xrightarrow{\beta} 2I,$$

$$R + I \xrightarrow{\gamma} 2R,$$

$$2I \xrightarrow{\gamma} 2R.$$

The final two interactions correspond to infected individuals losing interest in spreading the rumour further after they find out that the person they are gossiping with already knows it. In an otherwise identical model by Maki and Thompson [44] called the MT rumour model, the final process is instead

$$2I \xrightarrow{\gamma} R + I,$$

and says that when two infected individuals gossip with each other only one of the individuals (the one doing the "telling") loses interest.

Finally, it is worth noting that there also exist a number of models specifically created to model social contagion, which aren't built as directly on epidemic models. For example, the Bass model is a growth model focusing on the timing

of people's initial purchases of new consumer products [45]. The model assumes that people are either innovators or imitators, and that the speed at which they purchase a product is dependent on the extent to which they fit these roles. There is also an argument to be made that social contagion models should often be non-linear, in the sense that an individual should instead adopt an idea (or product, etc.) only once some minimum threshold of their neighbours have also adopted it [46]. These kinds of models are called threshold models. They are thought to occur in a range of situations, such as the diffusion of innovations [47] (where people may be wary of, for example, new medical advancements until they see enough of their peers engaging), rumour spreading [48] (where an individual might only find a rumour credible once they have heard it from enough people), and deciding who to vote for [49] (partly due to social influence, partly due to 'tactical voting' pressures).

1.4 Modelling dynamics on networks

Analysis of these dynamical models on networks has been developed using a number of different approaches, but perhaps the standard method is that of mean-field approximations [50], which focus on the probabilities of nodes being in different dynamical states while typically ignoring local clustering, modularity, and dynamical correlations [51]. This thesis will focus heavily on these mean-field approximations, and I will go into further detail about their justification, derivation and accuracy in Chapter 2. As we will see, these approximations can generally be used to reduce the state space of the problem and allow us to solve for any steady states, or identify further properties such as the epidemic threshold.

For completeness, note that there are alternative approaches to mean-field approximations. One such approach uses generating functions [52] to study models without steady-states, such as the SIR model. Solving the SIR dynamics for the late-time static properties can be translated into an equivalent bond percolation problem, as in a key paper by Newman [52]. Percolation problems essentially involve looking at the removal of nodes or edges from a system, which Newman explains parallels the removal of susceptible nodes during an SIR infection. He

defines various generating functions for distributions related to the degree distribution, and uses these to find the outbreak size distribution (or equivalently the size distribution of the connected component in the percolation problem). Further work has used the generating function approach on SIR-like dynamics to evaluate immunisation strategies [53] and containment measures [54, 55].

We can also get information about how the process will run on a network by looking at the spectra of the adjacency matrix and graph Laplacian. The Laplacian $L = D - A$ is formed by the difference between the degree matrix D (defined as the matrix that has the degree of each node along the diagonal, and zero elsewhere) and the adjacency matrix A. This spectral approach can be used to find an alternative, more accurate approximation for the epidemic threshold of SIS dynamics on any given network, by evaluating the eigenvalues of the Laplacian as in Wang et al [56]. They show that the condition for a stable disease-free state is

$$1 - \gamma + \beta\lambda_{1,A} < 1$$

where γ is the recovery parameter, β is the infection parameter, and $\lambda_{1,A}$ is the largest eigenvalue of the adjacency matrix. This gives us an expression for the epidemic threshold:

$$\frac{\beta}{\gamma} = \frac{1}{\lambda_{1,A}}.$$

Spectral methods have also given insight as to exactly where the epidemic is sustained if the epidemic threshold is reached. Goltsev et al [57] investigate whether the eigenvector associated with the largest adjacency matrix eigenvalue is localised. If the eigenvector is localised, then when the infection-recovery ratio just reaches the epidemic threshold the infection is localised to a finite number of vertices, and the actual fraction of nodes infected is negligibly small. As the ratio increases, the infection smoothly spreads to a finite fraction of vertices. If the eigenvector isn't localised, then the infection is sustained amongst a finite fraction of vertices as soon as the ratio reaches the threshold.

1.5 Thesis structure

As previously stated, in this thesis I will investigate both how to approximate and how to infer dynamics on networks. I have already referred to several different types of dynamical process, and several different ways of approximating these processes on networks. Chapters 2 and 3 will go into further detail, studying the error of some of these approximations both qualitatively and quantitatively. Chapter 2 will show specifically how to formulate the mean-field approach for SIS epidemic dynamics on networks. I also perform an investigation into the accuracy of various different mean-field approximations when compared to simulation results on a range of different networks, to see in which situations each of the different mean-field approximations are appropriate. I have chosen to focus on SIS dynamics and mean-field approximations for a number of reasons. Epidemic dynamics have gained particular relevance due to the recent COVID-19 pandemic, and the SIS model is one of the simplest epidemic models to study. My discussion of several different social contagion models has also shown how, despite its simplicity, the SIS model can provide the basis for numerous other models of interest in different fields. Since mean-field approximations are generally presented as the standard approach for these types of problems, our emphasis on these approximations throughout this thesis will give it relevance to a wide variety of network science research.

In Chapter 3, I present a novel coarse-graining of SIS dynamics on a network using a technique called approximate lumping. I will show how this technique can be applied to recover one of the mean-field approximations presented in Chapter 2, providing a novel justification for this approximation. This work aims to showcase the insights that this lumping approach can offer us. This chapter also constitutes the basis of an article that is currently in the process of being submitted to a peer reviewed journal.

Chapters 4 and 5 turn to the subject of inference in network science. Chapter 4 presents an overview of the work already done in this area, with a particular focus on the approaches used to infer the dynamics on a network when there is missing information: either an unknown dynamical process, or a known dynamical process and missing observations. This chapter aims to explain the advantages

and disadvantages of these different inference approaches, and also introduces several tools such as Bayesian Markov Chain Monte Carlo (MCMC) methods, and Approximate Bayesian Computation (ABC).

Several of these tools are then incorporated into the analysis performed in Chapter 5, where I look at a novel application of these ideas to a multilayer network system. This chapter provides an introduction to the concept of a multilayer network, before performing a case study on the problem of inferring SIS dynamics on a multilayer network where only one of the layers is visible. I look at simulated data of SIS dynamics on a two-layer network, and then use two different inference frameworks to try and recover the infection parameters when the information about the second 'hidden' layer is removed. This chapter aims to flag pitfalls that analysts confronted with a system like this need to be wary of. The work also aims to identify what specific situations will allow us to perform accurate inference, and in what situations we are forced to collect further information about the hidden layer before we can draw useful conclusions.

Finally, Chapter 6 concludes the thesis, providing both a summary of each chapter's findings, and a discussion of some potential industry applications related to my CASE partnership.

Chapter 2

Mean-field approximations for SIS dynamics on networks

2.1 Introduction

As indicated in the previous chapter, there are a range of different dynamical processes that can be applied to networks, across a wide variety of fields. I have also touched upon a number of different ways of solving these systems, both in terms of their formulation, and approximations that enable analytical tractability. In this chapter I will focus on SIS dynamics in particular, and explore in more detail how mean-field approximations can be used to solve the Markovian stochastic formulation of this dynamical model on networks.

The focus on SIS dynamics in this chapter leads into work in Chapters 3 and 5, both of which also involve SIS dynamics in some way. I've chosen to focus on SIS dynamics because it is arguably one of the simplest epidemic models, while offering slightly more complexity than the SI model due to the presence of a meaningful endemic steady state. In this way SIS dynamics gives us a fairly simple and relevant framework within which to present some novel ideas on approximate lumping (Chapter 3) and multilayer networks (Chapter 5). We might also expect that some of the approximations and methods used to analyse SIS dynamics throughout this thesis can be extended to more complex epidemic models, such as the previously mentioned SIR and SEIR models, as well as social infection models.

The SIS model also notably contains both an absorbing disease-free state, and a quasi-stationary endemic state, which will be studied later in this section. We can eliminate the absorbing state by simply adding a constant background infection rate, dynamics referred to here as the SISa model [58]. This adaptability will prove useful in Chapter 3, where it will allow me to easily consider how my analysis changes for situations with or without an absorbing state.

I've chosen to study mean-field approximations because they offer significant analytical tractability, as I will demonstrate during this chapter. While they will prove to be inaccurate under certain conditions, the ability to derive threshold conditions for the endemic state, and quickly explore the solution space to see how the dynamical model parameters influence the solution, will prove useful in predicting interesting results in Chapter 5. I will also show how approximate lumping can be used to estimate the error of some of these approximations in Chapter 3.

In this chapter I will first introduce the deterministic compartmental model for SIS dynamics in Section 2.2. In Section 2.3 I detail the formulation of SIS dynamics on a network of contacts, and I demonstrate the need for ways to simulate and approximate these dynamics beyond the exact formulation. Sections 2.4 focuses on simulation methods, while in Sections 2.5 and 2.6 I outline the derivations and analytical results for a range of different mean-field approximations. Finally, in Section 2.7 I investigate how the results of these mean-field approximations compare to simulation results for various different networks and parameter values.

Overall, this chapter provides an overview of the standard mean-field models describing SIS dynamics, setting up the principles behind these models and detailing the basic results in anticipation of future chapters. The homogeneous mean-field model at the single level will reappear in Chapter 3, where I derive it using novel arguments that are arguably more rigorous than the physical reasoning presented in this chapter. The results of the final section, looking at the accuracy of the various mean-field approximations, will also help guide work in Chapter 5, where I study a homogeneous mean-field model of SIS dynamics on a multilayer system.

2.2 Deterministic SIS compartmental model

First we construct the deterministic compartmental model of the SIS process, which ignores any network structure. Individuals can be in one of two states or 'compartments': susceptible (S) or infected (I). There are two processes, infection and recovery, governing the transitions of individuals between compartments. These possible transitions are illustrated in Figure 2.1. We assume that individuals meet each other at some fixed rate, and that the rate of infection in the population is dependent on the probability of a given meeting involving a susceptible individual and an infected individual. Given a well-mixed population, this probability, and therefore the rate of infection, is proportional to the number of individuals in the infected compartment and the number of individuals in the susceptible compartment. Since individuals recover independently of their interactions with other individuals, we assume that the rate of recovery in the population is proportional to simply the number of individuals in the infected compartment. If we also denote the number of susceptible and infected individuals by S and I respectively (it will be clear based on context whether I mean the number of individuals in a state, or the state itself), this leads to the governing equations:

$$\dot{S} = -\beta \frac{S}{N} I + \gamma I$$
$$\dot{I} = \beta \frac{S}{N} I - \gamma I.$$

The constants of proportionality β and γ are known as the infection rate and recovery rate respectively. We can think of β as the rate at which infected individuals make contact with others in a way that could transmit infection.

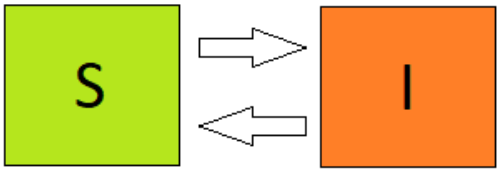


Figure 2.1: Compartmental model for SIS, showing the possible transitions from the susceptible compartment (S) to the infected compartment (I), and from I to S.

We can see from the conservation identity $S + I = N$ that this system can be reduced to a single equation, describing the time evolution of the number of infected individuals. This equation can be solved directly, using the substitution $y = I^{-1}$ [39]. We find that there are two steady state solutions: a disease-free state with $I = 0$, and an endemic state with $I = N\left(1 - \frac{\gamma}{\beta}\right)$. This latter state has physical significance only if $\frac{\beta}{\gamma} > 1$. We call this ratio the 'basic reproductive ratio', denoted by R_0. R_0 can be thought of here as the ratio of infections to recoveries that happen in a short time interval. If $R_0 = \frac{\beta}{\gamma} > 1$, the endemic steady state has physical meaning and is stable, while the disease-free steady state is unstable. For $R_0 < 1$, the endemic steady state no longer has physical meaning (and is unstable), and the disease-free steady state is stable. At $R_0 = 1$ a transcritical bifurcation occurs, where the stability of the fixed points is exchanged. In this way the value of R_0 determines the epidemic threshold of the system. We can also define this point using the critical value of the infection parameter $\beta_c = \gamma$.

Although this model gives simple analytical results, the assumption of a fully-mixed population is extremely crude. Often we have access to information about the way a population is connected, and the next section considers how to incorporate that information into the dynamics.

2.3 Exact stochastic formulation of SIS dynamics on networks

We now consider a stochastic formulation of the process, and account for an explicit network of contacts between the individuals. Consider initially general dynamics on a network with N vertices, where each vertex can be in one of M vertex-states $\{\Sigma_1, \Sigma_2, ..., \Sigma_M\}$. A given state s_i of the system can be described by a vector of vertex-states $s_i = (\sigma_1, \sigma_2, ..., \sigma_N)$, where the vertex-state for a given vertex $\sigma_v \in \{\Sigma_1, \Sigma_2, ..., \Sigma_M\}$. In this way the state space of the dynamical system consists of $n = M^N$ states, $\mathcal{S} = \{s_1, s_2..., s_n\}$. The transition rate between two states s_i and s_j can be expressed as $h(s_i, s_j)$, with $h(s_i, s_i) = -\sum_{j \neq i} h(s_i, s_j)$ so that $-h(s_i, s_i)$ is the rate at which the system transitions out of state s_i to any other state. We make a number of assumptions here. We consider time to be continuous, and assume that the process is stochastic, with exponentially distributed inter-event times. We also consider transitions at different nodes to be independent events. These assumptions allow us to structure the problem as a continuous-time Markov chain as follows.

Let $X_i(t)$ be the probability that the system is in state s_i at time t, i.e. $X_i(t) = P(s(t) = s_i)$. Then the vector $\mathbf{X}(t) = (X_1(t), X_2(t), ...X_{M^N}(t))$ forms the time-dependent Markov chain probability distribution over the state-space $\mathcal{S}$. Using the transition probabilities above, the probability of transitioning from s_i to s_j in a short time interval $(t, t + \delta t)$ is $h(s_i, s_j)\delta t + o(\delta t)$, where the notation $o(\delta t)$ represents an error that behaves as $o(\delta t)/(\delta t) \to 0$ as δt is decreased to 0 [59]. The probability of the transition not occurring in the same interval is likewise $1 - h(s_i, s_j)\delta t + o(\delta t)$. Therefore, if the system starts in state s_i at time t,

$$P(s(t + \delta t) = s_j \mid s(t) = s_i) = h(s_i, s_j)\delta t + o(\delta t).$$

Using the law of total probability, we find

$$\begin{aligned} X_j(t+\delta t) &= P(s(t+\delta t) = s_j) \\ &= \sum_{i=1}^{n} P(s(t+\delta t) = s_j \mid s(t) = s_i) P(s(t) = s_i) \\ &= \left(\sum_{i \neq j} h(s_i, s_j) \delta t X_i(t) \right) + (1 - \sum_{k \neq j} h(s_j, s_k) \delta t) X_j(t) + o(\delta t) \\ &= \left(\sum_{i \neq j} h(s_i, s_j) \delta t X_i(t) \right) + (1 + h(s_j, s_j) \delta t) X_j(t) + o(\delta t). \end{aligned}$$

If we subtract $X_j(t)$, divide by the time interval δt and take $\delta t \to 0$, we find

$$\dot{X}_j(t) = \left(\sum_{i=1}^{n} h(s_i, s_j) X_i(t) \right) + h(s_j, s_j) X_j(t).$$

Therefore the master equation describing the time evolution of $\mathbf{X}(t)$ is given by

$$\dot{\mathbf{X}} = \mathbf{Q}^{\mathrm{T}} \mathbf{X},$$

where $\mathbf{Q}$ is the infinitesimal generator formed by the instantaneous transition rates, with off-diagonal elements $\mathbf{Q}_{ij} = h(s_i, s_j)$ and diagonal elements $\mathbf{Q}_{ii} = h(s_i, s_i) = -\sum_{k \neq i} h(s_i, s_k)$ such that each row sums to zero.

The SIS model of epidemics is an example of a more general class of dynamical models defined by Ward and López-García as single-vertex transition (SVT) models [60]. In a SVT model, the transition rate $h(s_i, s_j)$ between any two states $s_i = (\sigma_1, \sigma_2, ...\sigma_N)$ and $s_j = (\xi_1, \xi_2, ...\xi_N)$ is non-zero only if the vertex-states are different for at most one vertex v: if $\sigma_v \neq \xi_v$ and $\sigma_u = \xi_u$ for all $u \neq v$. Kiss et al. actually restrict to a smaller subclass of models in their work [59]: they consider SVTs where the rate at which a vertex transitions to another vertex-state is the same for all vertices that are in the same state and which have the same number of neighbours in each state. If we let $\mathbf{n}(v) = (n_{\Sigma_1}, n_{\Sigma_2}, ...n_{\Sigma_M})$ represent the number of neighbours vertex v has in each vertex-state, then we can define $f_{A,B}(\mathbf{n}(v))$ as the rate at which vertex v transitions from a state A to a state B. Therefore the transition rates $h(s_i, s_j)$ satisfy

$$h(s_i, s_j) = \begin{cases} f_{A,B}(n_{\Sigma_1}, n_{\Sigma_2}, ...n_{\Sigma_M}) & \text{if } s_i,\ s_j \text{ differ in one vertex-state } v \\ 0 & \text{otherwise.} \end{cases},$$

For these SVT models, it can be helpful to define the classes or levels of the system. Each level $\mathcal{C}^{k_{\Sigma_1},k_{\Sigma_2},...k_{\Sigma_M}}$ is the subset of states that have $k_{\Sigma_1}, k_{\Sigma_2}, ...k_{\Sigma_M}$ vertices in vertex-states $\Sigma_1, \Sigma_2, ...\Sigma_M$ respectively.

For SIS dynamics, we have just two node-states: S for susceptible, and I for infected. There are likewise two types of node-state transition that can occur for a node, from $S \to I$ and from $I \to S$. The rate at which an infected node v recovers depends solely on the recovery rate γ. In this way the transition probability $h(s_i, s_j) = \gamma$ if s_i is identical to s_j except for node v which recovers from infection. However the rate at which a susceptible node v becomes infected depends both on the infection rate τ, and on $n_I(v)$, the number of infected neighbours node v has in the current state. The infection rate τ here is more precisely the per-contact infection rate, the rate at which transmission occurs across an edge between an infected node and a susceptible node. Therefore the transition probability $h(s_i, s_j) = \tau n_I(v)$ if s_i is identical to s_j except for node v, which has $n_I(v)$ infected neighbours in state s_α and which becomes infected. Since we can deduce the number of susceptible nodes from the number of infected nodes, the levels of the system can be written as $\mathcal{C}^{k_I}$ with just a single index k_I.

Since there are two node-states, the infinitesimal generator $\mathbf{Q}$ for this system is an 2^N by 2^N matrix, and equivalently there are 2^N linear differential equations in the system of master equations. Due to this exponential scaling, the dynamics quickly become intractable on networks with even a modest number of vertices. This motivates the use of methods and approximations to reduce the state space to a more feasible size.

2.4 Simulations of SIS dynamics

In anticipation of these equations becoming unwieldy for larger networks, it is helpful to develop an algorithm to instead simulate the dynamical system. We can then run a large ensemble of these simulations and analyse the ensemble average and spread. Performing these simulations for smaller networks, where we can still solve the full system of master equations, will allow us some insight into how the simulation results compare with the true solution.

To perform simulations of SIS dynamics on networks (and later on multilayer networks in Chapter 5), I decided to use the Gillespie algorithm. I first present the general form of the algorithm, before presenting my results of applying this algorithm to SIS dynamics.

2.4.1 Gillespie algorithm

The Gillespie algorithm is a general algorithm for simulating Markovian stochastic processes involving reaction-dependent populations [61]. The method involves generating two random numbers uniformly between 0 and 1, which determine at what time the next event will happen, and which event will happen at that time. This lets the stochastic process be modelled as a continuous-time process, and avoids simulating time steps where nothing happens. Due to this, the simulation time tends to be faster than for a step-by-step discrete-time simulation. The algorithm steps are presented below for a general network process:

1. Generate two random numbers r_1, r_2, picked from a uniform distribution on $[0, 1]$.

2. Compute the propensity function $\alpha_i^m(x)$ for each node i and event m. The propensity function is such that the probability that event m will happen to node i in the time interval dt, given the current state x of the entire network, is $\alpha_i^m(x)dt$ [62]. From this we can compute the total propensity for the system:
$$\alpha_0 = \sum_{m=1}^{M}\sum_{i=1}^{N}\alpha_i^m(x).$$

3. Using the fact that the inter-event times are exponentially distributed for Markovian stochastic processes, compute the time until the next event takes place as
$$\tau = \frac{1}{\alpha_0}\log\frac{1}{r_1}.$$

4. Compute which event takes place at $t + \tau$, i.e find k, j such that
$$r_2 \geq \frac{1}{\alpha_0}\sum_{m=1}^{k-1}\sum_{i=1}^{j-1}\alpha_i^m(x) \text{ and } r_2 \leq \frac{1}{\alpha_0}\sum_{m=1}^{k}\sum_{i=1}^{j}\alpha_i^m(x).$$

5. Update the node states and propensities accordingly.

6. Repeat from step 1, starting at time $t + \tau$.

To apply this algorithm to SIS dynamics, we note that there are two types of node events: infection events and recovery events. The propensity function for a node i being infected is $\alpha_i^I(x) = \tau n_I$, where τ is the infection rate and n_I is the number of infected neighbours of node i. The propensity function for a node i recovering is $\alpha_i^R(x) = \gamma$, where γ is the recovery rate. In order to efficiently calculate the propensities at each time step, I have designed the algorithm so that I store the number of infected neighbours of each node. With each node that is either infected or recovers, the 'infected neighbour' counts for the node's neighbours are updated. The algorithm terminates when either the total number of infected nodes reaches zero (in which case the infection can no longer spread), or the time exceeds some input time.

2.4.2 Simulated results

I ran these simulations on an Erdős-Rényi network of $N = 15$ nodes ($p = \frac{4}{15}$), with $\tau = 0.8$ and $\gamma = 1$. I ran an ensemble of 1000 simulations, and each simulation began with the entire network initially infected. I also found the exact expected solution by solving the full system of Kolmogorov equations, using the 'ode' function in the R *deSolve* package with its default settings. The default method is an LSODA integrator that can switch between stiff and non-stiff methods. The default absolute and relative tolerances are 1×10^{-6}. I chose a network with $N = 15$ nodes because this was the largest network where the Kolmogorov equations were tractable.

Figure 2.2 shows the ensemble of simulations plotted as a heat map. The thick grey line shows the ensemble average, and the black line shows the exact expected solution. Here it is easy to see that the simulated solutions each follow a general behaviour: they fluctuate around some quasi-stationary state (close to $I = 10$ infected nodes), before the stochastic nature of the process means they inevitably fall into the absorbing state of $I = 0$ infected nodes. Eventually, all the simulations will reach the absorbing state.

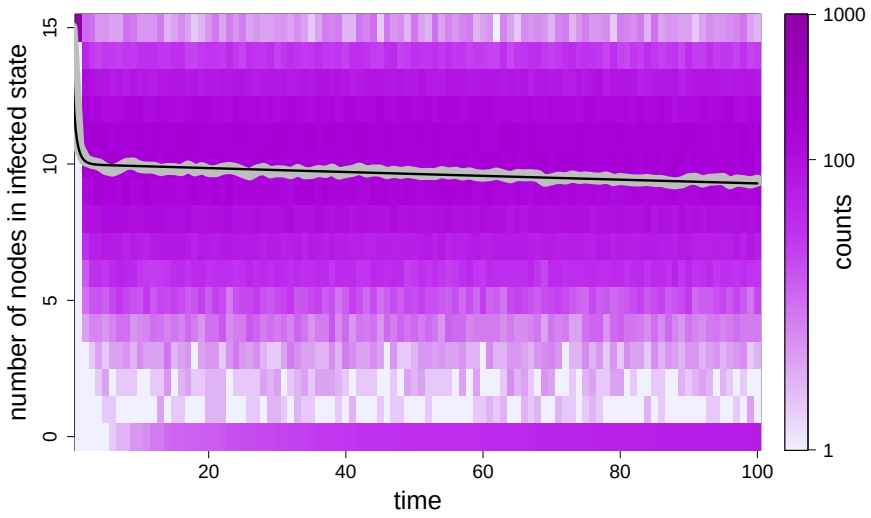
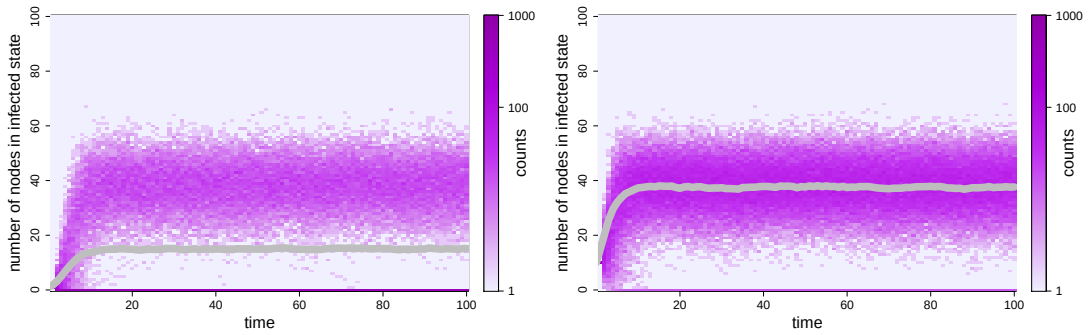


Figure 2.2: A heatmap showing the results from an ensemble of 1000 simulations of SIS dynamics on an Erdős-Rényi network of $N = 15$ nodes, starting with the entire network initially infected. The heatmap shows how each of the 1000 simulations evolved until the time $t = 100$. The counts are plotted on a log scale, where I have added 1 to the number of counts before taking logs (to avoid infinities). The darker cells around $I = 10$ infected nodes shows how the simulations tend to cluster around this steady state value. As time progresses, more of the solutions get absorbed by the disease-free steady state, and so the cells representing $I = 0$ infected nodes gradually get darker. The thick grey line shows the ensemble average. The black line shows the exact solution that is computed by solving the full system of Kolmogorov equations.

The presence of this absorbing state creates some questions around how we should seed the infection. If we seed the infection with only a few infected nodes, we can sometimes generate an ensemble that is divided into two outcomes: those where the infection quickly dies out in the first few events, and those where the infection survives and reaches the steady state value. To demonstrate this, I simulated an ensemble of 1000 runs on an Erdős-Rényi network of 100 nodes ($p = 0.4$), running the simulations until $t = 100$. The parameters were $\tau = 0.04$ and $\gamma = 1$. Note that in this case the network is too large to compute the exact solution from the master equations. Initially I seeded the simulations with just 1

infected node. The heatmap in Figure 2.3(a) shows the results of this simulation ensemble. The darker spots around $I = 40$ show the runs that reached the steady state, while the dark line at $I = 0$ show the runs that were absorbed into the disease-free state. The thick grey line shows the ensemble average. In this case the ensemble average fails to capture either of the two outcomes. It is also computationally wasteful to be running a large number of simulations that simply die out.

However, this divide in simulation outcomes is avoided if we seed the infection with a larger number of initially infected nodes. The heatmap in Figure 2.3(b) shows the results of a simulation ensemble on the same network, with the same infection parameters, but with 10 initially infected nodes rather than 1. There are 1000 simulations in the ensemble, and for each simulation the 10 initially infected nodes are chosen at random. Here far fewer simulations are absorbed into the steady state before $t = 100$, and the ensemble average tends to a state more closely matching the stationary state. Therefore if we want the ensemble average to be informative with regards to the stationary state value, it is important to seed the simulations with a sufficiently large number of infected nodes.



(a) 1 initially infected seed node

(b) 10 initially infected seed nodes

Figure 2.3: Two heatmaps showing the results from simulation ensembles of SIS dynamics on an Erdős-Rényi graph of $N = 100$ nodes. The counts are plotted on a log scale, where I have added 1 to the number of counts before taking logs (to avoid infinities). In (a), the epidemic is seeded with just 1 infected node. We can see that in a large number of solutions the epidemic dies out straight away, represented by the dark $I = 0$ cells. A significant number of solutions see the number of infected nodes rising to the steady state value, around $I = 40$, represented by the darker cells surrounding this value. The thick grey line shows the ensemble average, which notably here does not meaningfully capture either of these two types of solution. In (b), the epidemic is seeded instead with 10 infected nodes. Here the $I = 0$ cells are slightly lighter than in (a), showing how in this case far fewer of the solutions see the epidemic dying out quickly. The majority of solutions see the number of infected nodes rising to the steady state value. The thick grey line shows the ensemble average, which this time captures the steady state value well.

As we can see by the close agreement of the simulated ensemble average and the exact solution in Figure 2.2, the mean behaviour of the ensemble is accurate. However, generating a large enough ensemble to reliably find this expected behaviour can become computationally demanding for larger networks. As such, a variety of different approximations have been developed, which describe the system at a coarser-grain level than the exact master equations. These will inevitably be less accurate than the exact system, but the tractability they offer makes them useful nonetheless, and in many cases they still offer a useful in-

sight into the system behaviour. The following two sections 2.5 and 2.6 introduce first homogeneous mean-field approximations to SIS dynamics, and then a number of alternative approximations, before I compare the results of applying these approximations with simulation results in Section 2.7.

2.5 Homogeneous mean-field approximations

In this section I will detail several homogeneous mean-field approximations. They are so-called 'homogeneous' because they involve the assumption that each node's degree can be well-approximated by the mean degree. These homogeneous mean-field approximations in particular will resurface later in the thesis: first in Chapter 3, where we find that rates involved in the homogeneous mean-field at the single level can be derived directly from the full continuous-time Markov chain formulation, and then in Chapter 5, where we use a homogeneous mean-field approximation at the single level to guide an investigation into multilayer SIS dynamics. This section therefore acts as an introduction to these approximations in anticipation of their derivation in Chapter 3, and provides justification for their use in Chapter 5. The choice to use the homogeneous mean-field approximation, rather than any alternative approximation, will be further justified in the investigation of results in Section 2.7.

The goal of the mean-field construction is to find a population-level description of the system. In other words, we would like to track the number of susceptible and infected nodes in the system at each time. Following the derivation presented by Simon et al [63], we can construct this population-level description by returning to the continuous-time Markov chain description of the system, and ordering the states by ascending number of infected nodes. This then leads to a block-diagonal infinitesimal generator Q of the form

$$Q = \begin{pmatrix} B^0 & A^0 & 0 & 0 & 0 & 0 \\ C^1 & B^1 & A^1 & 0 & 0 & 0 \\ 0 & C^2 & B^2 & A^2 & 0 & 0 \\ 0 & 0 & \ddots & \ddots & \ddots & 0 \\ 0 & 0 & 0 & C^{N-1} & B^{N-1} & A^{N-1} \\ 0 & 0 & 0 & 0 & C^N & B^N \end{pmatrix}. \tag{2.1}$$

We also divide the state vector X into sub-vectors $X = (X^0, X^1, ...X^N)$, where X^k is the sub-vector whose entries all correspond to the states with k infected nodes belonging to the class $\mathcal{C}^k$. We also denote a state belonging to the class $\mathcal{C}^k$ as s^k. Structuring the dynamics in this way leads to a master equation of the form

$$\dot{X}^k = A^k X^{k-1} + B^k X^k + C^k X^{k+1},$$

where the A^k matrices capture the transition of a node in the system from S to I, the C^k matrices capture the transition of a node from I to S, and the B^k matrices account for the rate of $s_i^k \to s_j^k$ transitions. If we sum the elements in the jth column of matrix A^k, this is the equivalent to summing over all possible infection transitions out of the state s_j^{k-1}. The probability of a susceptible node is a function of the number of its infected neighbours, which can also be thought of as the number of (S, I) edges that are connected to the node. Therefore the A^k matrices satisfy the identities

$$\sum_{i=1}^{c_k} A_{i,j}^k = \tau N_{SI}(s_j^{k-1}), k = 0, 1, ..., N$$

where $N_{SI}(s_j^{k-1})$ is the total number of (S, I) edges in state s_j^{k-1}. If we perform the same sum over the elements in the jth column of the matrix C^k, this time we are effectively summing over all possible recovery transitions out of the state s_j^{k+1}. In this state we have $(k+1)$ infected nodes, and so the C^k matrices satisfy the identities

$$\sum_{i=1}^{c_k} C_{i,j}^k = \gamma(k+1), k = 0, 1, ..., N.$$

We can also note that the rate of transition from s_i^k to s_j^k is zero if $s_i \neq s_j$. In this way B^k is a diagonal matrix, where the diagonal elements

$$B_{i,i}^k = -\sum_{j=1}^{c_{k+1}} A_{j,i}^{k+1} - \sum_{j=1}^{c_{k-1}} C_{j,i}^{k-1}$$

because the sum of the entries in any column is 0. Since we are interested in the population-level counts, it is worth expressing the expected number of infected

and susceptible nodes in terms of the Markov chain vector. We can see that, for the expected number of infected nodes:

$$[I](t) = \sum_{k=0}^{N} \sum_{j=1}^{c_k} k X_j^k(t) = \sum_{k=0}^{N} k e_k X^k,$$

where e_k is a row vector of ones with c_k entries. Likewise, for the expected number of susceptible nodes:

$$[S](t) = \sum_{k=0}^{N} \sum_{j=1}^{c_k} (N-k) X_j^k(t) = \sum_{k=0}^{N} (N-k) e_k X^k.$$

Finally, we can express the expected number of SI edges (the number of edges incident to both a susceptible node and an infected node):

$$[SI](t) = \sum_{k=0}^{N} \sum_{j=1}^{c_k} N_{SI}(s_j^k) X_j^k(t).$$

Since we are interested in the time evolution of the system, we look at the time derivative of the expected number of infected nodes, and we obtain

$$\begin{aligned}
[\dot{I}] &= \sum_{k=0}^{N} k e_k \dot{X}^k && (2.2)\\
&= \sum_{k=0}^{N} k e_k (A^k X^{k-1} + B^k X^k + C^k X^{k+1}) && (2.3)\\
&= \sum_{k=1}^{N} k e_k A^k X^{k-1} + \sum_{k=0}^{N} k e_k B^k X^k + \sum_{k=0}^{N-1} k e_k C^k X^{k+1} && (2.4)\\
&= \sum_{k=0}^{N-1} (k+1) e_{k+1} A^{k+1} X^k + \sum_{k=0}^{N} k e_k B^k X^k + \sum_{k=1}^{N} (k-1) e_{k-1} C^{k-1} X^k && (2.5)\\
&= \sum_{k=0}^{N} \left((k+1) e_{k+1} A^{k+1} + k e_k B^k + (k-1) e_{k-1} C^{k-1} \right) X^k. && (2.6)
\end{aligned}$$

It can be shown that

$$e_{k+1} A^{k+1} + e_k B^k + e_{k-1} C^{k-1} = 0$$

for all $k = 0, 1, ..., N$. Using this identity, we can simplify the above expression for $[\dot{I}]$ and we find

$$[\dot{I}] = \sum_{k=0}^{N} \left(e_{k+1} A^{k+1} - e_{k-1} C^{k-1} \right) X^k.$$

Then, by using

$$\left(e_{k-1} C^{k-1} \right)_j = \sum_{i=1}^{c_{k-1}} C_{i,j}^{k-1} = \gamma k,$$

which implies $e_{k-1} C^{k-1} = \gamma k e_k$, we find

$$\sum_{k=0}^{N} e_{k-1} C^{k-1} X^k = \sum_{k=0}^{N} \gamma e_k X^k = \gamma [I].$$

Similarly, by using

$$\left(e_{k+1} A^{k+1} \right)_j = \sum_{i=1}^{c_{k+1}} A_{i,j}^{k+1} = \tau N_{SI}(s_j^k),$$

we find

$$\sum_{k=0}^{N} e_{k+1} A^{k+1} X^k = \sum_{k=0}^{N} \sum_{j=1}^{c_k} \left(e_{k+1} A^{k+1} \right)_j X_j^k = \tau \sum_{k=0}^{N} \sum_{j=1}^{c_k} N_{SI}(s_j^k) X_j^k(t) = \tau [SI].$$

Putting this together, we have the exact mean-field equations:

$$[\dot{I}] = \tau [SI] - \gamma [I]$$
$$[\dot{S}] = \gamma [I] - \tau [SI],$$

where we have found the equivalent equation for $[\dot{S}]$ by making use of the fact that $[\dot{S}] + [\dot{I}] = 0$. We can similarly derive the following differential equations for the various edge pairings [64]:

$$[\dot{SI}] = \gamma([II] - [SI]) + \tau([SSI] - [ISI] - [SI]),$$
$$[\dot{SS}] = 2\gamma[SI] - 2\tau[SSI],$$
$$[\dot{II}] = -2\gamma[II] + 2\tau([ISI] + [SI]).$$

To solve these equations in their exact form, we theoretically need to find equations governing the time evolution of the triplets $[SSI]$ and $[ISI]$. These equations

would in turn involve terms involving four node-states, and the process continues until we have effectively rewritten the full master equations. In order to reduce the state space and make the mean-field approximations useful, we can apply closures to the mean-field equations [65]. Closure approaches all involve expressing higher order structures in terms of lower order ones, either exactly or approximately. These approaches rely on the assumption that, at some scale, certain variables can be treated as independent. In the following subsections I will show how we can apply closures to the exact SIS mean-field model, at two different levels: the first involving an approximation for the expected number of pairs $[SI]$, $[II]$ and $[SS]$ in terms of the singles $[S]$ and $[I]$, and the second involving an approximation for the expected number of triples $[SSI]$ and $[ISI]$ in terms of the pairs. Following the convention of Kiss, Miller and Simon [59], we call these the closures at the pair and triple levels, leading to mean-field approximations at the single level and pairwise level respectively.

2.5.1 Closure at the pair level

In order to make a closure at the pair level and express $[SI]$ in terms of $[S]$ and $[I]$, we make a number of assumptions. We first assume that the network is homogeneous: that is, each node has the same number of neighbours n. We also assume that infected nodes are distributed randomly. This means that if there are I infected nodes making up $\frac{[I]}{N}$ of the population, an average susceptible node will have $\frac{n[I]}{N}$ infected neighbours. Using this, the total number of SI edges is approximated by

$$[SI] \approx \frac{n}{N}[S][I].$$

In reality infected nodes are more likely to be in contact with other infected nodes due to how the infection propagates, and in general the network will not be homogeneous. Using this approximation for $[SI]$ will therefore result in an inexact closed system of equations, and the values of $[S]$ and $[I]$ found by solving the closed system will only be approximations of the true expected values. The discrepancies between these values will vary based on the network properties and the dynamical parameters, and I investigate this in Section 2.7. However, making

this closure allows us to limit the number of equations which describe the system. Starting with the exact system,

$$[\dot{I}] = \tau[SI] - \gamma[I]$$
$$[\dot{S}] = \gamma[I] - \tau[SI],$$

and making the closure at pair level, these equations become

$$[\dot{S}] = -\tau\frac{n}{N}[S][I] + \gamma[I]$$
$$[\dot{I}] = \tau\frac{n}{N}[S][I] - \gamma[I].$$

As stated above, we refer to this system as the homogeneous mean-field at the single level. This is essentially a stochastic equivalent of the deterministic compartmental model described earlier, with β replaced by $n\tau$. These equations can be easily solved, and the results mirror those for the compartmental model: the system has two steady states at $[I] = 0$, where the disease dies out entirely, and $[I] = N(1 - \frac{\gamma}{n\tau})$, where the system reaches an endemic state. The latter solution is only physical for $\gamma < n\tau$. Stability analysis reveals that for $\gamma > n\tau$ the disease-free equilibrium point is stable, and for $\gamma < n\tau$ the endemic equilibrium point is stable. There is therefore a transcritical bifurcation at the epidemic threshold $\gamma = n\tau$, at the critical value $\tau_c = \frac{\gamma}{n}$.

It is useful here to return to the concept of the basic reproduction ratio R_0. In the stochastic context, the reproduction ratio is defined as the average number of secondary infections caused by a primary infection introduced in a fully susceptible population [66]. Given that an infected node will be infected for an average time of $\frac{1}{\gamma}$, and the rate of transmission from one infected node in a susceptible population will be $n\tau$, this means that

$$R_0 = \frac{n\tau}{\gamma}.$$

The epidemic threshold can then be defined in terms of the reproduction ratio. If $R_0 > 1$ then there exists an endemic steady state. If $R_0 < 1$ then the disease-free steady state is the only physical steady state in existence.

2.5.2 Closure at the triple level

To potentially improve the accuracy of the approximation, we can instead apply a closure at the triple level. To do this we need to find approximations for the expected number of triples $[SSI]$ and $[ISI]$ in terms of singles and pairs. Again, we can use physical reasoning to find sensible approximations. Considering still a homogeneous system with each node having degree n, the number of edges starting from susceptible nodes is $n[S]$. Given a total number of SI edges $[SI]$, then a proportion $[SI]/n[S]$ of the edges starting from susceptible nodes will lead to infected nodes. Using the same reasoning, a proportion $[SS]/n[S]$ will lead to susceptible nodes. Therefore if we choose a susceptible node u and two of its neighbours v and w, the probability of the nodes forming a $[SSI]$ triplet is $[SS][SI]/n^2[S]^2$. Since there are $n(n-1)$ ways we can choose the neighbours v and w, we can conclude that

$$[SSI] \approx \frac{n-1}{n}\frac{[SS][SI]}{[S]}.$$

We can apply equivalent analysis to find

$$[ISI] \approx \frac{n-1}{n}\frac{[SI]^2}{[S]}.$$

This approximation again does not fully account for correlations between the node-states of neighbouring nodes: a previously infected, recently recovered node is more likely to have infected neighbours than other susceptible nodes. However, it again allows us to reduce the system of equations down to a tractable state space:

$$\begin{aligned}
[\dot{S}] &= -\tau[SI] + \gamma[I] \\
[\dot{I}] &= \tau[SI] - \gamma[I] \\
[\dot{SI}] &= \gamma([II] - [SI]) + \tau\frac{n-1}{n}\frac{[SI]([SS] - [SI])}{[S]} - \tau[SI] \\
[\dot{SS}] &= 2\gamma[SI] - 2\tau\frac{n-1}{n}\frac{[SI][SS]}{[S]} \\
[\dot{II}] &= -2\gamma[SI] + 2\tau\frac{n-1}{n}\frac{[SI]^2}{[S]} + 2\tau[SI].
\end{aligned}$$

We can actually reduce this system even further by recognising that the following conservation relations hold:

$$[S] + [I] = N$$
$$2[SI] + [SS] + [II] = nN$$
$$[SS] + [SI] = n[S]$$
$$[SI] + [II] = n[I].$$

Using these, the system simplifies down to a two-dimensional system:

$$[\dot{S}] = \gamma N - (\gamma + n\tau)[S] + \tau[SS]$$
$$[\dot{SS}] = 2(n[S] - [SS])\big(\gamma - \tau(n-1)\frac{[SS]}{n[S]}\big).$$

We can solve these equations, again revealing a disease-free steady state and an endemic steady state. The disease-free state $[I] = 0, [SS] = nN$ is stable for $\tau(n-1) < \gamma$, and the endemic steady state $[S] = N\frac{\gamma(n-1)}{\tau n(n-1)-\gamma}, [SS] = \frac{\gamma n}{\tau(n-1)}[S]$ is stable for $\tau(n-1) > \gamma$.

We can label the single level approximation with the subscript '1' (indicating the first approximation), and the pair level approximation with the subscript '2' (indicating the second approximation). It can be shown (as in Kiss, Miller and Simon [59]), that the following inequalities hold:

$$\frac{n}{N}[S]_2[I]_2 > [SI]_2, [SS]_2 > \frac{n}{N}[S]_1^2, [II]_2 > \frac{n}{N}[I]_2^2.$$

These can be used to further show that $[I]_2 < [I]_1$, i.e. the single level approximation of the infection prevalence provides an upper bound for the pair level approximation. We will see confirmation of this in Section 2.7, where I will explore the accuracy of these mean-field approximations on a range of different networks.

2.6 Alternative approximations

There are a large number of more sophisticated approximations, which try to capture a wider range of system behaviours. In this section I will briefly summarise some of the more commonly used approximations, discuss the assumptions inherent in their formulation, and present the resulting system equations for each.

