Stochastic Epidemics Conditioned On Their Final Outcome

Simon Richard White, MMath (Hons)

Thesis submitted to the University of Nottingham for the degree of Doctor of Philosophy

October 2009

Abstract

This thesis investigates the representation of a stochastic epidemic process as a directed random graph; we use this representation to impute the missing information in final size data to make Bayesian statistical inference about the model parameters using Markov Chain Monte Carlo (MCMC) techniques.

The directed random graph representation is analysed, in particular its behaviour under the condition that the epidemic has a given final size. This is used to construct efficient updates for MCMC algorithms.

The MCMC method is extended to include two-level mixing models and two-type models, with a general framework given for an arbitrary number of levels and types. Partially observed epidemics, that is, where the number of susceptibles is unknown or where only a subset of the population is observed, are analysed. The method is applied to several well known data sets and comparisons are made with previous results.

Finally, the method is applied to data of an outbreak of Equine Influenza (H3N8) at Newmarket in 2003, with a comparison to another analysis of the same data. Practical issues of implementing the method are discussed and are overcome using parallel computing (GNU OpenMP) and arbitrary precision arithmetic (GNU MPFR).

Dedicated to Mr. and Mrs. Spoon

keep reaching for the moon

Acknowledgements

The first thank you is to Phil, for being such a supportive and patient supervisor, for his many helpful comments and guidance over the years.

All my friends who have been there to keep me sane and on (or at least near) the path. To the Maths PhDs and Postdocs, past and present, who gave me their friendship and wisdom: Chris, Dave, Fraser, Irina, Kelly, Kim, Simon, Simon, Theo and Tom. For all the Wednesday afternoons and evenings: Dave, Gareth and Rich. To Graham and Andy for all the trips home to London and everyone there. To all my friends I met in Nottingham over the years: Brendan, Clare, Collette, Emma, Izzy, Jemma, Jennie, John, Leah, Linda, Lisa, Liz, Louise, Matt, Pippa, Phil and Rich. To Helen, Vicky and all the friends of the family. Thank you to all the people I've not mentioned who have been there.

To Chris and Matt for being the best house mates; for all the parties, holidays, film marathons and general insanity. To Adam, Andy, Simon and Tom for the many wonderful memories of gaming, holidays together and friendship.

Finally my family: Mum, Reg, Rain, James and baby Freddie; my Dad and Grandparents who are no longer with us; for their support and love, I owe them so much.

Contents

List of Figures

List of Tables

List of Theorems

List of Algorithms

Chapter 1

Introduction

1.1 Overview

This thesis aims to develop methods for Bayesian statistical inference for stochastic epidemic models, in particular, inference for final size data using a multi-level multi-type model. The analysis of final size data presents challenges to inference techniques, especially for more realistic models which are commonly of high dimensionality and analytically difficult. However, the need for proper statistical inference is paramount to make informed and rigorous analysis.

The remainder of this chapter provides background theory, including definitions and notation, that will be used later. Initially we define an epidemic model, the stochastic version is the focus of this thesis, two standard results for the stochastic model, the threshold and final size equations, and several extensions to the simple model. Then we outline Bayesian statistical inference and present the implementation of a technique known as Markov Chain Monte Carlo, theoretical results on validity and convergence are omitted. References to key papers are given in the appropriate sections, as well as a brief review of recent work in the area of statistical inference for epidemic models. Finally, an outline of the following chapters is given.

1.2 Epidemic Models

There is a long history of applying mathematics to the study of infectious disease data. It is thought to have originated with Bernoulli (1766), a study of the effect of vaccination on smallpox mortality. Bailey (1975) provides a further discussion of the early history and development of epidemic modelling.

Much of the following background is also presented in summary by Andersson and Britton (2000). We reproduce the outline here but refer the reader back to the fuller descriptions within that book and supplementary references.

We begin with a population of individuals, some of whom are initially infected with the remainder susceptible to the disease. Individuals become infected after contact with an infective individual, they will become an infective themselves. Infective individuals remain so for a time called their infectious period. At the end of an individuals infectious period they recover or become immune and are called removed. Thus, removed individuals play no further part in the epidemic, this may represent immunity or mortality depending on the disease being modelled.