2.6.1 Heterogeneous mean-field approximations

The heterogeneous mean-field is formulated similarly to the homogeneous case, but instead of focussing on the expected number of infected (and susceptible) nodes, we focus on the expected number of infected (and susceptible) nodes that have degree k, for $k = 1, 2, ..., k_{\max}$, where $k_{\max}$ is the maximum degree. We can also introduce N_k, the number of nodes with degree k, and so $N_1 + N_2 + ... + N_M = N$. This allows us to relax the assumption that every node has the same degree, and therefore we can account for a heterogeneous degree distribution. The system of master equations becomes

$$[\dot{S}_k] = \gamma[I_k] - \tau[S_k I]$$
$$[\dot{I}_k] = \tau[S_k I] - \gamma[I_k]$$

for $k = 1, 2, ..., k_{\max}$. We can also derive equations for the time evolution of the pairs $[S_k I_l]$, $[S_k S_l]$ and $[I_k I_l]$. As with the homogeneous mean-field approximation, we need to apply a closure to this system in order to make it useful. We start with a closure at the pair level. Heuristic reasoning gives us that

$$[S_k I] \approx k[S_k]\pi_I,$$

where $k[S_k]$ is the expected number of edges leaving susceptible nodes of degree k, and $\pi_I = \sum_{l=1}^{k_{\max}} l[I_l] / \sum_{l=1}^{k_{\max}} l N_l$ is the probability that a randomly chosen edge leaving any node is connected to an infected node. This allows us to write the system using $2k_{\max}$ equations:

$$[\dot{S}_k] = \gamma[I_k] - \tau k[S_k]\pi_I$$
$$[\dot{I}_k] = \tau k[S_k]\pi_I - \gamma[I_k]$$
$$\pi_I = \sum_{l=1}^{k_{\max}} l[I_l] / \sum_{l=1}^{k_{\max}} l N_l.$$

This closure assumes the "annealed network" assumption of random neighbour selection: in other words, it assumes that the neighbours of all nodes are interchangeable (or that the nodes are changing their neighbours rapidly). As in the homogeneous closure of pairs, this approximation fails to account for correlations between neighbouring nodes in static networks.

Using $[S_k]+[I_k] = N_k$, we can reduce this system to $k_{\max}$ equations for $[\dot{I}_k]$, and solve the system for the steady states. We find again a disease-free steady state and an endemic steady state. Stability analysis reveals there is a transcritical bifurcation occurring at

$$\frac{\tau}{\gamma} = \frac{\langle k \rangle}{\langle k^2 \rangle},$$

and so the critical value of τ is given by

$$\tau_c = \gamma \frac{\langle k \rangle}{\langle k^2 \rangle}.$$

There also exists a heterogeneous pairwise model, although this has $\mathcal{O}(k_{\max}^2)$ equations, and for practical reasons a compact pairwise model with $2k_{\max} + 3$ equations is often used instead:

$$\begin{aligned}
[\dot{S}_k] &= \gamma[I_k] - \tau k[S_k]\frac{[SI]}{[SX]} \\
[\dot{I}_k] &= \tau k[S_k]\frac{[SI]}{[SX]} - \gamma[I_k] \\
[\dot{SI}] &= \gamma([II] - [SI]) + \tau([SS] - [SI])[SI]Q_c - \tau[SI] \\
[\dot{SS}] &= 2\gamma[SI] - 2\tau[SS][SI]Q_c \\
[\dot{II}] &= 2\tau[SI] - 2\gamma[II] + 2\tau[SI]^2 Q_c,
\end{aligned}$$

where $[SX] = \sum_{k=1}^{k_{\max}} k[S_k]$ and $Q_c = \frac{1}{[SX]^2}\sum_{k=1}^{k_{\max}}(k-1)k[S_k]$. The approximations required for this model assume that the neighbours of all susceptible nodes are interchangeable. However, neighbours of a susceptible degree k node that has been previously infected are more likely to be infected than neighbours of a susceptible degree k node that has never been infected.

2.6.2 Individual-based mean-field approach

The approximations so far have been derived using what Kiss, Miller and Simon describe as a 'top-down' approach, where the system equations are formulated by initially tracking the vector $X(t)$, containing the probabilities of the entire system being in each system-state. An alternative 'bottom-up' approach is possible, where we instead track the probabilities of each node being in each node-state.

We denote the probability of node i being susceptible or infected at time t by $\langle S_i(t)\rangle$ and $\langle I_i(t)\rangle$ respectively. We then seek to find the time evolution of these probabilities. The resulting equation for $\langle I_i\rangle$ is

$$\langle \dot{I}_i\rangle = \tau \sum_{j=1}^{N} g_{ij}\langle S_i I_j\rangle - \gamma\langle I_i\rangle,$$

where g_{ij} is the ijth element of the adjacency matrix G describing the connections between nodes, and γ is the recovery rate. This formulation theoretically allows for the value of the recovery parameter to vary for different nodes, by introducing separate recovery rates γ_i for each node i. In this chapter I consider only the simpler case where the recovery rate is the same for each node.

This equation for $\langle I_i\rangle$ depends on $\langle S_i I_j\rangle$, and so we also need to include the equation describing the time evolution of this term too:

$$\langle \dot{S_i I_j}\rangle = \tau \sum_{k=1,k\neq i}^{N} g_{jk}\langle S_i S_j I_k\rangle - \tau \sum_{k=1,k\neq j}^{N} g_{ik}\langle I_k S_i I_j\rangle - \tau g_{ij}\langle S_i I_j\rangle - \gamma\langle S_i I_j\rangle + \gamma\langle I_i I_j\rangle.$$

This equation in turn depends on triples as well as the further pairs, and we find the same problem as in the homogeneous and heterogeneous mean-field approximations. In order to meaningfully reduce the state space of the exact system, we need to apply a closure. The simplest choice is to apply a closure at the pair level, and the resulting individual-based mean-field model at the single level is described by the system of equations

$$\langle \dot{Y}_i\rangle = \left(\tau \sum_{j=1}^{N} g_{ij}(1-\langle Y_i\rangle)\langle Y_j\rangle\right) - \gamma\langle Y_i\rangle,$$

where $\langle X_i\rangle$ and $\langle Y_i\rangle$ are approximations to $\langle S_i\rangle$ and $\langle I_i\rangle$ respectively. This system involves N equations, and is the same system as that referred to by Van Mieghem et al as the N-intertwined mean-field approach (NIMFA) [67]. This closure at the pair level assumes neighbouring node-states are independent, and so the system cannot account for correlations between neighbouring nodes.

In fact, it can be proved that NIMFA will give overestimates of the true infection probabilities. To show this, we use the fact that Markovian SIS epidemics are

non-negatively correlated [68], so $\langle S_i I_j \rangle \leq \langle S_i \rangle \langle I_j \rangle$ for $t > 0$. If we then consider the exact individual-based model:

$$\begin{aligned}
\langle \dot{I}_i \rangle &= \tau \sum_{j=1}^{N} g_{ij} \langle S_i I_j \rangle - \gamma \langle I_i \rangle \\
&= \tau \sum_{j=1}^{N} g_{ij} \langle S_i \rangle \langle I_j \rangle - \gamma \langle I_i \rangle + \tau \sum_{j=1}^{N} g_{ij} (\langle S_i I_j \rangle - \langle S_i \rangle \langle I_j \rangle) \\
&= \tau \sum_{j=1}^{N} g_{ij} (1 - \langle I_i \rangle) \langle I_j \rangle - \gamma \langle I_i \rangle + \tau \sum_{j=1}^{N} g_{ij} (\langle S_i I_j \rangle - \langle S_i \rangle \langle I_j \rangle) \\
&\leq \tau \sum_{j=1}^{N} g_{ij} (1 - \langle I_i \rangle) \langle I_j \rangle - \gamma \langle I_i \rangle.
\end{aligned}$$

Assuming that the exact individual-based model, and the approximate closed model both start with the same initial conditions, i.e. $\langle I_i \rangle(0) = \langle Y_i \rangle(0)$ for $i = 1, 2, ..., N$, then $\langle I_i \rangle(t) \leq Y_i(t)$ for $i = 1, 2, ..., N$.

2.6.3 Approximate master equations

The final approximation I will consider is that of the approximate master equations derived by Gleeson [69]. In Gleeson's approach, we track the variables $S_{k,m}(t)$ and $I_{k,m}(t)$, the fractions of k-degree nodes that have m infected neighbours and are either susceptible or infected respectively. Note that in this section S and I denote fractions of susceptible and infected nodes, rather than absolute numbers. This model is a SVT model, and in its most general form Gleeson allows the probability of a node changing state to depend on its degree as well as on the number of its infected neighbours. Therefore the probability that a k-degree susceptible node with m infected neighbours becomes infected in a time interval dt is denoted by $F_{k,m}dt$. The probability of a k-degree infected node with m infected neighbours recovering within a time dt is similarly denoted $R_{k,m}dt$. Gleeson derives the master equations for the evolution of the variables $S_{k,m}$ and $I_{k,m}$, and finds

$$\begin{aligned}
\dot{S}_{k,m} &= -F_{k,m} S_{k,m} + R_{k,m} I_{k,m} - \beta^s (k-m) S_{k,m} + \beta^s (k-m+1) S_{k,m-1} - \gamma^s m S_{k,m} + \gamma^s (m+1) S_{k,m+1} \\
\dot{I}_{k,m} &= -R_{k,m} I_{k,m} + F_{k,m} S_{k,m} - \beta^i (k-m) I_{k,m} + \beta^i (k-m+1) I_{k,m-1} - \gamma^i m I_{k,m} + \gamma^i (m+1).
\end{aligned}$$

The terms including the rates $F_{k,m}$ and $R_{k,m}$ describe transitions where a k-degree node is either infected or recovers. The remaining terms represent situations where the neighbour of a k-degree node changes state. The rates $\beta^s, \beta^i, \gamma^s$ and γ^i are variable, and are approximated by counting the number of edges of each type and seeing how they change as the system evolves. For example, the probability $\beta^s dt$ is calculated by finding the number of $S-S$ edges which transition to $S-I$ edges in the small time interval dt, and dividing this by the total number of $S-S$ edges at time t. We can find $I_k(t)$, the total fraction of k-degree nodes that are infected, by summing over all possible values of m: $I_k(t) = \sum_{m=0}^{k} I_{k,m}$. The total fraction of nodes that are infected, $I(t)$, can also be found, by summing over all possible degrees: $I(t) = \sum_k P_k I_k(t)$.

If we have nodes varying in degree from $k = 0$ to some maximum degree $k = k_{\max}$, there are $(k_{\max}+1)(k_{\max}+2)$ equations, and so the state space grows as $\mathcal{O}(k_{\max}^2)$. For networks with a small value of $k_{\max}$, these master equations are tractable, and for $k_{\max} \ll N$, the state space is significantly smaller than that of the standard master equations. I will show in the next section how, for a number of different networks, the solution to these approximate master equations is more accurate than the solutions found using the previous approximations. For networks of this size, we can also use the approximate master equations to find the SIS epidemic threshold. An appropriate linearisation of the master equations reveals a link between the largest eigenvalue of a matrix of order $k_{\max}^2$ and the epidemic threshold.

However, for systems with larger maximum degree, this full system is not tractable and the matrix used in determining the epidemic threshold is too large to construct or analyse. We again look at ways we can apply a closure to the system, to reduce the state space. To do this we consider instead the parameters $p_k(t)$ and $q_k(t)$, the probabilities of a randomly-chosen neighbour of a susceptible or infected k-degree node respectively being infected at time t. These parameters can be expressed in terms of $S_{k,m}$ and $I_{k,m}$. For example, $p_k = \sum_{m=0}^{k} m S_{k,m} / \sum_{m=0}^{k} k S_{k,m}$. Then we can apply a closure approximation to $S_{k,m}$ and $I_{k,m}$, and approximate the two variables as being proportional to binomial distributions: $S_{k,m} \approx (1-I_k) B_{k,m}(p_k)$, $I_{k,m} \approx I_k B_{k,m}(q_k)$. This sys-

tem ultimately results in a system of $3k_{\max}+1$ differential equations, with time evolution equations for I_k, p_k and r_k for all values of k.

An even cruder approximation replaces both p_k and q_k with ω, where $\omega = \langle \frac{k}{z} I_k \rangle$ is the probability that one end of a randomly-chosen edge is infected. This gives us a system of just $k_{\max}+1$ differential equations for I_k:

$$\dot{I}_k = I_k \sum_{m=0}^{k} R_{k,m} B_{k,m}(\omega) + (1 - I_k) \sum_{m=0}^{k} F_{k,m} B_{k,m}(\omega).$$

Inputting for $F_{k,m}$ and $R_{k,m}$ the specific infection and recovery rates for the SIS model, this system reduces to that used by Pastor-Satorras and Vespignani [50].

2.7 Accuracy of approximations compared to simulation

In this section I will examine the accuracy of the approximations detailed above, comparing the results from these approximations with results from simulated ensembles. The main aims of this investigation are as follows:

- To assess the accuracy of the homogeneous mean-field approximation at the single level, and to see how this compares to the other approximations. Given the approximation will be the focus of much of Chapter 3, and will be used during the investigation in Chapter 5, it is necessary to know when it will give practical results.
- To identify whether there is a significant improvement in accuracy by using the approximations with larger state spaces, and whether this improvement is large enough to outweigh any sacrifices in analytical tractability.

To do this, I have simulated SIS dynamics running on a number of different networks. I explored the effects of the system parameters, such as the network size N and the reproductive ratio R_0, and the effects of structural properties of the networks, such as the average degree $\langle k \rangle$ and the degree distribution variance $\langle k^2 \rangle - \langle k \rangle^2$. To do this I also used several different structures of networks, several of which were introduced in Chapter 1:

- Erdős-Rényi networks: these are random networks, defined by some probability p of each edge being present. I generated these networks in R using the 'erdos.renyi.game' function in the *igraph* package.

- Scale-free networks: these are networks where the degree distribution approximately follows a power-law distribution. I generated these networks in R using the 'barabasi.game' function in the *igraph* package.

- Regular random networks: these are random networks where each node has the same degree. I generated these networks in R using the 'sample_k_regular' function in the *igraph* package.

- Bimodal random networks: these are random networks where each node can have one of two different degrees, labelled d_1 and d_2. I generated these networks in R using the 'sample_degseq' function in the *igraph* package.

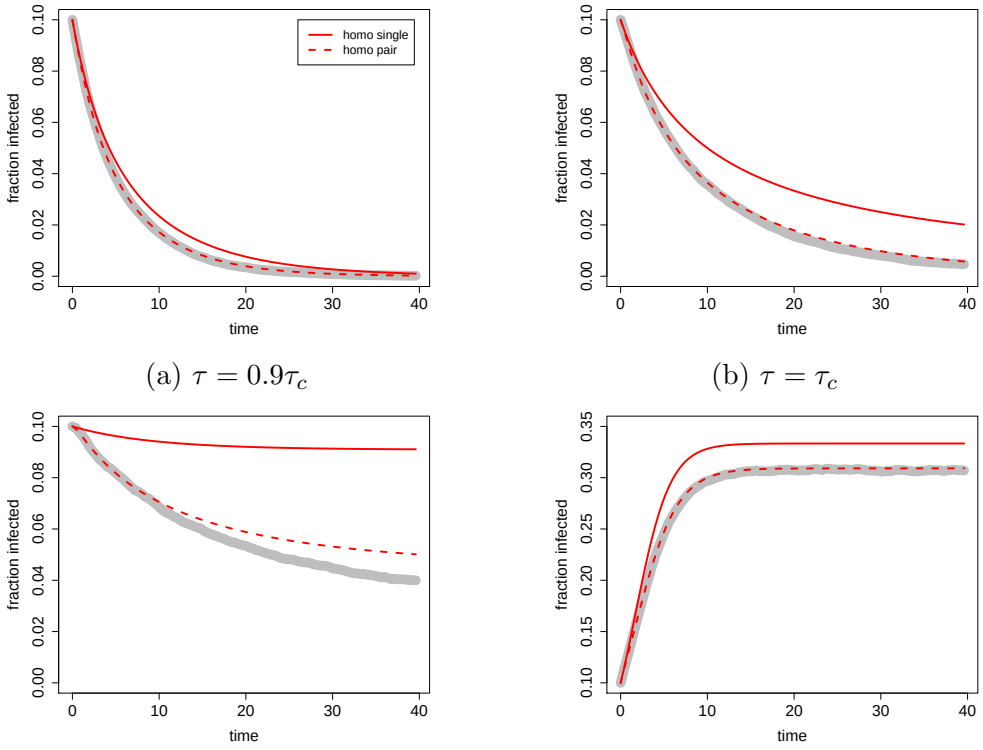
The latter two network structures can be generated by using methods based on the configuration model method. The configuration model generates random networks, given a set degree sequence. Each degree in the given degree sequence is assigned to a node. These degrees can be thought of as half-edges or 'stubs', which need to be connected with another degree in order to form an edge between two nodes. If we sample randomly from the unpaired degrees in the network, we can randomly place the edges on the network while preserving the degree distribution. This model can result in self-edges or multi-edges, but the R functions aim to avoid this through more sophisticated algorithms.

For this investigation I worked exclusively with networks that were too large for me to solve the SIS dynamics analytically (ranging from $N = 100$ to $N = 1000$). I wanted to see how well the mean-field results capture both the transitional period as well as the steady state value. To this end, I started the simulations from a state of low infection where 10% of nodes are initially infected. I chose to start with 10% of nodes initially infected, rather than simply a single infected node, so that the majority of simulations would reach the endemic state, and avoid the absorbing state. This means the ensemble average is more likely to meaningfully represent the endemic steady state, and this allowed me to compare the accuracy of the mean-field steady state with the ensemble result.

First, I investigated the performance of both the homogeneous single level approximation and the homogeneous pairwise level approximation. This section closely follows the work done by Kiss, Miller and Simon, and confirms many of their results [59]. I started by examining the dynamics on a regular random network with $N = 1000$ and $\langle k \rangle = 20$ for different values of R_0, or equivalently different values of τ relative to the homogeneous single level critical value $\tau_c = \frac{\gamma}{n}$. Plots showing the results for $R_0 = 0.9, 1, 1.1, 1.5$, equivalently $\tau = 0.9\tau_c, \tau_c, 1.1\tau_c, 1.5\tau_c$, are shown in Figure 2.4. The thick grey line indicates the ensemble average, and the homogeneous single level and pairwise level approximations are indicated by the solid and dashed red lines respectively. We can see that the approximations perform better for values further from the predicted epidemic threshold $\tau = \tau_c$. The single level approximation in particular seems sensitive to this. These simulations (and all further ones during this investigation) also confirm the previously-proved result that the homogeneous mean-field pairwise model is bounded by the single level model. The single level model appears to perform worse than the pairwise model, a result we might anticipate from the fact that the pairwise model is a larger system including more information about the pairwise behaviour.

To study the dependence on the size of the network, I performed simulations on regular random networks of size $N = 100, 200, 400$ and 800. For each of the networks I kept the node degree the same at $\langle k \rangle = 10$. I set $R_0 = 1.5$. We can see in Figure 2.5 that the simulation result more closely matches the approximations for larger systems. Notably, for $N = 100$ we can see that the ensemble average reaches an apparent stationary state before decaying towards (at some future time) the disease-free steady state. This behaviour cannot be captured by the mean-field approximations, which ignore this stochastic effect.

I also found that the denser the network, the more accurate the approximations. Figure 2.6 shows the results for sparse and dense regular random networks (with $\langle k \rangle = 5$ and $\langle k \rangle = 50$ respectively), and for sparse and dense Erdős-Rényi networks (again, with $\langle k \rangle = 5$ and $\langle k \rangle = 50$). In both cases the homogeneous approximations perform better for the denser network. The approximations perform better for the regular random networks than the Erdős-Rényi networks, an initial



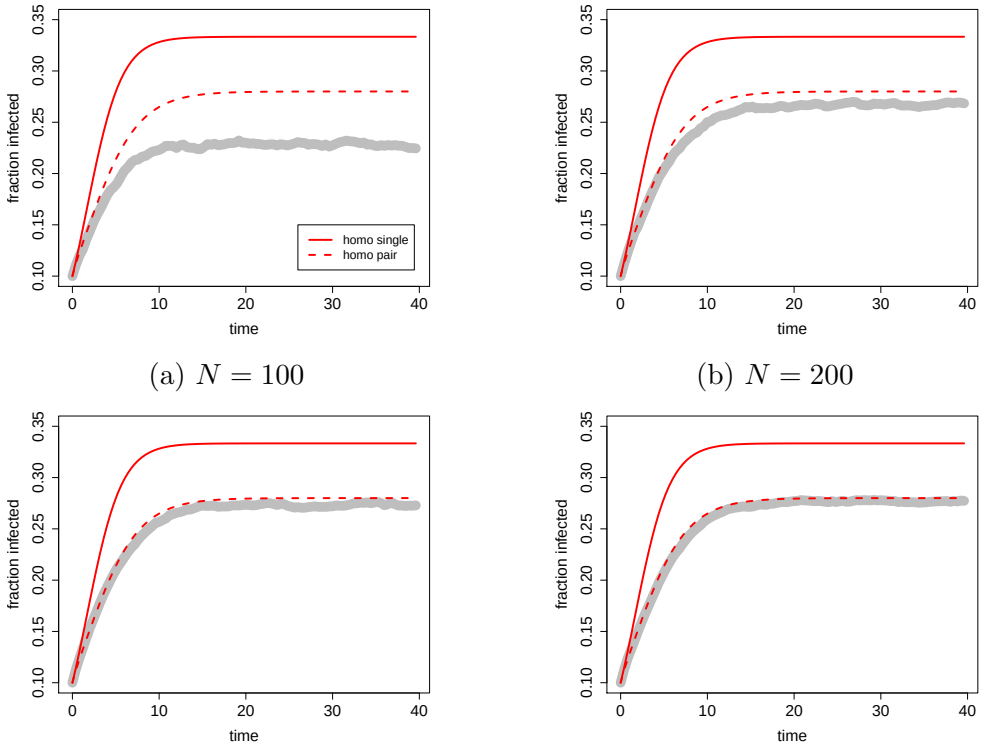
(a) $\tau = 0.9\tau_c$

(b) $\tau = \tau_c$

(c) $\tau = 1.1\tau_c$

(d) $\tau = 1.5\tau_c$

Figure 2.4: The results of simulating SIS dynamics on a regular random network for varying values of τ (thick grey line), compared with the homogeneous single level model (solid red line) and the homogeneous pairwise model (dashed red line). The values of the parameters are $N = 1000, \langle k \rangle = 20, \gamma = 1$, and $\tau = 0.9\tau_c, \tau_c, 1.1\tau_c$, and $1.5\tau_c$ for plots (a), (b), (c) and (d) respectively.



(a) $N = 100$ (b) $N = 200$

(c) $N = 400$ (d) $N = 800$

Figure 2.5: The results of simulating SIS dynamics on regular random networks of varying size (thick grey line), compared with the homogeneous single level model (solid red line) and the homogeneous pairwise model (dashed red line). The sizes of the networks are $N = 100, 200, 400$ and 800 for plots (a), (b), (c) and (d) respectively. The networks all have $\langle k \rangle = 10$, and the epidemic parameters are $\gamma = 1, \tau = 1.5\tau_c$.

example of how the accuracy breaks down once the network degree distribution shows some heterogeneity.

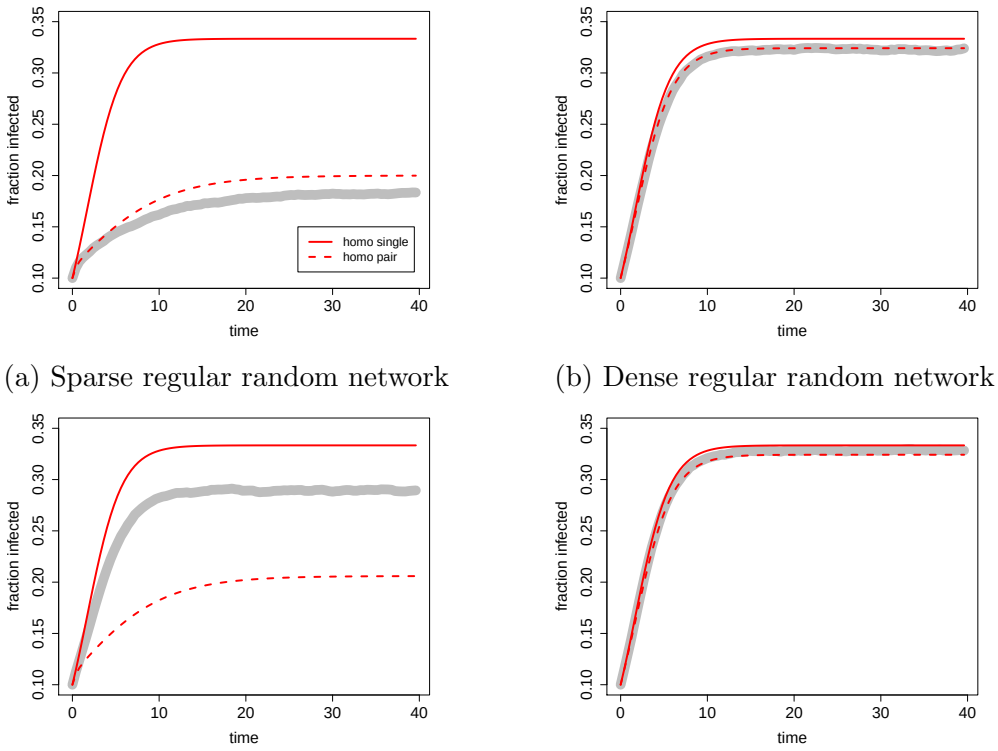
To further explore this, I widened my consideration to include the two heterogeneous approximations, and NIMFA at the single level. Since these three approximations include information about the degree distribution, we might expect them to outperform the homogeneous mean-field approximations for networks with large degree heterogeneity. I looked at the dynamics on bimodal random networks, where I could adjust the two different possible degrees to generate networks with varying degree heterogeneity.

Figure 2.7 shows the results as I moved from a degree variance of $\langle k^2 \rangle - \langle k \rangle^2 = 4$ (Figure 2.7(a)) to a degree variance of $\langle k^2 \rangle - \langle k \rangle^2 = 225$ (Figure 2.7(c)). The two homogeneous approximations are again marked in red lines (a solid line for the single level, dashed for pairwise). As we move from more homogeneous to more heterogeneous degree distributions, these two approximations start to perform worse. The pairwise approximation does a better job, in general, but still fails to accurately predict the dynamics. This is to be expected, since the key assumption of homogeneity no longer holds for the bimodal networks with larger degree variance.

In comparison, the heterogeneous approximations shown in blue lines (solid for single level, dashed for compact pairwise) perform better for the most heterogeneous case. However, we can see that for the most homogeneous case, the heterogeneous single level approximations in fact performs worse than the homogeneous pairwise level approximation. This suggests that, for more homogeneous networks, it becomes more important to capture the pairwise correlations than the (low) heterogeneity of the network.

The NIMFA single level approximation (illustrated with a dashed green line) shows similar behaviour to the heterogeneous single level approximation: it peforms relatively poorly for the more homogeneous networks, since it is a single level closure, but performs relatively well for the heterogeneous case, where the inclusion of the adjacency matrix allows it to better capture the heterogeneous degree distribution.

I also performed simulations for the dynamics on a power law network, generated using the Barabasi-Albert model with constant degree of 10 for all newly-



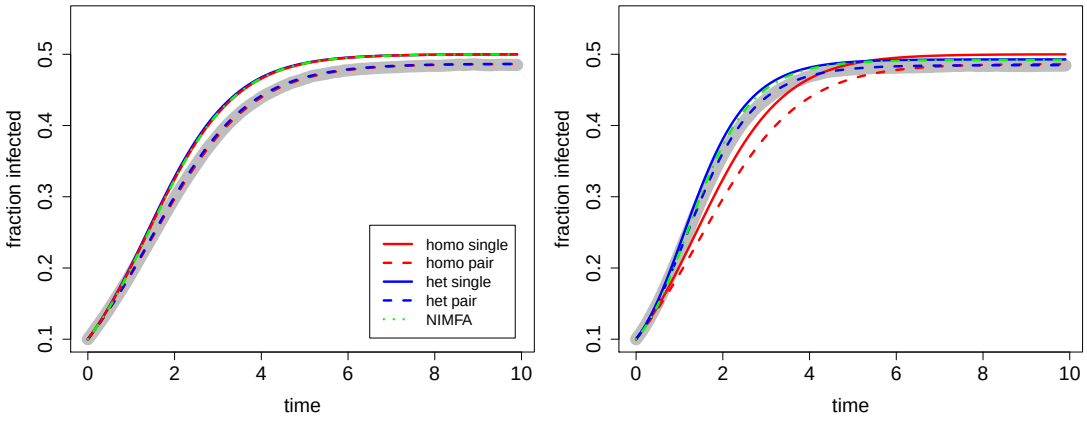
(a) Sparse regular random network

(b) Dense regular random network

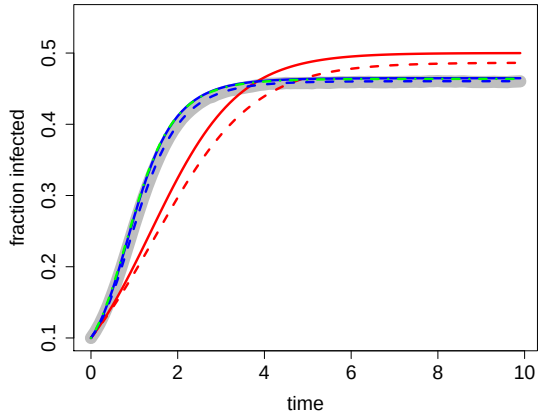
(c) Sparse Erdős-Rényi network

(d) Dense Erdős-Rényi network

Figure 2.6: The results of simulating SIS dynamics (thick grey line), first on (a) sparse and (b) dense regular random networks, and then on (c) sparse and (d) dense Erdős-Rényi networks, compared with the homogeneous single level model (solid red line) and the homogeneous pairwise model (dashed red line). The networks all have $N = 1000$, with $\langle k \rangle = 5$ for the sparse networks and $\langle k \rangle = 50$ for the dense networks. The epidemic parameters are $\gamma = 1$ and $\tau = 1.5\tau_c$.



(a) $d_1 = 18, d_2 = 22$ (b) $d_1 = 10, d_2 = 30$



(c) $d_1 = 5, d_2 = 35$

Figure 2.7: The results of simulating SIS dynamics on a range of different bimodal networks (thick grey line), compared to the homogeneous single level model (solid red line), the homogeneous pairwise model (dashed red line), the heterogeneous single level model (solid blue line), the heterogeneous compact pairwise model (dashed blue line), and NIMFA (dashed green line). The networks all have the same size ($N = 1000$) and average degree ($\langle k \rangle = 20$), but the network in (a) has $d_1 = 18, d_2 = 22$, and degree variance $\langle k^2 \rangle - \langle k \rangle^2 = 4$ the network in (b) has $d_1 = 10$, $d_2 = 30$ and degree variance $\langle k^2 \rangle - \langle k \rangle^2 = 100$, and the network in (c) has $d_1 = 5$, $d_2 = 35$ and degree variance $\langle k^2 \rangle - \langle k \rangle^2 = 225$. The epidemic parameters in all three cases are $\gamma = 1$ and $\tau = 1.5\tau_c$.

attached vertices. This led to a network with maximum degree 161, and degree variance $\langle k^2 \rangle - \langle k \rangle^2 = 302$. As shown in Figure 2.8, the homogeneous approximation accuracy completely breaks down in this large variance situation. Again, the results from NIMFA closely follow the heterogeneous single level model, while the heterogeneous compact pairwise model performs best. These results suggest that the homogeneous approximations are not a reasonable choice of approximation for networks that have high degree variance.

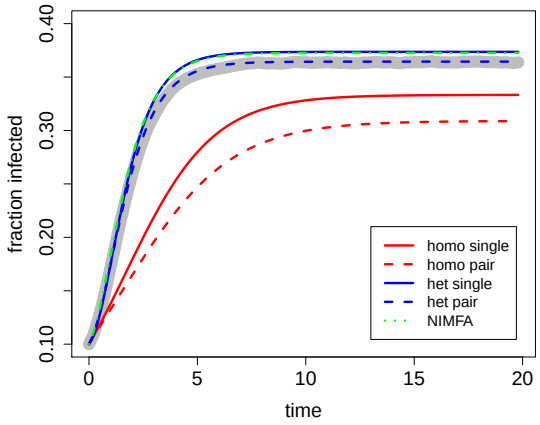


Figure 2.8: The result of simulating SIS dynamics on a Barabási-Albert network (thick grey line), compared to the homogeneous single level model (solid red line), the homogeneous pairwise model (dashed red line), the heterogeneous single level model (solid blue line), the heterogeneous compact pairwise model (dashed blue line) and NIMFA (dashed green line). The network parameters are $N = 1000$ with constant degree 10 for each newly-attached vertex, max degree 161, degree variance $\langle k^2 \rangle - \langle k \rangle^2 = 302$. The epidemic parameters are $\gamma = 1$ and $\tau = 1.5\tau_c$.

I also explored the same three dependences as before, now with these heterogeneous approximations: how their accuracy varies with proximity of the infection parameter to the critical value τ_c, density of the network, and size of the network. Taking again a bimodal network with $N = 1000$ and $d_1 = 5, d_2 = 35$, Figure 2.9 shows how the approximations are more accurate further from the heterogeneous single level critical value of $\tau_c = \gamma\langle k \rangle / \langle k^2 \rangle$. Figure 2.10 shows the results for a sparse Erdős-Rényi network and a dense Erdős-Rényi ($N = 1000$, $\langle k \rangle = 10$ and 50 respectively), and we see better agreement for the denser network. Finally I

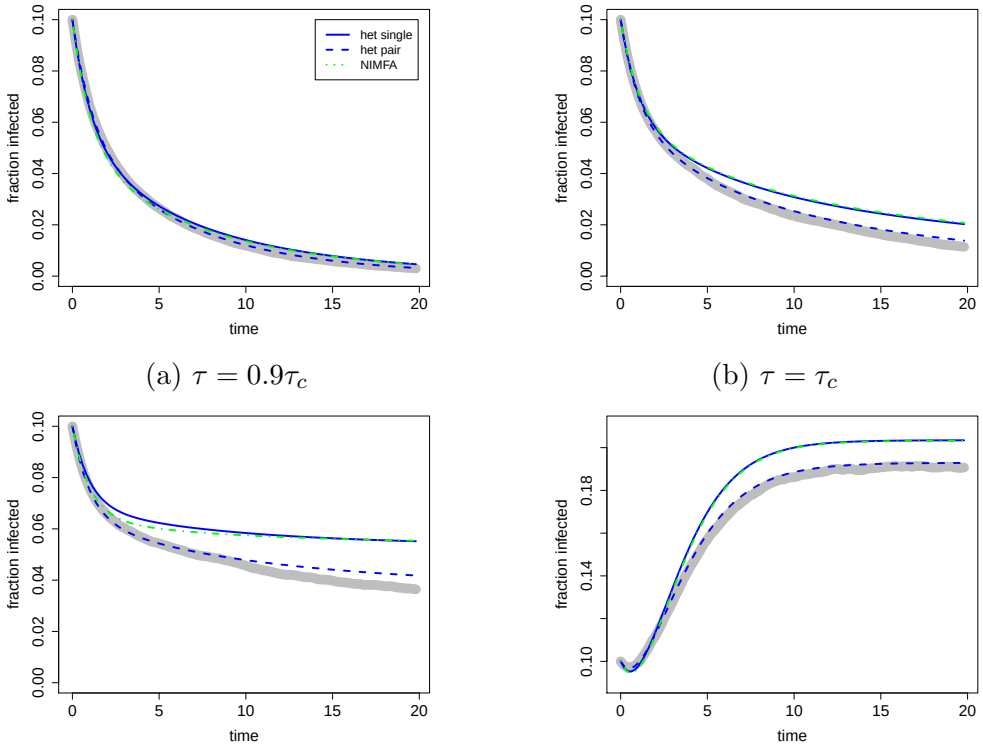
looked at the dynamics on bimodal networks of varying size: $N = 100, 200, 400$ and 800, all with $d_1 = 5, d_2 = 15$. Figure 2.11 shows the results, and we can see how the approximations gradually become more accurate as the network size increases. These three results are all in line with my findings for the homogeneous approximations.

Finally, the approximate master equations performed well in the situations I tested. Figure 2.12 shows how closely the result (the magenta dashed line) matches the simulation (the thick grey line) for a range of different networks, including very heterogeneous networks and a small network. This accuracy is predictable, as the state space for this system is much larger than the other approximations. However, this larger state space means that the computational demands of solving the system are non-trivial. I used the 'ode' function in R, again with the default settings, to solve the system, and I found that calculating the solution was a struggle for networks with maximum degree $k_{\max} > 50$. Figure 2.12 also shows the result of the heterogeneous compact pairwise model (again a dashed blue line) for the same networks and parameters. For all except the simulation on the Barabási-Albert network, the heterogeneous compact pairwise solution and the approximate master equations solution are almost indistinguishable. Given the heterogeneous compact pairwise model has only $2k_{\max}+3$ equations, and involves far fewer operations to solve, this seems a more sensible choice of approximation.

Summarising these results in relation to the main aims of this investigation:

- The homogeneous mean-field approximation at the single level performs well on networks with homogeneous degree distributions. Its accuracy increases with the size of the network, the density of the network, and for values of τ far from the critical value τ_c. The pair level closure generally gives more accurate results, but for the case of the power-law network, the single level closure actually performed better.

- The two approximations with the largest state spaces, the heterogeneous compact pairwise model and the approximate master equations, give the most accurate results across all the networks and parameters studied in this section. However, the heterogeneous compact pairwise model performs



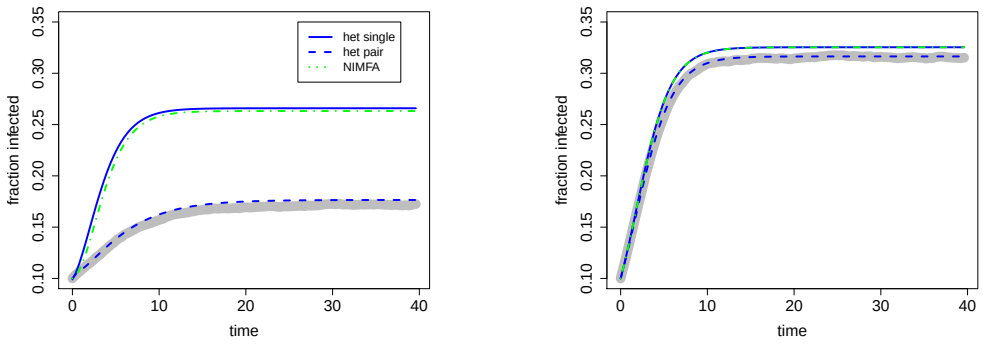
(a) $\tau = 0.9\tau_c$

(b) $\tau = \tau_c$

(c) $\tau = 1.1\tau_c$

(d) $\tau = 1.5\tau_c$

Figure 2.9: The results of simulating SIS dynamics on a bimodal network for varying values of τ (thick grey line), compared with the heterogeneous single level model (solid blue line), the heterogeneous compact pairwise model (dashed blue line), and NIMFA (dashed green line). The values of the parameters are $N = 1000, d_1 = 5, d_2 = 35, \langle k \rangle = 20, \gamma = 1$, and $\tau = 0.9\tau_c, \tau_c, 1.1\tau_c$, and $1.5\tau_c$ for plots (a), (b), (c) and (d) respectively.



(a) Sparse Erdős-Rényi network

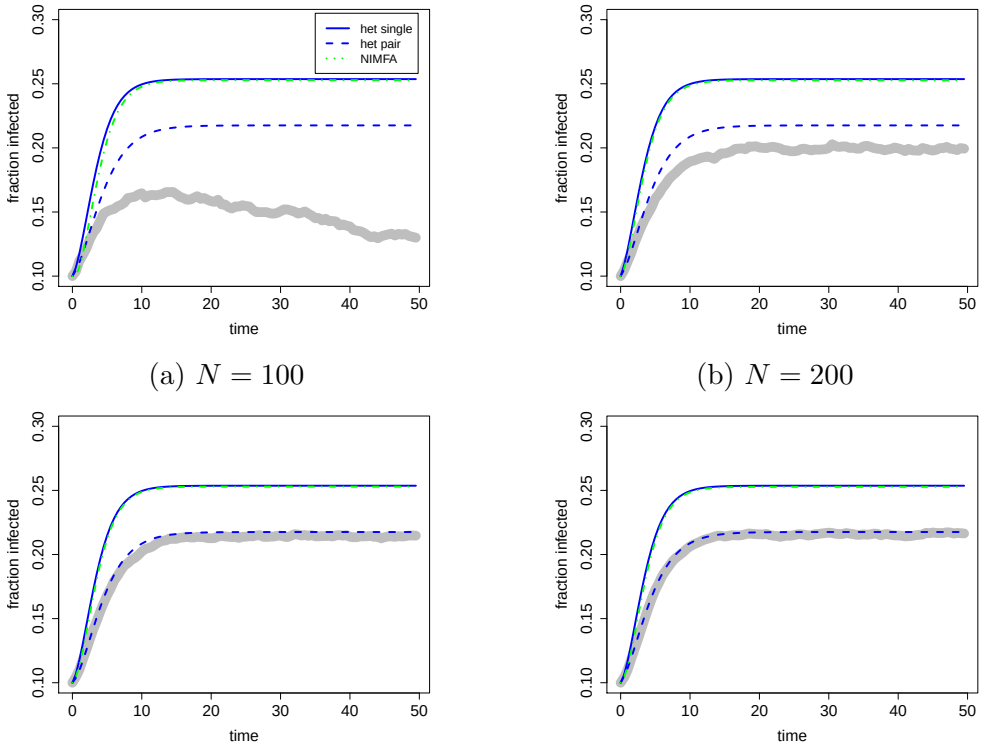
(b) Dense Erdős-Rényi network

Figure 2.10: The results of simulating SIS dynamics on (a) sparse and (b) dense Erdős-Rényi networks (thick grey line), compared with the heterogeneous single level model (solid blue line), the heterogeneous compact pairwise model (dashed blue line) and NIMFA (dashed green line). The networks both have $N = 1000$, with $\langle k \rangle = 5$ for the sparse network and $\langle k \rangle = 50$ for the dense network. The epidemic parameters are $\gamma = 1$ and $\tau = 1.5\tau_c$.

almost identically to the approximate master equations, despite having a much smaller state space in general and being much less computationally expensive to solve. This shows that the gains in accuracy from using a more detailed model are not always significant enough to justify the increased computational demands.

2.8 Summary

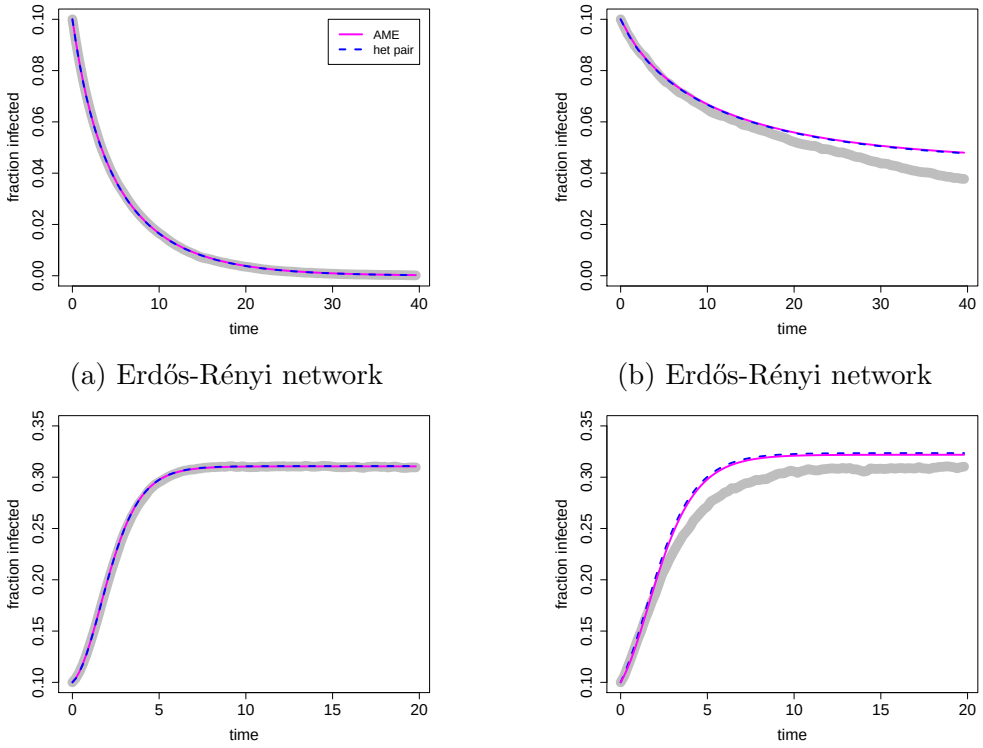
This chapter has outlined a number of different ways of approximation and solving SIS dynamics on a network. The compartmental model is the simplest model, but does not include any information about the underlying network. A stochastic Markov Chain treatment allows us to incorporate network information, but this system is only tractable for very small networks. Simulation methods offer one way of solving the system for larger networks, but they can be computationally expensive. I have therefore presented and explored a number of different mean-field approximations, which can be used to reduce the state space of the system.



(a) $N = 100$ (b) $N = 200$

(c) $N = 400$ (d) $N = 800$

Figure 2.11: The results of simulating SIS dynamics on bimodal networks of varying size, compared with the heterogeneous single level model (solid blue line), the heterogeneous compact pairwise model (dashed blue line) and NIMFA (dashed green line). The sizes of the networks are $N = 100, 200, 400$ and 800 for plots (a), (b), (c) and (d) respectively. The networks all have $d_1 = 5$ and $d_2 = 15$ with $\langle k \rangle = 10$, and the epidemic parameters are $\gamma = 1, \tau = 1.5\tau_c$.



(a) Erdős-Rényi network

(b) Erdős-Rényi network

(c) Bimodal network

(d) Barabási-Albert network

Figure 2.12: The results of simulating SIS dynamics on several different networks (thick grey line), compared with the approximate master equations solution (magenta dashed line) and the heterogeneous compact pairwise model solution (dashed blue line). The two solutions are almost identical, and so almost indistinguishable. The networks and parameters are respectively: (a) an Erdős-Rényi network with $N = 1000$, $\langle k \rangle = 20$ and $\gamma = 1$, $\tau = 0.9\tau_c$; (b) an Erdős-Rényi network with $N = 1000$, $\langle k \rangle = 20$ and $\gamma = 1$, $\tau = 1.1\tau_c$; (c) a bimodal network with $N = 1000$, $d_1 = 5, d_2 = 35$ where $\tau = 2\tau_c$; (d) a Barabási-Albert network with $N = 100$, constant degree 5 for each newly-added node and $\tau = 2\tau_c$.

As expected, the accuracy of the homogeneous mean-field approximations breaks down when applied to networks with large degree heterogeneity, such as Barabási-Albert networks. In these situations, approximations that can take into account the heterogeneous degree distribution, such as the heterogeneous mean-field approximations and NIMFA, are the most appropriate approximations to use. If the network is fairly homogeneous, however, then the homogeneous mean-field approximations perform well, and it becomes more important to apply a closure at the level of pairs than to capture information about the degree distribution. The approximate master equations generally outperform the five other approximations investigated, but these equations aren't practical for networks that have a maximum degree $k_{\max} > 50$. The heterogeneous compact pairwise model also gives incredibly close results to the approximate master equations for the networks explored in this investigation, and is in general less computationally demanding, providing a more attractive choice in most situations. This shows how systems with a larger state space are not always significantly more accurate than simpler, smaller systems.

Notably, this investigation has shown how it is hard to know *a priori* which approximation will perform best, even when we know the general trends in approximation accuracy. For example, when considering the sparse Erdős-Rényi network shown in Figure 2.6(c), the single level homogeneous approximation actually outperformed the pairwise homogeneous approximation. Likewise this result occurred for the Barabási-Albert network in Figure 2.8. It is also possible for the homogeneous pairwise level approximation to outperform the heterogeneous single level approximation, as in the fairly homogeneous network shown in Figure 2.7(a). This shows the need to develop a better understanding of how the error depends on network properties. My work in the next chapter starts to tackle this problem.

Chapter 3

Approximate lumping of dynamics on networks

3.1 Introduction

The mean-field approximations explored in the previous chapter are frequently applied to dynamics on networks, but many of the assumptions that underpin these approximations, such as the absence of local clustering or dynamical correlations, are routinely violated by dynamical processes on real-world networks [51]. As shown in the previous chapter, their motivation is also typically based on intuitive probabilistic reasoning rather than rigorous mathematics. This means it is generally difficult to predict how well these approaches will work, beyond the general trends based on the dynamical parameters, and the size, density and homogeneity of the network already investigated in the previous chapter. Due to these issues, Pellis et al identified the problem of giving rigorous theoretical understanding to approximation schemes as one of the core current challenges facing network research [70].

In this chapter I focus on an approximation method called the approximate lumping method, and I show how this method provides us with a novel theoretical underpinning to mean-field approaches. I define a quantity called the lumping error, which measures the extent to which the approximate lumping violates the condition for an exact lumping. I minimise this quantity for a broad range of dynamical processes on networks, and I demonstrate how this approach allows us

to derive an expression for the error of the approximate lumped solution. This error expression relies in part on the steady state solution of the dynamics, and therefore is of limited usefulness without further work. A full investigation of the error analysis for larger networks was beyond the scope of this thesis, but I discuss some preliminary ideas which feature in a submitted paper [71] in Section 3.6.3 of this chapter.

Section 3.2 describes the Markov Chain formulation of the system, and shows how approximate lumping can be applied, and the lumping error minimised, for a general set of network dynamics. In Section 3.3 I solve this minimisation process for the case of single-vertex transition models with binary vertex-states, and show that we recover the homogeneous mean-field approximation transition rates. In Section 3.4 I analyse the error of the approximate lumped solution in the binary vertex-state case, for processes with and without absorbing states. Section 3.5 extends some of this work to models with non-binary vertex-states. Section 3.6 concludes and outlines some potential extensions to this work.

3.2 Approximate lumping optimisation

In this chapter I will focus on finite simple networks, i.e. undirected, unweighted networks that have no self-edges. Consider such a network with N vertices, where each vertex can be in one of M vertex-states $\{\Sigma_1, \Sigma_2, ...\Sigma_M\}$. We can then construct the same Markov chain formulation of general dynamics as in Section 2.3, which is summarised again here.

The Markov chain state space consists of $n = M^N$ states, $\mathcal{S} = \{s_1, s_2...s_n\}$. We denote the vertex-state of a given vertex v as σ_v, where $\sigma_v \in \{\Sigma_1, \Sigma_2, ...\Sigma_M\}$. A given state s_i can then be described by a vector of these vertex-states $s_i = (\sigma_1, \sigma_2, ...\sigma_N)$. The transition rate between two states s_i and s_j can be expressed as a general function of the two states $h(s_i, s_j)$. The vector $\mathbf{X}(t) = (X_1(t), X_2(t), ...X_{M^N}(t))$ contains the probabilities $X_i(t)$ that the system is in state s_i at time t, and forms the time-dependent Markov chain probability distribution over the state-space $\mathcal{S}$. The master equation describing the time evolution of $\mathbf{X}(t)$ is given by

$$\dot{\mathbf{X}} = \mathbf{Q}^{\mathrm{T}}\mathbf{X},$$

where $\mathbf{Q}$ is the infinitesimal generator formed by the instantaneous transition rates [59]. $\mathbf{Q}$ is an M^N by M^N matrix, and equivalently there are M^N linear differential equations in the system of master equations. Due to this exponential scaling, the dynamics quickly become intractable on networks with even a modest number of vertices. This motivates the use of methods and approximations to reduce the state space to a more feasible size, as initially explored in Chapter 2.

Lumping is one method of reducing the state space, in which states are partitioned into a smaller number of subsets referred to as cells. Importantly, lumping preserves the Markov property, which holds when the probability of observing each possible future state of a stochastic process is dependent only on the present state of the system, and not on the past states. More formally, a partitioning of the state space $\mathcal{S}$ is defined as $\mathcal{L} = \{L_1, L_2, ...L_r\}$, where a cell L_k is a non-empty subset of $\mathcal{S}$, $L_k \cap L_l = \emptyset$, $\bigcup_k L_k = \mathcal{S}$. We say that a state space $\mathcal{S}$ is strongly lumpable with respect to a given partitioning $\mathcal{L}$ if the lumped system also obeys the Markov condition [72]. A necessary and sufficient condition for this is that for any pair of cells L_i and L_j, the transition rate of going from any $s_k \in L_i$ to any state $s_l \in L_j$ is the same for all s_k, i.e. there exists a matrix $\mathbf{R}_{ij}$ such that

$$\mathbf{R}_{ij} = \sum_{s_l \in L_j} \mathbf{Q}_{kl} \quad \text{for all } s_k \in L_i. \tag{3.1}$$

We can define the collector matrix $\mathbf{C}$ [73] for a given partitioning $\mathcal{L}$:

$$\mathbf{C}_{ij} = \begin{cases} 1 & \text{if } s_i \in L_j \\ 0 & \text{otherwise.} \end{cases}$$

We can also define the distributor matrix $\mathbf{D}$ [73] for a given partitioning:

$$\mathbf{D}_{ij} = \begin{cases} \frac{1}{\#L_i} & \text{if } s_j \in L_i \\ 0 & \text{otherwise,} \end{cases}$$

where $\#L_i$ is the number of states in the cell L_i. This matrix satisfies $\mathbf{DC} = \mathbf{I}$. If the state space is strongly lumpable, then the matrix $\mathbf{R}$ can be constructed as $\mathbf{R} = \mathbf{DQC}$, and satisfies the lumping condition $\mathbf{QC} = \mathbf{CR}$. This lumping condition is equivalent to (3.1). This matrix $\mathbf{R}$ functions as the infinitesimal generator governing the time evolution of a vector $\mathbf{Y} = (Y_1(t), Y_2(t), ...Y_r(t))$, which contains the probabilities $Y_i(t)$ of being in each cell L_i at time t:

$$\dot{\mathbf{Y}} = \mathbf{R}^{\mathrm{T}}\mathbf{Y}.$$

For a strongly lumpable system, $\mathbf{Y}(t) = \mathbf{C}^{\mathrm{T}}\mathbf{X}(t)$ for all time t.

The possible lumpings of a system are intrinsically linked with the symmetries of the network [59]. This makes lumping an effective tool for use on very small networks or larger networks with obvious patterns of symmetry, but less practical for the more complex networks generally found in real-life. Therefore I chose to explore approximate lumping, where the condition of preserving the Markov property is no longer imposed. An approximate lumping is therefore any lumping of the state space for which there does not exist a matrix $\mathbf{R}$ satisfying condition (3.1).

I aimed to seek the most appropriate infinitesimal generator $\mathbf{R}$ to govern the time evolution of the vector $\mathbf{Y}$, in the approximate lumping case where the lumping condition $\mathbf{QC} = \mathbf{CR}$ is violated. Since $\mathbf{QC} - \mathbf{CR} = 0$ for exact lumpings, I would like to construct the generator $\mathbf{R}$ such that a measure of $\mathbf{QC} - \mathbf{CR}$ is small. Using the Frobenius norm, I therefore sought to minimise the quantity $\|\mathbf{QC} - \mathbf{CR}\|_F$, which I call the *lumping error*. From the definition of the Frobenius norm:

$$\|\mathbf{QC} - \mathbf{CR}\|_F = \sum_{i=1}^{M^N} \sum_{j=1}^{r} (\mathbf{QC} - \mathbf{CR})_{ij}^2 \tag{3.2}$$

$$= \sum_{i=1}^{r} \sum_{s_k \in L_i} \sum_{j=1}^{r} (\mathbf{QC} - \mathbf{CR})_{kj}^2. \tag{3.3}$$

Using the previous definition of the collector matrix

$$\mathbf{C}_{ij} = \begin{cases} 1 & \text{if } s_i \in L_j \\ 0 & \text{otherwise} \end{cases}$$

then we can see that

$$(\mathbf{CR})_{kj} = \sum_{i=1}^{r} \mathbf{C}_{ki}\mathbf{R}_{ij} = \mathbf{R}_{ij} \quad \text{if } s_k \in L_i,$$

and so

$$\|\mathbf{QC} - \mathbf{CR}\|_F = \sum_{i=1}^{r} \sum_{s_k \in L_i} \sum_{j=1}^{r} [(\mathbf{QC})_{kj} - \mathbf{R}_{ij}]^2.$$

For ease of notation we can also define the matrix $\mathbf{P} = \mathbf{QC}$, and then the condition to minimise the lumping error corresponds to minimising the sum of square

differences

$$\sum_{i=1}^{r}\sum_{j=1}^{r}\sum_{s_k\in L_i}(\mathbf{P}_{kj}-\mathbf{R}_{ij})^2.$$

This is minimised if we choose $\mathbf{R}_{ij}$ to be the average of the sum of rates out of states in the ith level and into the jth level, i.e.

$$\mathbf{R}_{ij}=\frac{1}{\#L_i}\sum_{s_k\in L_i}\mathbf{P}_{kj}. \tag{3.4}$$

This infinitesimal generator is in fact still structured as $\mathbf{R}=\mathbf{DQC}$. Therefore using this construction for the generator $\mathbf{R}$, even when the lumping condition is not satisfied, is still optimal if we wish to minimise the lumping error. The structure for the generator $\mathbf{R}$ for SIS dynamics on a four-node network, along with the collector matrix $\mathbf{C}$, distributor matrix $\mathbf{D}$, and an illustration of the matrix multiplication $\mathbf{DQC}=\mathbf{R}$, are shown in Figure 3.1(a).

This optimisation applies in the case of general transition rates $h(s_i,s_j)$. For the following sections, I limited my consideration to the set of SVT models considered by Kiss et al in their work [59]. As previously defined in Chapter 2, these models only allow for transitions where one vertex-state changes, and the transition rates are the same for nodes that have the same number of neighbours in each vertex-state:

$$h(s_i,s_j)=\begin{cases} f_{A,B}(n_{\Sigma_1},n_{\Sigma_2},...n_{\Sigma_M}) & \text{if } s_i,\ s_j \text{ differ in one vertex-state } v \\ 0 & \text{otherwise.}\end{cases},$$

Recall that we can define the levels or classes of the system, where each level $\mathcal{C}^{k_{\Sigma_1},k_{\Sigma_2},...k_{\Sigma_M}}$ is the subset of states that have $k_{\Sigma_1},k_{\Sigma_2},...k_{\Sigma_M}$ vertices in the vertex-states $\Sigma_1,\Sigma_2,...\Sigma_M$ respectively. Here I choose these levels as the cells for our lumped space, so that the lumped system forms a population model. I consider first the elements of the new lumped infinitesimal generator $\mathbf{R}$ in the case of binary vertex-states, and then consider the matrix in the more general case of $M\geq 2$ vertex-states.

3.3 Binary vertex-state dynamics

To illustrate the results for binary vertex-state dynamics, let us consider the SIS epidemic model. The two states here are susceptible (S) and infected (I). We

note that the levels can now be written as $\mathcal{C}^k$, using just a single index k, the number of infected vertices. There are $N+1$ levels, and each level $\mathcal{C}^k$ directly corresponds to a cell L_k for $k = 1, 2, ..., N+1$.

The only entries in the approximate lumped infinitesimal generator $\mathbf{R}$ will be the tri-diagonal entries. To compute the entries $\mathbf{R}_{k,k+1}$ for all k, we need to calculate the average transition rates from every state in L_k to every state in L_{k+1}. The number of neighbours a vertex i has is equal to its degree d_i. For a given vertex i, there are $\binom{d_i}{m}\binom{N-1-d_i}{k-m}$ states in L_k where vertex i is susceptible and m of its neighbours are infected. In these states, the rate at which the vertex i becomes infected by its neighbours is τm, as in the standard SIS model. Summing this rate over all N vertices and all possible values of m, and using the fact that there are $\binom{N}{k}$ level k states, we find that the average of transition rates from states in L_k to states in L_{k+1} is

$$\mathbf{R}_{k,k+1} = \tau \frac{k!(N-k)!}{N!} \sum_{i=1}^{N} \sum_{m=0}^{k} m \binom{d_i}{m} \binom{N-1-d_i}{k-m}.$$

If we expand the binomial coefficient, we can simplify this to

$$\mathbf{R}_{k,k+1} = \tau \frac{k!(N-k)!}{N!} \sum_{i=1}^{N} d_i \sum_{m=0}^{k} \binom{d_i-1}{m-1} \binom{N-1-d_i}{k-m},$$

and then, applying the Chu-Vandermonde identity to the sum over m, we find

$$\mathbf{R}_{k,k+1} = \tau \frac{k!(N-k)!}{N!} \binom{N-2}{k-1} \sum_{i=1}^{N} d_i \tag{3.5}$$

$$= \tau k(N-k) \frac{z}{N-1}, \tag{3.6}$$

where $z = \frac{\sum_{i=1}^{N} d_i}{N}$ is the mean degree of all N vertices. Note that this is very similar to the homogeneous mean-field transition rate, usually derived from the exact stochastic models by applying a moment closure at the single level [74], as in Chapter 2. This result therefore provides a satisfying mathematical argument for this closure. We can also analyse the error of the approximate lumped solution, as shown in the following section. This gives us an alternative framework within which to analyse the error of the homogeneous mean-field approximation.

This approximate lumping method is applied to SIS dynamics on a small four-node network in Figure 3.1. The first panel illustrates the matrix multiplication $\mathbf{DQC} = \mathbf{R}$, showing the structure of these matrices. The second and third panels show the possible transitions out of a specific state in level $\mathcal{C}^2$, and the lumped transitions out of level $\mathcal{C}^2$ respectively. The fourth panel compares the results of the exact system of Kolmogorov equations and the approximate lumped equations.

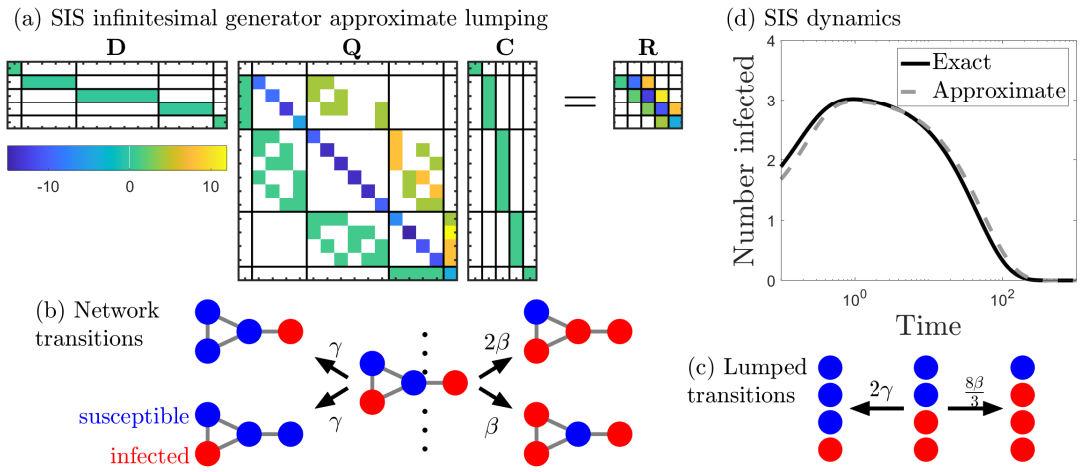


Figure 3.1: An illustration of the approximate lumping method applied to SIS dynamics on a small four-node network. Panel (a) shows the structure of the distributor matrix $\mathbf{D}$, the infinitesimal generator $\mathbf{Q}$ and the collector matrix $\mathbf{C}$, and shows how the multiplication $\mathbf{DQC}$ generates the smaller lumped generator $\mathbf{R}$. The colour scale shows the value of the matrix entries for $\tau = 4, \gamma = 1$. Panel (b) shows the structure of the network, and shows the possible transitions out of a specific state in level $\mathcal{C}^2$, where $k = 2$ (i.e. two infected nodes). The blue nodes are susceptible, and the red nodes are infected. There are two possible recovery transitions (in which either of the two infected nodes recover), and two possible infection transitions (in which either of the two susceptible nodes are infected by their infected neighbours). The arrows are annotated with the transition rates. The vertical dots indicate that there are other states possible that have two infected nodes. Panel (c) shows the corresponding transitions rates for the lumped system. Finally panel (d) compares the solution to the full system of Kolmogorov equations (the exact solution) with the solution to the lumped system ODEs (the approximate solution).

3.4 Error analysis of binary state dynamics

In this section I show how the approximate lumping method allows us to directly quantify the error of the approximate lumped solution. Since the generator $\mathbf{R}$ does not satisfy the lumping condition $\mathbf{QC} = \mathbf{CR}$ for this partitioning $\mathcal{L}$, the

solution $\mathbf{Y}$ to the equation $\dot{\mathbf{Y}} = \mathbf{R}^{\mathrm{T}}\mathbf{Y}$ will not contain the true probabilities of being in each lumped cell. $\mathbf{Y} = \mathbf{C}^{\mathrm{T}}\mathbf{X}$ will no longer hold for all time t. As such I defined the error of the approximate lumped solution as $\mathbf{Z} = \mathbf{C}^{\mathrm{T}}\mathbf{X} - \mathbf{Y}$, which I will analyse in this section.

To determine how the error changes in time, consider the error commutator $\boldsymbol{\Delta} = \mathbf{QCD} - \mathbf{CDQ}$. For exact lumping, $\mathbf{CD}$ commutes with $\mathbf{Q}$, and so this commutator is equal to zero. This does not hold in general for approximate lumping, however. Differentiating $\mathbf{Z}$ with respect to time, we find

$$\begin{aligned}
\dot{\mathbf{Z}} &= \mathbf{C}^{\mathrm{T}}\dot{\mathbf{X}} - \dot{\mathbf{Y}}, \\
&= \mathbf{C}^{\mathrm{T}}\mathbf{Q}^{\mathrm{T}}\mathbf{X} - \mathbf{R}^{\mathrm{T}}\mathbf{Y}, \\
&= \mathbf{C}^{\mathrm{T}}\mathbf{D}^{\mathrm{T}}\mathbf{C}^{\mathrm{T}}\mathbf{Q}^{\mathrm{T}}\mathbf{X} - \mathbf{R}^{\mathrm{T}}\mathbf{Y}, \\
&= \mathbf{C}^{\mathrm{T}}\left[\mathbf{Q}^{\mathrm{T}}\mathbf{D}^{\mathrm{T}}\mathbf{C}^{\mathrm{T}} + \boldsymbol{\Delta}^{\mathrm{T}}\right]\mathbf{X} - \mathbf{R}^{\mathrm{T}}\mathbf{Y}, \\
&= \mathbf{R}^{\mathrm{T}}\mathbf{Z} + \mathbf{C}^{\mathrm{T}}\boldsymbol{\Delta}^{\mathrm{T}}\mathbf{X},
\end{aligned}$$

where I have used the fact that $\mathbf{DC} = \mathbf{C}^{\mathrm{T}}\mathbf{D}^{\mathrm{T}} = \mathbf{I}$ and the decomposition $\mathbf{R} = \mathbf{DQC}$. If we now integrate, using the variation of constants formula [75], we find that

$$\mathbf{Z}(t) = \exp(\mathbf{R}^{\mathrm{T}}t)\mathbf{Z}(0) + \int_0^t \exp(\mathbf{R}^{\mathrm{T}}(t-s))\mathbf{C}^{\mathrm{T}}\boldsymbol{\Delta}^{\mathrm{T}}\mathbf{X}(s)\,\mathrm{d}s.$$

I assume that the initial lumped state $\mathbf{Y}(0) = \mathbf{C}^{\mathrm{T}}\mathbf{X}(0)$ is known, and so $\mathbf{Z}(0) = 0$. This allows the error to be written as

$$\mathbf{Z}(t) = \int_0^t \exp(\mathbf{R}^{\mathrm{T}}s)\mathbf{H}(t-s)\,\mathrm{d}s,$$

where I have applied a change of integration variable s to $t - s$, and grouped the final terms $\mathbf{C}^{\mathrm{T}}\boldsymbol{\Delta}^{\mathrm{T}}\mathbf{X}(t-s) = \mathbf{H}(t-s)$.

This expression has two main parts: one consisting of the exponential of the lumped infinitesimal generator $\mathbf{R}$, and one consisting of the factor $\mathbf{H}$, which is related to the commutator $\boldsymbol{\Delta}$ and the system state $\mathbf{X}(t-s)$. The following subsections break down the analysis by exploring each of these terms in turn. Section 3.4.1 and 3.4.2 investigate the lumped infinitesimal generator $\mathbf{R}$ and the error commutator matrix $\boldsymbol{\Delta}$ respectively. Section 3.4.3 and Section 3.4.4 then apply these results to two situations: first the situation where the system has a

non-zero stationary state, and then the situation where the system has only a non-zero quasi-stationary state. These two situations impact the way I treat the $\mathbf{X}(t-s)$ term and create differences for the time integral.

3.4.1 Lumped infinitesimal generator R matrix

To calculate the $\exp\left(\mathbf{R}^{\mathrm{T}}s\right)$ term, we need to know the structure of the $\mathbf{R}$ matrix, specifically its eigenvalues and eigenvectors. For SVT binary-state processes the $\mathbf{R}$ matrix is of the form

$$\mathbf{R} = \begin{pmatrix} \beta_0 & \alpha_0 & 0 & 0 & 0 \\ \gamma_1 & \beta_1 & \alpha_1 & 0 & 0 \\ 0 & \ddots & \ddots & \ddots & 0 \\ 0 & 0 & \gamma_{N-1} & \beta_{N-1} & \alpha_{N-1} \\ 0 & 0 & 0 & \gamma_N & \beta_N \end{pmatrix},$$

where the α_i entries describe the lumped rate of moving from cell L_i to L_{i+1}, the γ_i entries describe the lumped rate of moving from cell L_i to L_{i-1}, and $-\beta_i = \alpha_i + \gamma_i$. In situations where $\beta_0 = 0 = \alpha_0 = 0$ (as in SIS dynamics), or where $\gamma_N = \beta_N = 0$, we have absorbing states. For situations that do not have an absorbing state, such as SISa dynamics, $-\beta_0 = \alpha_0 \neq 0$ and $-\beta_N = \gamma_N \neq 0$. In both situations, we can see by eye that $\lambda_0 = 0$ is an eigenvalue of $\mathbf{R}$ with a corresponding right eigenvector $\mathbf{u}_0 = (1, 1, ..., 1)^T$. The Gershgorin circle theorem tells us that there can be no positive eigenvalues, since each Gershgorin disc is centred at $-\beta_i$ with radius $\alpha_i + \gamma_i = \beta_i$ [76].

We look to see if we can transform $\mathbf{R}$ into a symmetric matrix, since symmetric matrices have special properties. In the case of a real tridiagonal asymmetric matrix

$$\mathbf{T} = \begin{pmatrix} a_1 & b_1 & 0 & 0 & 0 \\ c_1 & a_2 & b_2 & 0 & 0 \\ 0 & c_2 & \ddots & \ddots & 0 \\ 0 & 0 & \ddots & \ddots & b_{N-1} \\ 0 & 0 & 0 & c_{N-1} & a_N \end{pmatrix},$$

it is possible to perform a similarity transformation and generate a symmetric tridiagonal matrix, if the off-diagonal products are all strictly positive, i.e. $b_i c_i >$

0 for all i. This is done by defining a transformation matrix $\mathbf{F}$

$$\mathbf{F} = \mathrm{diag}(\delta_1, ..., \delta_N)$$

where

$$\delta_i = \begin{cases} 1 & i = 1 \\ \sqrt{\frac{c_{i-1}...c_1}{b_{i-1}...b_1}} & i = 2, ..., N \end{cases}.$$

The transformed symmetric tridiagonal matrix is then given by

$$\mathbf{J} = \mathbf{F}^{-1}\mathbf{T}\mathbf{F} = \begin{pmatrix} a_1 & \sqrt{b_1 c_1} & 0 & 0 & 0 \\ \sqrt{b_1 c_1} & a_2 & \sqrt{b_2 c_2} & 0 & 0 \\ 0 & \sqrt{b_2 c_2} & \ddots & \ddots & 0 \\ 0 & 0 & \ddots & \ddots & \sqrt{b_{N-1} c_{N-1}} \\ 0 & 0 & 0 & \sqrt{b_{N-1} c_{N-1}} & a_N \end{pmatrix}.$$

In the case where $-\beta_0 = \alpha_0 \neq 0$ and $-\beta_N = \gamma_N \neq 0$, $\mathbf{R}$ is a real tridiagonal matrix that satisfies the condition of having strictly positive off-diagonal products. When $\beta_0 = \alpha_0 = 0$, or $\beta_N = \gamma_N = 0$, this is no longer true. As an example to show what we can learn about the eigenvalues and eigenvectors, I take here the example of SIS dynamics with $\beta_0 = \alpha_0 = 0$. In this case we can instead consider the matrix $\mathbf{R}_+$ formed by omitting the first row and first column of $\mathbf{R}$, which does satisfy the condition. The transformation matrix $\mathbf{F}$ can be constructed as defined above, giving the symmetric matrix

$$\mathbf{F}^{-1}\mathbf{R}_+\mathbf{F} = \begin{pmatrix} \beta_1 & \sqrt{\alpha_1 \gamma_2} & 0 & 0 & 0 \\ \sqrt{\alpha_1 \gamma_2} & \beta_2 & \sqrt{\alpha_2 \gamma_3} & 0 & 0 \\ 0 & \sqrt{\alpha_2 \gamma_3} & \ddots & \ddots & 0 \\ 0 & 0 & \ddots & \ddots & \sqrt{\alpha_{N-1} \gamma_N} \\ 0 & 0 & 0 & \sqrt{\alpha_{N-1} \gamma_N} & \beta_N \end{pmatrix}.$$

This is a real, symmetric matrix with positive above-diagonal entries, and so the eigenvectors of this matrix are orthonormal and the eigenvalues are real and distinct. Therefore we know that the eigenvalues of $\mathbf{R}_+$ are real, negative and distinct:

$$\lambda_0 = 0 > \lambda_1 > \lambda_2 > ... > \lambda_N.$$

These eigenvalues of $\mathbf{R}_+$ are the same as the eigenvalues of $\mathbf{R}$. To obtain the eigenvectors of $\mathbf{R}$, I first considered the orthonormal basis of eigenvectors

$\{\mathbf{w}_1, \mathbf{w}_2, ...\mathbf{w}_N\}$ of the symmetric matrix $\mathbf{F}^{-1}\mathbf{R}_+\mathbf{F}$. The right eigenvectors of $\mathbf{R}_+$ are therefore

$$\{\mathbf{F}\mathbf{w}_1, \mathbf{F}\mathbf{w}_2, ..., \mathbf{F}\mathbf{w}_N\},$$

and the left eigenvectors are

$$\{\mathbf{F}^{-1}\mathbf{w}_1^{\mathrm{T}}, \mathbf{F}^{-1}\mathbf{w}_2^{\mathrm{T}}, ..., \mathbf{F}^{-1}\mathbf{w}_N^{\mathrm{T}}\}.$$

These can then be transformed to give us the right and left eigenvectors of the full $\mathbf{R}$ matrix:

$$\mathbf{u}_k = \begin{pmatrix} 0 \\ \mathbf{F}^{-1}\mathbf{w}_k \end{pmatrix}$$

and

$$\mathbf{v}_k = \begin{pmatrix} \gamma_1\delta_1(\mathbf{w}_k)_1/\lambda_k, \mathbf{F}\mathbf{w}_k^{\mathrm{T}} \end{pmatrix}.$$

While the eigenvalues and eigenvectors could be computed without relating them to the symmetric case, this has shown that the eigenvalues are real and distinct, and the calculations may be computationally helpful. The lumped infinitesimal generator $\mathbf{R}$ can be expressed in Jordan normal form, $\mathbf{R} = \mathbf{U}\mathbf{\Lambda}\mathbf{V}$, where $\mathbf{U}$ is the matrix of right eigenvectors as columns, $\mathbf{V}$ is the matrix of left eigenvectors as rows, and $\mathbf{\Lambda}$ is the diagonal matrix of eigenvalues. This allows the exponential term in the error integral to be rewritten as

$$\exp\left(\mathbf{R}^{\mathrm{T}}s\right) = \mathbf{V}^{\mathrm{T}} \exp\left(\begin{bmatrix} 0 & 0 \\ 0 & \mathbf{\Lambda}_- \end{bmatrix} s\right) \mathbf{U}^{\mathrm{T}},$$

where $\mathbf{\Lambda}_- = \mathrm{diag}(\lambda_1, \lambda_2, ..., \lambda_N)$ is a submatrix of $\mathbf{\Lambda}$ containing the non-zero eigenvalues.

3.4.2 Error commutator Δ matrix

To construct the $\mathbf{\Delta}$ matrix, I examined the structure of the matrix $\mathbf{Q}$, and the collector and distributor matrices $\mathbf{C}$ and $\mathbf{D}$. For SVT models with binary states, it is possible to order the states such that the generator $\mathbf{Q}$ is block tridiagonal:

$$\mathbf{Q} = \begin{pmatrix} \mathbf{B}^0 & \mathbf{A}^0 & 0 & 0 & 0 & 0 \\ \mathbf{C}^1 & \mathbf{B}^1 & \mathbf{A}^1 & 0 & 0 & 0 \\ 0 & \mathbf{C}^2 & \mathbf{B}^2 & \mathbf{A}^2 & 0 & 0 \\ 0 & 0 & \ddots & \ddots & \ddots & 0 \\ 0 & 0 & 0 & \mathbf{C}^{N-1} & \mathbf{B}^{N-1} & \mathbf{A}^{N-1} \\ 0 & 0 & 0 & 0 & \mathbf{C}^N & \mathbf{B}^N \end{pmatrix}. \tag{3.7}$$

Note that this generator has the same tridiagonal structure as the generator defined in equation 2.1 for SIS dynamics, but here the matrices $\mathbf{A}^k$, $\mathbf{B}^k$ and $\mathbf{C}^k$ currently describe a more general set of transition rates. $\mathbf{A}^k$ is a matrix with $c_k = \binom{N}{k}$ rows and c_{k+1} columns. It contains the rates at which states with k nodes in a state A transition to states with $k+1$ nodes in state A. $\mathbf{C}^k$ is a matrix with c_k rows and c_{k-1} columns. It contains the rates at which states with k nodes in state A transition to states with $k-1$ nodes in state A. With the states ordered in this way, the collector matrix $\mathbf{C}$ takes the form

$$\mathbf{C} = \begin{pmatrix} \mathbf{e}_0 & 0 & 0 & 0 & 0 \\ 0 & \mathbf{e}_1 & 0 & 0 & 0 \\ 0 & 0 & \ddots & 0 & 0 \\ 0 & 0 & 0 & \mathbf{e}_{N-1} & 0 \\ 0 & 0 & 0 & 0 & \mathbf{e}_N \end{pmatrix}, \tag{3.8}$$

where $\mathbf{e}_k$ is here a column vector of length c_k with every entry equal to 1. The distributor matrix $\mathbf{D}$ takes the form

$$\mathbf{D} = \begin{pmatrix} \frac{1}{c_0}\mathbf{e}_0^{\mathrm{T}} & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{c_1}\mathbf{e}_1^{\mathrm{T}} & 0 & 0 & 0 \\ 0 & 0 & \ddots & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{c_{N-1}}\mathbf{e}_{N-1}^{\mathrm{T}} & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{c_N}\mathbf{e}_N^{\mathrm{T}} \end{pmatrix}. \tag{3.9}$$

It is easy to confirm from the structure of the collector matrix and the distributor matrix that the product $\mathbf{DC} = \mathbf{I}$. It is also straightforward to see that the product $\mathbf{CD}$ is a matrix with the following structure:

$$\mathbf{CD} = \begin{pmatrix} \frac{1}{c_0}\mathbf{E}^0 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{c_1}\mathbf{E}^1 & 0 & 0 & 0 \\ 0 & 0 & \ddots & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{c_{N-1}}\mathbf{E}^{N-1} & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{c_N}\mathbf{E}^N \end{pmatrix}, \tag{3.10}$$

where $\mathbf{E}^k = \mathbf{e}_k\mathbf{e}_k^{\mathrm{T}}$ is a $c_k \times c_k$ matrix with each term equal to 1.

The $\mathbf{\Delta}$ matrix is defined as $\mathbf{\Delta} = \mathbf{QCD} - \mathbf{CDQ}$ The matrix $\mathbf{\Delta}$ appears in the integral in the term $\mathbf{H}(t-s) = \mathbf{C}^{\mathrm{T}}\mathbf{\Delta}^{\mathrm{T}}\mathbf{X}(t-s)$. While we can calculate

$\mathbf{\Delta} = \mathbf{QCD} - \mathbf{CDQ}$ directly, it is more useful to calculate $\mathbf{\Delta C}$:

$$\mathbf{\Delta C} = \begin{pmatrix} \mathbf{b}_0 & \mathbf{a}_0 & 0 & 0 & 0 & 0 \\ \mathbf{c}_1 & \mathbf{b}_1 & \mathbf{a}_1 & 0 & 0 & 0 \\ 0 & \mathbf{c}_2 & \mathbf{b}_2 & \mathbf{a}_2 & 0 & 0 \\ 0 & 0 & \ddots & \ddots & \ddots & 0 \\ 0 & 0 & 0 & \mathbf{c}_{N-1} & \mathbf{b}_{N-1} & \mathbf{a}_{N-1} \\ 0 & 0 & 0 & 0 & \mathbf{c}_N & \mathbf{b}_N \end{pmatrix},$$

where

$$\begin{aligned} \mathbf{a}_k &= (\frac{1}{c_{k+1}}\mathbf{A}^k\mathbf{E}^{k+1} - \frac{1}{c_k}\mathbf{E}^k\mathbf{A}^k)\mathbf{e}_{k+1}, \\ \mathbf{b}_k &= \frac{1}{c_k}(\mathbf{B}^k\mathbf{E}^k - \mathbf{E}^k\mathbf{B}^k)\mathbf{e}_k \quad \text{and} \\ \mathbf{c}_k &= (\frac{1}{c_{k-1}}\mathbf{C}^k\mathbf{E}^{k-1} - \frac{1}{c_k}\mathbf{E}^k\mathbf{C}^k)\mathbf{e}_{k-1}. \end{aligned}$$

Here I introduce $\mathbf{q}_k = \mathbf{A}^k\mathbf{e}_{k+1}$. This vector relates to the characteristics of the original graph: its jth coordinate is equal to the probability of transitioning out of the jth state of class $\mathcal{C}^k$, to any state in class $\mathcal{C}^{k+1}$. For SIS dynamics, this is equal to the number of SI edges in the jth state, multiplied by the infection rate. I also define $\bar{q_k} = \frac{1}{c_k}\sum_{j=1}^{c_k}(\mathbf{q}_k)_j$ which is the average of the entries of $\mathbf{q}_k$, i.e. the average rate of transition from states in class $\mathcal{C}^k$ to any of the states in class $\mathcal{C}^{k+1}$. Using these expressions we can write

$$\mathbf{a}_k = \mathbf{q}_k - \bar{q_k}\mathbf{e}_k.$$

Equivalent expressions $\mathbf{p}_k = \mathbf{C}^k\mathbf{e}_{k+1}$ and $\bar{p_k} = \frac{1}{c_k}\sum_{j=1}^{c_k}(\mathbf{p}_k)_j$ let us also write

$$\mathbf{c}_k = \mathbf{p}_k - \bar{p_k}\mathbf{e}_k.$$

These terms will appear in the error integral. In this way I have highlighted how the error is dependent on the rates of transition out of each state in a level, and the difference between each of these rates and the average rate.

3.4.3 Processes with no absorbing state

Here I return to the integral for the error:

$$\mathbf{Z}(t) = \int_0^t \exp(\mathbf{R}^{\mathrm{T}}s)\mathbf{H}(t-s)\,\mathrm{d}s,$$

where $\mathbf{H}(t-s) = \mathbf{C}^{\mathrm{T}}\mathbf{\Delta}^{\mathrm{T}}\mathbf{X}(t-s)$.

I considered first the processes that have no absorbing state, for example SISa dynamics. These processes have a steady state $\mathbf{X}^*$ which will be reached after a finite time. If we assume that we begin the error calculation already in this steady state, then $\mathbf{H}(t-s)$ is some constant $\mathbf{H} = \mathbf{C}^{\mathrm{T}}\mathbf{\Delta}^{\mathrm{T}}\mathbf{X}^*$ for all time. Using this fact, and the Jordan normal form for $\mathbf{R}$ found in Section 3.4.1, the integration becomes straightforward:

$$\begin{aligned}
\mathbf{Z}(t) &= \int_0^t \mathbf{V}^{\mathrm{T}} \exp\left(\begin{bmatrix} 0 & 0 \\ 0 & \mathbf{\Lambda}_- \end{bmatrix} s \right) \mathbf{U}^{\mathrm{T}}\mathbf{H} \mathrm{d}s, \\
&= \mathbf{V}^{\mathrm{T}} \begin{bmatrix} t & 0 \\ 0 & (\exp(\mathbf{\Lambda}_- t) - \mathbf{I})(\mathbf{\Lambda}_-)^{-1} \end{bmatrix} \mathbf{U}^{\mathrm{T}}\mathbf{H}, \\
&= \left(t\mathbf{v}_0^{\mathrm{T}}\mathbf{u}_0^{\mathrm{T}} + \sum_{i=1}^{N} \frac{1}{|\lambda_i|} \left[1 - \mathrm{e}^{\lambda_i t}\right] \mathbf{v}_i^{\mathrm{T}}\mathbf{u}_i^{\mathrm{T}} \right) \mathbf{H}.
\end{aligned}$$

We assume here that we can calculate $\mathbf{v}_i, \mathbf{u}_i, \lambda_i$ (the right eigenvectors, left eigenvectors, and eigenvalues respectively) from the matrix $\mathbf{R}$. This appears a fair assumption, since $\mathbf{R}$ is a tridiagonal $N \times N$ matrix. However, since we cannot in general solve for $\mathbf{X}^*(s)$, we are limited to finding a bound rather than an exact value for the $\mathbf{H}$ matrix.

Noting the tridiagonal structure of $\mathbf{\Delta C}$, and the fact that $-\mathbf{b}_k = \mathbf{a}_k + \mathbf{c}_k$, we can write the factor $\mathbf{H}$ as

$$\mathbf{H} = \begin{pmatrix} -\mathbf{a}_0^{\mathrm{T}}\mathbf{X}_0^* + \mathbf{c}_1^{\mathrm{T}}\mathbf{X}_1^* \\ \mathbf{a}_0^{\mathrm{T}}\mathbf{X}_0^* - \mathbf{a}_1^{\mathrm{T}}\mathbf{X}_1^* + \mathbf{c}_2^{\mathrm{T}}\mathbf{X}_2^* - \mathbf{c}_1^{\mathrm{T}}\mathbf{X}_1^* \\ \vdots \\ \mathbf{a}_{N-2}^{\mathrm{T}}\mathbf{X}_{N-2}^* - \mathbf{a}_{N-1}^{\mathrm{T}}\mathbf{X}_{N-1}^* + \mathbf{c}_N^{\mathrm{T}}\mathbf{X}_N^* - \mathbf{c}_{N-1}^{\mathrm{T}}\mathbf{X}_{N-1}^* \\ \mathbf{a}_{N-1}^{\mathrm{T}}\mathbf{X}_{N-1}^* - \mathbf{c}_N^{\mathrm{T}}\mathbf{X}_N^* \end{pmatrix},$$

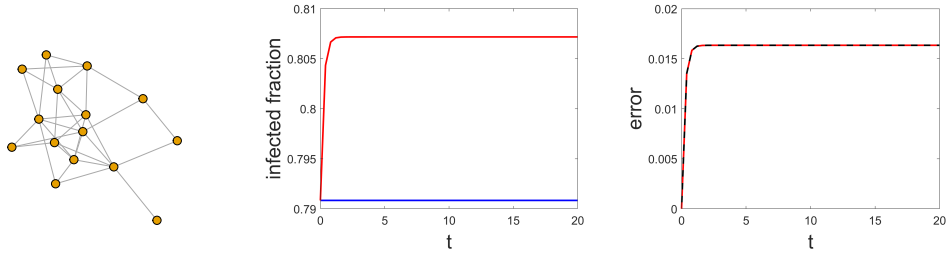
where $\mathbf{X}_j^*$ is a subset of the probability distribution $\mathbf{X}^*$, containing only the probabilities of being in the states in class $\mathcal{C}^j$. This shows further that the more likely a state in class $\mathcal{C}^j$ is to be observed in the steady state, the larger the contribution of the corresponding component of the $\mathbf{a}_j$ and $\mathbf{c}_j$ terms. We can also see that the sum of components in $\mathbf{H}$ is equal to zero. This is relevant since $\mathbf{u}_0 = (1, \ldots, 1)^T$, and so the term proportional to t vanishes. In this way the

expression for the error reduces to

$$\begin{aligned}\mathbf{Z}(t) &= \left(\sum_{i=1}^{N} \frac{1}{|\lambda_i|}\left[1 - \mathrm{e}^{\lambda_i t}\right]\mathbf{v}_i^{\mathrm{T}}\mathbf{u}_i^{\mathrm{T}}\right)\mathbf{H} \\ &= \sum_{i=1}^{N} \frac{1}{|\lambda_i|}\left[1 - \mathrm{e}^{\lambda_i t}\right]\mathbf{v}_i^{\mathrm{T}}\mathbf{u}_i^{\mathrm{T}}\mathbf{H}.\end{aligned}$$

This is an explicit solution of the error. However, the eigenvector decomposition can be unstable. For small networks, where $\mathbf{Q}$ can be feasibly constructed, it can be simpler to solve the ODEs for $\mathbf{Z}$ directly. For these small networks the exact solution from the full Kolmogorov equations can also be solved directly, and compared with the lumped solution. This allows us to check and confirm the validity of the solutions for $\mathbf{Z}$. I solved these equations for SISa dynamics on three small networks, all with $N = 15$ nodes: an Erdős-Rényi network, a star network, and a cycle network. I first solved the full Kolmogorov equations, to find the true steady state $\mathbf{X}^*$. I then solved the lumped system, assuming that the system starts in the state $\mathbf{Y}^* = \mathbf{C}^{\mathrm{T}}\mathbf{X}^*$. I then solved the ODEs for $\mathbf{Z}$, by constructing the appropriate $\mathbf{R}$ and $\mathbf{\Delta}$ matrices, and using the previously calculated $\mathbf{X}^*$ solution in the factor $\mathbf{H}$.

For each of the three networks, I have plotted the calculated approximation error alongside the difference between the lumped solution and the true solution, in Figures 3.2, 3.3 and 3.4. As expected, the two results agree for each network. Since both the lumped solution and the true solution have a non-zero steady-state, the error is constant once the lumped system has relaxed into its steady-state from its initial value $\mathbf{Y}_0 = \mathbf{Y}^*$. In all three cases the lumped solution is an overestimate of the true steady state value.