Individuals are thus classified as being in one of three states: susceptible, infective or removed, and an individual has transitions between these states according to some model. The simplest such model we shall consider is called the Susceptible-Infective-Removed (SIR) model.

We consider only closed populations, where the total number of individuals is constant. We shall use the following convention, unless noted otherwise, that the total population size is N, of which n are initial susceptibles and a are initial infectives, i.e. $n + a = N$.

This thesis will primarily be concerned with stochastic epidemic models, which we shall

define in Section 1.2.1. However, for some diffusion results for continuous time epidemic processes considered in Chapter 2, we shall need the deterministic model outlined in Section 1.2.1.

1.2.1 Deterministic Susceptible-Infective-Removed Model

Early work on epidemics was focused on deterministic models, as they were better understood and there existed known methods to analyse them. For example, the study of HIV was initially conducted using deterministic models. For an overview of other such examples and theoretical results see Anderson and May (1991).

The formal definition of the deterministic model was given by Kermack and McKendrick (1927), known as the deterministic general epidemic. Let $x(t)$, $y(t)$ and $z(t)$ denote the number of susceptible, infectives and removed at time t respectively. The initial state is $(x(0), y(0), z(0)) = (n, a, 0)$ and $x(t) + y(t) + z(t) = N$ for all $t \geq 0$. The model is defined by the following differential equations,

$$\begin{aligned} x'(t) &= -\alpha x(t) y(t) \\ y'(t) &= \alpha x(t) y(t) - \beta y(t) \\ z'(t) &= \beta y(t), \end{aligned}$$

where α and β denote the rate of new infections and removals respectively. The important term is the product of the number of susceptibles and infectives, $x(t)y(t)$, this is the so called mass action term, where the rate of new infections depends on the product.

The deterministic model is generally valid for large populations only. In particular, outcomes near the edge of the state space for small populations can be problematic

since the deterministic solution is continuous.

1.2.2 Stochastic Susceptible-Infective-Removed Model

The stochastic model was presented around the same time as the deterministic by McKendrick (1926), however it received much less attention. The focus at that time was on discrete-time stochastic models, namely the chain-binomial model proposed by Reed and Frost.

As before, we consider a population of N individuals with n initial susceptibles and a initial infectives. The infectious periods of the infective individuals are independent and identically distributed according to a random variable T with an arbitrary but specified distribution. While infectious, an individual makes contacts with each of the N individuals in the population at times given by the points of a Poisson process of rate λ/N If the contacted individual is susceptible, they immediately become an infective and can immediately begin infecting other individuals for the length of their infectious period. An individual is removed once their infectious period has ended. The epidemic ends once there are no infective individuals remaining. Following Ball (1995), we refer to this as the standard SIR epidemic model and denote the process by $E_{n,a}(\lambda, T)$. The rate of contacting an individual is normalised by the total population in order to keep the rate independent of the population size. The epidemic process is stochastic, thus it is applicable to small populations where the deterministic approximation fails.

The infectious period, T, is a defined distribution with mean, $\mathrm{E}[T] = \iota$ and variance, $\mathrm{Var}(T) = \sigma^2$, we shall consider various infectious distributions, though they are often parametric and of a form that gives rise to tractable expressions. The special case where the infectious period is an exponential distribution is known as the general stochastic epidemic, an unfortunate historical artifact. The exponential infectious period is com-

monly used for mathematical ease, though many realistic biological disease infectious periods are poorly approximated by such a model.

1.2.3 Threshold Results

Returning to the deterministic model, Kermack and McKendrick (1927) showed that y, the number of infectives, is initially decreasing unless $y(0)(\alpha x(0) - \beta) > 0$ or equivalently $x(0) > \beta/\alpha$, i.e. the ratio of the rate of removal to the rate of infection. The model exhibits different behaviour depending upon whether $x(0)$ is greater than β/α or not, this is said to be a threshold between the two behaviours and the inequality is a threshold result.

For the stochastic case, Ball (1983) derives several threshold theorems, specifically for the SIR model presented in Section 1.2.2, Theorem 7 states that a major epidemic occurs with non-zero probability if and only if $\lambda\iota > 1$ (using our notation).

We define R_0 as the basic reproductive number for the simple SIR model, which is defined as the expected number of infections caused by a typical infective individual in the early stages of the epidemic. For more complicated models care must be taken to define a typical individual. We call R_0 a threshold parameter, since it determines if a major outbreak is possible; if this is less than one, i.e. below threshold, then each infective is at best producing a single new infective and the epidemic will quickly die out.

Note that being above threshold does not mean a major outbreak will occur, only that there is a possibility that it will. Since the process is stochastic, there is a non-zero probability that the process will die out even if it is above threshold. Determining the probability of a major outbreak given the process is above threshold is also of great

interest in epidemic theory.

For the standard SIR epidemic model, the basic reproductive number is $R_0 = \lambda\iota$, where λ is as defined in Section 1.2.2 and ι is the expected length of the infectious period. The epidemic is above threshold if $R_0 > 1$ and a majour outbreak is possible. For details see Williams (1971) and Ball (1983).

1.2.4 Final Size Results

The SIR epidemic process ends when there are no more infective individuals, all that remain are susceptible and removed individuals. The final size of an epidemic is commonly denoted Z and is defined to be the total number of initial susceptibles that became infected during the course of the epidemic, i.e. $Z = S(0) - S(\infty)$. The final size does not include the initial infectives and hence $0 \leq Z \leq n$, where n is the number of initial susceptibles.

For the deterministic model, we have that $z(t) \to z_\infty < n$ as $t \to \infty$, where z_∞ is the solution of $z = n - x_0 \exp(-\frac{\alpha z}{\beta})$, i.e. the final size is less than n, meaning not everyone is infected. This is a very interesting result, that even for major outbreaks we do not expect the entire population to become infected under the SIR model. In the stochastic setting, there is a non-zero probability of all possible final sizes (for non-degenerate parameter values), so more care must be taken for the final size behaviour; this motivates our investigation in Chapter 2.

For the standard SIR epidemic model, denoted $E_{n,a}(\lambda, T)$, Ball (1986) derived the probability of a final size k ($0 \leq k \leq n$), denoted P_k^n, which satisfies the following set

of triangular equations,

$$\sum_{k=0}^{l} \frac{\binom{n-k}{l-k} P_k^n}{\left[\phi\left(\frac{\lambda(n-l)}{n}\right)\right]} = \binom{n}{l}, \quad 0 \leq l \leq n,$$

where $\phi(s) = \mathrm{E}[\exp(-sT)]$ for $s \geq 0$.

Using these equations we can theoretically calculate the final size probabilities, however there are issues of numerical stability in practice. In fact, the expressions become unstable using standard double precision for populations in the range $50 < n < 100$ (see Demiris (2004)), depending upon the parameter values and infectious period distribution.

Many limiting results have been derived for the final size of a stochastic epidemic as the population size tends to infinity, these account for a variety of models. The following results are derived rigorously in Scalia-Tomba (1990), though we present them using the form and notation of Andersson and Britton (2000).

For the standard SIR model, as defined in Section 1.2.2, there are two limiting results depending on the form of the number of initial infectives. We present only the case for a fixed number of initial infectives as the population size tends to infinity. The alternative, having the ratio of initial infectives to susceptibles tend to a constant will not be presented (see Andersson and Britton (2000, Theorem 4.1)).

Theorem 1.1 (Andersson and Britton (2000, Theorem 4.2))
Consider a sequence of epidemic processes $E_{n,a_n}(\lambda, T)$. Assume that $a_n = a$ for all n, and define ψ as the nontrivial solution to

$$1 - \exp(-\lambda\iota\psi) = \psi.$$

Also denote the final epidemic size by Z_n and write $Z_n' = Z_n + a$.