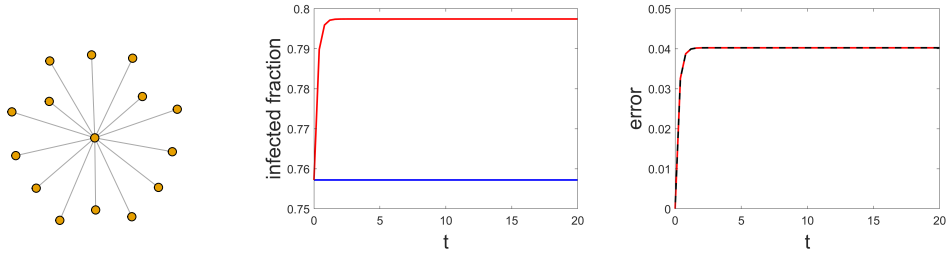


(a) Erdős-Rényi network

(b) Plot of lumped solution vs full solution

(c) Plot of the absolute error

Figure 3.2: Plots showing the results of the approximate lumping analysis of SISa dynamics on an Erdős-Rényi network. The network is shown in (a), and has $N = 15$ nodes, $p = \frac{4}{15}$. The SISa parameters are $\tau = 0.9091, \gamma = 1, \phi = 1$. The plot in (b) compares the infected fraction of the network found by solving the approximate lumped system (the red line) with the true fraction found by solving the exact full system (the blue line). The difference between the solutions looks substantial at this axis scale, but I have in fact zoomed in to show what is actually a small difference more clearly. The plot in (c) shows the absolute error between these two solutions with time, solved exactly using the calculation presented in this section (black line), and calculated by taking the difference of the two solutions shown in (b) (dashed red line).

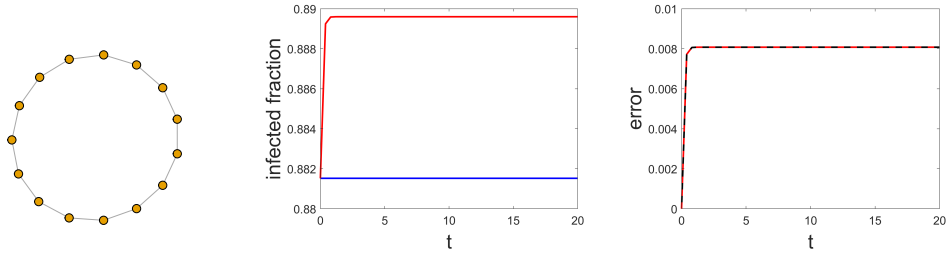


(a) Star network

(b) Plot of lumped solution vs full solution

(c) Plot of the absolute error

Figure 3.3: Plots showing the results of the approximate lumping analysis of SISa dynamics on an star network. The network is shown in (a), and has $N = 15$ nodes. The SISa parameters are $\tau = 2, \gamma = 1, \phi = 1$. The plot in (b) compares the infected fraction of the network found by solving the approximate lumped system (the red line) with the true fraction found by solving the exact full system (the blue line). As in Figure 3.2, the difference between the solutions looks substantial at this axis scale, but I have in fact zoomed in to show what is actually a small difference more clearly. The plot in (c) shows the absolute error between these two solutions with time, solved exactly using the calculation presented in this section (black line), and calculated by taking the difference of the two solutions shown in (b) (dashed red line).



(a) Cycle network

(b) Plot of lumped solution vs full solution

(c) Plot of the absolute error

Figure 3.4: Plots showing the results of the approximate lumping analysis of SISa dynamics on a cycle network. The network is shown in (a), and has $N = 15$ nodes. The SISa parameters are $\tau = 4, \gamma = 1, \phi = 1$. The plot in (b) compares the infected fraction of the network found by solving the approximate lumped system (the red line) with the true fraction found by solving the exact full system (the blue line). As in Figures 3.2 and 3.3, the difference between the solutions looks substantial at this axis scale, but I have in fact zoomed in to show what is actually a small difference more clearly. The plot in (c) shows the absolute error between these two solutions with time, solved exactly using the calculation presented in this section (black line), and calculated by taking the difference of the two solutions shown in (b) (dashed red line).

Note that this method for finding the error necessarily involves solving the full equations as its first step. While this makes the use of the approximation and the calculation of its error somewhat redundant in this case, this work is simply exploring how to formulate the error equation. This allows us to see how the error is connected to network structure and dynamics. For larger networks, the $\mathbf{Q}$ matrix becomes too computationally demanding to construct, and so we cannot solve the full system. The consequences of this are discussed in Section 3.6.3 of this chapter.

3.4.4 Processes with an absorbing state

Many common binary-state processes have an absorbing state, for example SIS dynamics. This means that the endemic state is quasi-stationary. If we return to

the expression for the error integral

$$\mathbf{Z}(t) = \int_0^t \mathbf{V}^{\mathrm{T}} \exp\left(\begin{bmatrix} 0 & 0 \\ 0 & \mathbf{\Lambda}_- \end{bmatrix} s \right) \mathbf{U}^{\mathrm{T}} \mathbf{H} \mathrm{d}s,$$

we see that we cannot simply take the factor $\mathbf{H}$ as constant when the state is in this quasi-stationary state. Instead we can examine the quasi-stationary state and see how this changes with time. For processes like this with an absorbing state, the infinitesimal generator $\mathbf{Q}$ takes the form

$$\mathbf{Q} = \begin{pmatrix} 0 & \mathbf{0}^{\mathrm{T}} \\ \tau & \mathbf{T} \end{pmatrix},$$

where $\mathbf{T}$ is a lossy generator defined on the subspace containing all the states with $k > 0$ [77]. This lossy generator governs a lossy Markov chain, and the limiting distribution to this Markov chain is the quasi-stationary distribution. By adding a sufficiently large positive constant to the diagonal of $\mathbf{T}$, we can apply the Perron-Frobenius theorem [78] to show that $\mathbf{T}$ has a strictly negative largest eigenvalue $\mu < 0$, and that there exists a unique probability vector $\mathbf{W}^*$ with strictly positive components which satisfies

$$\mu \mathbf{W}^{*\mathrm{T}} = \mathbf{W}^{*\mathrm{T}} \mathbf{T}.$$

The quasi-stationary state is therefore equivalent to the left eigenvector of the lossy generator $\mathbf{T}$ corresponding to this largest eigenvalue μ.

The (lossy) transition matrix of $\mathbf{X}(t)$ is given by

$$\mathbf{T}(t) = \exp(\mathbf{T}t) = \sum_{n=0}^{\infty} \frac{\mathbf{T}^n t^n}{n!},$$

and so the quasi-stationary state of the system will decay in time according to the exponential $\exp(\mu t)$:

$$\mathbf{W}^{*\mathrm{T}} \mathbf{T}(t) = \mathbf{W}^{*\mathrm{T}} \sum_{n=0}^{\infty} \frac{(\mu_1 t)^n}{n!} = \exp(\mu t) \mathbf{W}^{*\mathrm{T}}.$$

Assuming that the system starts in this quasi-stationary state, then after a time s the full state of the system will be

$$\mathbf{X}(s) = \begin{pmatrix} 1 - \exp(\mu s) \\ \exp(\mu s) \mathbf{W}^* \end{pmatrix},$$

where I have deduced the probability of being in the absorbing state by using the fact that the sum of components in $\mathbf{X}(s)$ must sum to 1 at all times. Here I also return to our expression for the factor $\mathbf{H}$. Note that the term $X_0 = 1 - \exp(\mu s)$ only appears with $\mathbf{a}_0$. Reminding ourselves of the definition of $\mathbf{a}_k$,

$$\mathbf{a}_k = \frac{1}{c_{k+1}}(\mathbf{A}^k\mathbf{E}^{k+1} - \mathbf{E}^k\mathbf{A}^k)\mathbf{e}_{k+1},$$

we can see that $\mathbf{A}^0 = \mathbf{0}^{\mathrm{T}}$ for systems with an absorbing state. Therefore $\mathbf{a}_0 = 0$, and we can write $\mathbf{H}(s) = \exp(\mu s)\mathbf{H}^*$, where

$$\mathbf{H}^* = \begin{pmatrix} \mathbf{c}_1^{\mathrm{T}}\mathbf{W}_1^* \\ -\mathbf{a}_1^{\mathrm{T}}\mathbf{W}_1^* + \mathbf{c}_2^{\mathrm{T}}\mathbf{W}_2^* - \mathbf{c}_1^{\mathrm{T}}\mathbf{W}_1^* \\ \vdots \\ \mathbf{a}_{N-2}^{\mathrm{T}}\mathbf{W}_{N-2}^* - \mathbf{a}_{N-1}^{\mathrm{T}}\mathbf{W}_{N-1}^* + \mathbf{c}_N^{\mathrm{T}}\mathbf{W}_N^* - \mathbf{c}_{N-1}^{\mathrm{T}}\mathbf{W}_{N-1}^* \\ \mathbf{a}_{N-1}^{\mathrm{T}}\mathbf{W}_{N-1}^* - \mathbf{c}_N^{\mathrm{T}}\mathbf{W}_N^* \end{pmatrix}.$$

The error integral is therefore very similar to the case with no absorbing state, except with the inclusion of the exponential $\exp(\mu(t-s))$ term:

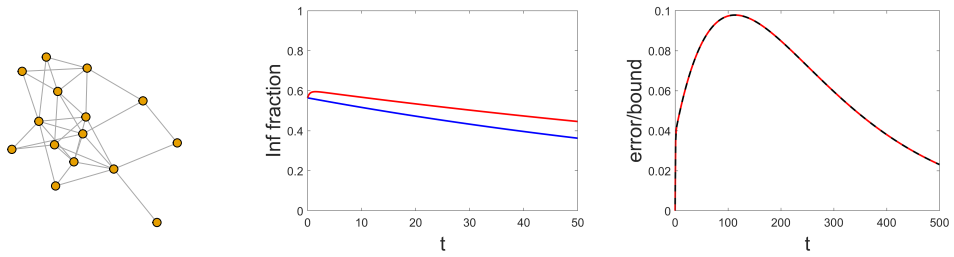
$$\begin{aligned} \mathbf{Z}(t) &= \int_0^t \mathbf{V}^{\mathrm{T}} \exp\left(\begin{bmatrix} 0 & 0 \\ 0 & \mathbf{\Lambda}_- \end{bmatrix} s \right) \mathbf{U}^{\mathrm{T}} \exp(\mu(t-s))\mathbf{H}^* \mathrm{d}s \\ &= \int_0^t \sum_{i=1}^{N} \mathbf{v}_i^{\mathrm{T}}\mathbf{u}_i^{\mathrm{T}} \exp(\lambda_i s) \exp(\mu(t-s))\mathbf{H}^* \mathrm{d}s \\ &= \sum_{i=1}^{N} \mathbf{v}_i^{\mathrm{T}}\mathbf{u}_i^{\mathrm{T}} \frac{1}{\lambda_i - \mu} \left(\exp(\lambda_i t) - \exp(\mu t)\right) \mathbf{H}^*. \end{aligned}$$

To check this result on small networks, some care has to be taken to find the quasi-stationary state $\mathbf{W}^*$ and the eigenvalue μ. First, I solved the full ODEs to a short time ($t = 50$) to get it near quasi-stationary. Then I normalised the resulting distribution (with the entry for the zero state being zero) to find $\mathbf{W}^*$ and solved the full ODEs for a long time ($t = 500$) with this as the starting distribution. Comparing successive results for the infected fraction of the network allowed me to estimate the value for μ. Finally, I used this to solve the ODEs for the error $\mathbf{Z}$, and compared this with the difference between the lumped solution and the full solution.

This analysis was performed for SIS dynamics on the same three networks as in Section 3.4.3: an Erdős-Rényi network, a star network, and a cycle network,

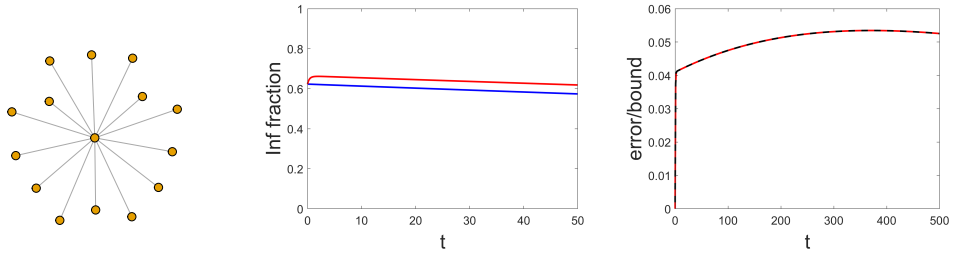
all with $N = 15$ nodes. The results are shown in Figures 3.5, 3.6, and 3.7. Again, as expected, the directly-computed error agrees with the difference between the lumped solution and full solution for each of the three networks. The lumped solution is also again an overestimate in these three examples. We see that, in the case of the Erdős-Rényi and star networks, the error starts from zero, grows but eventually turns over and starts to decay again. Ultimately, as the two solutions tend to the absorbing state zero, the error will tend to zero. This will be true for the cycle network too, but the process is happening more slowly and so isn't apparent by $t = 500$.

As in the non-absorbing state case, this analysis becomes non-trivial for large networks. Further discussion of this is in Section 3.6.3.



(a) Erdős-Rényi network (b) Plot of lumped solution vs full solution (c) Plot of the absolute error

Figure 3.5: Plots showing the results of the approximate lumping analysis of SIS dynamics on an Erdős-Rényi network. The network is shown in (a) and has $N = 15$ nodes, with $p = \frac{4}{15}$. The SIS parameters are $\tau = 0.9091, \gamma = 1, \phi = 1$. The plot in (b) compares the infected fraction of the network found by solving the approximate lumped system (the red line) with the true fraction found by solving the exact full system (the blue line), from $t = 0$ to $t = 50$. The plot in (c) shows the absolute error between these two solutions with time, for a longer time period of $t = 500$, solved exactly using the calculation presented in this section (black line), and calculated by taking the difference of the two solutions shown in (b) (dashed red line). Here we can see how, after reaching a maximum deviation of close to 0.1, the two solutions slowly move closer together as they both decay gradually to the absorbing state.

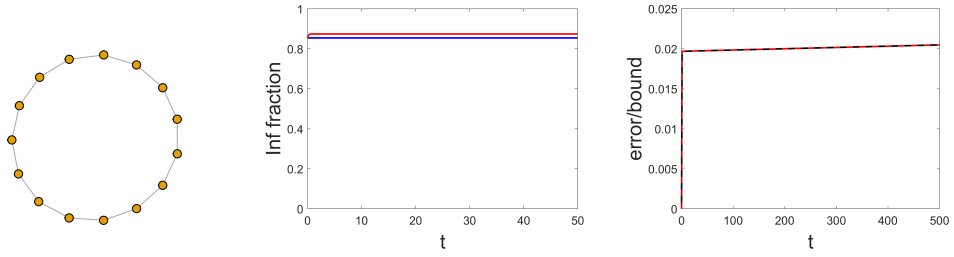


(a) Star network

(b) Plot of lumped solution vs full solution

(c) Plot of the absolute error

Figure 3.6: Plots showing the results of the approximate lumping analysis for SIS dynamics on a star network. The network is shown in (a), and has $N = 15$ nodes. The SIS parameters are $\tau = 2, \gamma = 1$. The plot in (b) compares the infected fraction of the network found by solving the approximate lumped system (the red line) with the true fraction found by solving the exact full system (the blue line), from $t = 0$ to $t = 50$. The plot in (c) shows the absolute error between these two solutions with time, for a longer time period of $t = 500$, solved exactly using the calculation presented in this section (black line), and calculated by taking the difference of the two solutions shown in (b) (dashed red line). We see that the difference between the two solutions reaches a peak before starting to decrease, as with the Erdős-Rényi network case in Figure 3.5.



(a) Cycle network

(b) Plot of lumped solution vs full solution

(c) Plot of the absolute error

Figure 3.7: Plots showing the results of the approximate lumping analysis for SIS dynamics on a cycle network. The network is shown in (a), and has $N = 15$ nodes. The SIS parameters are $\tau = 4, \gamma = 1$. The plot in (b) compares the infected fraction of the network found by solving the approximate lumped system (the red line) with the true fraction found by solving the exact full system (the blue line), from $t = 0$ to $t = 50$. The plot in (c) shows the absolute error between these two solutions with time, for a longer time period of $t = 500$, solved exactly using the calculation presented in this section (black line), and calculated by taking the difference of the two solutions shown in (b) (dashed red line). Unlike in Figures 3.5 and 3.6, here the time period is not long enough to see the difference between the two solutions reach a maximum and start to decrease. This is unsurprising, as we can see from the solutions in (b) that the rate of decay to the absorbing state is very slow.

3.5 Non-binary vertex-state dynamics

Here I lift the restriction of binary vertex-states, and study the more general set of dynamics where there are $M \geq 2$ vertex-states. One example is the SIR epidemic model, where $M = 3$. Vertices can transition from susceptible (S) to infected (I), and from infected to recovered (R). A more general case of non-binary vertex-state dynamics, which I consider in this section, lets a vertex v transition from any vertex-state A to any other vertex-state B at a transition rate $f_{A,B}(n_{\Sigma_1}, n_{\Sigma_2}, ..., n_{\Sigma_M}) = \tau_{AB} n_B(v)$.

Again, I considered the cells of the lumped state-space to align with the levels of the system $\mathcal{C}^{k_{\Sigma_1}, k_{\Sigma_2}, ... k_{\Sigma_M}}$. We can also think of this as assigning to each level

L_i a vector $\mathbf{k}_i = (k_{\Sigma_1}, k_{\Sigma_2}, ...k_{\Sigma_M})$ which lists the number of vertices in each of the M vertex-states. For a system of N vertices and M vertex-states there are $\binom{N+M-1}{N}$ levels, and so there are $\binom{N+M-1}{N}$ lumped cells. The optimisation process used before still holds, and so the lumped infinitesimal generator remains as

$$\mathbf{R}_{ij} = \frac{1}{\#L_i} \sum_{s_k \in L_i} \mathbf{P}_{kj}. \tag{3.11}$$

To demonstrate that $\mathbf{R}_{ij}$ can be calculated as in the binary state case, let us take as an example a cell L_i described by $\mathbf{k}_i = (k_A, k_B, ..., k_M)$ and a cell L_j described by $\mathbf{k}_j = (k_A - 1, k_B + 1, ..., k_M)$, where $\mathbf{k}_i$ and $\mathbf{k}_j$ differ only in the first two terms. A transition from a state in cell L_i to a state in the cell L_j is caused by a vertex in vertex-state A transitioning to be in vertex-state B.

For a given vertex v, there are $\binom{d_v}{m_B}\binom{N-1-d_v}{k_B-m_B}\binom{N-1-k_B}{k_A-1}\binom{N-k_A-k_B}{k_C,...,k_M}$ states in L_i where vertex v is in vertex-state A and m_B of its neighbours are in vertex-state B. In these states, the rate at which the vertex v transitions to vertex-state B due to its neighbours is $\tau_{AB} m_B$. Summing this rate over all N vertices and all possible values of m_B, and using the fact that there are $\binom{N}{k_A}\binom{N-k_A}{k_B}\binom{N-k_A-k_B}{k_C,...,k_M}$ states in L_i, we find that the average of transition rates from states in L_i to states in L_j is

$$\begin{aligned}
\mathbf{R}_{ij} &= \tau_{BA} \frac{\binom{N-1-k_B}{k_A-1}}{\binom{N}{k_A}\binom{N-k_A}{k_B}} \sum_{i=1}^{N} \sum_{m_B=0}^{k_B} m_B \binom{d_i}{m_B} \binom{N-d_i-1}{k_B-m_B} \\
&= \tau_{BA} \frac{\binom{N-1-k_B}{k_A-1}}{\binom{N}{k_A}\binom{N-k_A}{k_B}} \sum_{i=1}^{N} d_i \sum_{m_B=0}^{k_B} \binom{d_i-1}{m_B-1} \binom{N-d_i-1}{k_B-m_B}.
\end{aligned}$$

As before, we can apply the Chu-Vandermonde identity to simplify this and we find that

$$\begin{aligned}
\mathbf{R}_{ij} &= \tau_{BA} \frac{\binom{N-1-k_B}{k_A-1}}{\binom{N}{k_A}\binom{N-k_A}{k_B}} \binom{N-2}{k_B-1} \sum_{i=1}^{N} d_i \\
&= \tau_{BA} k_A \cdot \frac{k_B}{N-1} \cdot \frac{\sum_{i=1}^{N} d_i}{N}.
\end{aligned}$$

As one might expect from the binary vertex-state analysis, this result is again the same as that derived by applying moment closures at the single level to the exact master equations.

3.6 Discussion and future research directions

3.6.1 Summary

In this chapter I have applied approximate lumping to the class of single-vertex transition models on networks, and in doing so I have demonstrated how calculation of the lumped infinitesimal generator $\mathbf{R}$ recovers the homogeneous mean-field rates, in the case of both binary and non-binary vertex-states, providing rigorous mathematical reasoning for these rates.

I have identified three main areas of this work that I feel hold potential for future investigation. Firstly, it could be interesting to explore the results for non-linear infection rates. The following section 3.6.2 presents a derivation and discussion of the lumped infinitesimal generator $\mathbf{R}$ rates for the case where the vertex infection rates go as $\sim m^2$, where m is the number of infected neighbours. Secondly, I would like to extend the evaluation of the approximation error to larger networks. My initial investigation of this is discussed in Section 3.6.3. Finally, there might be scope to consider less coarse lumping, with fewer states per cell. This would hopefully allow for more accurate approximations than those derived in this chapter, while still improving the tractability of the system ODEs.

3.6.2 Non-linear functions of neighbour state

I have so far focused on dynamics where the vertex transition rates depend on either constant rates, or linearly on the states of their neighbours. However, there exist multiple SVTs with transition rates that depend non-linearly on neighbour states, such as the non-zero temperature Ising-Glauber model [79], the nonlinear q-voter model [80], and threshold models of opinion dynamics [81]. While I was unable to generalise to these more complex transition rates, I was able to adjust the binary-state calculation to consider a non-linear vertex infection rate that depends on m^q, where m is the number of infected neighbours and $q \geq 2$. Here I show how the calculation can be performed for $q = 2$. The method demonstrated can theoretically be extended for $q \geq 2$, although it quickly becomes unwieldy.

Using the same reasoning as before, the transition rate $\mathbf{R}_{k,k+1}$ is now given by

$$\mathbf{R}_{k,k+1} = \tau \frac{k!(N-k)!}{N!} \sum_{i=1}^{N} \sum_{m=0}^{k} m^2 \binom{d_i}{m} \binom{N-1-d_i}{k-m}.$$

We can make use of the absorption/extraction identity for binomial coefficients [82] to expand the $\binom{d_i}{m}$ coefficient out to $\mathcal{O}(\frac{1}{m^2})$, and simplify accordingly:

$$\begin{aligned}
\mathbf{R}_{k,k+1} &= \tau \frac{k!(N-k)!}{N!} \sum_{i=1}^{N} \sum_{m=0}^{k} m^2 \frac{d_i}{m} \frac{d_i - 1}{m-1} \binom{d_i - 2}{m-2} \binom{N-1-d_i}{k-m} \\
&= \tau \frac{k!(N-k)!}{N!} \left(\sum_{i=1}^{N} \sum_{m=0}^{k} d_i(d_i-1) \binom{d_i-2}{m-2} \binom{N-1-d_i}{k-m} \right. \\
&\left. + \sum_{i=1}^{N} \sum_{m=0}^{k} d_i \frac{d_i - 1}{m-1} \binom{d_i-2}{m-2} \binom{N-1-d_i}{k-m} \right).
\end{aligned}$$

Then, absorbing any remaining m terms back into the binomial coefficients, and again making use of the Chu-Vandermonde identity, we find

$$\mathbf{R}_{k,k+1} = \tau \frac{k!(N-k)!}{N!} \left(\binom{N-3}{k-2} \sum_{i=1}^{N} d_i(d_i-1) + \binom{N-2}{k-1} \sum_{i=1}^{N} d_i \right).$$

As expected, this expression collapses to be the same as the linear case if all $d_i \leq 1$. Ordering the terms in powers of d_i,

$$\mathbf{R}_{k,k+1} = \tau \frac{k!(N-k)!}{N!} \binom{N-3}{k-2} \left(\sum_{i=1}^{N} d_i^2 + \left(\frac{N-2}{k-1} - 1 \right) \sum_{i=1}^{N} d_i \right).$$

When $N - 2 < 2(k-1)$ the second moment of the vertex degrees is weighted higher than the first moment. However, which term contributes the most to the value of $\mathbf{R}_{k,k+1}$ depends on the relative magnitude of the two moments as well as this weighting.

The above method can be adapted to perform the calculation for general $q > 2$. For infection rate τm^q, the $\binom{d_i}{m}$ coefficient is expanded to $\mathcal{O}(\frac{1}{m^q})$ before simplifying, and the resulting expression will contain up to the qth moment of the vertex degrees. Unfortunately I was unable to find a straightforward formula for the weightings of these moments for a general q.

3.6.3 Error estimates for larger networks

Whereas for small networks it is possible to construct the full infinitesimal generator $\mathbf{Q}$ (and hence the error commutator matrix $\mathbf{\Delta}$), and solve the full Kolmogorov equations to find the stationary distribution $\mathbf{X}^*$ (or the quasi-stationary solution $\mathbf{W}^*$), these calculations are impossible with larger networks. We can see the practical implications of this if we consider the factor $\mathbf{H}$ in the error calculation for processes with no absorbing state:

$$\mathbf{H} = \begin{pmatrix} -\mathbf{a}_0^{\mathrm{T}}\mathbf{X}_0^* + \mathbf{c}_1^{\mathrm{T}}\mathbf{X}_1^* \\ \mathbf{a}_0^{\mathrm{T}}\mathbf{X}_0^* - \mathbf{a}_1^{\mathrm{T}}\mathbf{X}_1^* + \mathbf{c}_2^{\mathrm{T}}\mathbf{X}_2^* - \mathbf{c}_1^{\mathrm{T}}\mathbf{X}_1^* \\ \vdots \\ \mathbf{a}_{N-2}^{\mathrm{T}}\mathbf{X}_{N-2}^* - \mathbf{a}_{N-1}^{\mathrm{T}}\mathbf{X}_{N-1}^* + \mathbf{c}_N^{\mathrm{T}}\mathbf{X}_N^* - \mathbf{c}_{N-1}^{\mathrm{T}}\mathbf{X}_{N-1}^* \\ \mathbf{a}_{N-1}^{\mathrm{T}}\mathbf{X}_{N-1}^* - \mathbf{c}_N^{\mathrm{T}}\mathbf{X}_N^* \end{pmatrix}.$$

The values of $\mathbf{a}_j, \mathbf{c}_j$ and $\mathbf{X}_j^*$ are unknown. Some of the consequences of this are presented in a paper I have co-authored [71] (currently in submission) which details and extends the work in this chapter. In the paper we limit consideration to SISa dynamics, where all $\mathbf{c}_j = 0$. It can be shown that in this case if we can find an upper bound on the values of $\left|\mathbf{a}_j^{\mathrm{T}}\mathbf{X}_j^*\right|$ then this allows us to calculate an upper bound on the error.

Let $\mathbf{a}_j^+ = \max|\mathbf{a}_j| \geq 0$, then $\left|\mathbf{a}_j^{\mathrm{T}}\mathbf{X}_j^*\right| \leq \mathbf{a}_j^+ \sum_k (\mathbf{X}_j^*)_k$. Essentially we can define an upper bound for the expression $\left|\mathbf{a}_j^{\mathrm{T}}\mathbf{X}_j^*\right|$ by assuming that each entry of $\mathbf{a}_j$ is equal to the maximum entry of $\mathbf{a}_j$. We are now posed with the problem of finding the value of $\mathbf{a}_j^+$. It is possible to define a hard bound for this value which will always hold, but this bound is often much larger than desirable. For example, for SISa dynamics the value of $\mathbf{a}_j^+$ will correspond to the state in level $\mathcal{C}^j$ that has the highest number of SI edges. We can assume that the state with the highest number of SI edges is one where the j nodes with the highest degrees are infected, and all their neighbours are susceptible. If we sum their degrees, we find a hard upper bound on the number of SI edges. In reality, this state will often not exist: if any of the j nodes are neighbours to each other then the calculation of SI edges will be more complex than this simple sum. Finding a more intelligent estimate for the value of $\mathbf{a}_j^+$ which is closer to the true value than

this crude bound remains an area for future investigation, and the paper presents some ideas on this subject.

There is also the question of how to handle the expression $\sum_k (\mathbf{X}_j^*)_k$. The paper explores the assumption that $\sum_i (\mathbf{X}_j^*)_i \leq \mathbf{Y}_j^*$. If this assumption holds, then $\left|\mathbf{a}_j^{\mathrm{T}} \mathbf{X}_j^*\right| \leq \mathbf{a}_j^+ \mathbf{Y}_j^*$, and we can use the value of $\mathbf{a}_j^+ \mathbf{Y}_j^*$ in our construction of an upper bound. The paper investigates how often this assumption holds, and shows that even for some cases where the assumption does not hold and $\sum_i (\mathbf{X}_j^*)_i > \mathbf{Y}_j^*$, the value of the calculated bound is still found to correctly bound the error.

Chapter 4

Inference on networks

Previous chapters have explored approximations that are necessary due to computational limitations. In theory, we can solve the Kolmogorov equations precisely, and we are only inhibited by computational power and time. However, often in network science we also have to make approximations and estimations because we simply do not have all the necessary information available. Due to a number of factors, such as the feasibility or cost of data collection, it's often not possible to gather information on the entire network that is being studied. Sometimes we are unaware of what the relevant network might even be. If there is a dynamical process occurring on a network, we might not be able to observe the entire process, instead we might be limited to discrete-time observations, such as prevalence data collected during regular intervals during an epidemic [83]. Entire node-states or transition types might not be visible, such as in an SEIR model of a disease that has an asymptomatic 'incubation period'. In this situation we might not be able to model the exact time when an individual moves from the susceptible state (S) to the asymptomatic exposed state (E) [40]. These problems of missing information have led to the development of a number of inference techniques, which I briefly summarise in this chapter.

Section 4.1 discusses existing approaches to infer an unknown network, whereas Section 4.2 focuses on existing approaches to infer an unknown network process (which might be happening on a known or unknown network). I explore this literature in anticipation of my work in Chapter 5, where I will investigate the problem of inferring an SIS process on a multilayer network with a hidden layer.

This is a novel inference problem, due to the fact that the missing information corresponds to an entire hidden network layer. My choice of inference scheme for this problem, discussed and justified later in Section 5.4, was chosen with consideration of the different schemes outlined in this chapter.

4.1 Inferring the network

In situations where it is not possible to collect information on the entire network, a sampled subgraph of the network, formed by sampling certain nodes and edges of the network, is often studied instead. The forward problem is concerned with how the sampling method affects the properties of the sampled subgraph in relation to the original network. Different sampling procedures will induce different biases, and it is important to consider this when choosing the appropriate sampling procedure for a problem [84, 85]. Meanwhile the inverse problem attempts to infer the original network properties based on a sampled subgraph and a known sampling method. This problem is sometimes framed in terms of a 'missing data' problem as opposed to a sampling problem [86]. Handcock and Gile [87] cover the inverse problem, and define two different approaches to this inference problem: design-based inference and model-based inference.

4.1.1 Design-based inference

The design-based framework treats the network as fixed, and the interest focuses on determining the exact network ('population graph') based on partial observation. Any random variation is treated as due to sampling alone. No model for the data is needed, and the unobserved data values are the parameters of interest. Design-based inference has the advantage that no specific knowledge about the network being sampled is required, but there are limitations.

One such limitation can be seen when we try to infer the degree distribution of the population graph from the sample degree distribution, as explored by Zhang *et al* [88]. For certain sampling designs, such as random node and random edge sampling (where the nodes or edges of a network are sampled with a set probability α), we can actually express the relationship between the expected

degree distribution of the sampled subgraph $p'(k)$ and that of the population graph $p(k_0)$ precisely. Here k refers to the degree of a node in the sampled subgraph, and k_0 refers to the degree of a node in the population graph. In the case of random edge sampling, in order to observe a node of degree k in the subgraph that has degree k_0 in the population graph, we need to have sampled k edges incident to the node, and not sample the remaining $k_0 - k$. There are $\binom{k_0}{k}$ ways of selecting the edges like this. Therefore the probability of observing this node is given by the binomial formula, $\binom{k_0}{k}\alpha^{k_0}(1-\alpha)^{k-k_0}$, and the relationship between the degree distributions is given by

$$p'(k) = \sum_{k_0=k}^{\infty} p(k_0)\binom{k_0}{k}\alpha^k(1-\alpha)^{k_0-k},$$

where α is the probability of sampling each edge. We can also write this as $p' = Ap$, where A is a matrix with elements

$$A_{ij} = \binom{i}{j}\alpha^j(1-\alpha)^{i-j}.$$

Therefore to find p given p' we might naively expect we can invert the equation to give us an estimate of the true degree distribution, given the sampled subgraph degree distribution

$$p = A^{-1}p'.$$

However, if we perform this calculation using a known population graph and a corresponding sampled subgraph, we find that the naive estimate differs hugely from the true degree distribution. The reason for this result is that we can only write down an expression for the *expected* sample degree distribution. In reality the sample degree distribution will vary from this expected result, and it turns out that this small deviation from this expected result leads to a huge deviation in our naive estimate for the true degree distribution. In other words, the problem is ill-conditioned.

This can also be seen by studying the condition number $\kappa(A)$ of the matrix A. For a linear equation $Ax = b$, the condition number describes the extent to which an error in b causes an error in the solution of x. A large condition number means that even a small error in b can cause a large error in x. By considering

the ratio of the relative error in the solution x to the relative error in b, it can be shown that the condition number of a matrix $\kappa(A) = \|A\|\|A^{-1}\|$. As Zhang *et al* claim, in this case the condition number goes as

$$\kappa(A) \sim \frac{1}{\alpha^D},$$

where D is the maximum degree of the nodes in the population graph, and therefore also the number of rows and columns of matrix A. As the value of D increases, the condition number increases, and practically the matrix becomes almost singular. I derived this relationship to check Zhang *et al*'s claim, work which is included in Appendix 1.

It is only possible to even express this relationship between degree distributions for random node sampling and random edge sampling because for these sampling schemes we can state the probability of selecting a node of degree j and selecting i of its neighbours. However, it is not possible to state this probability for all sampling schemes, and here we identify the main flaw with design-based inference, which Handcock and Gile raise [87]. To make unbiased design-based inference without assuming anything about the distribution of the unobserved data, we need full knowledge of the sampling procedure. Typically, full knowledge of the sampling procedure includes knowledge of the sampling probability of each unit in the sample. In the context of networks, this means we need to know the sampling probability for each pair of nodes. Sampling methods where it is possible to know these sampling probabilities are known as 'probability sampling methods'. The simpler schemes mentioned above, like random node and random edge sampling, are probability sampling methods. However, link-tracing designs such as multiple-wave snowball sampling are so-called 'non-probability sampling methods', where it is not possible to know the sampling probabilities for each pair of nodes [89].

4.1.2 Model-based inference

Model-based inference frameworks, on the other hand, treat the network as a stochastic realisation of an underlying random process. The goal is then to determine a model for this underlying process, rather than the precise network itself.

Inference in this framework involves formulating a model for the population graph, and calibrating the model parameters against the sample data available [87].

This method is often applied using exponential random graph models (ERGMs, as introduced in Section 1.2.3) as the model for the population graph. An ERGM is defined by a probability distribution $P(G)$ over a set of possible networks $G \in \mathcal{G}$:

$$P(G) = \frac{e^{H(G)}}{Z},$$

where $H(G) = \sum_i \beta_i x_i(G)$ is the graph Hamiltonian and $Z = \sum_{G \in \mathcal{G}} e^{H(G)}$ is the normalising constant. The ERGM parameters β_i are conjugate to certain configurations of the network, which are measured by the function $x_i(G)$. To estimate the ERGM parameters for a given sampled network, we first have to understand how the parameters are estimated for a completely-observed network, and then adapt these methods to consider missing information.

To estimate the parameters for a completely-observed network, we apply Markov Chain Monte Carlo (MCMC) methods to sample from the probability density functions of these parameters. MCMC algorithms construct a Markov chain whose stationary distribution is the same as the desired target probability distribution [90]. The states of the chain then serve as a sample of the target distribution. This approach requires us to know some function $\pi(x)$ of the current Markov state x which is proportional to the target distribution.

There are a number of different algorithms for constructing such a chain. One general framework is the Metropolis-Hastings algorithm [91, 92], a method which uses a proposal kernel (also known as a proposal density) to generate the next potential step in the chain. This next potential step is then either rejected or accepted according to an acceptance ratio, which is a function of both the proposal kernel and the function $\pi(x)$.

If, at step t the Markov chain is in state x, i.e. $X(t) = x$, then one step of the algorithm is as follows [93]:

1. Draw a proposed state y from some proposal kernel $q(x, y)$. Note that the new proposed step depends on the current state of the Markov chain.

2. Calculate the acceptance ratio

$$\alpha(x,y) = \begin{cases} \min\left[\frac{\pi(y)q(y,x)}{\pi(x)q(x,y)}\right] & \text{if } \pi(x)q(x,y) > 0 \\ 1 & \text{otherwise} \end{cases},$$

where $\pi(x)$ is a function proportional to the target distribution.

3. Generate a uniform random number $u \in [0,1]$, and compare this to the acceptance ratio.

   - If $u \leq \alpha(x,y)$ then we accept the move to state y: $X(t+1) = y$.
   - If $u > \alpha(x,y)$ then we reject the move to state y, and stay at state x: $X(t+1) = x$.

Starting the Markov chain in an initial state x_0, the above steps are repeated until enough states have been generated for the chain to settle into its equilibrium distribution. When this happens the Markov chain is said to have converged. Practically convergence is often identified by eye, and the samples before this are considered part of a 'burn-in' period and are simply discarded. In principle however, one can look at the internal variation along the chain, or check for convergence between multiple chains generated from different starting points [94].

We can use this algorithm to perform maximum likelihood estimation on the ERGM parameter values of a network, as long as we use a function $\pi(x)$ in the acceptance ratio that is proportional to the likelihood of the network being generated as a function of the ERGM parameters. In other words, the Markov chain states x under consideration become some vector of ERGM parameters θ, and $\pi(x)$ in the algorithm above becomes $\pi(\theta) = p(g \mid \theta) = \frac{e^{H(g)}}{Z}$, with $H(g) = \sum_i \theta_i x_i(g)$ and $Z = \sum_{g \in \mathcal{G}} e^{H(g)}$, where g is the observed network. The acceptance ratio similarly becomes

$$\alpha(\theta_0,\theta_1) = \begin{cases} \min\left[\frac{p(g|\theta_1)q(\theta_1,\theta_0)}{p(g|\theta_0)q(\theta_0,\theta_1)}\right] & \text{if } p(g \mid \theta_0)q(\theta_0,\theta_1) > 0, \\ 1 & \text{otherwise.} \end{cases}$$

Note that the normalising constant Z depends on the ERGM parameters θ. This means that the value of Z that features in the distribution $p(g \mid \theta_1)$ will not be the same as that featurng in $p(g \mid \theta_0)$, and so the normalising constants will not

cancel in the fraction $\frac{p(g|\theta_1)}{p(g|\theta_0)}$. Due to the fact that these normalising constants Z are often intractable, calculating the acceptance ratio is not trivial.

Initial solutions to this problem involve an approximation for the acceptance ratio which requires generating a random sample of networks from the set of configuration parameter values under consideration [95]. These random sample networks also have to be simulated using an MCMC routine. Snijders distinguishes between these two routines in his discussion of MCMC estimation of ERGMs [96], identifying an "MCMC simulation algorithm" which is used to generate the sample of networks from a set of parameter values, and an "MCMC estimation algorithm", which estimates the parameters for a given observed network, and which involves repeated use of the MCMC simulation algorithm. When a model is not a good representation of the observed network, the MCMC simulated networks may be far enough away from the observed network that the estimation process is affected [97]. In the worst case scenario, the simulated networks will be so different that the algorithm fails altogether. This happens in instances such as the two-star model ERGMs, where both high density networks and low density networks appear unpredictably. The model does not converge properly, and is said to be degenerate [98].

Koskinen [99] considers an initial Bayesian treatment of the problem, attempting to estimate the posterior probability distributions for the ERGM parameters. Bayes theorem states in this context that

$$p(\theta \mid g) = \frac{p(g \mid \theta)p(\theta)}{p(g)},$$

where $p(\theta)$ is a prior we can define, and $p(g \mid \theta) = \frac{e^{H(g)}}{Z}$ again for the observed network g. The function $\pi(x)$ in the algorithm definition here becomes $\pi(\theta) = p(\theta \mid g) = \frac{p(g|\theta)p(\theta)}{p(g)}$, thereby allowing us to incorporate the prior $p(\theta)$ into the MCMC inference. This distribution is "doubly-intractable", in the sense that both the normalising constant $p(g)$, and the normalising constant Z involved in $p(g \mid \theta) = \frac{e^{H(g)}}{Z}$, are intractable. Fortunately the denominator $p(g)$ is independent of the ERGM parameters θ, and so is cancelled out in the acceptance ratio.

Caimo and Friel [100] make use of the exchange algorithm presented in Murray et al [101], in which the intractable normalising constants Z that feature in $p(g \mid \theta)$

also cancel out in the acceptance ratio. This method offers improved efficiency, but still involves a step which requires a network to be generated from a given set of ERGM parameters, so the nested structure of an MCMC simulation algorithm within an MCMC estimation algorithms remains.

Finally, Koskinen et al [86] show how to extend this Bayesian MCMC algorithm to the inference question at hand: the situation where we only have access to a sample of the data (or equivalently a situation with missing data). Each iteration of their algorithm draws proposed values for the unobserved data, alongside proposed values for the configuration parameters and a generated network for use in the exchange algorithm. The unobserved data can also be generated by an MCMC sampler, where the sampler is constrained so that the observed parts of the network are never updated. This method presumes we know and can observe all the nodes of the network, and it is simply some edge values that are missing.

4.2 Inferring the dynamics on a network

When a dynamical process is occurring on a network, further inference problems arise. Given an assumed model for the process, we can attempt to infer the values of the model parameters. Since any model for the dynamical process that we might use for inference is inevitably an approximation, we can also ask the larger question: can we determine which model is the most accurate, or most appropriate, for investigating the dynamics in question? These questions have become particularly relevant due to the recent outbreak of coronavirus. If we can better understand how a disease spreads, we are more equipped to predict how the spread will progress, and to predict the result of interventions.

In some trivial cases, where we have complete information about the underlying contact network and the time and types of event that are happening, inference is straightforward. For example, Becker and Britton [102] consider observation of an SIR process where we know the infection and recovery times of every node. Direct likelihood maximisation can be used in this case, since the likelihood function for a given set of observations in terms of the model parameters can be expressed and maximised exactly.

If we are missing event observations, or do not have full knowledge of the underlying network, then the inference problem becomes more complicated. These systems do not generally have straightforward likelihoods, and so we have to take a more complicated approach. Here Bayesian methods are particularly useful, as they allow us to treat all unknowns equally and properly track the corresponding uncertainties. The sequence of events, and potentially some information about the underlying network, can be inferred along with the process parameters.

O'Neill outlines a number of different ways to approach these problems with intractable likelihoods [103]. Here I will summarise three of these: using an approximate model, data augmentation methods, and likelihood-free methods.

4.2.1 Using an approximate model

If the likelihood for a more complex model is proving intractable, one solution is to use a simpler model that has a computable likelihood instead.

One very simple example of this is considering the infected period in e.g. an SIS or SIR model to be a fixed constant, rather than exponentially or otherwise distributed [104]. This allows a data set that is missing either the infection event times or the recovery event times to be treated as a complete data set instead, although it means that the process is no longer Markovian. Another example of an approximate model features in Becker and Britton's study of disease spreading between households of individuals [105]. They take a model of household transmission that explicitly models between-household interactions, and replace these explicit interactions with a fixed probability that each individual avoids between-household infection. This allows the individual households to be treated as independent, and the likelihood becomes tractable.

Using a simpler approximate model is often an efficient method, but one must be careful in choosing appropriate approximations. If the model is too far from reality, the results may not be meaningful or helpful.

4.2.2 Data augmentation methods

In situations where evaluation of the likelihood requires information about the epidemic that we do not have, we can use data imputation to provide that in-

formation. Phillip O'Neill et al [106] provide a tutorial introduction outlining this method. They make use of a Bayesian MCMC routine which has the joint posterior density of the model parameters as its target density. Each iteration of the routine involves proposing a potential sequence of events using the current parameter values, and then selecting a set of parameter values based on the likelihood of the sequence of events.

The proposed potential sequence of events has to be compatible with the existing observations. A number of different methods exist to generate such a sequence of events. O'Neill and Roberts [107] feature one such method in their work studying an SIR process occurring in a well-mixed (network-free) population. They take a set of recovery times as the observed data set, and treat the unobserved infection times as further unknown model parameters. Each iteration of the MCMC algorithm requires sampling a potential sequence of infection times. To do this they take an initial set of infection times and use a Metropolis-Hastings algorithm that has three possible moves (either moving an infection time, removing an infection time, or adding a new infection time), with appropriate acceptance probabilities for given moves. The new infection times are uniformly sampled from the time interval (I_1, T), where I_1 is the infection time of the first individual and T is the time of the final observed recovery event.

We can also attempt to simulate a potential set of events, ideally conditioning on the observed events. Hobolth and Stone [108] consider this in a more general form of simulating events from any continuous-time Markov chain conditioned on fixed endpoints. They outline the most basic, inefficient method: using a Gillespie algorithm to generate a set of events, and then accepting this set of events only if it aligns with the observed events. This is time-consuming, and might never find compatible solutions for certain systems.

Hobolth and Stone go on to describe several further, more sophisticated methods, finally outlining the method of uniformization. This method involves constructing an auxiliary stochastic process, defined by an adjusted transition matrix that allows for virtual state changes, where a jump occurs but the state is unchanged. This method allows us to sample first the number of state changes (including virtual changes) that occur between a pair of observations, and then sample the times for the state changes and the nature of the changes (including

identifying which changes are virtual). Calculating this information involves taking the exponential of the adjusted transition matrix. This matrix has the same dimensions as the original transition matrix governing the master equations of the system. Therefore for systems with a large state space, the uniformization method can be too time-consuming and is not always numerically stable [109]. However, Choi and Rempala [110] successfully applied this method to a partially-observed SIRS epidemic, by considering the system as a well-mixed population. This meant the transition matrix involved was only an $(N+1)$ by $(N+1)$ matrix, where N is the total population size, rather than the 3^N by 3^N matrix describing the dynamics on the full network structure.

Britton and O'Neill [111] extend the idea of using data augmentation within an MCMC routine to perform inference on an unknown network. They study an SIR process, and consider the situation where only the recovery times of the individuals are known. They show that in this case, the parameters describing the network (given a presumed random network model in this case) can be inferred alongside the sequence of events and the dynamic parameters. Here each iteration of the MCMC algorithm generates a potential realisation of the network G by generating edges conditioned on the sequence of events, before then sampling a potential 'infection path' (i.e. select the edges (i, j) between nodes i and j along which the infection passes) by sampling nodes i uniformly from the possible infected neighbours of all ultimately infected nodes j. Finally a Metropolis-Hastings algorithm is used to update potential infection times, conditioned on the infection path.

4.2.3 Likelihood-free methods

If the likelihood cannot be evaluated conveniently using the previous methods, we can turn to likelihood-free methods, in particular Approximate Bayesian Computation (ABC) methods [112]. In ABC, an initial set of parameters are chosen and used to perform a simulation of the dynamical process. The results of the simulation are then reduced to summary statistics and compared with the data, and the parameters are either rejected or accepted according to the distance between the

simulated and the true observed summary statistics [113] and a pre-defined tolerance. In this way the likelihood is essentially replaced by this 'quasi-likelihood' distance measure.

The choice of summary statistics will generally depend on the particular inference question we would like to address, since it is the summary statistics that in a sense describe how 'good' a proposed set of parameters are. In choosing a set of summary statistics we also need to be careful to avoid 'the curse of dimensionality', which Prangle discusses in a review of the subject in relation to ABC [114]. Here the term dimension refers to the number of summary statistics. If we use too many summary statistics, then it becomes more likely for the distance between the simulated and observed summary statistics to exceed the tolerance. We are forced to either increase the tolerance, which will affect the accuracy of the posterior distribution, or to use fewer summary statistics, which reduces the amount of detail we can incorporate. These considerations must be balanced when selecting summary statistics and defining the tolerance.

Kypraios et al. [115] present a tutorial where they use ABC methods to estimate the parameters of a homogeneously mixing, network-free SIR model. The data being compared are the counts of how many individuals are in each state at each time. They use the sum-of-squared differences between observed and simulated counts in several time intervals as a summary statistic. McKinley et al [116] study an SEIR process in a homogeneously mixed population. They posit a criterion that involves placing an envelope around the observed data such that any simulated epidemic is rejected if it deviates beyond the envelope at any point. However, they judge this to be too sensitive to spurious fluctuations, and go on to suggest a sum-of-squared difference as in Kypraios et al. In the end they use a chi-squared goodness-of-fit criterion, which scales the contribution at each time point by the observed data, and as such incorporates the fact that the variation will change as the epidemic plays out.

Where the process occurs on a network, more complicated summary statistics can be used, such as those used in Dutta et al [117]. They consider both a simple SI model and a more complex general contagion model. They ran the processes on known networks: synthetic networks generated using BA and ER models, each with 100 nodes, and larger empirical networks, including a social network with

more than 4000 nodes. To accurately capture the state of the system at each discretely observed time step, they use multiple summary statistics. The first is the proportion of exposed nodes at different time steps. The second involves the subgraphs induced by the infected nodes and those induced by the exposed nodes at each time step. The shortest path lengths between pairs of nodes in the induced subgraphs are calculated. This allows the summary statistics to capture the differences in the locations of the infected and exposed nodes, as well as simply the proportions of the nodes that are in these states. These are high-dimensional summary statistics, but their interdependence helps avoid the curse of dimensionality problem.

4.3 Summary

In this chapter I have provided a brief summary of the different kinds of inference problems that have been explored in the context of network analysis. The first section discussed how to infer the population graph, given a sampled subgraph of the network. Design-based inference, where we seek to find out information about the exact population graph, can be applied in the case of certain probability sampling methods such as random node sampling and random edge sampling. However, some of this inference faces problems of ill-conditioning. Model-based inference is an alternative that can be used for nonprobability sampling methods. This inference scheme involves trying to identify a suitable model and model parameters for the population network, rather than the exact network itself. ERGMs are often used for this type of inference, as they can be calibrated against a number of different configurations, but in some cases these models face problems of convergence.

The second section looked at the problem of performing inference of dynamical systems on networks in situations with missing information. Using an approximate model can drastically simplify the inference process, but we risk oversimplifying and rendering the results less useful. Data augmentation methods are an alternative to this, where we impute the data required to make the likelihood tractable. This can be time-consuming and computationally demanding. In particular the method of uniformization requires taking the exponent of the

system transition matrix, which is not always possible. ABC methods eliminate the need to evaluate the likelihood, replacing it with a 'quasi-likelihood' in the form of summary statistics. However, choosing appropriate summary statistics can be a challenge.

In the next chapter I will analyse a system that has a huge amount of missing information: an SIS process occurring on a two-layer network, where one of the layers is hidden entirely. This system contains both missing information about the network (the hidden layer), and missing observations of the dynamical process (events on the hidden layer). I will attempt to infer the parameters of the dynamical system, assuming that every event on the visible layer is observed. In choosing an inference scheme, I considered the various different methods described in this chapter. A discussion around which inference scheme I selected, and why, features in section 5.4.

Chapter 5

Multilayer network with a latent layer

5.1 Problem set-up and motivation

So far the research discussion and investigation have only involved systems which are naturally described by a single network. However, sometimes data can be more appropriately described by a multilayer system, where each layer either represents a different network entirely, or represents the same or a similar set of nodes interacting in a different way. These types of network can be used to describe social networks, where people generally interact with different groups of people in different ways [118]. They have also been used to model situations in epidemiology, for example the situation where the two processes of a disease and disease-awareness spread across the same nodes on different networks [119]. Even society infrastructure can be thought of as coordination across a large number of interrlated networks such as energy supply and transport networks [120]. The effect of this multilayer structure on phenomena that have been well studied on single-layer networks is an important emerging area of interest [121].

At the heart of this topic is the core question of whether multilayer networks are really needed, or whether we can 'flatten' the multilayer structure into an *aggregated* network. To aggregate two network layers, the nodes and edges on both layers are considered to instead lie on the same, single layer. Any differences in the type of node or edge on the two layers are disregarded. Initial research

in this area has found instances where ignoring the multilayer structure skews measures such as the centrality of the nodes [122], and can affect things such as the epidemic threshold of the network dynamics [123]. However, since multilayer networks are inherently a more complex structure than single-layer networks, it is useful to identify and clarify any situations in which the network does not need to truly be modelled as multilayer, or instances where layers can be aggregated without affecting our analysis. The nature of these situations depends on the kind of analysis we would like to perform, and exactly how we define the concept of similarity in network science. De Dominico *et al.* [124] use the application of Von Neumann entropy to networks, and compare the entropy of the completely aggregated multilayer network to the entropy of various different partial aggregations of the layers. Sanchez *et al.* [125] instead choose to compare the eigenvalue spectra of the adjacency and Laplacian matrices of the full multilayer network with those of the aggregrated network. Meanwhile Diakovona *et al.* [126] take the specific example of voter model dynamics on a multilayer network, and compare simulation results of the dynamics on the complete multilayer network with analytic predictions of the dynamics on various aggregated single-layer networks. They conclude that consideration of the full structure is necessary in this context for all but either extremely connected or extremely unconnected layers.

As a way to further explore this question, we decided to study SIS dynamics occurring on a two-layer network, where the dynamics on one layer are hidden. This might occur in real life if data collection on one layer is not possible, impractical or expensive. In attempting to infer both the dynamical model and the parameters of this model, we can investigate when, if ever, it is possible to approximate the multilayer structure and dynamics with simpler models. This will highlight instances where the multilayer structure is redundant, and instances where it cannot be ignored or simplified. We will also need to adapt the inference approaches discussed in Chapter 4 that have been previously applied to one-layer systems, exploring the problem of how we can perform inference when there is an entire hidden layer.

Section 5.2 outlines the basic multilayer definitions and representations that will allow us to study this problem. Section 5.3 describes further the two-layer SIS dynamics, and provides a basic mean-field analysis of the problem. This

section also includes some preliminary investigation using simulations, to identify interesting combinations of parameters and types of network structure to study. In Section 5.4 I discuss the possible ways of approaching this inference problem, and justify my choice to use approximate models to perform inference. First in Section 5.5 I use simulated data to determine whether the two-layer dynamics can be approximated using a naive one-layer SISa model. Section 5.6 then looks at fitting the simulated data using a more complicated 'latent variable' model which involves approximating the hidden layer dynamics using just a single node on the hidden layer. These latter two sections will use MCMC methods of inference, with Section 5.6 adapting a method used in hidden Markov models. Section 5.7 summarises and discusses ideas for future investigation.

5.2 Multilayer networks

To formally define multilayer networks, we look to Boccaletti *et al.* [127], which outlines a number of different multilayer structures. The most general form of a multilayer network is described by a pair $\mathcal{M} = (\mathcal{G}, \mathcal{C})$, a set of layers $\mathcal{G}$ and a set of interconnections $\mathcal{C}$ between nodes of different layers respectively. The set of layers $\mathcal{G} = \{G_\alpha; \alpha \in \{1, ..., M\}\}$ is a family of graphs $G_\alpha = (X_\alpha, E_\alpha)$, where $X_\alpha = \{x_1^\alpha ..., x_{N_\alpha}^\alpha\}$ is the set of nodes on layer α and E_α is the set of edges that connect nodes within layer α. The edges E_α are known as *intralayer* edges. The set of interconnections $\mathcal{C} = \{E_{\alpha\beta} \subseteq X_\alpha \times X_\beta; \alpha, \beta \in \{1, ..., M\}, \alpha \neq \beta\}$ contains the edges $E_{\alpha\beta}$ that are known as *interlayer* edges. In some types of multilayer network, the same nodes will feature on multiple layers. We call these node replicas.

Multilevel networks are a type of multilayer network where interlayer edges only exist between node replicas, but not all nodes necessarily have replicas. An example of this might be a two-layer network of a group of users interacting on different social media platforms, where some but not all have Facebook or Twitter accounts. Interlayer edges represent the switching between different platforms. *Multiplex* networks are a type of multilevel network where all nodes feature on all layers. An example of this might be a system describing the different kind of

relations between colleagues: one layer might contain the edges connecting colleagues who consider each other friends, while another might contain the edges between colleagues who have worked directly on projects together. Multiplex networks have also frequently been used to model systems where a disease is spreading alongside awareness of the disease [119]. One layer contains the edges along which the epidemic spreads between individuals, while a second layer contains the edges along which knowledge of the epidemic spreads. This thesis has so far shown a focus on epidemic dynamics, and so in this section I have chosen to focus on multiplex networks. However, for completeness, there also exist *interconnected* networks, where the layers represent different networks, and the interlayer edges correspond to interactions between networks. An example of this might be a two-layer network representing sexual relationships, with one layer formed of males and one layer formed of females. Heterosexual relationships are represented by interlayer edges, while homosexual relationships are represented by intralayer edges.

With these more complex network systems comes the question of how best to represent them. One natural way of representing the connections in a multilayer network is to extend the standard adjacency matrix representation of one-layer networks to a tensor representation [128]. Alternatively the tensor representation can be flattened by combining all of the layers and node indices to obtain additional node indices, arriving at the supra-adjacency representation. Sanchez *et al.* [125] describe the resulting supra-adjacency matrix $A_M = A_{M_G}$, which has diagonal blocks A_α, the adjacency matrices of the layers G_α, and has off-diagonal blocks $A_{\alpha\beta}$, matrices representing the inter-layer connectivity. Note that the supra-adjacency matrix representation alone cannot be used to identify which nodes are replicas of each other. Sanchez *et al.* also define the supra-node set as the set of nodes representing the same object, to contain the information about node replicas.

The multilayer Laplacian tensor and supra-Laplacian matrix can be constructed from the adjacency tensor and supra-adjacency matrix as in the single-layer case [129]. The supra-Laplacian matrix has diagonal blocks encoding the Laplacian matrices for the corresponding network layers, while the off-diagonal

blocks enocode the interlayer connections. Spectral analysis of multilayer networks involves the eigenvalues of these supra- matrices [130].

Single-layer network properties such as the degree distribution, associativity, and clustering coefficient can be generalised and applied to multilayer networks, a comprehensive study of which can be found in Battiston *et al.* [131]. There are also properties specific to multilayer networks and their structure, for example interdependence, which compares the number of shortest paths in which multiple layers are traversed to the total number of shortest paths [132]. Another measure, overlap, can be used to describe the common connections between the layers. The total overlap between two layers is defined as the total number of links that are common between two layers [133], while the local overlap of a node i is defined as the total number of neighbours of node i that are neighbours in both layers [133].

5.3 Mean-field analysis and simulation results

In order to explore the problem of inferring SIS dynamics on a two-layered multiplex network, we first need to extend the one-layer SIS model to two layers. This introduces further epidemic parameters: two separate intralayer spreading rates τ_{11} and τ_{22} are required to describe how the epidemic spreads between nodes on layer 1 and layer 2 respectively, while the interlayer spreading rates τ_{12} and τ_{21} describe how frequently the epidemic spreads from nodes on layer 1 to layer 2, and vice versa. Two recovery rates γ_1 and γ_2 allow for nodes on layer 1 to recover at a different rate to nodes on layer 2. An example of a two-layer network with the infection rates labelled is shown in Figure 5.1.

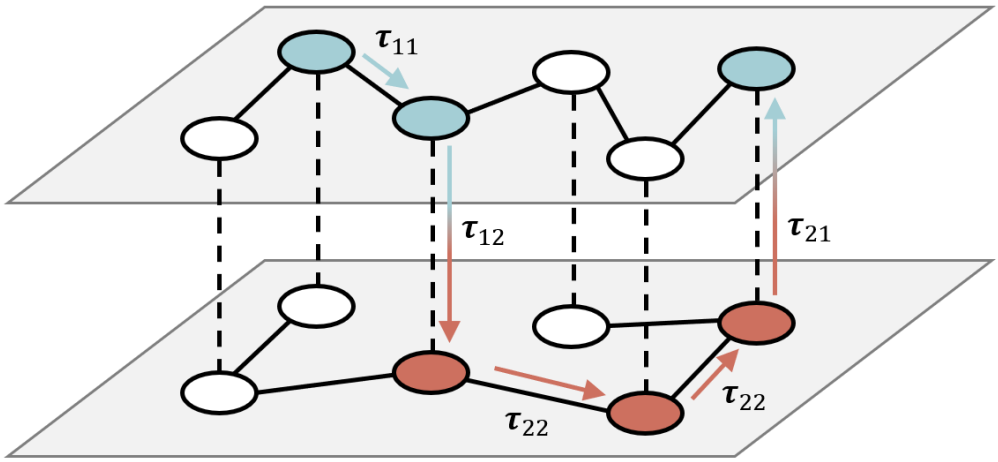


Figure 5.1: A pictorial representation of a two-layer multiplex network, showing how the infection might spread between nodes. The infection spreads within each layer with intralayer spreadings τ_{11} and τ_{22}, and between layers with interlayer spreadings τ_{12} and τ_{21}.

In some models of multilayer SIS dynamics, nodes that are infected on one of the layers are taken to be instantly infected on the other layers. However, we have chosen to model the interlayer spreading as non-instantaneous, to allow for a 'layer-switching cost', in agreement with [134]. This better approximates situations such as people travelling through a transport network, where switching mode of transport can take time. There may also be a layer-switching cost in social media. For example, it takes very little effort for someone to repost information on the same social network where they first observed it, but the process of copying the information or link across to another site might require more effort. In situations where cross-layer infection is best modelled as instantaneous, it may be that these dynamics can be recovered in the limit of large interlayer spreading rates. There are fewer good examples in the context of infection dynamics, although a layer-switching cost might be appropriate if the nodes on different layers do not strictly model the same thing. For example, nodes on one layer may represent people, and nodes on another layer may represent the households people belong to. We might not want to immediately classify a household as

infected if only one person is infected, and similarly we may not want to classify all the people in a household as infected if their household is classified as infected.

We can now derive a mean-field approximation for these dynamics. We denote the number of susceptible and infected nodes on layer 1 as S_1 and I_1 respectively. We denote the equivalent quantities on layer 2 as S_2 and I_2. Drawing from the single-layer mean-field approach discussed in Chapter 2, we can then describe the evolution in time of infected nodes on layer 1 and layer 2 as:

$$[\dot{I}_1] = \tau_{11}[S_1 I_1] + \tau_{21}[S_1 I_2] - \gamma_1[I_1],$$

$$[\dot{I}_2] = \tau_{22}[S_2 I_2] + \tau_{12}[S_2 I_1] - \gamma_2[I_2].$$

The first term in each equation represents the increase in infected nodes due to infection from a same-layer neighbour. The second term represents an increase due to infection from a replica in the other layer. The final term models the decrease in infected nodes due to recovery.

We can make a closure at the single level:

$$[S_1 I_1] \approx [S_1] n_{11} \frac{[I_1]}{N_1},$$

$$[S_2 I_2] \approx [S_2] n_{22} \frac{[I_2]}{N_2},$$

$$[S_1 I_2] \approx [S_1] n_{12} \frac{[I_2]}{N_2},$$

$$[S_2 I_1] \approx [S_2] n_{21} \frac{[I_1]}{N_1},$$

where n_{ab} is the average number of connections between a node in layer a and a node in layer b, and N_a is the number of nodes in layer a. This closure can be justified with the same physical reasoning as in Chapter 2: the expected number of (S_1, I_1) links will equal the number of susceptible nodes on layer 1 ($[S_1]$), multiplied by the fraction of nodes on layer 1 that are infected ($\frac{[I_1]}{N_1}$) and the average number of layer-1 connections that a given susceptible node has (n_{11}). Equivalent reasoning gives us expressions for the other three terms.

In a multilevel network the $[S_1 I_2]$ and $[S_2 I_1]$ terms will correspond to the number of nodes that are susceptible in one layer but infected in the other. In

a multiplex network, $n_{12} = 1$. In a multilevel network with overlapping fraction p, $n_{12} = p$. In an interconnected network, n_{12} can take any number of values (limited only by the number of nodes in each layer). Since this chapter focuses on multiplex networks, I will use $n_{12} = n_{21} = 1$ for all networks that are mentioned.

Applying this closure, and using the conservation identities $[S_1] = N_1 - [I_1]$ and $[S_2] = N_2 - [I_2]$, the mean-field equations can be written:

$$\begin{aligned}[\dot{I}_1] &= \tau_{11}\frac{n_{11}}{N_1}(N_1 - [I_1])[I_1] + \tau_{21}\frac{n_{12}}{N_2}(N_1 - [I_1])[I_2] - \gamma_1[I_1] \\ [\dot{I}_2] &= \tau_{22}\frac{n_{22}}{N_2}(N_2 - [I_2])[I_2] + \tau_{12}\frac{n_{21}}{N_1}(N_2 - [I_2])[I_1] - \gamma_2[I_2].\end{aligned} \tag{5.1}$$

Solving for $[\dot{I}_1] = 0$ and $[\dot{I}_2] = 0$, there exists a disease-free solution $I_1 = I_2 = 0$. The remaining solutions for I_1 and I_2 satisfy

$$I_2 = \frac{\gamma_1 I_1 - \hat{\tau}_{11}(N_1 - I_1)I_1}{\hat{\tau}_{21}(N_1 - I_1)},$$

where $\hat{\tau}_{11} = \tau_{11}\frac{n_{11}}{N_1}$ and $\hat{\tau}_{21} = \tau_{21}\frac{n_{12}}{N_2}$. Likewise we define $\hat{\tau}_{22} = \tau_{22}\frac{n_{22}}{N_2}$ and $\hat{\tau}_{12} = \tau_{12}\frac{n_{21}}{N_1}$. To find the solution for I_1 involves solving the following cubic:

$$\begin{gathered}[\hat{\tau}_{11}\hat{\tau}_{21}\hat{\tau}_{12} - \hat{\tau}_{22}\hat{\tau}_{11}^2]I_1^3 + \\ [\gamma_2\hat{\tau}_{11}\hat{\tau}_{21} + \gamma_1\hat{\tau}_{21}\hat{\tau}_{12} + \hat{\tau}_{21}^2\hat{\tau}_{12}N_2 - 2\hat{\tau}_{11}\hat{\tau}_{21}\hat{\tau}_{12}N_1 + 2\hat{\tau}_{11}^2\hat{\tau}_{22}N_1 - \hat{\tau}_{11}\hat{\tau}_{22}\hat{\tau}_{21}N_2 - 2\hat{\tau}_{11}\hat{\tau}_{22}\gamma_1]I_1^2 + \\ [2\hat{\tau}_{11}\hat{\tau}_{22}\hat{\tau}_{21}N_2N_1 - \gamma_1\hat{\tau}_{22}\hat{\tau}_{21}N_2 - \gamma_1^2\hat{\tau}_{22} + 2\hat{\tau}_{11}\hat{\tau}_{22}\gamma_1N_1 - \hat{\tau}_{11}^2\hat{\tau}_{22}N_1^2 - 2\hat{\tau}_{21}^2\hat{\tau}_{12}N_1N_2 - \hat{\tau}_{21}\hat{\tau}_{12}\gamma_1N_1 + \\ \hat{\tau}_{11}\hat{\tau}_{21}\hat{\tau}_{12}N_1^2 + \gamma_1\gamma_2\hat{\tau}_{21} - 2\hat{\tau}_{11}\hat{\tau}_{21}\gamma_2N_1]I_1 + \\ [\gamma_1\hat{\tau}_{22}\hat{\tau}_{21}N_1N_2 - \hat{\tau}_{11}\hat{\tau}_{22}\hat{\tau}_{21}N_2N_1^2 + \hat{\tau}_{21}^2\hat{\tau}_{12}N_1^2N_2 - \gamma_1\gamma_2\hat{\tau}_{21}N_1 + \gamma_2\hat{\tau}_{11}\hat{\tau}_{21}N_1^2] = 0.\end{gathered} \tag{5.2}$$

We can consider the discriminant $\Delta_3 = 18abcd - 4b^3d + b^2c^2 - 4ac^3 - 27a^2d^2$ of a cubic $ax^3 + bx^2 + cx + d = 0$. If the discriminant $\Delta_3 > 0$, then the cubic has three distinct roots. If $\Delta_3 = 0$ the cubic has repeated roots, and if $\Delta_3 < 0$ then the cubic has one real root and two non-real complex conjugate roots. Computing Δ_3 for the cubic 5.2 with different values of the dynamical parameters revealed that Δ_3 can be both positive or negative. However, for all the parameter combinations I tested, I found there was at most one solution for I_1 and I_2 that was physical, i.e. both values were real and positive. It appears there are two possible steady state solutions: the disease-free absorbing state, and, if the dynamical parameters allow, an endemic state with $I_1 > 0$, $I_2 > 0$.

To examine the stability of these steady states analytically, we can analyse the Jacobian for the system 5.1:

$$J = \begin{bmatrix} \frac{\partial \dot{I_1}}{\partial I_1} & \frac{\partial \dot{I_1}}{\partial I_2} \\ \frac{\partial \dot{I_2}}{\partial I_1} & \frac{\partial \dot{I_2}}{\partial I_2} \end{bmatrix} = \begin{bmatrix} \hat{\tau}_{11}(N_1 - 2I_1) - \hat{\tau}_{21}I_2 - \gamma_1 & \hat{\tau}_{21}(N_1 - I_1) \\ \hat{\tau}_{12}(N_2 - I_2) & \hat{\tau}_{22}(N_2 - 2I_2) - \hat{\tau}_{12}I_1 - \gamma_2 \end{bmatrix}.$$

First I investigated the stability of the case where $I_1 > 0, I_2 > 0$. Given the relations $I_2 = \frac{\gamma_1 I_1 - \hat{\tau}_{11}(N_1 - I_1)I_1}{\hat{\tau}_{21}(N_1 - I_1)}$ and $I_1 = \frac{\gamma_2 I_2 - \hat{\tau}_{22}(N_2 - I_2)I_2}{\hat{\tau}_{12}(N_2 - I_2)}$, we know that for these solutions $\gamma_1 - \hat{\tau}_{11}(N_1 - I_1) > 0$ and $\gamma_2 - \hat{\tau}_{22}(N_2 - I_2) > 0$. This means that the trace of the Jacobian, $\mathrm{Tr}(J) = -(\gamma_1 - \hat{\tau}_{11}(N_1 - I_1)) - \hat{\tau}_{11}I_1 - \hat{\tau}_{21}I_2 - (\gamma_2 - \hat{\tau}_{22}(N_2 - I_2)) - \hat{\tau}_{22}I_2 - \hat{\tau}_{12}I_1 < 0$. For this steady state solution to be stable, we also have the condition that the determinant of the Jacobian $\det(J)$ must be positive. Knowing that the diagonal terms of the Jacobian are each sums of negative values, we can write

$$\det(J) = \text{identically positive terms} + \hat{\tau}_{12}\hat{\tau}_{21}I_1I_2 - \hat{\tau}_{12}\hat{\tau}_{21}(N_1 - I_1)(N_2 - I_2).$$

Therefore at least for cases where $I_1I_2 > (N_1 - I_1)(N_2 - I_2)$, $\det(J) > 0$ and the solution is stable. For example, solutions where $I_1 > \frac{N_1}{2}, I_2 > \frac{N_2}{2}$ will be stable. Numerically $\det(J) > 0$ held for all the solutions with $I_1 > 0, I_2 > 0$ that I explored, but I was unable to analytically show if this was true for all possible solutions.

Analysing the Jacobian for the disease-free stationary point $I_1 = I_2 = 0$ proved more straightforward. The stationary point will be stable if the real part of both eigenvalues $\Lambda_\pm$ of the Jacobian evaluated at this point are negative. Equivalently, the point will be unstable if $\Lambda_+ > 0$. Redefining the infection rates as $\hat{\tau}_{11} = \tau_{11}n_{11}$, $\hat{\tau}_{22} = \tau_{22}n_{22}$, $\hat{\tau}_{12} = \tau_{12}n_{21}$ and $\hat{\tau}_{21} = \tau_{21}n_{12}$, then at $I_1 = I_2 = 0$ the eigenvalues $\Lambda_\pm$ satisfy

$$\Lambda_\pm^2 - ((\hat{\tau}_{11} - \gamma_1) + (\hat{\tau}_{22} - \gamma_2))\Lambda_\pm + (\hat{\tau}_{11} - \gamma_1)(\hat{\tau}_{22} - \gamma_2) - \hat{\tau}_{12}\hat{\tau}_{21} = 0.$$

Therefore $\Lambda_+ > 0$ is satisfied when

$$(\hat{\tau}_{11} - \gamma_1) + (\hat{\tau}_{22} - \gamma_2) + \sqrt{((\hat{\tau}_{11} - \gamma_1) + (\hat{\tau}_{22} - \gamma_2))^2 - 4((\hat{\tau}_{11} - \gamma_1)(\hat{\tau}_{22} - \gamma_2) - \hat{\tau}_{12}\hat{\tau}_{21})} > 0.$$

If $(\hat{\tau}_{11} - \gamma_1) + (\hat{\tau}_{22} - \gamma_2) > 0$ (**condition 1**) is true, i.e. if the trace of the Jacobian is positive, then this is automatically satisfied. If condition 1 is not

true, we can manipulate the inequality further to reach **condition 2**: $\hat{\tau}_{12}\hat{\tau}_{21} > (\hat{\tau}_{11} - \gamma_1)(\hat{\tau}_{22} - \gamma_2)$, i.e. the discriminant is negative.

If the two terms $(\hat{\tau}_{11} - \gamma_1)$ and $(\hat{\tau}_{22} - \gamma_2)$ are both positive, then condition 1 is satisfied. If the terms have opposite signs, then condition 2 is satisfied. Therefore a necessary (but not sufficient) condition for the disease-free stationary point to be stable is if $(\hat{\tau}_{11} - \gamma_1)$ and $(\hat{\tau}_{22} - \gamma_2)$ are both negative. If we note here that $(\hat{\tau}_{11} - \gamma_1) > 0$ and $(\hat{\tau}_{22} - \gamma_2) > 0$ are the conditions for an endemic steady state on the isolated network layers 1 and 2 respectively, this result means that if either one of the layers can sustain an endemic steady state by itself, then the disease-free state will not be stable. The layer capable of supporting an endemic state will constantly reseed the other layer, and in this way an epidemic can occur on a layer which would not otherwise be sustained, a situation I will refer to in this work as an *induced epidemic*.

These conditions are illustrated in a diagram in Figure 5.2, which shows the different regions of stability for the disease-free state as $\hat{\tau}_{11}$ and $\hat{\tau}_{22}$ are varied. The values of the other parameters are: $\hat{\tau}_{12} = 0.5$, $\hat{\tau}_{21} = 1$, $\gamma_1 = 1$ and $\gamma_2 = 2$. The red regions indicate where condition 1 holds. In this region an endemic steady state could be sustained on at least one of the layers in isolation, and we find the disease-free state is unstable. The blue regions indicate where condition 2 holds. Note that the regions are shaded with transparent colour, and the purple region indicates where both conditions hold. In the section where only condition 2 holds, neither of the layers could sustain an endemic steady state in isolation, but the connections with the other layer and the values of the interlayer spreading are such that the disease-free state is still unstable. The white region indicates the only area where $\Lambda_+ < 0$, and the disease-free steady state is stable.

To explore how well the mean-field results matched the results from simulating the stochastic system, I considered the dynamics running on a fully connected two-layer multiplex network (i.e. each node is connected to every other node on the same layer, and each node is connected to its replica in the other layer) that has 100 nodes on each layer. I chose this network because I expected it to represent a 'best-case scenario' multiplex network where the two-layer homogeneous mean-field approximation would perform the best: it is the densest network with $N = 100$ possible, and every node has the same number of neighbours. I

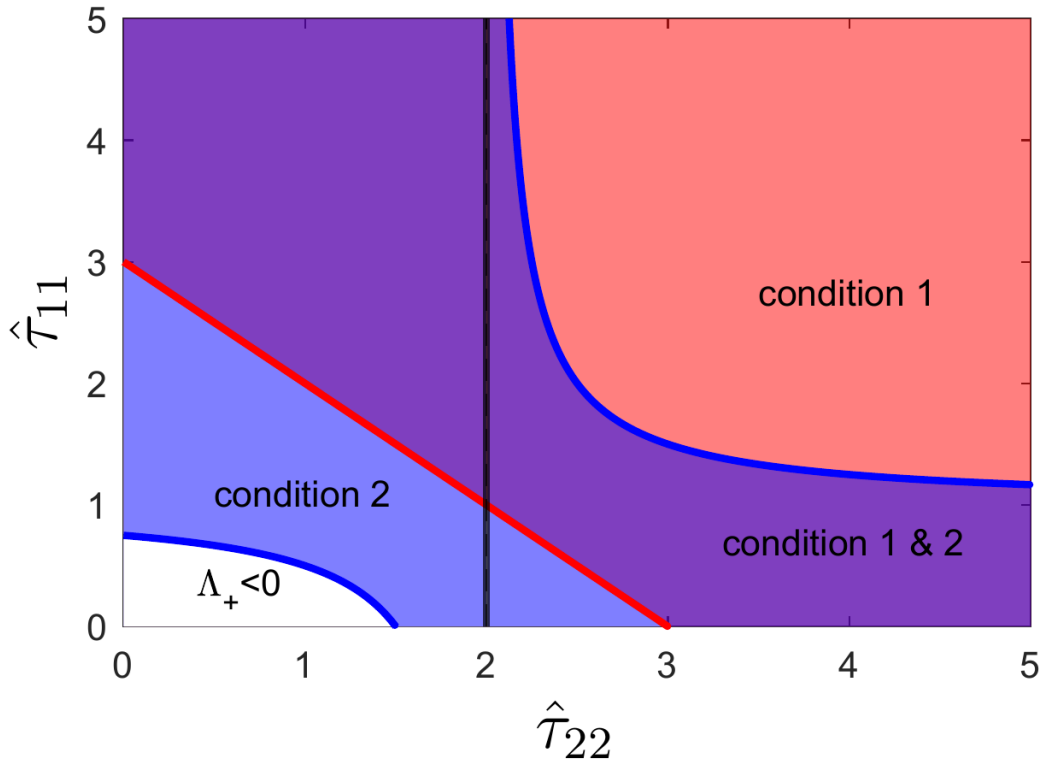
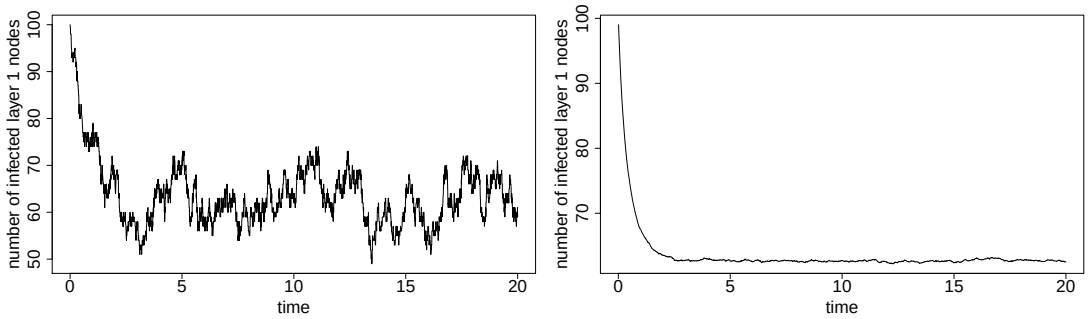


Figure 5.2: Diagram showing the different regions of stability for the disease-free steady state as $\hat{\tau}_{11}$ and $\hat{\tau}_{22}$ vary. The other parameters are kept fixed at the values: $\hat{\tau}_{12} = 0.5$, $\hat{\tau}_{21} = 1$, $\gamma_1 = 1$ and $\gamma_2 = 2$. The red line indicates where $(\hat{\tau}_{11} - \gamma_1) + (\hat{\tau}_{22} - \gamma_2) = 0$, and the red regions indicate where the condition $(\hat{\tau}_{11} - \gamma_1) + (\hat{\tau}_{22} - \gamma_2) > 0$ (condition 1) holds. The blue line indicates where $\hat{\tau}_{12}\hat{\tau}_{21} = (\hat{\tau}_{11} - \gamma_1)(\hat{\tau}_{22} - \gamma_2)$, with the black line indicating the asymptote for this plot, and the blue regions indicate where the condition $\hat{\tau}_{12}\hat{\tau}_{21} > (\hat{\tau}_{11} - \gamma_1)(\hat{\tau}_{22} - \gamma_2)$ (condition 2) holds. Note that I have made these colours transparent, and so the purple region indicates where both of these conditions hold. In these coloured regions $\Lambda_+ > 0$, and the disease-free steady state becomes unstable. The white region marks the values of $\hat{\tau}_{11}$ and $\hat{\tau}_{22}$ where the disease-free steady state is stable.

compared the mean-field steady state value to the estimated steady state value of the stochastic simulations for various different parameter values. To estimate steady state values for the simulations, I started the simulations with an entirely infected network, and played out the Gillespie algorithm until the number

of infected nodes had either reached zero or seemed to be fluctuating around a constant value. I judged this 'burn-in' time by visual inspection. For each set of parameter values, 1000 simulations were performed. I computed the mean of these 1000 simulations, and the value of this mean after the 'burn-in' time was taken as the estimated steady state value. An example of a single simulation, and the mean of an ensemble of 1000 such simulations, for the parameters $\hat{\tau}_{11} = \hat{\tau}_{22} = 0.02, \hat{\tau}_{12} = \hat{\tau}_{21} = 0.8$, and $\gamma_1 = \gamma_2 = 1$ are shown in Figure 5.3.



(a) Single run

(b) Ensemble mean, 1000 runs

Figure 5.3: Plots showing the number of infected layer 1 nodes with time on a fully connected two-layer multiplex network with 100 nodes on each layer. The multilayer SIS parameters were $\hat{\tau}_{11} = \hat{\tau}_{22} = 0.02, \hat{\tau}_{12} = \hat{\tau}_{21} = 0.8$, and $\gamma_1 = \gamma_2 = 1$. The plot in (a) shows the result of a single simulation, while the plot in (b) shows the mean result of an ensemble of 1000 simulations. I judged the burn-in time for this system to be $t = 10$.

In the first case, the values of $\hat{\tau}_{11}$ and $\hat{\tau}_{22}$ were varied while keeping the other parameter values fixed at the values $\hat{\tau}_{12} = \hat{\tau}_{21} = 0.8$, $\gamma_1 = \gamma_2 = 1$. Figure 5.4 shows a comparison of the simulated steady-state values with the mean-field steady-state solutions. The left dotted line marks the point at which $\hat{\tau}_{12}\hat{\tau}_{21} = (\hat{\tau}_{11} - \gamma_1)(\hat{\tau}_{22} - \gamma_2)$, and the right marks the point at which $\hat{\tau}_{11} = \gamma_1$, $\hat{\tau}_{22} = \gamma_2$. We find that the simulation and mean-field predictions agree better at higher $\hat{\tau}_{11} = \hat{\tau}_{22}$. Particularly at lower $\hat{\tau}_{11} = \hat{\tau}_{22}$, the predicted value for the fraction infected in the steady state is much higher than the simulation. This could be due to the difficulty in judging the burn-in time for these simulations. When the

steady state is close to $I_1 = 0$, running the simulation for too long results in most of the solutions falling into the absorbing disease-free state.

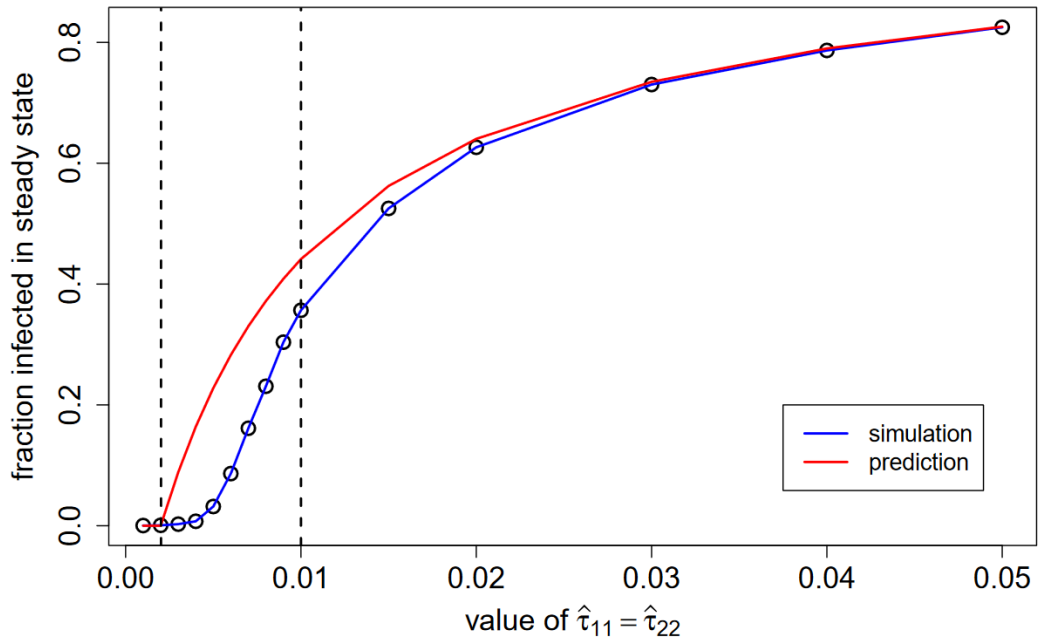


Figure 5.4: Plot showing how the fraction of infected nodes on layer 1 in the steady state of the dynamics varies with the value of $\hat{\tau}_{11} = \hat{\tau}_{22}$. The simulation results are plotted in black markers, with the blue line showing the curve joining these values, while the curve joining the prediction values is plotted in red. The left dotted line marks the point at which condition 2 is met: $\hat{\tau}_{12}\hat{\tau}_{21} = (\hat{\tau}_{11} - \gamma_1)(\hat{\tau}_{22} - \gamma_2)$. The right dotted line indicates where $\hat{\tau}_{11} = \gamma_1 = \hat{\tau}_{22} = \gamma_2$. Between these lines, the disease-free steady state is unstable, and we see that there exists an endemic steady state with $I_1 > 0$. The existence of these endemic steady state values between these lines confirms that the system can sustain an endemic solution even when an endemic solution would not be sustained on the individual layers in isolation. We can see that the predicted results and the simulation results agree better at higher values of $\hat{\tau}_{11} = \hat{\tau}_{22}$.

In the other case the values $\hat{\tau}_{12}$ and $\hat{\tau}_{21}$ were varied while the other values stayed fixed. The results are shown in Figure 5.5. Here $\hat{\tau}_{11} = \hat{\tau}_{22} = 0.5 < \gamma_1 = \gamma_2 = 1$, so condition 1 is never fulfilled as we vary $\hat{\tau}_{12} = \hat{\tau}_{21}$. The dotted line

marks the value of $\hat{\tau}_{12}\hat{\tau}_{21} = (\hat{\tau}_{11} - \gamma_1)(\hat{\tau}_{22} - \gamma_2)$. Again, for low predicted fractions of infected nodes, the simulation steady state is very close to 0. This could again be in part due to the difficulty in judging the burn-in time for these simulations.

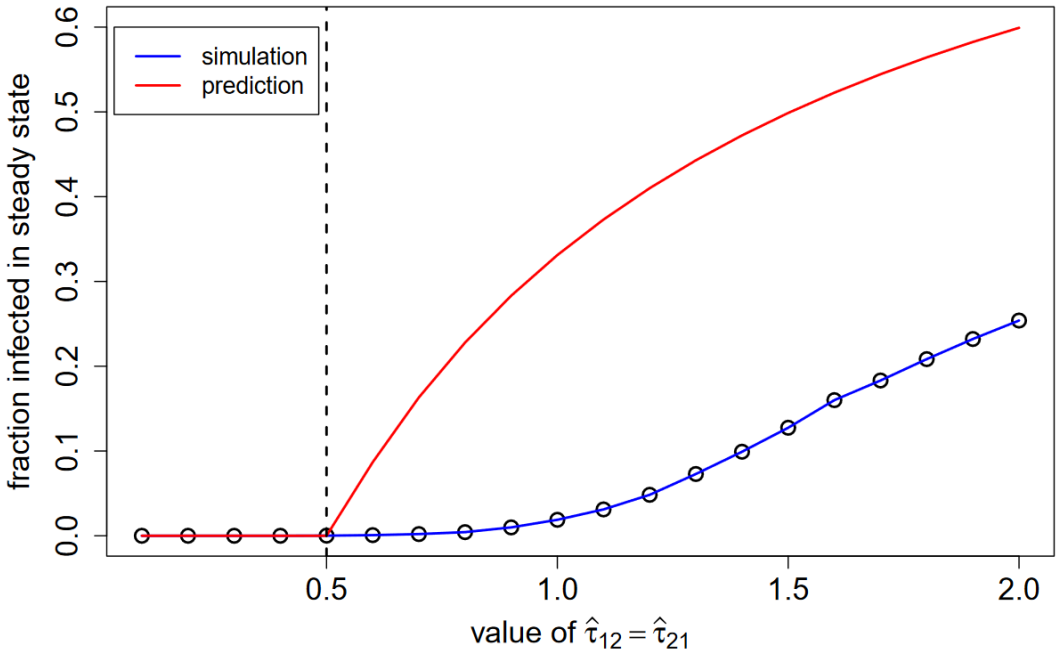


Figure 5.5: Plot showing how the fraction of infected nodes on layer 1 in the steady state of the dynamics varies with the value of $\hat{\tau}_{12} = \hat{\tau}_{21}$. The simulation results are plotted in black markers, with the blue line showing the curve joining these values, while the curve joining the prediction values is plotted in red. The dotted line marks the point at which condition 2 is met: $\hat{\tau}_{12}\hat{\tau}_{21} = (\hat{\tau}_{11} - \gamma_1)(\hat{\tau}_{22} - \gamma_2)$. To the right of this point, the disease-free state is unstable, and we find that there is an endemic steady state.