If $\lambda\iota \leq 1$ *then* $Z_n \to Z$ *almost surely, where* $\mathrm{P}(Z < \infty) = 1$ *and* Z *is the total progeny in a continuous time branching process* $E_a(\lambda, T)$*, initiated by a ancestors, in which individuals give birth at the rate* λ *during a lifetime distributed according to* T*.*

If $\lambda\iota > 1$ *then* Z_n *still converges to* Z*, but now* $\mathrm{P}(Z < \infty) = q^m$*, where* q^m *is the extinction probability of the branching process* $E_a(\lambda, T)$*. With probability* $1 - q^m$*, the sequence* $\sqrt{n}(Z'_n/n - \psi)$ *converges to a normally distributed random variable with mean* 0 *and variance*

$$\frac{\rho(1-\rho) + \lambda^2\sigma^2\psi\rho^2}{(1-\lambda\iota)^2},$$

where $\rho = 1 - \psi$*.*

From the result, it follows that the limiting final size behaviour is dependent upon whether the process is above threshold. In the case where it is, there is still uncertainty of a major outbreak. The probability of a major outbreak can be calculated using the branching process approxmation to the early stages of an epidemic.

Theorem 1.1 derives an importanting limiting result for the final size of an epidemic, the behaviour is dependent upon whether the process is above threshold and is inherently stochastic. It is an important result in epidemic theory, much work has been done to extend these results to alternative models. For example, Ball and Clancy (1993) derive an asymptotic result for the final size of a multitype epidemic where individuals move among a fixed number of groups; Ball et al. (1997) derive asymptotics for a population of households with local and global contacts.

1.2.5 Model Extensions

The general stochastic Susceptible-Infective-Removed model described so far has many limitations, specifically for application to actual epidemic data. The model assumptions usually do not reflect real-life diseases characteristics. The following selection of extensions gives an overview of some of the progress made to adapt the simple SIR model.

1.2.5.1 New States

Some diseases cannot be explained by the Susceptible-Infective-Removed sequence of states. For example, for the common cold a more appropriate model is the Susceptible-Infective-Susceptible (SIS), since the virus adapts quickly and so individuals do not become immune. In such models the epidemic process will stop with probability one since for stochastic models there is a non-zero probability of the process reaching the state of having zero infectives. This is a so called absorbing state, though the time to absorption may be infinity. Thus, to consider the behaviour of SIS models it is common to consider the quasi-stationary distribution of the epidemic, i.e. the distribution of the number of infectives conditional on the process not being extinct.

As another extension, we can consider that individuals have a latent period between being infected and becoming infectious. During this stage they are exposed but cannot spread. This new exposed state can be added to form the Susceptible-Exposed-Infective-Removed (SEIR) model. In fact, we can build arbitrary sequences of states from among: susceptible, infective, removed and exposed; as well as others. These become compartmental stochastic processes, with transitions between states according to a given model.

The SEIR model, including a latent period is an important extension in biological terms, since many real-life diseases have a latent period. Interestingly, if we restrict our attention to the final size distribution of an SIR model, then the distribution is invariant to the inclusion of a latent period, see Ludwig (1975) for details. Thus the final size analysis of this thesis apply equally to SIR and SEIR models.

1.2.5.2 Two Level Mixing And Multiple Type Models

Thus far we have only considered homogeneous populations of homogeneously mixing individuals. A natural extension is to relax these restrictions and allow multiple types of individual to mix, i.e. make contacts, in more complicated ways.

Human populations generally exhibit a structure that is of importance when modelling an epidemic. Within a population individuals will be grouped, the most obvious such grouping is within households. It is reasonable to expect the disease to spread at a different rate between individuals within the same household compared to between individuals in different households.

In this example we have two levels of mixing, within household and between household. Assigning each individual to a single household we then consider infectious contacts as either global or local depending on if they originate outside or within the individuals household respectively.

It is desirable to obtain threshold results for these models as for the simple SIR, see Ball et al. (1997) and Ball and Neal (2002) for details. Firstly, the definition of the basic reproductive number needs to be adjusted for this new setting. For the two level mixing model Ball et al. (1997) defines R_* as a generalisation of R_0, such that a major outbreak is possible only if $R_* > 1$ (recall that all threshold results are derived in a

limiting sense, for any finite population a stochastic epidemic process has a non-zero probability of any final size), where $R_* = R_G\mu$, the product of the reproductive ratio R_G for global contacts and the mean size μ of local outbreaks. Thus R_* is a clump-to-clump reproductive ratio, the expected number of clumps contacted by a clump. For clumps of size one, $R_* = R_0$.