If we keep the values of n_{11}, n_{22}, n_{12} and n_{21} the same, we have 6 parameters which can vary: $\tau_{12}, \tau_{21}, \tau_{11}, \tau_{22}, \gamma_1, \gamma_2$. If we fix γ_1 and γ_2, we can vary the τ values relative to these fixed γ values. We identify τ_{11} and τ_{22} as high or low by comparing them to the infection threshold for the single-layer case, i.e. τ_{11} is high means

$$\tau_{11} \gg \frac{\gamma_1}{n_{11}},$$

and τ_{11} is low means

$$\tau_{11} \ll \frac{\gamma_1}{n_{11}}.$$

We can extend this to define high interlayer spreading rate τ_{12} as

$$\tau_{12} \gg \frac{\gamma_1}{n_{12}},$$

and low interlayer spreading rate as

$$\tau_{12} \ll \frac{\gamma_1}{n_{12}},$$

and likewise for τ_{21}. We can label the different scenarios as in Table 5.1: i.e., numbered from scenario 1 to scenario 16 based on whether the parameters $\tau_{11}, \tau_{22}, \tau_{12}$ and τ_{21} are high or low.

				τ_{22}			
				high		low	
				τ_{21}		τ_{21}	
				high	low	high	low
τ_{11}	high	τ_{12}	high	1	2	9	10
			low	3	4	11	12
	low	τ_{12}	high	13	14	5	6
			low	15	16	7	8

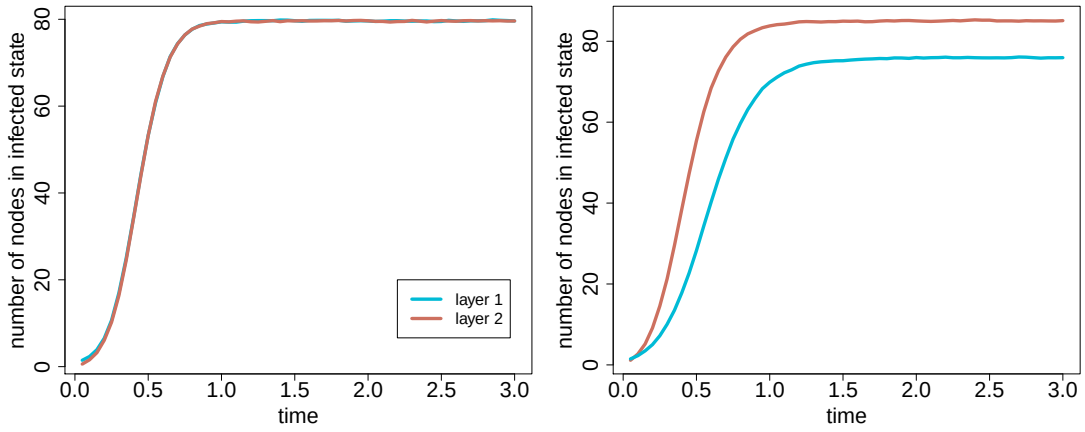
Table 5.1: Table highlighting the different parameter combinations possible. These parameter combinations are numbered, and the corresponding scenarios are referred to in the text by the same number, e.g. scenario 1 refers to a scenario where $\tau_{11}, \tau_{22}, \tau_{12}$ and τ_{21} are all high.

To explore these different scenarios I analysed the situation where the network is the same Erdős-Rényi network on both layers, generated to have 100 nodes with $p = 0.5$. I used $p = 0.5$ so that all graphs with $N = 100$ are equally likely. This resulted in a network on each layer with an average degree of 49.56. I assumed each node was connected to itself (i.e. $n_{12} = n_{21} = 1$, each node was capable of spreading the infection between layers by infecting its replica). I performed 1000 simulations and, in contrast to the previous investigation where the simulations

started with all nodes infected, here we seed the infection by infecting a randomly selected node in layer 1. This creates an asymmetry in the initial system. Epidemics will generally occur in this way, with perhaps one or more simultaneous seeds, and the results will highlight how stochasticity early on in the process can hugely affect the outcome in some situations. The results for each scenario are summarised below.

Scenarios 1-4 have high intralayer spreading on both layers. Due to the high spreading on layer 1 in all of these instances, an epidemic will always be sustained on layer 1. If the epidemic reaches layer 2, an epidemic will also be sustained on layer 2. Therefore in scenarios 1 and 2, where τ_{12} is high, the infection spreads quickly down to layer 2 and quickly reaches an endemic state on both layer 1 and layer 2. Figure 5.6 shows plots demonstrating this behaviour for examples of scenarios 1 (5.6(a)) and 2 (5.6(b)). The blue lines in these plots show the number of infected nodes on layer 1, while the orange lines show the number of infected nodes on layer 2. In both of these scenarios, the infection spreads quickly onto layer 2 due to the high value of τ_{12}. However, in scenario 2, the low value of τ_{21} means that layer 1 has a lower endemic steady state value than layer 2.

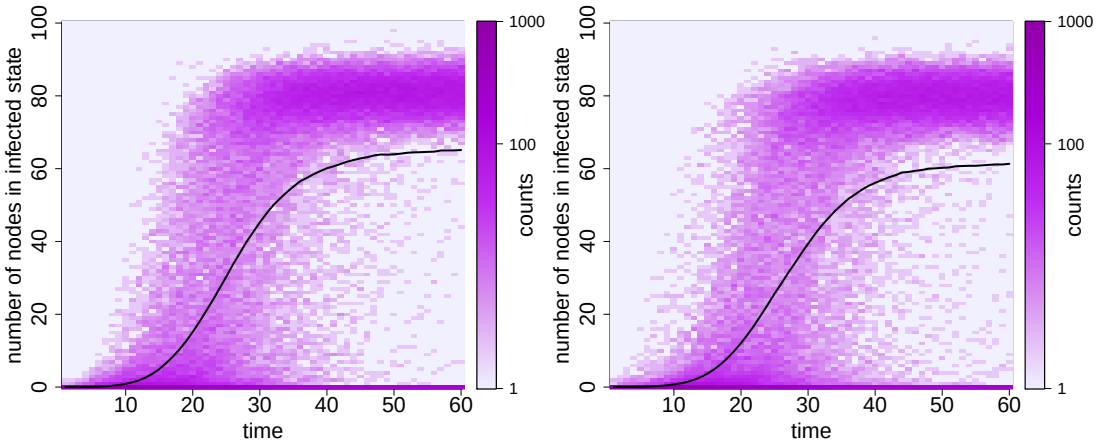
However, in scenarios 3 and 4, where τ_{12} is low, the infection only occasionally spreads down to layer 2, and does so after varying times. The ensemble is therefore composed of simulations showing two very different outcomes: either the infection spreads to layer 2 and spreads quickly to reach an endemic state, or the infection never spreads to layer 2. The ensemble average for the number of infected nodes on layer 2 loses physical meaning, and is perhaps not the best way to show the results of the simulations. Instead the ensemble results for the spread on layer 2 for these two scenarios are shown as a heatmaps in Figure 5.7. Both of these heatmaps show the two different types of outcome: there is a large concentration of results around the non-zero steady state, and also a large concentration of results in the absorbing infection-free state.



(a) Scenario 1

(b) Scenario 2

Figure 5.6: Plots showing the number of infected nodes in two scenarios (1 and 2) where the intralayer spreading is high. The blue lines indicate the number of infected nodes on layer 1, while the orange lines indicate the number of infected nodes on layer 2. In both scenario 1 (panel (a)) and 2 (panel (b)) τ_{12} is high, and so the infection quickly spreads to layer 2 and reaches an endemic state on both layers. In scenario 1, due to the networks on each layer being the same and the parameters $\tau_{11} = \tau_{22}$ and $\tau_{12} = \tau_{21}$, the endemic steady state value is the same for both layers, and so the lines lie on top of each other. In scenario 2 τ_{21} is low, and so layer 1 has a lower endemic steady state value than layer 2.



(a) Scenario 3

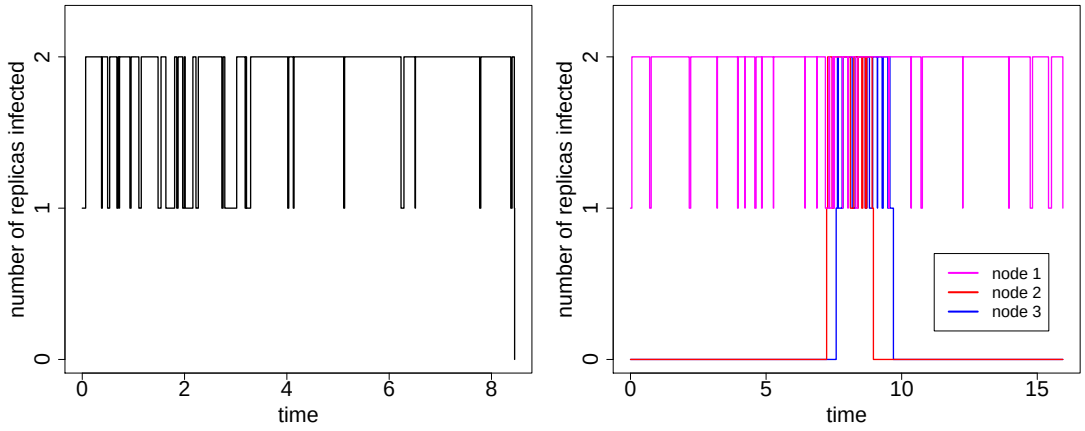
(b) Scenario 4

Figure 5.7: Heatmaps showing the number of infected nodes on layer 2 for two scenarios (3 and 4) where the intralayer spreading is high. Scenario 3 is illustrated in panel (a), and scenario 4 is illustrated in panel (b). For both these scenarios the interlayer spreading τ_{12} is low, and we see a large spread of results for layer 2 in the ensemble. In some cases the infection reaches layer 2, and subsequently spreads on layer 2 to reach an endemic state, while in others the infection never spreads to layer 2 within the course of the simulation. The black lines show the ensemble averages, making the point that the ensemble average does not capture either of the two extreme outcomes.

When there is low intralayer spreading on both layers, as in **scenarios 5-8**, the intralayer spreading is below each of the single-layer epidemic thresholds. Therefore neither layer would be able to sustain an epidemic in isolation. As suggested by the steady state stability condition 2, it is possible for non-zero steady state values to still be achieved in these scenarios if the interlayer spreading rates are large enough. In reality, a node can infect its replica in the other layer very quickly, but neither copy of the node is then able to infect any of its neighbours before it recovers due to low τ_{11} and τ_{22}. It is possible that the node and its replica can get into an extended cycle of recovering and then being re-infected by their still-infected replica. I have illustrated this behaviour for two different simulations in Figure 5.8. Each plot shows, for each node which is infected at some point in the simulation, the number of its infected replicas with

time. The maximum value this can be is 2, when the replicas on layer 1 and layer 2 are both infected. Figure 5.8(a) shows an extreme situation where an initially infected node enters into a cycle of infection/recovery with its replica, and no other nodes are infected. Figure 5.8(b) shows a similar situation, where a second and third node are infected at some point, and also enter into infection/recovery cycles with their replicas.

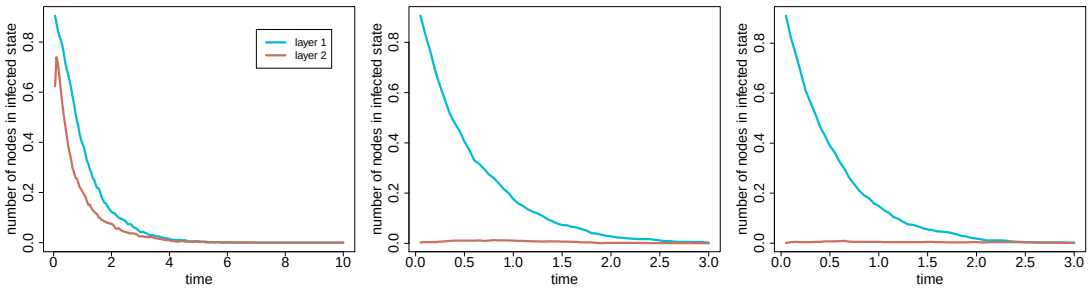
In scenarios 6 and 7 one interlayer infection rate is high, but due to the asymmetry and low intralayer infection rates the infection still dies out on both layers. In scenario 8 all the infection rates are low, and so the infection dies out very quickly. These three scenarios are shown in Figure 5.9.



(a) Scenario 5, one instance

(b) Scenario 5, another instance

Figure 5.8: Plots illustrating the infection dynamics for two examples of scenario 5, where intralayer spreading is low but interlayer spreading is high. The plots show, for nodes which are infected at some point during the simulation, the number of its replicas that are infected. In (a) the initially seeded node infects its replica, one then recovers, is reinfected, and the cycle continues until both node replicas happen to recover. In (b) this same pattern occurs, but by chance the initial node infects one of its neighbours, and then a third node is infected. Each of these nodes enters into a recovery/infection cycle with its replica, until both replicas happen to recover before they can be reinfected.



(a) Scenario 6 (b) Scenario 7 (c) Scenario 8

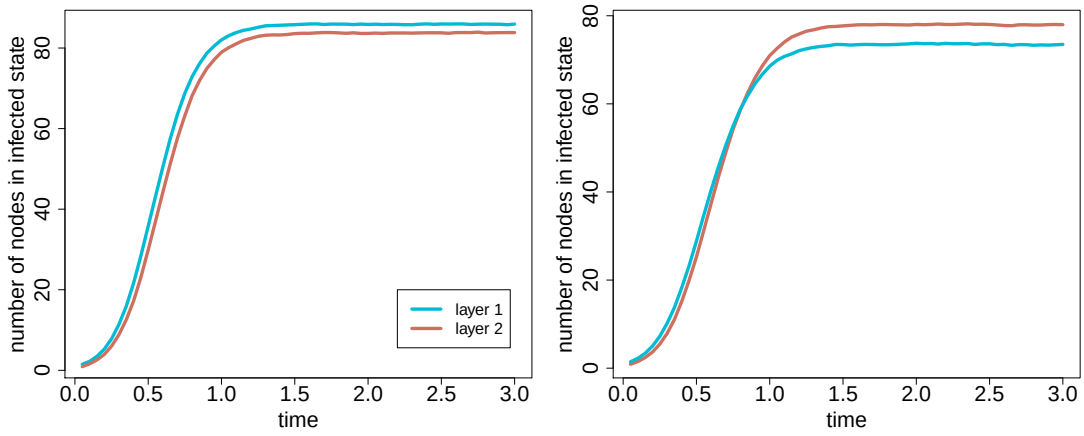
Figure 5.9: Plots showing the number of infected nodes for scenarios (6,7 and 8) where intralayer spreading is low on both layers. The blue lines indicate the number of infected nodes on layer 1, and the orange lines indicate the number of infected nodes on layer 2. In scenario 6, illustrated in panel (a), and scenario 7, illustrated in panel (b), one direction of interlayer spreading is high and the other low. In scenario 8, illustrated in panel (c), interlayer spreading is low in both directions. The interlayer spreading is not enough for any of these cases to sustain some sort of epidemic, and the infection quickly dies out.

In **scenarios 9, 10, 13 and 15** we have potential for an induced epidemic to occur, where one of the layers wouldn't sustain an epidemic in isolation, but sustains an epidemic due to high interlayer infection. In scenarios 9 and 10, there is high infection rate on layer 1, and so the infection quickly reaches the endemic steady state. The high τ_{12} rate means that the infection continuously spreads to layer 2, and the infection is sustained on layer 2 despite the low rate of infection between nodes on layer 2. Figure 5.10 shows the number of infected nodes on layer 2 for both of these cases.

Scenarios 13 and 15 are a little more subtle, but the same phenomenon is at play. The infection rate on layer 1 is low, so in some situations the infection will simply die out. However, if the infection spreads to layer 2 quickly enough - which it will do frequently if τ_{12} is high, as in scenario 13 - then the infection will spread and reach a significant endemic steady state on layer 2. Due to the high value of τ_{21} this infection will then feed layer 1, and induce an epidemic on layer 1. Figure 5.11(a) shows the number of infected nodes on layer 1 for this scenario, which is clearly non-zero despite the low value of τ_{11}. This behaviour

can also occur in scenario 15, although less frequently, because the lower value of τ_{12} means the probability of the infection first reaching layer 2 before it has died out in layer 1 is much lower. Here a heatmap is used in Figure 5.11(b), to again more clearly distinguish between these two outcomes: one where the infection reaches layer 2 before it dies out in layer 1 (and so we get an induced epidemic on layer 1), and one where the infection dies out before it reaches layer 2 (and so we do not get an induced epidemic on layer 1).

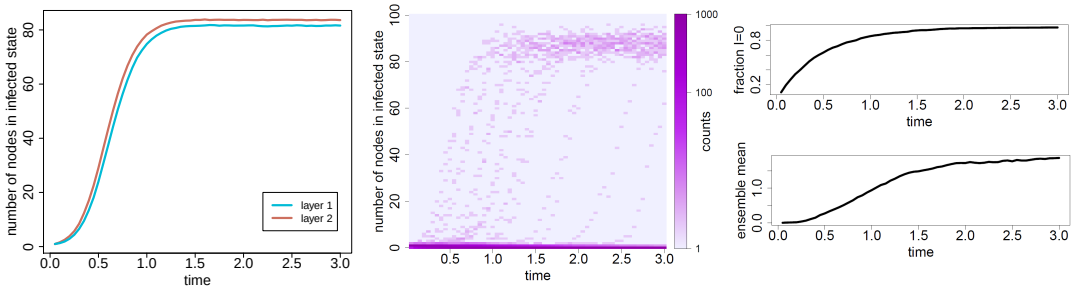
These situations of induced epidemics are notable. If viewed naively, without knowledge of the second endemic layer, these situations will act strangely. If nodes are infected on the visible layer that have no infected neighbours on the visible layer, we might suspect a hidden endemic layer, or a background ambient infection rate. However, if infections observed on the visible layer are driven by the spread on the hidden layer, and consequent intralayer infections from the hidden layer to the visible layer, then we might deduce a much larger infection/recovery ratio on the visible layer than the reality.



(a) Scenario 9

(b) Scenario 10

Figure 5.10: Plots showing the number of infected nodes for two scenarios (9 and 10), where an induced epidemic occurs on layer 2. The blue lines indicate the number of infected nodes on layer 1, and the orange lines indicate the number of infected nodes on layer 2. In both cases τ_{11} is high, and so an epidemic is sustained on layer 1 regardless of layer 2. Scenario 9, illustrated in panel (a), shows the case where interlayer spreading is high in both directions, whereas in scenario 10, illustrated in panel (b) τ_{12} is high but τ_{21} is low. In both cases the intralayer spreading on layer 2 τ_{22} is low, and so the epidemic on layer 2 is sustained due to interlayer infections rather than intralayer infections.



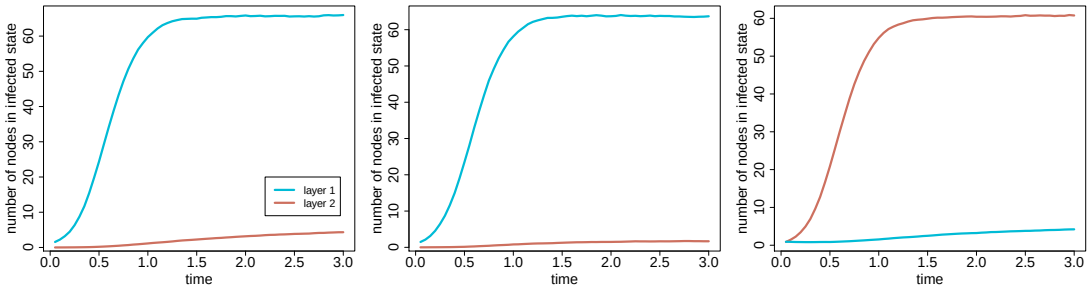
(a) Scenario 13

(b) Scenario 15

Figure 5.11: Plots showing the number of infected nodes for two scenarios (13 and 15), where an induced epidemic occurs on layer 1. Scenario 13 is illustrated in panel (a), and the number of infected nodes on layer 1 and layer 2 are indicated by the blue and orange lines respectively. The high τ_{12} parameter means the infection is quick to spread to layer 2, where it reaches an endemic steady state and induces an infection on layer 1. In scenario 15, τ_{12} is low, and so the scenario has two different outcomes: either the infection dies out on layer 1 before it spreads to layer 2, or it spreads to layer 2, where it again reaches an endemic steady state and induces an infection on layer 1. These two different outcomes are best illustrated in panel (b) with a heatmap showing the number of infected nodes on layer 1. Two other plots are shown in panel (b): the fraction of the simulations which are in the state with zero infected nodes on layer 1, and the ensemble mean of infected nodes on layer 1. The first helps to show the change in the dark line at $I = 0$ in the heatmap. The ensemble simulations that move into this state are ones where the infection never reaches an endemic state on layer 2, and so the epidemic is never induced on layer 1. Most of the simulations (973 out of 1000) reach this state within the simulation time. The final plot in panel (b) allows us to see the ensemble mean more clearly than if this is plotted over the heatmap. Note the reduced axis scale on this plot.

In **scenarios 11, 12, 14 and 16** one layer has a large spreading rate, while the other has a small spreading rate. In each scenario the interlayer τ_{ij} value associated with the infection spreading from nodes in the layer with high infection rate to nodes in the layer with low infection rate is low. Therefore in these scenarios we will only see instances where one layer is in an endemic state, and

the other layers only sees very rare interlayer infections. Examples of scenarios 11, 12 and 14 are shown in Figure 5.12. In scenarios 14 and 16 we will sometimes see instances where the infection dies out completely, as the infection can die out before it spreads down to layer 2. This is especially likely in scenario 16, where τ_{11} and τ_{12} are both low, as shown in the heatmap in Figure 5.13.



(a) Scenario 11 (b) Scenario 12 (c) Scenario 14

Figure 5.12: Plots showing the number of infected nodes for scenarios (11,12 and 13) where the infection reaches a high steady state on one layer, but not the other. The blue lines indicate the number of infected nodes on layer 1, and the orange lines indicate the number of infected nodes on layer 2. These large discrepancies in the steady state value between the layers is due to each scenario having high intralayer spreading on one layer, low intralayer spreading on the other, and low interlayer spreading.

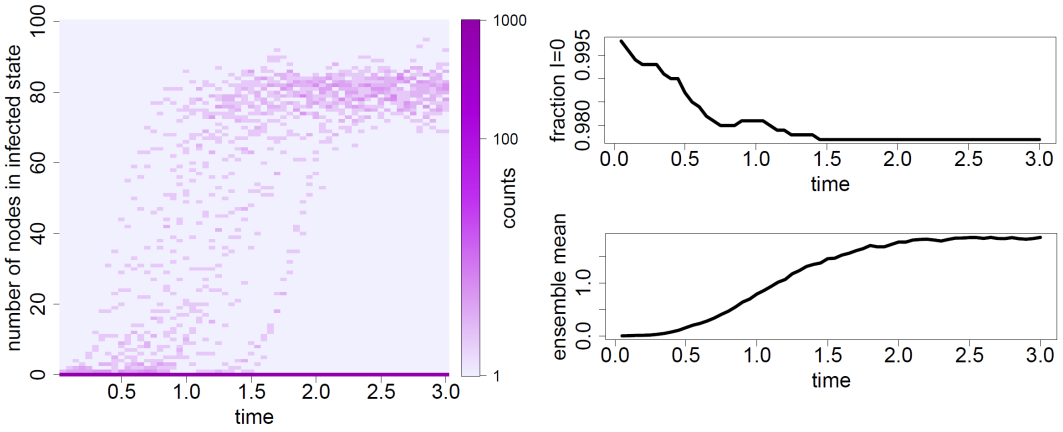


Figure 5.13: Heatmap (left) showing the two different types of outcome on layer 2 for scenario 16, where τ_{11} is low, τ_{22} is high, and τ_{12} and τ_{21} are both low. In some cases the infection simply dies out on layer 1 before it spreads to layer 2. However, in the rare event that the infection spreads to layer 2, then it will take off and reach an endemic state. The plots on the right show the fraction of the simulations that are in a state with zero infected nodes on layer 2 (top), and the ensemble mean of infected nodes on layer 2 (bottom). The top plot helps to show the change in the dark line at $I = 0$ in the heatmap. The ensemble simulations that move into this state are ones where the infection never spreads to layer 2, or spreads to layer 2 but does not reach an endemic state, before the infection dies out on layer 1. Most of the simulations (977 out of 1000) follow this behaviour. The bottom plot allows us to see the ensemble mean more clearly than if this is plotted over the heatmap. Note the reduced axis scale on this plot.

5.4 Selecting an inference scheme

In order to infer the parameters governing the dynamics, an inference scheme has to be selected. First, it is important to note that the likelihood for the full two-layer SIS model as described above is intractable when one of layers is entirely hidden. Therefore, we are forced to consider methods such as those described in Chapter 4.

The fact that we are assuming no knowledge about the hidden layer makes the

inference problem challenging. Any data augmentation method would require us to make assumptions about the network structure of the hidden layer, in order to impute the hidden infection and recovery events. The chances of assuming an accurate enough structure such that the imputed events are compatible with the observed events seemed low. Similarly, to use a likelihood-free ABC method, we would again have to assume the network structure of the hidden layer. Again, it seems unlikely that we would be able to assume an accurate enough hidden layer network for the summary statistics of the simulated events to be within a reasonably small tolerance.

Due to these factors, I decided to use some approximate models for inference instead. The first model I considered was the SISa model, where I have assumed that the effects of the intralayer infections due to the hidden layer can be effectively modelled as a constant ambient background infection rate. The second model is a so-called 'latent variable' (LV) model, where I have assumed that the hidden layer consists of a single node, which is connected to every node on the visible layer. The following sections 5.5 and 5.6 discuss these models in more depth. I show how the likelihood is tractable in both cases, and present some results of trying to infer the parameters of these models in different situations.

As well as having tractable likelihoods, these approximate models help to answer the main question posed in this chapter: in which situations can the multilayer structure of the network be modelled using a cruder structure? If these models can effectively infer the parameters of the system in certain situations, this suggests that the more complicated structure of the hidden layer is not necessary information for these instances. However, as these models are hugely simplified, I expect them to be inappropriate in a large number of situations. These limitations are also discussed in the following sections.

5.5 SISa approximation

Viewing an SIS process across a two-layer network where one of the layers is hidden can cause some nodes to appear as if they are spontaneously infected. A one-layer SIS process cannot account for these events. However an SISa model, which has an extra constant background ambient infection rate ϕ that applies

to every node, might be able to. We would like to explore whether, if we only observed one-layer data of this two-layer SIS process, and naively assumed we were observing a one-layer SISa process, there are any situations where we will accurately infer the infection and recovery parameters.

To investigate this, I first simulated an SIS process on two-layers using the Gillespie algorithm and a known set of parameters. I then flattened the two-layer data to one-layer data by simply removing any information about events occurring on the second layer, taking it to be 'hidden'. Under a one-layer SISa model, events that originally involved nodes on the hidden layer infecting nodes on the visible layer (interlayer infection events) can now be interpreted as either infection events due to nodes on the visible layer, or ambient infection events.

The task is then to fit an SISa model to this flattened one-layer data, and see how well the values for the visible layer infection parameter and recovery parameter are estimated. We can fit the data using an MCMC routine.

5.5.1 MCMC routine

In order to use an MCMC routine, it is necessary to know the probability of a set of observed events, given a set of parameters. To begin calculating this probability, we assume we know all the information about a particular epidemic realisation. This includes knowledge of each event time (and hence the interevent times), and complete knowledge of each event (i.e. which node is affected). Since the SISa model is a Markov process, the information of the state at each time forms a Markov chain. Moreover, since each transition of the Markov process is independent, we can find the probability of the entire visible Markov chain given certain epidemic parameters simply by finding the probabilities of each observed event.

We define the state of the system $X(t)$ at time t, which describes the node-state of each node. The system is observed from $t = 0$ to $t = T$, and we denote the full Markov chain of observed system states between these times as $X_{0:T}$. We label the times at which events e_i are observed t_i, where $i = 1...E$ and E is the total number of events. The set $\mathcal{I}$ contains all I events where an infection occurs, either through same-layer infection or ambient infection, and the set $\mathcal{R}$

contains all R events where a recovery occurs. We label the node affected in the event by n_i, and the number of infected neighbours the affected node has at t_i by $N_I(n_i)$. The system has three dynamical parameters: the infection rate τ which determines the likelihood of infection spreading from an infected node to its susceptible neighbours, the recovery rate γ, and the background ambient infection rate ϕ affecting all susceptible nodes.

Using the fact that the interevent times are exponentially distributed (i.e. the recovery and infection processes can be modelled as Poisson processes with corresponding rates), then the probability of k events happening in any continuous time interval δt is

$$p(k \text{ events in interval } \delta t) = \exp(-r\delta t)\frac{(r\delta t)^k}{k!},$$

where r is the rate at which events occur [135]. Therefore the probability of an event (of any kind) occurring in a given interevent time is

$$p(\text{a single event occuring in interval } \delta t_i) = (r_i \delta t_i)\exp(-r_i \delta t_i),$$

where r_i is the rate of an event occurring in interval $\delta t_i = t_i - t_{i-1}$, defined by the sum of propensities of the nodes at state $X(t_{i-1})$:

$$\begin{aligned} r_i = \gamma \times (\text{number of infected nodes at } t_i) + \tau \sum_{\substack{\text{susceptible} \\ \text{nodes } s}} N_I(s) \\ + \phi \times (\text{number of susceptible nodes at } t_i). \end{aligned} \tag{5.3}$$

This is the probability of any event occurring. To find the probability of a specific observed event occurring, we then need to multiply this value by the probability of the observed event occurring, conditional on the fact that we know an event (of any kind) has occurred. For an infection event, this probability is $\frac{\tau N_I(n_i)+\phi}{r_i}$. For a recovery event this probability is $\frac{\gamma}{r_i}$. To find the total probability of the entire Markov chain being observed, given certain parameter values for τ, γ and ϕ, we also need to calculate the product of the probabilities of each individual

event. In this way we find that

$$\begin{aligned} p(X_{0:T} \mid \tau, \gamma, \phi) \propto & \prod_{i=1}^{E} (r_i \delta t_i) \exp(-r_i \delta t_i) \\ & \times \prod_{e_i \in \mathcal{I}} \frac{\tau N_I(n_i) + \phi}{r_i} \\ & \times \prod_{e_i \in \mathcal{R}} \frac{\gamma}{r_i}. \end{aligned} \tag{5.4}$$

If we focus on the terms which depend on the parameters we find

$$p(X_{0:T} \mid \tau, \gamma, \phi) \propto (\gamma)^R \prod_{e_i \in \mathcal{I}} (\tau N_I(n_i) + \phi) \prod_{i=1}^{E} \exp(-r_i \delta t_i).$$

And so we can write the log-likelihood of the observed Markov chain as

$$\mathcal{L} = \log(p(X_{0:T} \mid \tau, \gamma, \phi)) = R \log(\gamma) + \sum_{e_i \in \mathcal{I}} \log(\tau N_I(n_i) + \phi) - \sum_{i=1}^{E} r_i \delta t_i + C,$$

where C is the normalisation constant. Differentiating this likelihood with respect to the recovery rate parameter, we find

$$\frac{\partial \mathcal{L}}{\partial \gamma} = \frac{R}{\gamma} - \sum_{i=1}^{E} (\text{number of infected nodes at } t_i) \delta t_i.$$

Setting this derivative to zero we find that the maximum likelihood value

$$\hat{\gamma} = \frac{R}{\sum_{i=1}^{E} (\text{number of infected nodes at } t_i) \delta t_i}.$$

However, the equivalent maximising processes for the parameters τ and ϕ result in simultaneous equations for $\hat{\tau}$ and $\hat{\phi}$ that are not solvable analytically. The maximum likelihood values also fail to give us clear information about the error of these estimations. In contrast, Bayesian methods allow us to incorporate prior beliefs of these parameter values, as well as giving us easily interpretable posterior probability distributions over the three parameters. We can apply a straightforward MCMC algorithm, using the above expression for the log-likelihood to accept or reject each proposed set of parameter values. For now, we use an uninformative prior.

We label the parameters of the SIS two-layer system as $\tau_{11}, \tau_{22}, \tau_{12}, \tau_{21}, \gamma_1$ and γ_2 as before. The three parameters of the SISa approximation are simply labelled τ, γ and ϕ. The recovery rate γ here has a direct parallel with the SIS parameter γ_1. Since the visible layer infections recover independently of events on the hidden layer, we expect the MCMC will recover the value for γ_1 accurately. The ambient rate ϕ has no direct parallel, and captures the effective background rate due to the multiple parameters $\tau_{22}, \tau_{12}, \tau_{21}$, and γ_2 affecting the hidden layer. We can draw comparisons between τ and τ_{11}, but it must be noted that since we are fitting a model that is ultimately incorrect, we may find combinations of the parameters τ and ϕ that best describe the dynamics even when the value for τ is far from the actual value of τ_{11}.

5.5.2 Results

From numerical experiments, I have found that the results broadly lie between three extreme regimes, based on the SIS parameters and the relationship between the layers. I will refer in this section to 'systems', by which I mean a given combination of parameters and network structure. For each system I first simulated an SIS epidemic on the full network structure, each time seeded with 1 infected node on the visible layer, before discarding the information about the hidden layer. I applied the MCMC routine described in the previous section to fit SISa dynamics to the remaining 1-layer data. For each inference experiment, I drew 100000 samples from the posterior distribution for each parameter. It is also worth noting that in all of these examples, the MCMC routine was initialised several times: first with values for τ and γ exactly equal to the true values for τ_{11} and γ_1 respectively, and then for several values further away from the true values. In all cases, regardless of the initial parameter values, the Markov chain appeared to converge on the same values, as the resulting distributions were all centred on the same values. The different regimes, and the accuracy of the converged values in each case, will be discussed in turn.

The first regime contains the systems where there are either no or very few interlayer events affecting the visible layer (case 1). The effective ambient rate is

negligible, and so the posterior density for τ will accurately match the value of τ_{11}, while the posterior density for ϕ will suggest a value close to zero. The SISa inference results of an example realisation falling within this regime are shown in Figure 5.14. The epidemic was simulated on a network where both layers are the same Erdős-Rényi network with 100 nodes, $p = 0.3$. The low values of $\tau_{12} = \tau_{21} = 0.00005$ meant that the epidemic simulations typically showed no interlayer events, and I chose one such simulation for inference. The figure shows the posterior probability densities for the parameters, and (in all future plots as in this plot) the thick black lines illustrate the true parameter values. Notably the true value of $\tau_{11} = 0.6$ lies well within the posterior distribution for the τ parameter, and the distribution for ϕ is skewed heavily towards a zero or low value.

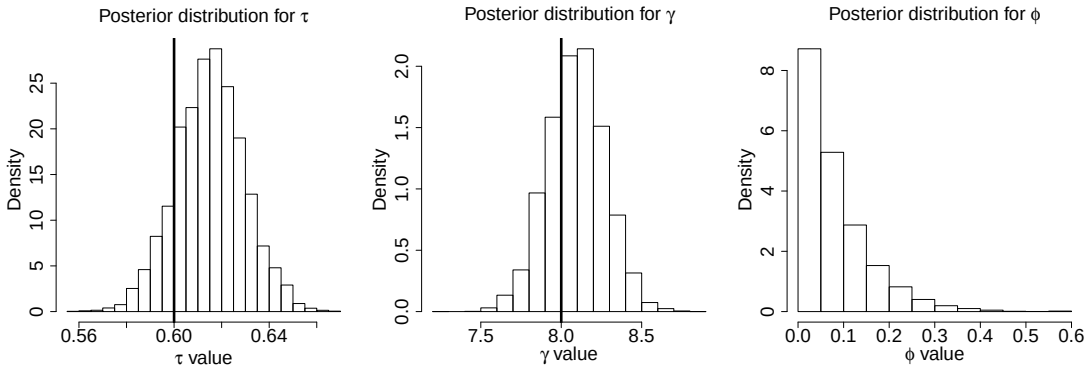


Figure 5.14: Plot showing example posterior probability densities for case 1. The parameters used to generate the case 1 simulation were $\tau_{11} = 0.6, \tau_{22} = 0.1, \tau_{12} = \tau_{21} = 0.05, \gamma_1 = \gamma_2 = 8$, and the simulation ran until $t = 5$. The true value of $\tau_{11} = 0.6$ (indicated by the thick black line) lies within the distribution for the τ parameter, and the distribution for ϕ is skewed heavily towards a zero or low value, suggesting that this case is best captured by an SISa model that is essentially a 1-layer SIS model. This is not surprising, given the lack of interlayer events.

The second regime contains systems where the interlayer infections from the hidden layer to the visible layer are well-modelled by a value for τ that is higher

than τ_{11}. In this regime we would expect to find a value $\tau > \tau_{11}$, and a low value of ϕ. This can occur in systems where the perceived ambient infections are frequently affecting nodes that already have infected neighbours on the visible layer. This might happen if there is a large overlap between the neighbours of each node's replicas, and the value of τ_{22} is low compared to the value of τ_{21} and τ_{11} (case 2). Infected nodes on the visible layer will infect their replica on the hidden layer. This replica can then infect their neighbouring hidden nodes, which can then infect their replicas on the visible layer. If the neighbourhoods of the two replicas are similar, this will lead to interlayer infections which are easily explained by an increased number of visible layer intralayer infections. The results of an example of a realisation falling within this regime are shown in Figure 5.15. This realisation is again simulated on a network where both layers are the same random network with 100 nodes and $p = 0.3$. This guarantees that, for each pair of node replicas, their neighbourhoods are the same. The values of $\tau_{12} = \tau_{21} = 10$ are high but the value of $\tau_{22} = 0.1$ is low. This time the true value of $\tau_{11} = 0.6$ is not contained in the posterior probability distribution for τ, and the distribution for ϕ appears centred on a low value of ϕ. This system seems best described by an SISa model where some of the interlayer events are attributed to intralayer infection instead.

Another system that can lead to this phenomenon is if the visible layer is quickly saturated with infected nodes before the hidden layer sees many infections, or before the hidden nodes have a chance to infect many visible nodes ($\tau_{11} > \tau_{12}, \tau_{11} > \tau_{22}$, or $\tau_{11} > \tau_{21}$) (case 3). This situation also requires that the majority of visible layer nodes have a large number of infected neighbours. This might happen if the steady state value on the visible layer is high ($\tau_{11} > \gamma_1$), or if the nodes all have very high degree. If these conditions are true then any interlayer infection events on the visible layer will be hard to distinguish from intralayer infection events. Figure 5.16 shows the results of an example of this case, using the same network with identical random layers. Here the fact that the node layers are identical is not important. The distribution of τ is centred on a higher value (1.12) than the true value of $\tau_{11} = 1$, and does not contain the value 1. The distribution of ϕ suggests a low value of the parameter. Again, the best fit for the SISa model seems to attribute some of the interlayer events to the

intralayer parameter τ.

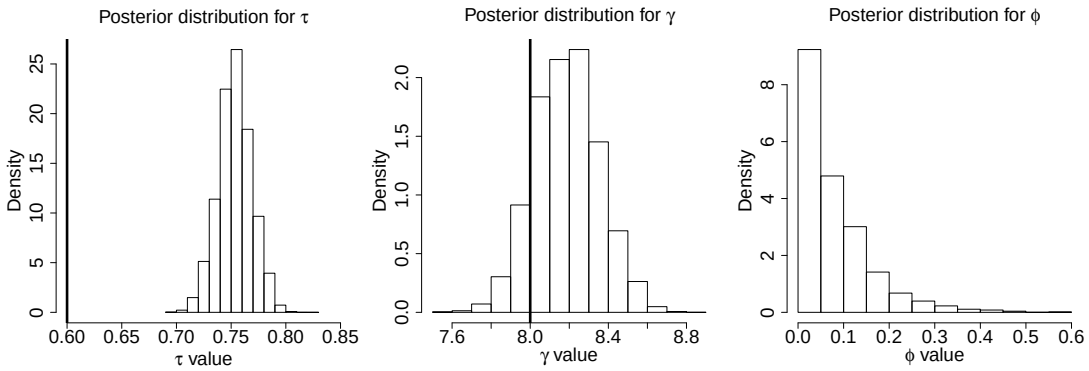


Figure 5.15: Plot showing example posterior probability densities for case 2. The parameters used to generate the case 2 simulation were $\tau_{11} = 0.6, \tau_{22} = 0.1, \tau_{12} = \tau_{21} = 10, \gamma_1 = \gamma_2 = 8$, and the simulation ran until $t = 5$. The two network layers are identical, and the high values of τ_{12} and τ_{21} compared to τ_{22} mean that the interlayer infections often affect the neighbours of infected nodes on the visible layer. In this way the interlayer infections are hard to distinguish from intralayer infection events. As such, the posterior probability densities suggest a value of τ that is higher than the true value of τ_{11}, and a very low value of ϕ.

The third regime describes systems where the effective ambient parameter is high, and cannot be modelled accurately by simply increasing τ relative to τ_{11}. In this regime we would expect to accurately recover $\tau = \tau_{11}$, and find a non-zero estimate for ϕ. This can occur when the interlayer events are highly uncorrelated with the behaviour of the visible layer. For example, this might be a system consisting of two very different network on the two layers (i.e. two network layers with low overlap) (case 4). In a system like this a node is unlikely to share the same neighbours on each of the two layers. If the infection is spreading out from part of the network on the hidden layer, the visible layer replicas of these infected hidden nodes (which could therefore be infected with interlayer infections) are unlikely to be connected. The posterior probability densities for an example of case 4 are shown in Figure 5.17.

Another instance belonging to this regime is the case where we have the same

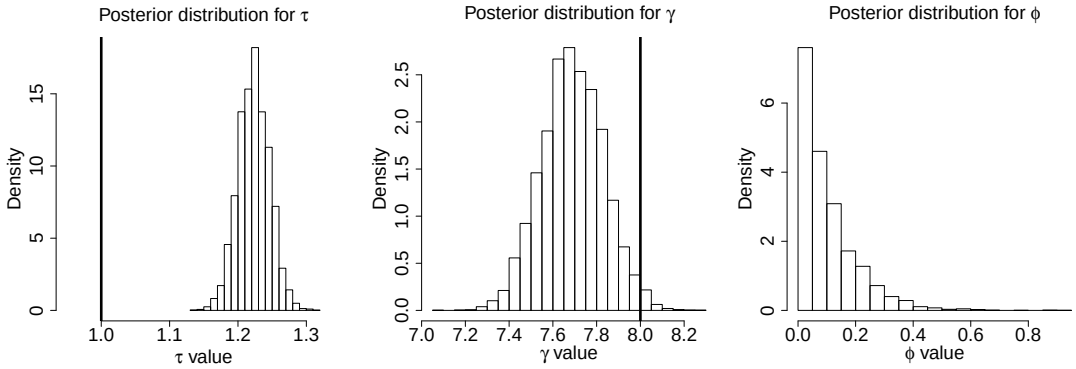


Figure 5.16: Plot showing example posterior probability densities for case 3. The parameters used to generate the case 3 simulation were $\tau_{11} = 1, \tau_{22} = 0.6, \tau_{12} = 0.02, \tau_{21} = 10, \gamma_1 = \gamma_2 = 8$, and the simulation ran until $t = 5$. The visible layer is quickly infected, and reaches a high steady state, before many interlayer events take place. In this way the interlayer events tend to affect nodes that already have infected neighbours on the visible layer. The interlayer infection events are therefore again hard to distinguish from intralayer infection events, and again the posterior probability densities suggest a value of τ higher than the true value of τ_{11}, and a low value of ϕ.

or very similar network layers, but the hidden layer gets saturated with a high steady state value much faster than the visible layer (case 5). The results for an example of case 5 are shown in Figure 5.18. Yet again, we use the same network with two identical random layers. This time the value of $\tau_{22} = 2$ is larger than the value of $\tau_{11} = 0.4$. The posterior distribution for the τ parameter includes the true value of τ_{11}, and the distribution of ϕ is centred around a non-zero (if small) value for ϕ, showing that the intralayer and interlayer events are indeed distinguishable.

These results are summarised in comparison with the latent variable model results in the final section 5.7.1 of this chapter.

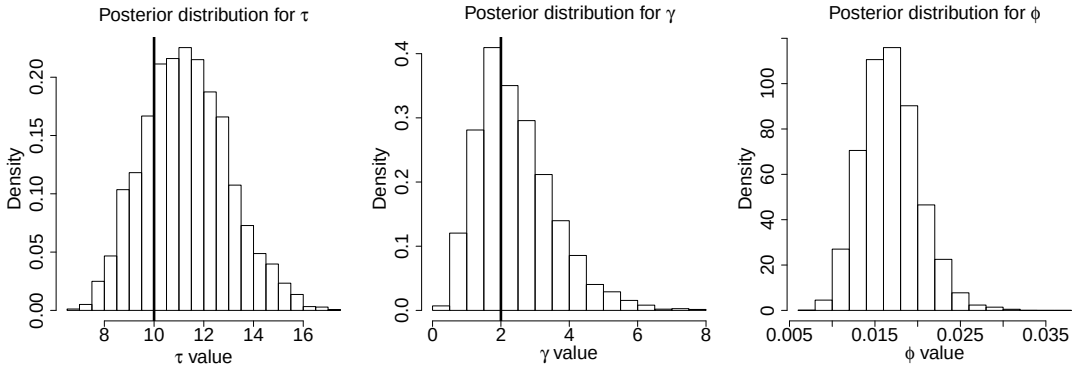


Figure 5.17: Plot showing example posterior probability densities for case 4. The parameters used to generate the case 4 simulation were $\tau_{11} = \tau_{22} = 10, \tau_{12} = \tau_{21} = 50, \gamma_1 = \gamma_2 = 2$, and the simulation ran until $t = 0.15$.The two network layers are very different: both network layers have 9999 edges, but only 22 edges are common to both layers. Therefore as the infection spreads on the hidden layer, the visible node replicas that the hidden nodes can infect are unlikely to be connected. Any interlayer infections will therefore be more likely to appear as ambient infections, and we find a non-zero estimate of ϕ and a distribution for τ that includes the true value of $\tau_{11} = 10$.

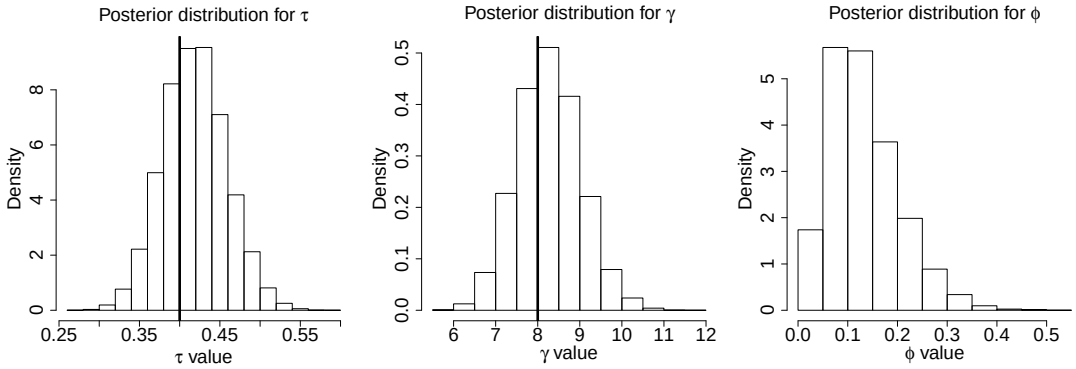


Figure 5.18: Plot showing example posterior probability densities for case 5. The parameters used to generate the case 5 simulation were $\tau_{11} = 0.4, \tau_{22} = 2, \tau_{12} = 1.7, \tau_{21} = 0.05, \gamma_1 = 8, \gamma_2 = 1$, and the simulation ran until $t = 1$.The hidden layer quickly reaches its steady state value, ahead of the visible layer. As with case 4, this means that the interlayer infections are likely to affect visible nodes with no (or very few) infected neighbours. The distribution for τ includes the true value for $\tau_{11} = 0.4$, and the model correctly identifies a set of interlayer events by fitting a non-zero value of ϕ.

5.6 Latent variable model

Considering these regimes, we can identify situations when the SISa model will be ill-fitting. If the hidden layer reaches a high steady state value, the interlayer infections will begin to behave ambiently. However, it might take the system considerable time to reach this point, during which the interlayer infections might instead be strongly correlated with the intralayer infections. This is one example where a static value of the ambient infection rate will describe the system poorly. The limitations of a static ambient infection rate led us to consider a more complex model where the ambient infection rate can change.

In this model we assume that the hidden layer is composed of just a single node that is connected to every node on the visible layer. A picture of this model is shown in Figure 5.19. The hidden node can be in either a susceptible or infected state, and the state of this node can now be thought of as a latent variable. This is equivalent to an SISa system where the ambient parameter is either 'on' or 'off' dependent on the state of the latent node. Effectively we have coupled the ambient parameter to the state of the nodes in the visible layer. We can assess how well this latent variable (LV) model fits the data by using an MCMC scheme.

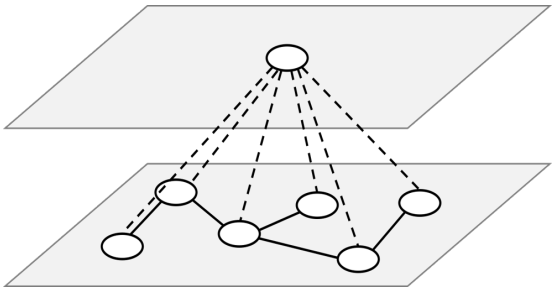


Figure 5.19: A pictorial representation of the latent variable model. The hidden layer has just a single node, which is connected to every node on the visible layer.

5.6.1 Hidden Markov model approach

The MCMC approach requires us to calculate or estimate the likelihood of the observed events, given a certain set of parameters. Assuming the LV variable

model, the state of the hidden layer (i.e. the hidden node) will always be unobserved, while the state of the visible layer is always observed. This led us to consider hidden Markov model (HMM) methods. Below I have explained how we used ideas from the HMM forward algorithm to calculate the likelihood of the Markov chain of observed events.

The forward algorithm is used to calculate the probability of each possible hidden state at a certain time, given a history of observed visible 'motifs'. Here the visible motif is the set of visible-layer nodes and their corresponding node-states. The hidden states refer to the entire state of the system (i.e. the states of both the visible nodes and the hidden node). For each visible motif M_n observed, there are two possible states of the entire system: one where the hidden node is infected which we label $M_n(I)$, and one where it is susceptible which we label $M_n(S)$. In both cases the node-states of the visible-layer nodes in the full-system states $M_n(I)$ and $M_n(S)$ match the node-states of the visible-layer nodes in the corresponding visible motif M_n.

Consider a pair of consecutively observed motifs: M_1 observed at time T_1 and M_2 observed at time T_2. An example is shown in Figure 5.20. We can label $M_1(S), M_1(I), M_2(S), M_2(I)$ as states $1, 2, 3, 4$ respectively. Prior to time T_2 the system can transition between $M_1(S)$ and $M_1(I)$ without our knowledge. The instantaneous rate matrix for the transitions between $M_1(S)$ and $M_1(I)$ is

$$Q_1 = \begin{pmatrix} -R_1 & \tau_{12} I_1(M_1) \\ \gamma_2 & -R_2 \end{pmatrix},$$

where $I_1(M_1)$ is the number of infected nodes on layer 1 (the visible layer) in M_1 (abbreviated to just I_1 in the following), and R_1 and R_2 are the total rates of transition out of states 1 and 2 respectively.

Given the probability vector $\alpha_1 = (\alpha_1(S), \alpha_1(I))$ of being in $M_1(S)$ and $M_1(I)$ at time T_1, then the probabilities of being in states $M_1(S)$ and $M_1(I)$ at time $T_1 + t < T_2$ are described by $\alpha_1 \cdot B_1$, where the transition matrix $B_1 = \exp(Q_1 t)$. Using the eigenvector decomposition of Q_1 where $Q_1 = U D_\lambda U^{-1}$,

$$U = \begin{pmatrix} -\tau_{12} I_1 & -\tau_{12} I_1 \\ -R_1 - \lambda^+ & -R_1 - \lambda^- \end{pmatrix}, D_\lambda = \begin{pmatrix} \lambda^+ & 0 \\ 0 & \lambda^- \end{pmatrix},$$

$$\lambda^\pm = -\frac{(R_1 + R_2)}{2} \pm \frac{1}{2}\sqrt{(R_1 - R_2)^2 + 4\gamma_2(\tau_{12} I_1)},$$

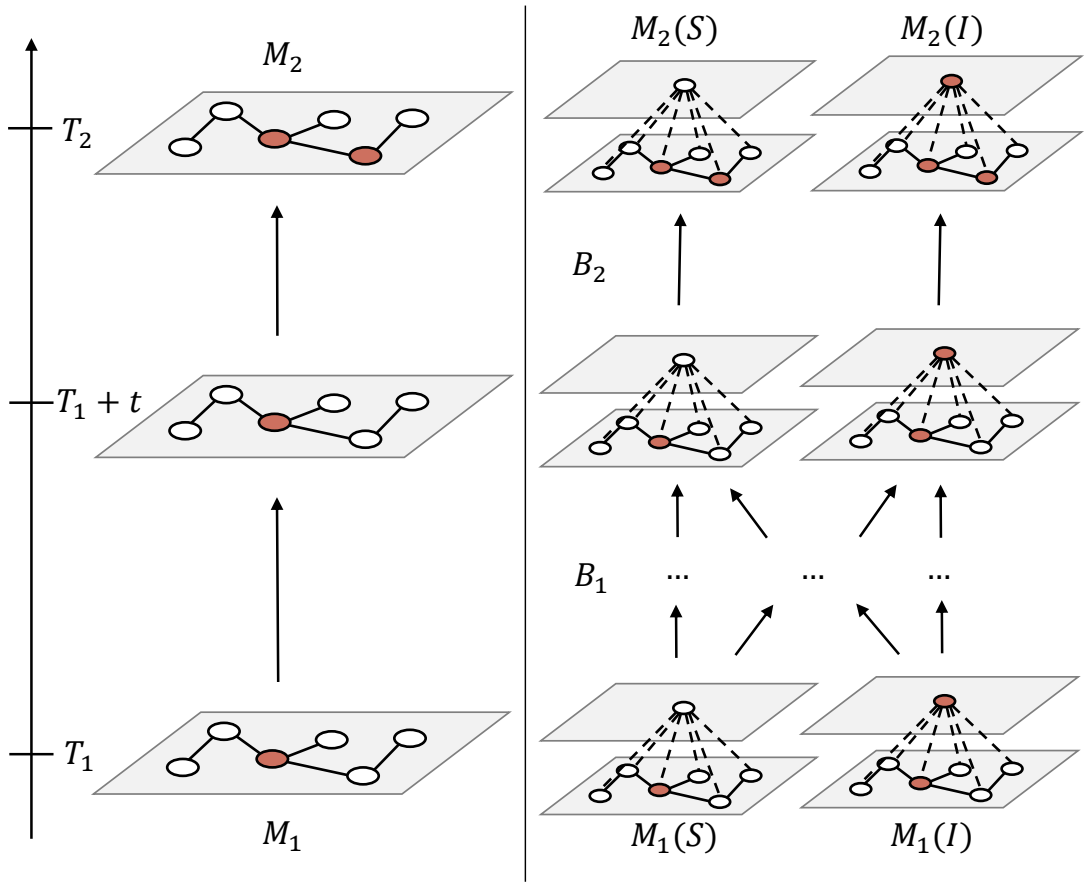


Figure 5.20: This picture demonstrates the different stages considered when calculating the probabilities of observing motif M_2 at time T_2, of being in $M_1(S)$ or $M_1(I)$ at time T_1. The left-hand images show the observed progression of the system. The right-hand images show the possible hidden states of the system corresponding to the observed motifs. Following an observation of motif M_1 at time T_1, we allow the system to transition between $M_1(S)$ and $M_1(I)$ for time t. The probabilities of these transitions are governed by transition matrix B_1, which is derived in the text. A transition must then happen after the time interval $T_2 - (T_1 + t)$ between either $M_1(S)$ and $M_2(S)$ or $M_1(I)$ and $M_2(I)$, so that the system matches the observed motif M_2 at time T_2. The probability of this transition is described by transition matrix B_2. Multiplying the two transition matrices together, and integrating over all possible values of t, gives us the final transition matrix.

then the transition matrix can be written as

$$
\begin{aligned}
B_1 &= \exp(Q_1 t) \\
&= U \exp(t D_\lambda) U^{-1} \\
&= \frac{1}{\tau_{12} I_1 (\lambda^- - \lambda^+)} \times \\
&\begin{pmatrix} \tau_{12} I_1 (-R_1 - \lambda^+) e^{\lambda^- t} - \tau_{12} I_1 (-R_1 - \lambda^-) e^{\lambda^+ t} & (\tau_{12} I_1)^2 (e^{\lambda^- t} - e^{\lambda^+ t}) \\ (-R_1 - \lambda^-)(-R_1 - \lambda^+)(e^{\lambda^+ t} - e^{\lambda^- t}) & \tau_{12} I_1 (-R_1 - \lambda^+) e^{\lambda^+ t} - \tau_{12} I_1 (-R_1 - \lambda^-) e^{\lambda^- t} \end{pmatrix}.
\end{aligned}
$$

However, we also know that at T_2 a transition occurs to a node on the visible layer such that the visible layer matches the observed motif M_2. We have to allow for a transition from $M_1(S)$ and $M_1(I)$ to $M_2(S)$ and $M_2(I)$ after the waiting time $T_2 - (t + T_1) = (T_2 - T_1) - t = \delta T - t$. We denote the rate of transition between state $M_1(S)$ and $M_2(S)$ as r_{SS}, and likewise r_{SI}, r_{IS} and r_{II} for the other three possible transitions. A second matrix describing the transition after the waiting time $\delta T - t$ can then be constructed:

$$
B_2 = \begin{pmatrix} r_{SS}(\delta T - t) e^{-R_1(\delta T - t)} & r_{SI}(\delta T - t) e^{-R_1(\delta T - t)} \\ r_{IS}(\delta T - t) e^{-R_2(\delta T - t)} & r_{II}(\delta T - t) e^{-R_2(\delta T - t)} \end{pmatrix}.
$$

Since we are assuming a SVT model where only transitions involving one node changing can occur, $r_{SI} = r_{IS} = 0$. Multiplying the two transition matrices gives us

$$
\begin{aligned}
A = B_1 B_2 &= \frac{1}{\gamma_2 + \tau_{12} I_1} \times \\
&\begin{pmatrix} r_{SS}(\delta T - t) e^{-R_1(\delta T - t)} \left(\gamma_2 + \tau_{12} I_1 e^{-(\gamma_2 + \tau_{12} I_1) t}\right) & r_{II}(\delta T - t) e^{-R_2(\delta T - t)} \left(\tau_{12} I_1 - \tau_{12} I_1 e^{-(\gamma_2 + \tau_{12} I_1) t}\right) \\ r_{SS}(\delta T - t) e^{-R_1(\delta T - t)} \left(\gamma_2 - \gamma_2 e^{-(\gamma_2 + \tau_{12} I_1) t}\right) & r_{II}(\delta T - t) e^{-R_2(\delta T - t)} \left(\tau_{12} I_1 + \gamma_2 e^{-(\gamma_2 + \tau_{12} I_1) t}\right) \end{pmatrix}.
\end{aligned}
$$

We now have to integrate this matrix product over potential times $0 \leq t \leq \delta T$, to sum over all potential transition possibilities. Taking the first term $A_{1,1}$ in the matrix as an example, we first expand the expression, grouping the terms by powers of t:

$$
\begin{aligned}
\int_{t=0}^{t=\delta T} A_{1,1} dt &= \int_{t=0}^{t=\delta T} r_{SS}(\delta T - t) e^{-R_1(\delta T - t)} \left(\gamma_2 + \tau_{12} I_1 e^{-(\gamma_2 + \tau_{12} I_1) t}\right) dt \\
&= r_{SS} \int_{t=0}^{t=\delta T} \gamma_2 e^{-R_1 \delta T} \delta T e^{R_1 t} + \tau_{12} I_1 e^{-R_1 \delta T} \delta T e^{(R_1 - \gamma_2 - \tau_{12} I_1) t} - \\
&\quad \gamma_2 e^{-R_1 \delta T} t e^{R_1 t} - \tau_{12} I_1 e^{-R_1 \delta T} t e^{(R_1 - \gamma_2 - \tau_{12} I_1) t} dt.
\end{aligned}
$$

We can then perform the integration, applying integration by parts to the third and fourth terms:

$$\int_{t=0}^{t=\delta T} A_{1,1}dt = r_{SS}\left[\gamma_2 e^{-R_1\delta T}\frac{\delta T}{R_1}e^{R_1 t} + \tau_{12}I_1 e^{-R_1\delta T}\frac{\delta T}{R_1-\gamma_2-\tau_{12}I}e^{(R_1-\gamma-\tau_{12}I_1)t}\right]_0^{\delta T} -$$
$$r_{SS}\Bigg[\gamma_2 e^{-R_1\delta T}\frac{1}{R_1^2}e^{R_1 t}(R_1 t-1)+$$
$$\tau_{12}I_1 e^{-R_1\delta T}\frac{1}{(R_1-\gamma-\tau_{12}I_1)^2}e^{(R_1-\gamma_2-\tau_{12}I_1)t}\left((R_1-\gamma-\tau_{12}I_1)t-1\right)\Bigg]_0^{\delta T}.$$

Evaluating this at the limits we find

$$\int_{t=0}^{t=\delta T} A_{1,1}dt = r_{SS}\Bigg[\gamma_2 e^{-R_1\delta T}\frac{\delta T}{R_1}e^{R_1\delta T} - \gamma_2 e^{-R_1\delta T}\frac{\delta T}{R_1}$$
$$+\tau_{12}I_1 e^{-R_1\delta T}\frac{\delta T}{R_1-\gamma_2-\tau_{12}I_1}e^{(R_1-\gamma_2-\tau I)\delta T} - \tau_{12}I_1 e^{-R_1\delta T}\frac{\delta T}{R_1-\gamma_2-\tau_{12}I_1}\Bigg] -$$
$$r_{SS}\Bigg[\gamma_2 e^{-R_1\delta T}\frac{1}{R_1^2}e^{R_1\delta T}(R_1\delta T-1) + \gamma e^{-R_1\delta T}\frac{1}{R_1^2}$$
$$+\tau_{12}I_1 e^{-R_1\delta T}\frac{1}{(R_1-\gamma_2-\tau_{12}I_1)^2}e^{(R_1-\gamma_2-\tau_{12}I_1)\delta T}\left((R_1-\gamma_2-\tau_{12}I_1)\delta T-1\right)$$
$$+\tau_{12}I_1 e^{-R_1\delta T}\frac{1}{(R_1-\gamma_2-\tau_{12}I_1)^2}\Bigg],$$

which simplifies to

$$\int_{t=0}^{t=\delta T} A_{1,1}dt = r_{SS}\Bigg(\frac{\gamma_2}{R_1^2}(1-(1+R_1T_2)\exp^{-R_1T_2})$$
$$+\frac{\tau_{12}I_1}{(R_1-\gamma_2-\tau_{12}I_1)^2}(\exp^{-(\gamma_2+\tau_{12}I_2)T_2} - (1+(R_1-\gamma_2-\tau_{12}I_1))\exp^{-R_1T_2})\Bigg).$$

The results for the other three terms are

$$\int_{t=0}^{t=\delta T} A_{1,2}dt = r_{II}\Bigg(\frac{\tau_{12}I_1}{R_2^2}(1-(1+R_2T_2)e^{-R_2T_2}) \\ -\frac{\tau_{12}I_1}{(R_2-\gamma_2-\tau_{12}I_1)^2}(e^{-(\gamma_2+\tau_{12}I_2)T_2}-(1+(R_2-\gamma_2-\tau_{12}I_1))e^{-R_2T_2})\Bigg)$$

$$\int_{t=0}^{t=\delta T} A_{2,1}dt = r_{SS}\Bigg(\frac{\gamma_2}{R_1^2}(1-(1+R_1T_2)e^{-R_1T_2}) \\ -\frac{\gamma_2}{(R_1-\gamma_2-\tau_{12}I_1)^2}(e^{-(\gamma_2+\tau_{12}I_2)T_2}-(1+(R_1-\gamma_2-\tau_{12}I_1))e^{-R_1T_2})\Bigg)$$

$$\int_{t=0}^{t=\delta T} A_{2,2}dt = r_{II}\Bigg(\frac{\tau_{12}I_1}{R_2^2}(1-(1+R_2T_2)e^{-R_2T_2}) \\ +\frac{\gamma_2}{(R_2-\gamma_2-\tau_{12}I_1)^2}(e^{-(\gamma_2+\tau_{12}I_2)T_2}-(1+(R_2-\gamma_2-\tau_{12}I_1))e^{-R_2T_2})\Bigg).$$

Now we can calculate the probabilities of being in states $M_2(S)$ and $M_2(I)$ at time T_2, given the probabilities of being in states $M_1(S)$ and $M_1(I)$ at time T_1. These probabilities are conditional, so tildes are used to differentiate between conditional and unconditional probabilities:

$$\begin{pmatrix}\tilde{\alpha}_2(S)\\ \tilde{\alpha}_2(I)\end{pmatrix} = \begin{pmatrix}\alpha_1(S)\\ \alpha_1(I)\end{pmatrix}\cdot\int A dt.$$

So, the total probability of seeing motif M_2 at time T_2, given all previous observations, is the sum $a_2 = \tilde{\alpha}_2(S) + \tilde{\alpha}_2(I)$. If we now want to move on to consider the next time interval and event, we need to know the probability of being in $M_2(S)$ and $M_2(I)$, given that we observe M_2. Therefore we need to normalise the probabilities

$$\begin{pmatrix}\alpha_2(S)\\ \alpha_2(I)\end{pmatrix} = \frac{1}{a_2}\begin{pmatrix}\tilde{\alpha}_2(S)\\ \tilde{\alpha}_2(I)\end{pmatrix},$$

before performing the same calculations as above with the next interval.

For a given set of observed motifs $M_1, M_2, ..., M_n$ measured at times $T_1, T_2, ..., T_n$, we can perform this process iteratively for all pairs of consecutively observed motifs. This gives us the probabilities $a_1, a_2, ..., a_n$ of seeing the observed motifs $M_1, M_2, ..., M_n$ respectively. The total probability of seeing the sequence of observed motifs $P(M_1...M_n)$ is then the product of these probabilities, or, if we take logs:

$$\log P(M_1...M_n) = \sum_{i=1}^{n}\log a_i.$$

This is the probability of seeing all the observed events, over the full time in which observations are measured. This probability is conditional on the chosen parameters, and can be used as the likelihood measure in the MCMC algorithm.

5.6.2 Results

Again, I performed a range of numerical experiments to explore the results of using this LV model to infer the dynamical parameters of the system, fitting only the visible layer event data for each simulation. In each case I set the probability vector $\alpha_1 = (\alpha_1(S), \alpha_1(I)) = (1, 0)$ in the HMM forward algorithm, i.e. I assumed the hidden node is initially susceptible. Since the MCMC routine requires us to calculate the transition rates r_{SS}, r_{II}, R_1 and R_2 for each pair of motifs, for each set of proposed parameter values, this MCMC scheme is much more computationally demanding. I found I could only practically draw 5000 samples from the posterior distributions for each inference experiment. I initialised the MCMC routines several times again: first with values for τ_{11} and γ_1 exactly equal to the true values for τ_{11} and γ_1, and then for several values futher away from the true values. Due to the small number of samples I could draw, the further away values struggled to converge on a solution effectively. This highlights an area of improvement, in finding ways to further optimise the MCMC routine and allow for more samples within a reasonable time. Note that in the following plots showing the MCMC posterior probability distributions, I have included a result showing a distribution for τ_{22}. This is somewhat misleading: since the LV model does not feature any possible intralayer infections on layer 2, and the HMM forward algorithm does not depend on the parameter τ_{22}, the acceptance ratio in the MCMC does not depend on the new proposed value of τ_{22}. We would therefore not expect to see any obvious single peak in the distribution. This meant that including the distribution for τ_{22} was still useful: if there were clear gaps and very isolated peaks in the distribution for τ_{22}, this was a quick indicator of potentially poor mixing.

In some situations, the effect of the hidden layer is consistent throughout the vast majority of the simulation. For example, if the nodes on the hidden layer never get infected (if τ_{12} is low) and there are solely intralayer events, then the

hidden layer effectively does not contribute to the dynamics. Figure 5.21 shows the result of simulating an epidemic where this happens and fitting it using the LV variable model, on a network where both layers are the same random Erdős-Rényi network ($N = 100$, $p = 0.3$). There are no interlayer events, and so the nodes on the hidden layer are never infected. The HMM MCMC finds posterior distributions for τ_{11}, τ_{12} and τ_{21} that contain the true, very low values (indicated by the thick black lines). There is a parallel to make here: the LV variable model with effectively-zero values for τ_{12} and τ_{21} is essentially a 1-layer SISa model with $\phi = 0$. Since the 1-layer SISa model accurately recovered the value of τ_{11} with a value $\phi = 0$ in a similar situation in Section 5.5.2 (see Figure 5.14), it is unsurprising that the LV model also successfully recovers the value of τ_{11} here, and finds τ_{12} and τ_{21} to be low.

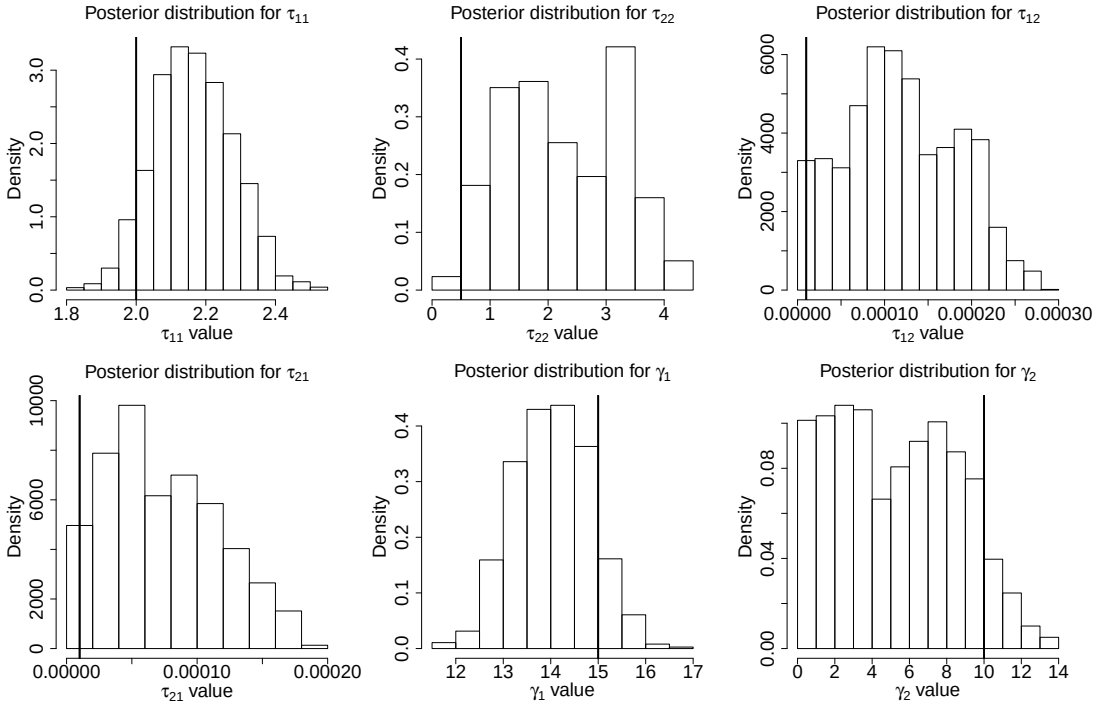


Figure 5.21: Plot showing the LV model results for the case of no interlayer events. The parameters used to generate this case were $\tau_{11} = 2, \tau_{22} = 0.5, \tau_{12} = \tau_{21} = 0.00001, \gamma_1 = 15, \gamma_2 = 10$, and the simulation ran until $t = 0.3$. The posterior probability distribution for τ_{11} contains the true value for the parameter (marked by the thick black lines), and the distributions for $\tau_{12} = \tau_{21} = 0.00001$ both seem to centre on a value extremely low compared to the other rates.

Likewise if the nodes on the hidden layer are all very quickly infected and reach a high steady state value (high τ_{12}, $\tau_{22} >> \tau_{11}$) then the rate of interlayer infections from layer 2 to layer 1 is consistent (at a high value) for the entire run. Figure 5.22 shows the result of using the LV model to fit a simulation of an epidemic where this happens, again on a network where both layers are the same random network ($N = 100$, $p = 0.3$). Here both τ_{12} and τ_{21} have posterior distributions suggesting non-zero values. The true value of τ_{11} falls within the posterior distribution. Again, we have a parallel with the SISa model here. An LV model where the hidden node is infected early and stays infected is equivalent to the SISa model with a non-zero value of ϕ. Since the SISa model performed

well applied to a similar example in Section 5.4, it is again unsurprising that the LV model recovers the true value of τ_{11} well here.

In these situations, where the effect of the hidden layer is consistent throughout the majority of the simulation, both the LV model and the SISa model perform well, with the computationally expensive LV model not obviously offering much advantage over the simpler and quicker SISa inference scheme.

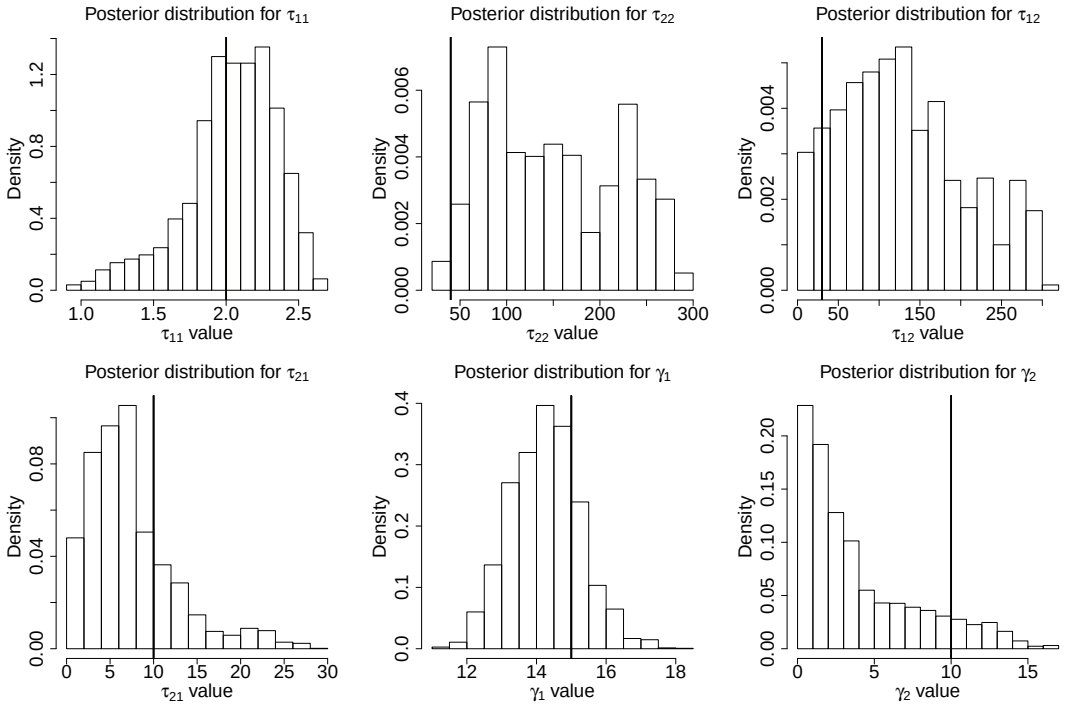


Figure 5.22: Plot showing the LV model results for the case where the hidden layer is quickly saturated with infected nodes. The parameters used to generate the simulation in this case were $\tau_{11} = 2, \tau_{22} = 40, \tau_{12} = 30, \tau_{21} = 10, \gamma_1 = 15, \gamma_2 = 10$, and the simulation ran until $t = 0.2$. The model produces posterior distributions for τ_{11}, τ_{12} and τ_{21} that all contain the true values of these parameters. This makes sense: a system where the majority of the nodes on the hidden layer become infected will act very similar to a LV model with a hidden node that is infected.

Alternatively, there are situations where the effect of the hidden layer changes during the simulation. For example, if we observe a system with the exact struc-

ture that the LV model assumes (i.e. one hidden node on the second hidden layer connected to every node on the visible layer), and set τ_{12} and γ_2 low, it is possible to simulate a situation where the hidden node is susceptible for the majority of the first half of the simulation, and infected for the second half. The results of fitting this situation, simulated with a path graph of 20 nodes on the visible layer, with the LV model are presented in Figure 5.23. The LV model captures the nuance of this situation well, correctly estimating the value for τ_{11} and the high value of τ_{21}. The results for the same data fitted with the SISa model are shown in Figure 5.24. The SISa results show a posterior distribution for τ_{11} that does not contain the true value of 1. This is in contrast to the LV model, and shows a clear example of where the LV model has an advantage.

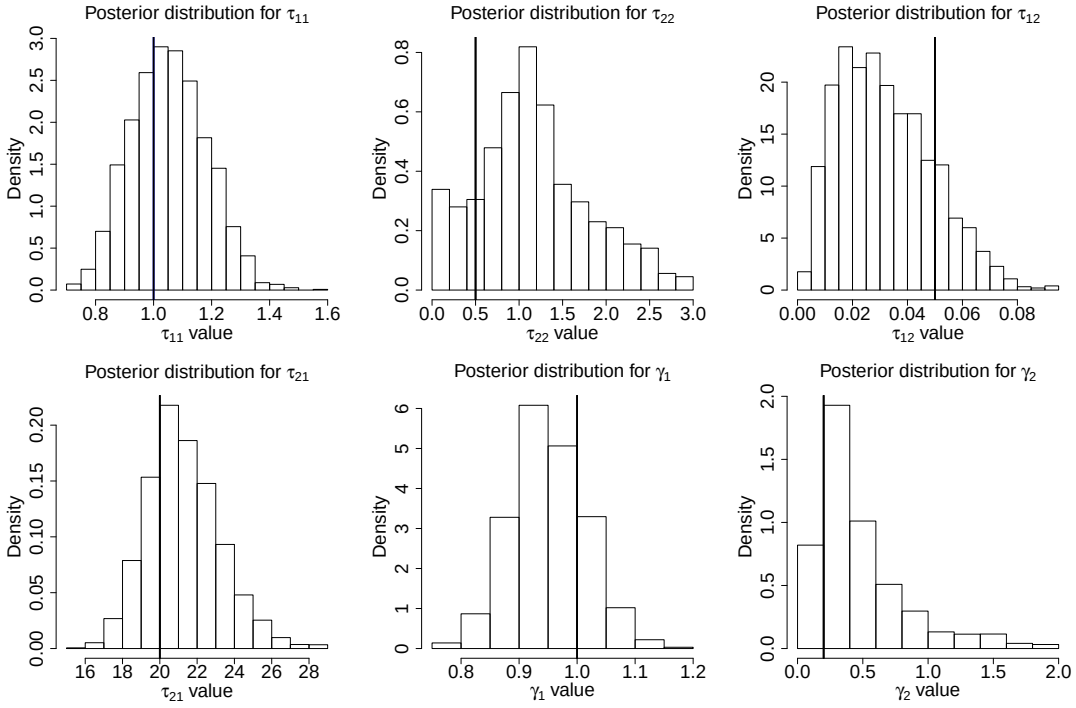


Figure 5.23: Plot showing the LV model results for the case where the hidden node is susceptible for half the simulation, and infected for the other. The parameters used to generate the case 2 simulation were $\tau_{11} = 1, \tau_{22} = 0.5, \tau_{12} = 0.05, \tau_{21} = 20, \gamma_1 = 1, \gamma_2 = 0.2$, and the simulation ran until $t = 20$. The LV model produces a posterior distribution for τ_{11} that contains the true value, as well as a non-zero τ_{12} and a distribution of τ_{21} that is centred on the true value.

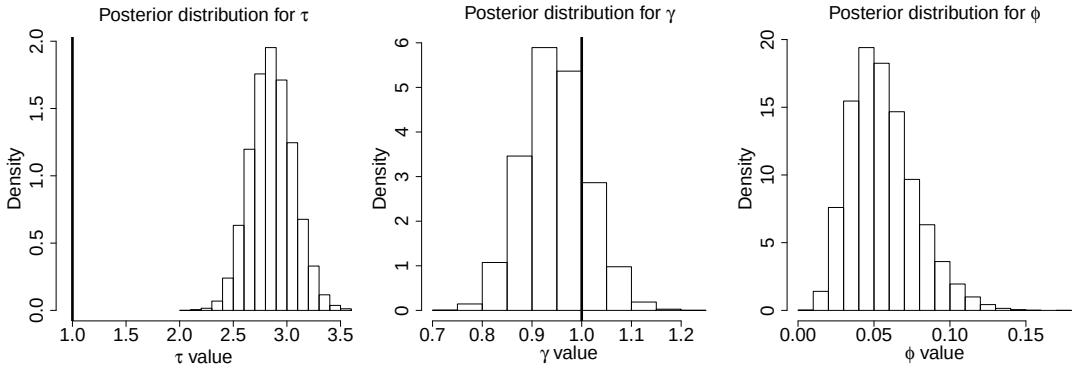


Figure 5.24: Plot showing the SISa model results for the same data as in 5.23, where the hidden node is susceptible for half the simulation, and infected for the other. Unlike the LV model, the posterior distribution for the τ parameter in the SISa model case does not include the true value of $\tau_{11} = 1$. The SISa model results also suggest a low value of ϕ, suggesting very few seemingly ambient infections.

We can also look at the same system with the parameters changed such that the hidden node is constantly changing, being infected and recovering many times throughout the epidemic. The results are shown for the LV model in Figure 5.25. Comparing these to the SISa results in Figure 5.26, we notice that both methods do poorly at recovering the true value for τ_{11}. In both cases they overestimate the value of τ_{11}. The latent variable method finds very low results for τ_{12}.

These results are summarised and discussed further in the final section 5.7.1 of this chapter.

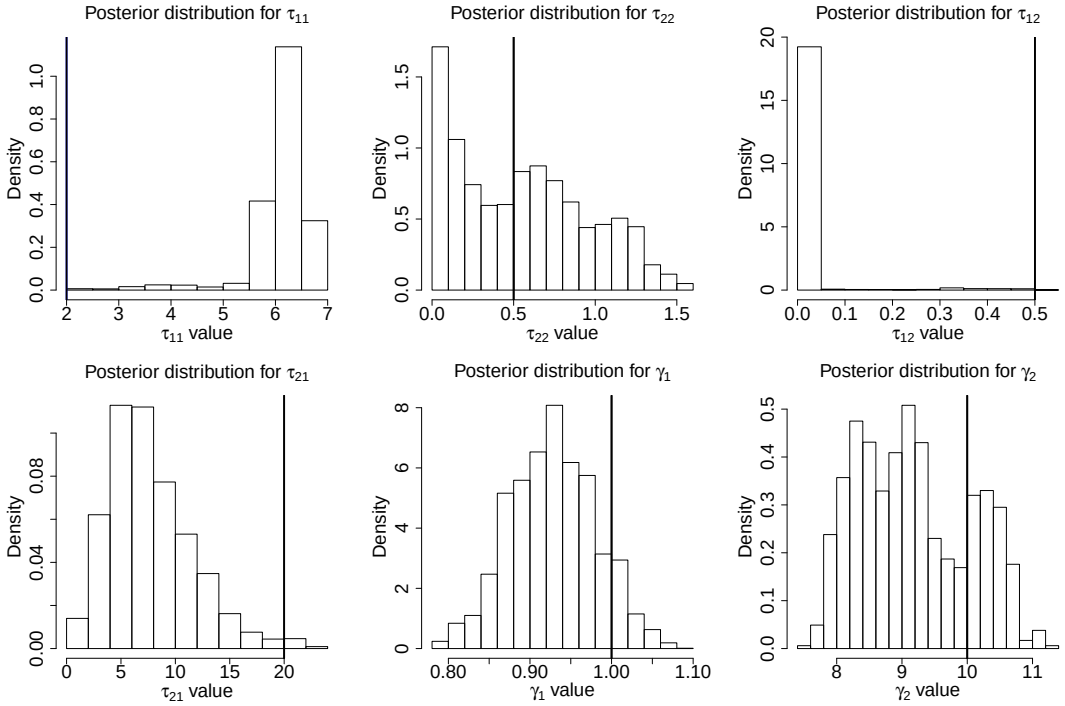


Figure 5.25: Plot showing the results of the LV model for the case where the hidden node is frequently alternating between susceptible and infected. The parameters used to generate the simulation in this case were $\tau_{11} = 2, \tau_{22} = 0.5, \tau_{12} = 0.5, \tau_{21} = 20, \gamma_1 = 1, \gamma_2 = 10$, and the simulation ran until $t = 20$. The LV model inference scheme seems to perform poorly here, as it does not manage to recover the true values for τ_{11}, τ_{12} or τ_{21}. The distribution of τ_{11} centring on a value so much larger than the true value $\tau_{11} = 2$ suggests that the effects of the hidden node layers are being attributed to an increased intralayer infection rate.

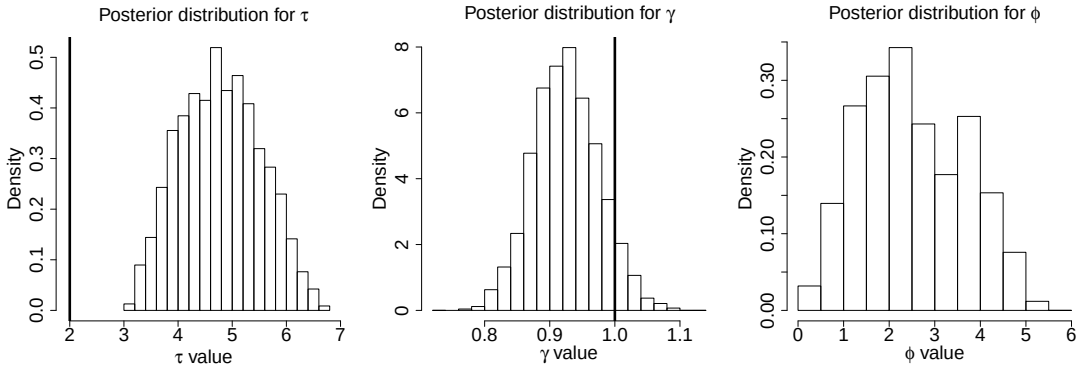


Figure 5.26: Plot showing the results of the SISa model, for the same data as in 5.25 where the hidden node is frequently alternating between susceptible and infected. As with the LV model, the SISa model fits a larger value of τ than the true value of $\tau_{11} = 2$. As the results are similar for the two models, there seems to be no convincing motivation as to why the LV model is more useful in this instance.

5.7 Discussion and future research directions

5.7.1 Summary

In this section, I have attempted to infer a two-layer SIS process where one of the layers is hidden. An initial exploration confirmed that there are certain phenomena unique to the multilayer SIS process, such as the induced epidemic where an epidemic is only sustained on one layer due to interlayer infections spreading from the other layer. These are situations where analysis must be performed carefully, so we do not simply apply naive inference assuming a single layer.

Due to the fact that the likelihood is intractable for the full two-layer SIS model, this inference problem called for an alternative approach. Data augmentation seemed infeasible: given that we are assuming no knowledge about the hidden layer network, inferring the structure of the hidden layer while also simulating a set of events on the hidden layer that are compatible with the observed events seemed unwieldy. Likewise, I anticipated that simulating a set of events us-

ing ABC that had summary statistics within a reasonably small tolerance would be challenging. As such, I attempted inference using two simpler approximate models instead: the SISa model and the LV model.

I found that the SISa model was able to recover a distribution for its parameter τ that contained the true value of the intralayer spreading on the visible layer (τ_{11}) for a number of regimes. The first regime contained systems that had no interlayer infections. Since these systems are essentially SISa dynamics with a zero value for the ambient parameter, the accurate results are expected. The second regime contains situations where the interlayer infections of the system frequently occur in an 'ambient' fashion, i.e. the interlayer infections seem random, unrelated to the spread of the nodes on the visible layer, and affect nodes on the visible layer that have no infected neighbours.

The LV model was similarly able to recover a distribution for its parameter τ that contained the true value of τ_{11} for these regimes. I did manage to find a single situation where the LV model recovered a distribution containing τ_{11} and the SISa model did not: the situation where the hidden layer is in fact a single node, which becomes infected halfway through the epidemic. This situation exploits the fact in the LV model, the effect of the hidden layer can change with time (as the latent node becomes infected and recovers), unlike in the SISa model where the ambient parameter (representing infections from the hidden layer) is static. However, it is unsurprising that the LV model does better at recovering the true value of τ_{11} in this situation, as the LV model very accurately represents this network structure. Arguably, given its increased complexity and computational time, there appears to be little benefit to using the LV model instead of the SISa model.

In cases where the interlayer infection events do not perform 'ambiently', both models generated distributions that do not contain the true value of τ_{11}. They also struggled with the situation where there was a single hidden node fluctuating quickly between susceptible and infected. These results have helped to highlight that in these suspected scenarios, gathering information about the hidden layer might be the only way to effectively perform inference.

All of these findings point to a key question: given that the two models perform so differently across the different regimes, how can we know *a priori* which regime we are in, and therefore what to expect from the inference results?

One situation at least is easy to identify: if there are nodes on the visible layer that are frequently being infected, even when they have no infected neighbours, then these infections are clearly due to the hidden layer, and we might hope that the interlayer infections can be modelled 'ambiently'. However, in all situations there will be a certain number of educated assumptions required in order to interpret the results, perhaps developed alongside expert consultation and careful consideration of the field, e.g. social networks are rarely structured like random networks.