Two-level mixing models can be investigated using an independent household model, where individuals within a household are subject to a local epidemic process and a constant global infection. Addy et al. (1991) consider maximum likelihood procedures to estimate community infection rates from final size data under an independent household model. We shall apply our approach, where the household final sizes are not independent, to the same data in Chapter 3.

For many diseases, there is a need to consider sub-populations that will have varying characteristics, i.e. to consider non-homogeneous populations. The addition of multiple types of infectives, where the mixing rate between types may differ, requires an alternative threshold result and final size analysis, Ball and Clancy (1993) derive examples of these using multi-type branching processes.

1.2.5.3 Epidemics On Random Graphs

In Chapter 2 we shall consider an epidemic process represented as a directed random graph, and use this representation in Chapters 3 and 4 in our inference technique.

There is a different area of epidemic modelling using random graphs to represent the contact structure of the population, on which an epidemic process is then begun. This is related to non-homogeneously mixing multi-level models as discussed by Ball and Neal (2002), however the contact network is itself random.

Many interesting papers exist for such models, see Andersson (1997, 1999), Newman (2002) and Kenah and Robins (2007). The random graph social networks have been extended to include so called 'casual contacts', equivalent to the homogeneous mixing and two level mixing effects, see Ball and Neal (2008).

1.2.6 Final Size Data, Missing Data And Partially Observed

Consider a completely observed SIR epidemic model. For each individual, i, we observe the time they are infected t_{i_1} and their subsequent removal time t_{i_2}, i.e. the length of their infectious period is $t_{i_2} - t_{i_1}$. Also, we observe the individuals that are contacted by individual i during its infectious period. From these complete data, we can make inference about the infection and removal rates in the model using techniques such as maximum likelihood (since the likelihood can be expressed given the complete data) or using a Bayesian approach, for example O'Neill (2002) use Markov Chain Monte Carlo (MCMC) (see Section 1.3.2) for the case of missing data, but their method applies equally to complete data.

Complete data are generally not available for real diseases, partly due to the difficulty in detecting the exact times of infection and removal biologically, also the effort to record all the observations for a moderately large population is prohibitive.

Epidemic data usually consist of the times the disease was detected in an individual, and the period over which symptoms (or a positive result for some medical test, for example positive swabs for MRSA) were observed. As such, the actual infection and removal times are unknown. The scale of the epidemic in time can also be an issue, observations will usually be recorded as daily counts; if the disease cycle occurs on a shorter time scale then such censoring will affect any inference.

In the extreme case, for an epidemic we record only those individuals who were infected during the course of the outbreak, i.e. the removed individuals in an SIR model, without any further details of when they were infected or for how long. It is thus simple to collate the detection times into a single time point, the end of the epidemic, and each detected individual has been infected. We then have the final size of the outbreak. See Longini et al. (1988) for an example of final size data.

For final size data, care must be taken as to when the epidemic has ended. Obviously, final size data is not defined for a disease where individuals are not removed, i.e. SIS or equivalent models, nor is it complete if the end of the epidemic cannot be determined satisfactorily.

An epidemic is said to have missing data or be partially observed if the complete information on each individual is not recorded. The type of missing data considered so far concerns not observing all events for an individual. Alternatively, we may only observe a subset of the population (which may also have missing data), i.e. there are individuals who are completely unobserved. If the fraction of the population observed is small, then any inference about the epidemic must take this into account.

1.3 Inference And Markov Chain Monte Carlo

Given data about an outbreak of a disease, we would like to develop an epidemic model and then fit the model to data. The model will contain a number of parameters and we wish to infer the parameter values from the data.

There are many approaches to inference, for example the commonly used maximum likelihood method considers the likelihood function of the model parameters given the

data and then maximises this likelihood over the parameter space.

For any inference technique, care must be taken that the model is appropriate. Inference can be made even for a poor choice of model, but the applicability of the results is questionable. Model choice is not considered in this thesis, see Berger (1993) and Kass and Raftery (1995) for a discussion of topics in the field of model choice including Bayes factors and model averaging. For an application to epidemics, see Neal and Roberts (2004) and O'Neill and Marks (2005)

In the following sections we shall outline the Bayesian approach and methodologies used. For a more detailed background see for example Bernardo and Smith (1994) or Gelman et al. (2003).

1.3.1 Bayesian Inference

Bayesian inference is a modern statistical technique for parameter estimation for a model given data. The model must permit a likelihood and the parameters require prior distribution. The likelihood and prior distribution are combined, using Bayes' theorem, to compute the posterior distribution of the parameters given the data.

We introduce the following notation to express the concepts in this and following sections, alternate notation will be used for the specific algorithms in Chapters 3 and 4. Let θ denote the vector of parameters to the model, under the Bayesian framework the model parameters are considered as random variables. Let X denote the outcome of the model, itself a random variable. We may proceed given a realisation of X, namely x, together with a likelihood of x given a parameter set, $\mathrm{L}(x|\theta)$ (a common alternative notation for the likelihood is $\pi(x|\theta)$) and a prior density on the parameters, $\pi(\theta)$.

1.3.1.1 Bayes' Theorem

We wish to derive the posterior density of the parameters conditional on the data, i.e. $\pi(\theta|x)$, which we obtain from the following relation

$$\pi(\theta|x) = \frac{\mathrm{L}(x|\theta)\pi(\theta)}{\int_\theta \mathrm{L}(x|\theta)\pi(\theta)\ \mathrm{d}\theta} \propto \pi(x|\theta)\pi(\theta).$$

This formula is known as Bayes' Theorem. The formula can be expressed up to proportionality by ignoring the integral in the denominator, which is necessary to obtain the correct normalising constant for equality. The expression is commonly stated crudely in words as *"The posterior is proportional to the likelihood times the prior"*.

The integral may not permit a closed form in general. One technique to overcome this problem is to choose an appropriate prior for the likelihood, a so called conjugate prior. If such a prior cannot be found, then a numerical evaluation of the integral is necessary, one such technique is Monte Carlo integration. Alternatively, Markov Chain Monte Carlo (MCMC) is a technique that avoids the calculation of the integral and gives an approximation to the posterior density.

1.3.1.2 Prior Distributions

The choice of prior is a matter of controversy even among proponents of the Bayesian approach. Broadly there are two types of prior, non-informative and informative/elicit. The former is chosen when we have no information concerning θ and wish the prior to reflect our lack of knowledge, i.e. not to favour one value of θ over another. The latter are created using an expert's opinion, we shall not consider elicitation methods any further in this thesis.

As mentioned, priors are commonly chosen from parametric families for computational convenience. In particular, if the distribution is conjugate to the likelihood, that is the posterior density belongs to the same family as the prior, computation can be made simpler. To avoid confusion, the parameters of a prior distribution are termed hyperparameters.

Conjugate Prior Morris (1983) showed that exponential families, which are a common form of likelihoods, have conjugate priors. Bayesian inference techniques will work for any prior, however for speed of computation conjugate priors can be preferable. For example, consider observing x heads after n coin tosses, wanting to make inference about the probability of a head. Here $\theta = p$, the probability of a head. Thus the likelihood is

$$\mathrm{L}(x|p) = \binom{n}{x} p^x (1-p)^{n-x}, \qquad x = 0, 1, \ldots, n.$$

If we choose a beta distribution for the prior on p, with the hyperparameters α and β, thus $p \sim \mathrm{Beta}(\alpha, \beta)$

$$\pi(p) = \frac{1}{B(\alpha, \beta)} p^{\alpha-1}(1-p)^{\beta-1},$$

where $B(\alpha, \beta) = \frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha+\beta)}$. Then the posterior is of the form

$$\begin{aligned}\pi(p|x) &= \frac{\frac{1}{B(\alpha,\beta)}\binom{n}{x} p^x (1-p)^{n-x} p^{\alpha-1}(1-p)^{\beta-1}}{\int_{q=0}^{1} \frac{1}{B(\alpha,\beta)}\binom{n}{x} q^x (1-q)^{n-x} q^{\alpha-1}(1-q)^{\beta-1}} \\ &= \frac{p^{x+\alpha-1}(1-p)^{n-x+\beta-1}}{B(x+\alpha, n-x+\beta)}.\end{aligned}$$

Thus the posterior density is again a beta distribution. We say the beta prior is conjugate to a binomial likelihood.