5.7.2 Future research directions

Given the scope of the research question, there are numerous other directions for future study. To begin with, we could explore an entirely different approach to deal with the intractable likelihood of the full two-layer SIS dynamics. In this chapter, I decided to use some approximate models that had a tractable likelihood instead. This was due to the data augmentation and likelihood-free methods seeming unwieldy for the situation with an entirely hidden layer. Exploring these alternative approaches was beyond the scope of this thesis, but it would be interesting to see if they are feasible in any other related situations: for example, if we know the structure of the hidden network but cannot observe the hidden layer events.

Alternatively, we could explore extensions of the current approximate model approach, by formulating further models with increased complexity to try and fit the data. We briefly considered a 'latent mean-field model', where the hidden layer is approximated by a well-mixed population that is entirely connected to each node on the visible layer, using a simple homogeneous mean-field model. In essence, this is an SISa model where the ambient parameter can vary discretely between zero and some maximum throughout the simulation, rather than the binary on/off values that the LV model provides. However, the transition matrices involved in a system like this mean the likelihood function cannot be calculated as simply as in the latent variable case. The diagonalisation of Q_1 is complicated and I struggled to express B_1 symbolically to allow for integration.

A question I've touched on throughout this chapter is what we actually mean by a 'well-fitting' model in this situation. Finding a model that produces parameter values that are similar to the true parameter values might not necessarily suit the problem at hand. It might be that it is more useful to find parameter values that can be used to accurately predict future results, to model the initial spread accurately, or to achieve the correct steady state value. It would be interesting to see if there are many situations where the results for the SISa and LV model appear to diverge under these different measurements of 'goodness of fit'.

Related to this question is one of model selection. So far I have provided some descriptive examples of where the SISa and LV models work well and where they might run into pitfalls, but ideally one would perform some kind of model selection to quantify which model is the most suitable to use in a given situation. Any model selection method would require a clear choice of 'goodness of fit' measurement. Model selection can be based on various information criterion methods [136], which generally specify a penalty per parameter to discourage overfitting.

Another particularly pertinent question is whether there are multiple combinations of parameters that have similar likelihoods for the scenarios analysed. One way to probe this is to repeat the MCMC routine with a large number of different initial values for the parameters, and see whether this results in a large number of different solutions. I touched on this for the SISa inference model, where I initialised the MCMC routine with a handful of different values, and found good convergence to the solutions I have presented. However, this was more difficult for the LV inference model, since the limited number of iterations possible in the MCMC routine meant it struggled with convergence for initial parameter values far from the true values. A more rigorous investigation into this question might involve testing more initial values and running longer MCMC routines for the LV model.

It would also be helpful, if possible, to identify whether there are any basic multilayer properties that indicate when the different models are most appropriate, and what to expect from the results. Assuming we have a reasonable estimate of the hidden layer properties, this would make it easier to *a priori* select the most suitable model for our analysis, and draw considered conclusions. One possible

example of such a property might be the overlap of the network: we might expect situations where the layers have low overlap to have interlayer infections that are more easily identifiable as 'ambient' infections, leading to non-zero estimates for ϕ and more accurate estimations of τ_{11}. Another property to consider might be the average minimum path between pairs of nodes on the visible layer before and after taking the hidden layer into account. I would also like to explore whether there is a way to estimate the number of infected nodes that will become infected on the hidden layer before a hidden node infects a visible node. This will give some indication of how far the infection spreads on the hidden layer before it affects the visible layer. A rigorous relationship between these properties and the accuracy of the MCMC results is currently out of reach, but it might be possible to draw some general conclusions.

It is also worth noting that a lot of the work in this chapter assumes that we have some reasonable estimates of the hidden layer structure and behaviour, and working out how to make these reasonable estimates is an open question in itself.

Chapter 6

Conclusions

6.1 Thesis review and discussion

Ultimately, the conclusions of this thesis tie into the idea of uncertainty in network science, and how this changes our approach or limits what we can know. This chapter will provide a brief review of the work and explain its consequences in relation to this central theme. I will also discuss how this work might be used in industry applications, particularly those involving social media network analysis, motivated by my CASE PhD partnership with Jaywing Intelligence.

Chapter 2 focused on mean-field approximations of SIS dynamics. I introduced mean-field approximation methods ahead of future chapters, and investigated the accuracy of several different mean-field approximations. The results of this investigation confirmed that, while approximations such as the heterogeneous mean-field and the individual-based mean-field approach (NIMFA) are required to model SIS dynamics on networks with scale-free degree distributions, for networks such as regular networks and Erdős-Rényi networks homogeneous mean-field approximations are adequate. I demonstrated how systems such as the approximate master equations lead to greater accuracy, but come with an increased computational cost that sometimes renders them impractical. This highlights how different approximations of dynamics on networks can vary hugely in their accuracy and appropriateness based on the structure of the underlying network. If we know all the information about the network, we can make an informed decision as to which approximation to use. However, if we are looking

at a problem with an unknown network, such as those referenced in Chapter 4, choosing an appropriate approximation could pose a problem.

It is also true that, even if we can judge which approximation might be most accurate, the precise approximation error is often unknown *a priori*. The approximate lumping approach that I developed in Chapter 3 explores this problem. I showed how we can use approximate lumping to look at a population model for SIS dynamics, where the partitions correspond to the different levels of the system. I also showed how, by choosing an infinitesimal generator for the lumped system that minimises a quantity I refer to as the lumping error, we find that the elements of this generator are in fact similar to the transition rates of the homogeneous mean-field approximation closed at the single level. This provides a more rigorous mathematical motivation to the closure. I also expressed the error of the homogeneous mean-field approximation in terms of the lumped generator and related matrices. Unfortunately this error expression also relies on information about the full exact solution. This points to a possible limit in how precisely we can quantify the error of these approximations for most practical real-world situations.

Chapters 4 and 5 shifted focus to look at inference on networks. The two-layer system with a latent layer which is studied in Chapter 5 is an example of a situation where there is a huge amount of missing information. My initial investigation of simulating SIS dynamics on this system in Section 5.3 showed how a lack of awareness that the hidden layer existed could lead to confusion and erroneous conclusions regarding the dynamical parameters. This shows how important it is to thoroughly consider all the possible connections and network layers that could be at play in a network system before attempting analysis. It also shows how tenuous conclusions can be in situations where the possibility of further layers is unknown.

In Sections 5.5 and 5.6 I then attempted to infer the visible layer infection rate parameter by retaining and studying only the simulated events happening on the visible layer. I used two inference frameworks which both assumed an awareness that the hidden layer existed. The first was a simple SISa model, which effectively treated the hidden layer and any cross-layer infections as a constant background infection rate. The other was a more complex 'latent variable' model that I

developed, where I treated the hidden layer as a single node, attached to each of the nodes on the visible layer. I demonstrated how, for situations where the effect of the cross-layer infections from nodes on the hidden layer looked similar to a constant background rate, the SISa model was able to recover the true infection rate. I also showed how the latent variable model performed very similarly to the SISa model in most situations, but could outperform it in at least one situation that I managed to identify. On a practical level, this work shows that the SISa model is often a more appropriate choice for this inference problem, since it is computationally less expensive and often just as accurate as the latent variable model. On a more general level, this work highlights how, for certain structures and dynamical parameters, we can sometimes perform effective inference using models that are much simpler than the actual process. However, the fact that these models were accurate only in fairly specific situations shows how sensitive inference problems of this kind are. It is clear that sometimes we have to make large assumptions about the unknown behaviour on the hidden layer in order to even practically attempt inference. We have to be careful, and be aware that these assumptions, and therefore the inferred parameters, may not be accurate. If we suspect that the hidden layer contributes to the visible layer infection patterns in a more complex way, effective inference may prove to be out of reach.

To sum up, both approximating the dynamics and the presence of missing information about the network structure leads to uncertainty. It is vital to understand and quantify that uncertainty as far as possible, in order to know when specific approximations or models may be useful. One current example which shows the pressing need for further research into the technical and mathematical foundations of this area is the range of results from different forecasts of the COVID-19 epidemic. Different forecasts from different modelling teams show how sensitive such predictions are to different models and assumptions regarding homogeneity, even when using simple compartmental models without detailed network structure [137]. The results from these forecasts are particularly pertinent, since the UK's Scientific Advisory Group for Emergencies [138] lists stochastic transmission models as modelling inputs.

6.2 Industry applications and open questions

Social network analysis is used in advertising to study the spread of information and identify influential individuals or groups of people [139]. This information can then be used to more directly target products to certain people and to help monitor and influence things like brand perception and product awareness. In this section I am going to describe a few connections the work in this thesis might have with social networks, information spreading, and online data. I will also point to some difficulties and open questions in this area.

First, the work in Chapter 3 can be applied to SVT models used in social contagion, to better study the spread of information and ideas between people. For example, the Maki-Thompson (MT) rumour model can be formulated as an SVT model where the recovery rate, as well as the infection rate, depends linearly on the number of infected nodes [140]. This means that we could use the methods in Chapter 3 to perform approximate lumping and create a population model of the process, with the error of this model theoretically calculable. Mean-field approximations for the MT model have been studied [48], but ultimately this model does not benefit from the same wealth of research into these approximations and their accuracy that SIS dynamics does. Being able to analyse the error of a coarse-grained model of the dynamics using the methods from Chapter 3 is therefore a significant contribution.

There are also several questions around data collection and data quality when trying to measure information spreading online, which connect to the discussion in Chapter 4 on missing information. One problem is how we measure whether people are 'infected' with a rumour (actively spreading it) or not infected, i.e. 'susceptible' (not currently spreading the rumour). It is perhaps easier to see through an individual's posts when they become aware of a subject matter, brand or product. We might track which users are using certain keywords or phrases in their posts to find 'infected' individuals [141], or observe conversations and message exchanges between different users to observe spread [142]. This can show clear current interest, but how do we judge that an individual is no longer interested or aware of something? And how do we account for users who are more bystanders than active participants, and may be following a discussion closely

but without contributing in any noticeable way themselves? This links into the questions of who to include in the network, and how to sample the network in question. As explained in Chapter 4, if we sample the accounts connected to a given user using a non-probability sampling method, we lose the ability to use certain inference tools. However, it may only be useful or practical to sample in this way. As Lermand and Ghosh note [143], a large number of studies of social networks involve following a flow of information or connections between users rather than sampling a network and then observing the dynamics. For example, a common way of sampling data from Twitter is by forming the Twitter follower graph through identifying active users who are discussing the relevant content, and then sampling their followers [144, 145]. In this situation a directed edge exists between two users if one follows the other. Whether the edge should be directed towards the user that is being followed, or towards the follower, is dependent on the context of the problem. We might also trace a hashtag or phrase through a social network [141], or sample users who 'retweet' each other (share each others' posts) [146].

The work in Chapter 5 is relevant, as multilayer networks are rife in social structures [147]. There are a multitude of different social media websites that an individual can be active on, including websites such as Facebook, Twitter, Instagram, TikTok and Snapchat. People will often interact with the same people across different social media websites. The specific two-layer framework with one hidden layer that is studied in Chapter 5 might occur within a social network if we have access to data from a collection of users interacting on an online network, but no information about their offline connections. Likewise, we might have access to data from one social media website, but not from another. If we find that the connections on the hidden layer can be modelled effectively by using an ambient background infection rate as part of an SISa model as in Section 5.5, or by using the LV model described in Section 5.6, then we do not need to worry about collecting any further information relating to this extra behaviour. If we can avoid further data collection, we can save both time and money. It is also worth noting here that the work in Chapter 5 involved exclusively undirected networks. This is helpful for certain social connections, such as a mutual friendship connection on Facebook or a mutual co-worker relationship in real life, but is not as precisely

applicable to situations such as the directed Twitter follower network. I have at least allowed some flexibility by defining τ_{12} separate to τ_{21}, which would allow different spreading rates on edges directed from layer 1 to layer 2 and edges directed from layer 2 to layer 1. Accounting for directed edges would require changing the Gillespie algorithm to simulate the dynamics, and the inference schemes would need to take into account the directed edges when calculating the different rates of each kind of event. Extending this work to directed edges seems possible but was ultimately beyond the scope of this thesis.

A lot of the work in Chapter 5 also presumes that we have some general idea about the structure of the networks on each layer. In the context of social interactions, we can use information about common social network structures to make some assumptions about the case study in this context too: for example, social networks are rarely structured like random Erdős-Rényi networks, and are much more likely to feature degree distributions that display a heavy-tailed distribution [32]. More detailed observations are available through in-depth studies on specific social media websites, such as those looking at the structure of Twitter [148] and Facebook [149]. We might also be able to judge ahead of time whether the two layers are likely to have a large overlap (for example, someone's contacts on Snapchat and Facebook, where people tend to connect with their informal peers), versus low overlap (for example, someone's TikTok and LinkedIn contacts, where the former involves sharing mostly informal content and the latter centres on professional connections). This highlights the importance of studies looking at the structure of these kinds of networks.

These are just some of the possible applications of this work, in an ever-growing area of research. I hope that this work can help researchers to make careful approximations and inference, as well as prompting further developments and discussions around the idea of missing information in social network analysis.

Appendix A

Appendix 1 - Design-based inference matrix is ill-conditioned

The condition number $\kappa(A)$ of a matrix A is defined as

$$\kappa(A) = \|A\|\|A^{-1}\|,$$

where $\|A\|$ is the norm of matrix A [150] . If the condition number of a matrix is large, then the matrix is said to be ill-conditioned, and the solution of the linear system of equations represented by the matrix is prone to large numerical errors.

We are interested in the condition number of the matrix A involved in the naive estimate of the population degree distribution, given the sampled subgraph degree distribution, from section 4.1.1. This matrix has elements

$$A_{ij} = \binom{i}{j} \alpha^j (1-\alpha)^{i-j}.$$

We define the norm as

$$\|A\|_\infty = \max_i \sum_j^D |a_{ij}|.$$

For a matrix B,

$$\|B\|_\infty = \max_i \sum_j^D |b_{ij}| \geq \max_i |b_{ii}|.$$

Now let's set $B = A^{-1}$. Since A is an upper triangular matrix, B is also an upper triangular matrix with diagonal elements $b_{ii} = \frac{1}{a_{ii}}$. Therefore

$$\max_i |b_{ii}| = \max_i \left| \frac{1}{a_{ii}} \right| = \frac{1}{\min_i |a_{ii}|}.$$

So with

$$\|A\|_\infty = \max_i \sum_j^D |a_{ij}| \geq \max_i |a_{ii}|,$$

and

$$\|A^{-1}\|_\infty = \|B\|_\infty \geq \frac{1}{\min_i |a_{ii}|},$$

the norm becomes bounded:

$$\kappa(A) = \|A\| \|A^{-1}\| \geq \frac{\max_i |a_{ii}|}{\min_i |a_{ii}|}.$$

Since the value of $\max_i |a_{ii}| = 1$ and the value of $\min_i |a_{ii}| = \alpha^D$, where D is the maximum degree of the network:

$$\kappa(A) \geq \frac{1}{\alpha^D}.$$

This means that as α is decreased, and D increased, the matrix A becomes more ill-conditioned.

References

[1] E. Ravasz, A. L. Somera, D. A. Mongru, Z. N. Oltvai, and A.-L. Barabási, "Hierarchical organization of modularity in metabolic networks," *Science*, vol. 297, no. 5586, pp. 1551–1555, 2002. vi, 7

[2] D. Hao and C. Li, "The dichotomy in degree correlation of biological networks," *PloS One*, vol. 6, no. 12, p. e28322, 2011. vi, 8

[3] C.-Y. Huang, C.-T. Sun, and H.-C. Lin, "Influence of local information on social simulations in small-world network models," *Journal of Artificial Societies and Social Simulation*, vol. 8, no. 4, 2005. vi, 11

[4] M. E. Newman, "Power laws, Pareto distributions and Zipf's law," *Contemporary Physics*, vol. 46, no. 5, pp. 323–351, 2005. vi, 12, 13

[5] D. J. Watts and S. H. Strogatz, "Collective dynamics of small-world networks," *Nature*, vol. 393, no. 6684, p. 440, 1998. vii, 14, 15

[6] G. Robins, P. Pattison, Y. Kalish, and D. Lusher, "An introduction to exponential random graph (p*) models for social networks," *Social Networks*, vol. 29, no. 2, pp. 173–191, 2007. vii, 17

[7] B. D. Fath, U. M. Scharler, R. E. Ulanowicz, and B. Hannon, "Ecological network analysis: network construction," *Ecological Modelling*, vol. 208, no. 1, pp. 49–55, 2007. 1

[8] E. D. Levy and J. B. Pereira-Leal, "Evolution and dynamics of protein interactions and networks," *Current Opinion in Structural Biology*, vol. 18, no. 3, pp. 349–357, 2008. 1

[9] J. D. Westaby, D. L. Pfaff, and N. Redding, "Psychology and social networks: A dynamic network theory perspective.," *American Psychologist*, vol. 69, no. 3, p. 269, 2014. 1

[10] S. H. M. Lee, J. Cotte, and T. J. Noseworthy, "The role of network centrality in the flow of consumer influence," *Journal of Consumer Psychology*, vol. 20, no. 1, pp. 66–77, 2010. 1

[11] P. Grindrod, D. J. Higham, P. Laflin, A. Otley, and J. A. Ward, "Inverse network sampling to explore online brand allegiance," *European Journal of Applied Mathematics*, 2016. 1

[12] A.-L. Barabási, N. Gulbahce, and J. Loscalzo, "Network medicine: a network-based approach to human disease," *Nature Reviews Genetics*, vol. 12, no. 1, pp. 56–68, 2011. 1

[13] J. Wang, H. Mo, F. Wang, and F. Jin, "Exploring the network structure and nodal centrality of China's air transport network: A complex network approach," *Journal of Transport Geography*, vol. 19, no. 4, pp. 712–721, 2011. 1

[14] G. A. Pagani and M. Aiello, "The power grid as a complex network: a survey," *Physica A: Statistical Mechanics and its Applications*, vol. 392, no. 11, pp. 2688–2700, 2013. 1

[15] R. Pastor-Satorras, C. Castellano, P. Van Mieghem, and A. Vespignani, "Epidemic processes in complex networks," *Reviews of Modern Physics*, vol. 87, no. 3, p. 925, 2015. 1, 22

[16] V. Sood and S. Redner, "Voter model on heterogeneous graphs," *Physical Review Letters*, vol. 94, no. 17, p. 178701, 2005. 1

[17] J. Goldenberg, B. Libai, and E. Muller, "Talk of the network: A complex systems look at the underlying process of word-of-mouth," *Marketing Letters*, vol. 12, no. 3, pp. 211–223, 2001. 1

[18] J. P. Gleeson, J. A. Ward, K. P. O'sullivan, and W. T. Lee, "Competition-induced criticality in a model of meme popularity," *Physical Review Letters*, vol. 112, no. 4, p. 048701, 2014. 1

[19] A. Mellor, M. Mobilia, S. Redner, A. M. Rucklidge, and J. A. Ward, "Influence of Luddism on innovation diffusion," *Physical Review E*, vol. 92, no. 1, p. 012806, 2015. 1

[20] T. C. Schelling, "Dynamic models of segregation," *Journal of Mathematical Sociology*, vol. 1, no. 2, pp. 143–186, 1971. 1

[21] A. Baronchelli, M. Felici, V. Loreto, E. Caglioti, and L. Steels, "Sharp transition towards shared vocabularies in multi-agent systems," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2006, no. 06, p. P06014, 2006. 1

[22] M. Salathé and J. H. Jones, "Dynamics and control of diseases in networks with community structure," *PLoS Comput Biol*, vol. 6, no. 4, p. e1000736, 2010. 2

[23] M. Newman, *Networks: An Introduction*. Oxford University Press, Oxford, UK, 2010. 2, 18

[24] M. E. Newman, "The structure and function of complex networks," *SIAM review*, vol. 45, no. 2, pp. 167–256, 2003. 6

[25] M. E. Newman, "Assortative mixing in networks," *Physical Review Letters*, vol. 89, no. 20, p. 208701, 2002. 7

[26] J. Bollen, B. Gonçalves, G. Ruan, and H. Mao, "Happiness is assortative in online social networks," *Artificial Life*, vol. 17, no. 3, pp. 237–251, 2011. 8

[27] L. Zhang and W. Tu, "Six degrees of separation in online society," 2009. 8

[28] M. Šikić, A. Lančić, N. Antulov-Fantulin, and H. Štefančić, "Epidemic centrality – is there an underestimated epidemic impact of network peripheral

nodes?," *The European Physical Journal B*, vol. 86, no. 10, pp. 1–13, 2013. 10

[29] P. Laflin, A. V. Mantzaris, F. Ainley, A. Otley, P. Grindrod, and D. J. Higham, "Dynamic targeting in an online social medium," in *International Conference on Social Informatics*, pp. 82–95, Springer, 2012. 10

[30] G. Ferraz de Arruda, A. L. Barbieri, P. Martín Rodriguez, Y. Moreno, L. da Fontoura Costa, and F. Aparecido Rodrigues, "The role of centrality for the identification of influential spreaders in complex networks," *ArXiv e-prints*, pp. arXiv–1404, 2014. 10

[31] X. F. Wang and G. Chen, "Complex networks: small-world, scale-free and beyond," *IEEE Circuits and Systems Magazine*, vol. 3, no. 1, pp. 6–20, 2003. 12

[32] A.-L. Barabási and E. Bonabeau, "Scale-free networks," *Scientific American*, vol. 288, no. 5, pp. 60–69, 2003. 12, 168

[33] A. D. Broido and A. Clauset, "Scale-free networks are rare," *Nature Communications*, vol. 10, no. 1, pp. 1–10, 2019. 12

[34] A.-L. Barabási and R. Albert, "Emergence of scaling in random networks," *Science*, vol. 286, no. 5439, pp. 509–512, 1999. 13

[35] R. Toivonen, J.-P. Onnela, J. Saramäki, J. Hyvönen, and K. Kaski, "A model for social networks," *Physica A: Statistical Mechanics and its Applications*, vol. 371, no. 2, pp. 851–860, 2006. 14

[36] M. Shumate and E. T. Palazzolo, "Exponential random graph (p*) models as a method for social network analysis in communication research," *Communication Methods and Measures*, vol. 4, no. 4, pp. 341–371, 2010. 19

[37] R. Pastor-Satorras and A. Vespignani, "Epidemic dynamics and endemic states in complex networks," *Physical Review E*, vol. 63, no. 6, p. 066117, 2001. 21

[38] J. Gomez-Gardenes, V. Latora, Y. Moreno, and E. Profumo, "Spreading of sexually transmitted diseases in heterosexual populations," *Proceedings of the National Academy of Sciences*, vol. 105, no. 5, pp. 1399–1404, 2008. 21

[39] H. W. Hethcote, "Three basic epidemiological models," in *Applied Mathematical Ecology*, pp. 119–144, Springer, New York, NY, 1989. 21, 30

[40] P. E. Lekone and B. F. Finkenstädt, "Statistical inference in a stochastic epidemic SEIR model with control intervention: Ebola as a case study," *Biometrics*, vol. 62, no. 4, pp. 1170–1177, 2006. 21, 96

[41] J. M. Read, J. R. Bridgen, D. A. Cummings, A. Ho, and C. P. Jewell, "Novel coronavirus 2019-ncov: early estimation of epidemiological parameters and epidemic predictions," *MedRxiv*, 2020. 21

[42] K. Kandhway and J. Kuri, "How to run a campaign: Optimal control of SIS and SIR information epidemics," *Applied Mathematics and Computation*, vol. 231, pp. 79–92, 2014. 22

[43] D. J. Daley and D. G. Kendall, "Epidemics and rumours," *Nature*, vol. 204, no. 4963, p. 1118, 1964. 22

[44] J. Gani, "The Maki–Thompson rumour model: a detailed analysis," *Environmental Modelling & Software*, vol. 15, no. 8, pp. 721–725, 2000. 22

[45] F. M. Bass, "A new product growth for model consumer durables," *Management Science*, vol. 15, no. 5, pp. 215–227, 1969. 23

[46] M. Granovetter, "Threshold models of collective behavior," *American Journal of Sociology*, vol. 83, no. 6, pp. 1420–1443, 1978. 23

[47] T. W. Valente, "Social network thresholds in the diffusion of innovations," *Social Networks*, vol. 18, no. 1, pp. 69–89, 1996. 23

[48] M. Nekovee, Y. Moreno, G. Bianconi, and M. Marsili, "Theory of rumour spreading in complex social networks," *Physica A: Statistical Mechanics and its Applications*, vol. 374, no. 1, pp. 457–470, 2007. 23, 166

[49] W. H. Kaempfer and A. D. Lowenberg, "A threshold model of electoral policy and voter turnout," *Rationality and Society*, vol. 5, no. 1, pp. 107–126, 1993. 23

[50] R. Pastor-Satorras and A. Vespignani, "Epidemic spreading in scale-free networks," *Physical Review Letters*, vol. 86, no. 14, p. 3200, 2001. 23, 52

[51] J. P. Gleeson, S. Melnik, J. A. Ward, M. A. Porter, and P. J. Mucha, "Accuracy of mean-field theory for dynamics on real-world networks," *Physical Review E*, vol. 85, no. 2, p. 026106, 2012. 23, 67

[52] M. E. Newman, "Spread of epidemic disease on networks," *Physical Review E*, vol. 66, no. 1, p. 016128, 2002. 23

[53] D. Zhao, L. Wang, S. Li, Z. Wang, L. Wang, and B. Gao, "Immunization of epidemics in multiplex networks," *PloS One*, vol. 9, no. 11, p. e112018, 2014. 24

[54] L. Meyers, "Contact network epidemiology: Bond percolation applied to infectious disease prediction and control," *Bulletin of the American Mathematical Society*, vol. 44, no. 1, pp. 63–86, 2007. 24

[55] G. Bianconi and P. L. Krapivsky, "Epidemics with containment measures," *Physical Review E*, vol. 102, no. 3, p. 032305, 2020. 24

[56] Y. Wang, D. Chakrabarti, C. Wang, and C. Faloutsos, "Epidemic spreading in real networks: An eigenvalue viewpoint," in *22nd International Symposium on Reliable Distributed Systems, 2003. Proceedings.*, pp. 25–34, IEEE, 2003. 24

[57] A. V. Goltsev, S. N. Dorogovtsev, J. G. Oliveira, and J. F. Mendes, "Localization and spreading of diseases in complex networks," *Physical Review Letters*, vol. 109, no. 12, p. 128702, 2012. 24

[58] A. L. Hill, D. G. Rand, M. A. Nowak, and N. A. Christakis, "Emotions as infectious diseases in a large social network: the SISa model," *Proceedings of the Royal Society B: Biological Sciences*, vol. 277, no. 1701, pp. 3827–3835, 2010. 28

[59] I. Z. Kiss, J. C. Miller, P. L. Simon, *et al.*, "Mathematics of epidemics on networks," *Cham: Springer*, vol. 598, 2017. 31, 32, 43, 46, 54, 69, 70, 71

[60] J. A. Ward and M. López-García, "Exact analysis of summary statistics for continuous-time discrete-state markov processes on networks using graph-automorphism lumping," *Applied Network Science*, vol. 4, no. 1, pp. 1–28, 2019. 32

[61] D. T. Gillespie, "Exact stochastic simulation of coupled chemical reactions," *The Journal of Physical Chemistry*, vol. 81, no. 25, pp. 2340–2361, 1977. 34

[62] C. V. Rao and A. P. Arkin, "Stochastic chemical kinetics and the quasi-steady-state assumption: Application to the Gillespie algorithm," *The Journal of Chemical Physics*, vol. 118, no. 11, pp. 4999–5010, 2003. 34

[63] P. L. Simon, M. Taylor, and I. Z. Kiss, "Exact epidemic models on graphs using graph-automorphism driven lumping," *Journal of Mathematical Biology*, vol. 62, no. 4, pp. 479–508, 2011. 39

[64] M. Taylor, P. L. Simon, D. M. Green, T. House, and I. Z. Kiss, "From Markovian to pairwise epidemic models and the performance of moment closure approximations," *Journal of Mathematical Biology*, vol. 64, no. 6, pp. 1021–1042, 2012. 42

[65] C. Kuehn, "Moment closure – a brief review," *Control of Self-organizing Nonlinear Systems*, pp. 253–271, 2016. 43

[66] D. Smilkov, C. A. Hidalgo, and L. Kocarev, "Beyond network structure: How heterogeneous susceptibility modulates the spread of epidemics," *Scientific Reports*, vol. 4, no. 1, pp. 1–7, 2014. 44

[67] P. Van Mieghem, "The N-intertwined SIS epidemic network model," *Computing*, vol. 93, no. 2-4, pp. 147–169, 2011. 49

[68] E. Cator and P. Van Mieghem, "Nodal infection in Markovian susceptible-infected-susceptible and susceptible-infected-removed epidemics on networks are non-negatively correlated," *Physical Review E*, vol. 89, no. 5, p. 052802, 2014. 50

[69] J. P. Gleeson, "High-accuracy approximation of binary-state dynamics on networks," *Physical Review Letters*, vol. 107, no. 6, p. 068701, 2011. 50

[70] L. Pellis, F. Ball, S. Bansal, K. Eames, T. House, V. Isham, and P. Trapman, "Eight challenges for network epidemic models," *Epidemics*, vol. 10, pp. 58–62, 2015. 67

[71] J. A. Ward, A. Tapper, P. L. Simon, and R. P. Mann, "Micro-scale foundation with error quantification for the approximation of dynamics on networks." Submitted. 68, 94

[72] M. N. Jacobi and O. Goernerup, "A dual eigenvector condition for strong lumpability of Markov chains," *arXiv preprint arXiv:0710.1986*, 2007. 69

[73] P. Buchholz, "Exact and ordinary lumpability in finite Markov chains," *Journal of Applied Probability*, vol. 31, no. 1, pp. 59–75, 1994. 69

[74] P. L. Simon and I. Z. Kiss, "From exact stochastic to mean-field ODE models: a new approach to prove convergence results," *The IMA Journal of Applied Mathematics*, vol. 78, no. 5, pp. 945–964, 2013. 72

[75] J. Hale, *Ordinary differential equations*. Mineola, N.Y: Dover Publications, 2009. 75

[76] S. Gersgorin, "Uber die abgrenzung der eigenwerte einer matrix," *Bulletin del'Academie des Sciences de l'URSS. Classe des Sciences Mathematiques et na*, vol. 6, pp. 749–754, 1931. 76

[77] M. Kijima, "Continuous-time Markov chains," in *Markov processes for stochastic modeling*, pp. 167–241, Springer, New York, NY, 1997. 86

[78] J. P. Keener, "The Perron–Frobenius theorem and the ranking of football teams," *SIAM Review*, vol. 35, no. 1, pp. 80–93, 1993. 86

[79] R. J. Glauber, "Time-dependent statistics of the Ising model," *Journal of Mathematical Physics*, vol. 4, no. 2, pp. 294–307, 1963. 92

[80] C. Castellano, M. A. Muñoz, and R. Pastor-Satorras, "Nonlinear q-voter model," *Physical Review E*, vol. 80, no. 4, p. 041129, 2009. 92

[81] D. J. Watts, "A simple model of global cascades on random networks," in *The Structure and Dynamics of Networks*, pp. 497–502, Princeton University Press, 2011. 92

[82] N.-E. Fahssi, "Some identities involving polynomial coefficients," *arXiv preprint arXiv:1507.07968*, 2015. 93

[83] J. Fintzi, X. Cui, J. Wakefield, and V. N. Minin, "Efficient data augmentation for fitting stochastic epidemic models to prevalence data," *Journal of Computational and Graphical Statistics*, vol. 26, no. 4, pp. 918–929, 2017. 96

[84] E. Costenbader and T. W. Valente, "The stability of centrality measures when networks are sampled," *Social Networks*, vol. 25, no. 4, pp. 283–307, 2003. 97

[85] S. H. Lee, P.-J. Kim, and H. Jeong, "Statistical properties of sampled networks," *Physical Review E*, vol. 73, no. 1, p. 016102, 2006. 97

[86] J. H. Koskinen, G. L. Robins, P. Wang, and P. E. Pattison, "Bayesian analysis for partially observed network data, missing ties, attributes and actors," *Social Networks*, vol. 35, no. 4, pp. 514–527, 2013. 97, 103

[87] M. S. Handcock and K. J. Gile, "Modeling social networks from sampled data," *The Annals of Applied Statistics*, vol. 4, no. 1, p. 5, 2010. 97, 99, 100

[88] Y. Zhang, E. D. Kolaczyk, B. D. Spencer, *et al.*, "Estimating network degree distributions under sampling: An inverse problem, with applications to monitoring social media networks," *The Annals of Applied Statistics*, vol. 9, no. 1, pp. 166–199, 2015. 97

[89] D. D. Heckathorn, "Respondent-driven sampling II: deriving valid population estimates from chain-referral samples of hidden populations," *Social Problems*, vol. 49, no. 1, pp. 11–34, 2002. 99

[90] S. Brooks, "Markov chain Monte Carlo method and its application," *Journal of the Royal Statistical Society: Series D (The Statistician)*, vol. 47, no. 1, pp. 69–100, 1998. 100

[91] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, "Equation of state calculations by fast computing machines," *The Journal of Chemical Physics*, vol. 21, no. 6, pp. 1087–1092, 1953. 100

[92] W. K. Hastings, "Monte Carlo sampling methods using Markov chains and their applications," *Biometrika*, vol. 57, no. 1, pp. 97–109, 1970. 100

[93] S. Chib and E. Greenberg, "Understanding the Metropolis-Hastings algorithm," *The American Statistician*, vol. 49, no. 4, pp. 327–335, 1995. 100

[94] C. Geyer, "Introduction to Markov chain Monte Carlo," *Handbook of Markov chain Monte Carlo*, vol. 20116022, p. 45, 2011. 101

[95] D. R. Hunter, M. S. Handcock, C. T. Butts, S. M. Goodreau, and M. Morris, "ergm: A package to fit, simulate and diagnose exponential-family models for networks," *Journal of Statistical Software*, vol. 24, no. 3, p. nihpa54860, 2008. 102

[96] T. A. Snijders, "Markov chain Monte Carlo estimation of exponential random graph models," *Journal of Social Structure*, vol. 3, no. 2, pp. 1–40, 2002. 102

[97] T. A. Snijders, P. E. Pattison, G. L. Robins, and M. S. Handcock, "New specifications for exponential random graph models," *Sociological Methodology*, vol. 36, no. 1, pp. 99–153, 2006. 102

[98] S. J. Cranmer and B. A. Desmarais, "Inferential network analysis with exponential random graph models," *Political Analysis*, vol. 19, no. 1, pp. 66–86, 2011. 102

[99] J. Koskinen, "Bayesian analysis of exponential random graphs-estimation of parameters and model selection," in *Technical Report*, Stockholm University, 2004. 102

[100] A. Caimo and N. Friel, "Bayesian inference for exponential random graph models," *Social Networks*, vol. 33, no. 1, pp. 41–55, 2011. 102

[101] I. Murray, Z. Ghahramani, and D. MacKay, "MCMC for doubly-intractable distributions," *arXiv preprint arXiv:1206.6848*, 2012. 102

[102] N. G. Becker and T. Britton, "Statistical studies of infectious disease incidence," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 61, no. 2, pp. 287–307, 1999. 103

[103] P. D. O'Neill, "Introduction and snapshot review: relating infectious disease transmission models to data," *Statistics in Medicine*, vol. 29, no. 20, pp. 2069–2077, 2010. 104

[104] N. G. Becker, "Analysis of data from a single epidemic," *Australian Journal of Statistics*, vol. 25, no. 2, pp. 191–197, 1983. 104

[105] T. Britton and N. G. Becker, "Estimating the immunity coverage required to prevent epidemics in a community of households," *Biostatistics*, vol. 1, no. 4, pp. 389–402, 2000. 104

[106] P. D. O'Neill, "A tutorial introduction to Bayesian inference for stochastic epidemic models using Markov chain Monte Carlo methods," *Mathematical Biosciences*, vol. 180, no. 1-2, pp. 103–114, 2002. 105

[107] P. D. O'Neill and G. O. Roberts, "Bayesian inference for partially observed stochastic epidemics," *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, vol. 162, no. 1, pp. 121–129, 1999. 105

[108] A. Hobolth and E. A. Stone, "Simulation from endpoint-conditioned, continuous-time Markov chains on a finite state space, with applications to molecular evolution," *The Annals of Applied Statistics*, vol. 3, no. 3, p. 1204, 2009. 105

[109] L. S. T. Ho, F. W. Crawford, M. A. Suchard, *et al.*, "Direct likelihood-based inference for discretely observed stochastic compartmental models of infectious disease," *The Annals of Applied Statistics*, vol. 12, no. 3, pp. 1993–2021, 2018. 106

[110] B. Choi and G. A. Rempala, "Inference for discretely observed stochastic kinetic networks with applications to epidemic modeling," *Biostatistics*, vol. 13, no. 1, pp. 153–165, 2012. 106

[111] T. Britton and P. D. O'Neill, "Bayesian inference for stochastic epidemics in populations with random social structure," *Scandinavian Journal of Statistics*, vol. 29, no. 3, pp. 375–390, 2002. 106

[112] M. G. Blum and V. C. Tran, "HIV with contact tracing: a case study in approximate Bayesian computation," *Biostatistics*, vol. 11, no. 4, pp. 644–660, 2010. 106

[113] K. Csilléry, M. G. Blum, O. E. Gaggiotti, and O. François, "Approximate Bayesian computation (ABC) in practice," *Trends in Ecology & Evolution*, vol. 25, no. 7, pp. 410–418, 2010. 107

[114] D. Prangle, "Summary statistics in approximate Bayesian computation," *arXiv preprint arXiv:1512.05633*, 2015. 107

[115] T. Kypraios, P. Neal, and D. Prangle, "A tutorial introduction to Bayesian inference for stochastic epidemic models using Approximate Bayesian Computation," *Mathematical Biosciences*, vol. 287, pp. 42–53, 2017. 107

[116] T. McKinley, A. R. Cook, and R. Deardon, "Inference in epidemic models without likelihoods," *The International Journal of Biostatistics*, vol. 5, no. 1, 2009. 107

[117] R. Dutta, A. Mira, and J.-P. Onnela, "Bayesian inference of spreading processes on networks," *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 474, no. 2215, p. 20180129, 2018. 107

[118] M. Szell, R. Lambiotte, and S. Thurner, "Multirelational organization of large-scale social networks in an online world," *Proceedings of the National Academy of Sciences*, vol. 107, no. 31, pp. 13636–13641, 2010. 110

[119] S. Funk, E. Gilad, C. Watkins, and V. A. Jansen, "The spread of awareness and its impact on epidemic outbreaks," *Proceedings of the National Academy of Sciences*, vol. 106, no. 16, pp. 6872–6877, 2009. 110, 113

[120] K.-M. Lee, B. Min, and K.-I. Goh, "Towards real-world complexity: an introduction to multiplex networks," *The European Physical Journal B*, vol. 88, no. 2, p. 48, 2015. 110

[121] Y. Moreno and M. Perc, "Focus on multilayer networks," *New Journal of Physics*, vol. 22, no. 1, p. 010201, 2019. 110

[122] M. De Domenico, A. Solé-Ribalta, E. Omodei, S. Gómez, and A. Arenas, "Ranking in interconnected multilayer networks reveals versatile nodes," *Nature Communications*, vol. 6, no. 1, pp. 1–6, 2015. 111

[123] G. F. de Arruda, E. Cozzo, T. P. Peixoto, F. A. Rodrigues, and Y. Moreno, "Disease localization in multilayer networks," *Physical Review X*, vol. 7, no. 1, p. 011014, 2017. 111

[124] M. De Domenico, V. Nicosia, A. Arenas, and V. Latora, "Structural reducibility of multilayer networks," *Nature Communications*, vol. 6, no. 1, pp. 1–9, 2015. 111

[125] R. J. Sánchez-García, E. Cozzo, and Y. Moreno, "Dimensionality reduction and spectral properties of multilayer networks," *Physical Review E*, vol. 89, no. 5, p. 052815, 2014. 111, 113

[126] M. Diakonova, V. Nicosia, V. Latora, and M. San Miguel, "Irreducibility of multilayer network dynamics: the case of the voter model," *New Journal of Physics*, vol. 18, no. 2, p. 023010, 2016. 111

[127] S. Boccaletti, G. Bianconi, R. Criado, C. I. Del Genio, J. Gómez-Gardenes, M. Romance, I. Sendina-Nadal, Z. Wang, and M. Zanin, "The structure and dynamics of multilayer networks," *Physics Reports*, vol. 544, no. 1, pp. 1–122, 2014. 112

[128] M. De Domenico, A. Solé-Ribalta, E. Cozzo, M. Kivelä, Y. Moreno, M. A. Porter, S. Gómez, and A. Arenas, "Mathematical formulation of multilayer networks," *Physical Review X*, vol. 3, no. 4, p. 041022, 2013. 113

[129] M. De Domenico, C. Granell, M. A. Porter, and A. Arenas, "The physics of spreading processes in multilayer networks," *Nature Physics*, vol. 12, no. 10, pp. 901–906, 2016. 113

[130] E. Cozzo, G. F. de Arruda, F. A. Rodrigues, and Y. Moreno, "Multilayer networks: metrics and spectral properties," in *Interconnected Networks*, pp. 17–35, Springer, New York, NY, 2016. 114

[131] F. Battiston, V. Nicosia, and V. Latora, "Structural measures for multiplex networks," *Physical Review E*, vol. 89, no. 3, p. 032804, 2014. 114

[132] M. Kivelä, A. Arenas, M. Barthelemy, J. P. Gleeson, Y. Moreno, and M. A. Porter, "Multilayer networks," *Journal of Complex Networks*, vol. 2, no. 3, pp. 203–271, 2014. 114

[133] G. Bianconi, "Statistical mechanics of multiplex networks: Entropy and overlap," *Physical Review E*, vol. 87, no. 6, p. 062806, 2013. 114

[134] B. Min, S.-H. Gwak, N. Lee, and K.-I. Goh, "Layer-switching cost and optimality in information spreading on multiplex networks," *Scientific Reports*, vol. 6, p. 21392, 2016. 115

[135] G. Last and M. Penrose, *Lectures on the Poisson process*, vol. 7. Cambridge University Press, Cambridge, UK, 2017. 137

[136] H. Bozdogan, "Model selection and Akaike's information criterion (AIC): The general theory and its analytical extensions," *Psychometrika*, vol. 52, no. 3, pp. 345–370, 1987. 161

[137] J. P. Ioannidis, S. Cripps, and M. A. Tanner, "Forecasting for COVID-19 has failed," *International Journal of Forecasting*, 2020. 165

[138] Scientific Advisory Group for Emergencies, "Scientific evidence supporting the government response to coronavirus (COVID-19)." `https://www.gov.uk/government/collections/scientific-evidence-supporting-the-government-response-to-coronavirus-covid-19` (Accessed August 2021). 165

[139] P. Laflin, A. V. Mantzaris, F. Ainley, A. Otley, P. Grindrod, and D. J. Higham, "Discovering and validating influence in a dynamic online social network," *Social Network Analysis and Mining*, vol. 3, no. 4, pp. 1311–1323, 2013. 166

[140] D. P. Maki and M. Thompson, "Mathematical models and applications: with emphasis on the social life, and management sciences," tech. rep., 1973. 166

[141] Y. Wang and B. Zheng, "On macro and micro exploration of hashtag diffusion in twitter," in *2014 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2014)*, pp. 285–288, IEEE, 2014. 166, 167

[142] R. Kumar, M. Mahdian, and M. McGlohon, "Dynamics of conversations," in *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 553–562, 2010. 166

[143] K. Lerman and R. Ghosh, "Information contagion: An empirical study of the spread of news on digg and twitter social networks," in *Fourth international AAAI conference on weblogs and social media*, 2010. 167

[144] K. Lerman, R. Ghosh, and T. Surachawala, "Social contagion: An empirical study of information spread on digg and twitter follower graphs," *arXiv preprint arXiv:1202.3162*, 2012. 167

[145] E. Bakshy, J. M. Hofman, W. A. Mason, and D. J. Watts, "Everyone's an influencer: quantifying influence on twitter," in *Proceedings of the fourth ACM international conference on Web search and data mining*, pp. 65–74, 2011. 167

[146] M. Ten Thij, T. Ouboter, D. Worm, N. Litvak, H. van den Berg, and S. Bhulai, "Modelling of trends in twitter using retweet graph dynamics," in *International Workshop on Algorithms and Models for the Web-Graph*, pp. 132–147, Springer, 2014. 167

[147] M. E. Dickison, M. Magnani, and L. Rossi, *Multilayer social networks*. Cambridge University Press, 2016. 167

[148] H. Kwak, C. Lee, H. Park, and S. Moon, "What is twitter, a social network or a news media?," in *Proceedings of the 19th international conference on World wide web*, pp. 591–600, 2010. 168

[149] J. Ugander, B. Karrer, L. Backstrom, and C. Marlow, "The anatomy of the Facebook social graph," *arXiv preprint arXiv:1111.4503*, 2011. 168

[150] D. A. Belsley, E. Kuh, and R. E. Welsch, *Regression diagnostics: Identifying influential data and sources of collinearity*, vol. 571. John Wiley & Sons, Hoboken, NJ, 2005. 169