The hyperparameters of the prior are chosen to reflect our knowledge of the parameter. For particular families it is not always possible to form truely non-informative conjugate priors.

Proper Non-informative Priors For non-informative priors, care must be taken to ensure all expressions are well defined. If the parameter space is finite, either discrete or continuous, then a proper non-informative prior can be defined. In either case we use the uniform distribution, for a finite discrete parameter space, $\Theta = \{\theta_1, \ldots, \theta_n\}$,

$$\pi(\theta_i) = \frac{1}{n}, \qquad i = 1, \ldots, n.$$

Likewise in the case of a continuous bounded parameter space, $\Theta = [a, b]$ for $-\infty < a < b < \infty$,

$$\pi(\theta) = \frac{1}{b-a}, \qquad \theta \in \Theta.$$

Both priors are non-informative, giving no preference to any value of θ in the parameter space.

Improper Priors However, if the parameter space is unbounded then the prior may become ill-defined. For example, if $\Theta = (-\infty, \infty)$ and we choose a prior of $\pi(\theta) = c$ for all $\theta \in \Theta$, then clearly $\int \pi(\theta)\, \mathrm{d}\theta = \infty$.

This is a so called improper prior. Inference is still possible given the additional condition that $\int \pi(x|\theta)\, \mathrm{d}\theta = C < \infty$. In that case,

$$\pi(\theta|x) = \frac{\pi(x|\theta)c}{\int \pi(x|\theta)c\, \mathrm{d}\theta} = \frac{\pi(x|\theta)}{C},$$

then we may proceed as for a proper prior.

1.3.1.3 Sequential Bayes'

An important part of Bayesian inference is the ability to combine the analysis of multiple data sets. If x_1 and x_2 are independent data samples, then

$$\begin{aligned}\pi(\theta|x_1, x_2) &\propto \mathrm{L}(x_1, x_2|\theta)\pi(\theta) \\ &\propto \mathrm{L}(x_2|\theta)\mathrm{L}(x_1|\theta)\pi(\theta) \\ &\propto \mathrm{L}(x_2|\theta)\mathrm{L}(x_1|\theta)\pi(\theta) \\ &\propto \mathrm{L}(x_2|\theta)\pi(\theta|x_1).\end{aligned}$$

That is, we can obtain the full posterior of θ from x_1 and x_2 by first evaluating the posterior density of θ on the first data set x_1, $\pi(\theta|x_1)$, then use it as the prior on θ for the second data set x_2. Given an arbitrary number of independent data sets we can update the posterior for θ sequentially.

1.3.1.4 Posterior Estimation

Once the posterior distribution is obtained, we may plot the density function to represent the information about the parameters from the data. To summarise the density there are two common approaches, point estimation and interval estimation.

Point estimation is a single summary statistic, usually the mean, median or mode of $\pi(\theta|x)$. The appropriate measure to use, either the mean, mode or median, is dependent upon whether the density is symmetric, multi-modal or heavy tailed.

Interval estimation generally requires a numerical approach, for a given level α we obtain a subset C of Θ such that

$$1 - \alpha \leq \mathrm{P}[C|x] = \int_C \pi(\theta|x)\ \mathrm{d}\theta.$$

Such a subset C is not unique, a commonly used set is the highest posterior density (HPD) defined as

$$C_{HPD} = \{\theta \in \Theta : \pi(\theta|x) \geq \xi(\alpha)\}$$

where $\xi(\alpha)$ is the largest constant satisfying $\mathrm{P}[C|x] \geq 1 - \alpha$. Thus C_{HPD} consists of the most likely θ values.

An alternative simpler interval is the equal tail set. From $\pi(\theta|x)$ calculate the $\alpha/2$ and $1 - \alpha/2$ quantiles,

$$C_{ET} = \{\theta \in \Theta : \theta_{\alpha/2} \leq \theta \leq \theta_{1-\alpha/2}\}.$$

The equal tail interval can be misleading for multi-modal non-symmetric posterior densities. Clearly, $C_{HPD} = C_{ET}$ only for unimodal symmetric densities.

1.3.2 Markov Chain Monte Carlo

Markov Chain Monte Carlo (MCMC) methods are used to implement Bayesian inference by evaluating a Markov chain constructed such that its stationary distribution is the posterior density of interest.

An MCMC algorithm is used to simulate approximate samples from the posterior distribution by generating a Markov chain. The foundations of MCMC were developed by

Metropolis et al. (1953) and generalised by Hastings (1970). It was not until much later that the approach appeared in the statistical community, with the paper by Gelfand and Smith (1990). We shall now outline several MCMC algorithms. There is a vast literature on Bayesian techniques and MCMC, for more details on the limiting theory and other practical issues see for example Gilks et al. (1996) and Robert and Casella (1999).

Let $\{Y_0, Y_1, \dots\} = \{Y_t : t \geq 0\}$ be a sequence of random variables such that Y_{t+1} depends only on the current state, Y_t. Associated with the sequence are the transition probabilities, $\mathrm{P}[Y_{t+1}|Y_t]$. Such a sequence is called a Markov chain.

We wish to construct a chain such that once it reaches equilibrium, after a burn in period and regardless of the initial value Y_0, then the chain draws samples from $\pi(\theta|x)$. This is achieved by using an appropriate algorithm to sample the next element of the chain.

1.3.2.1 Metropolis-Hastings (MH) Algorithm

The posterior density, $\pi(\theta|x)$, is known up to proportionality, which is due to the normalising constant. We desire a method that does not require calculating the denominator in Bayes' Theorem. The MH algorithm draws approximate samples from the true posterior. The name is derived from Metropolis et al. (1953) and Hastings (1970) who first proposed and developed the method.

At each time t with current state $\theta^{(t)}$, the next state $\theta^{(t+1)}$ is chosen by first sampling a candidate ϕ from a proposal distribution $q(\cdot|\theta^{(t)})$. The candidate is then accepted with probability $\alpha(\theta^{(t)}, \phi)$ and then $\theta^{(t+1)} = \phi$; else rejected and the state remains the same, i.e. $\theta^{(t+1)} = \theta^{(t)}$. The acceptance probability α is the minimum of one and a

ratio of the posterior and proposal densities.

$$\alpha(\theta^{(t)}, \phi) = \min\left\{1\,,\ \frac{\pi(\phi|x)q(\theta^{(t)}|\phi)}{\pi(\theta^{(t)}|x)q(\phi|\theta^{(t)})}\right\}.$$

Thus the algorithm for a MH update is

Algorithm 1.1: Generic Metropolis-Hastings (MH) update for parameter vector θ.

Propose $\phi \sim q(\cdot|\theta^{(t)})$;
Evaluate $\alpha(\theta^{(t)}, \phi)$;
Draw $A \sim \mathrm{U}(0, 1)$;
if $\alpha < A$ **then**
    $\theta^{(t+1)} = \phi$
else
    $\theta^{(t+1)} = \theta^{(t)}$

Recall θ may represent a vector of parameters, in this case there are several possible updates. Firstly we can update all the parameters at the same time, thus all are accepted or rejected. Secondly, each parameter can be updated in turn. Each step of the chain consists of n updates, where n is the length of θ. Finally, the parameters can be updated in blocks, each parameter belonging to a single block.

1.3.2.2 Gibbs Algorithm

The Gibbs algorithm is a special case of the Metropolis-Hastings algorithm, the name is derived from Gibbs random fields where it was first used by Geman and Geman (1984).

Let θ be a vector of n parameters, $\theta_1, \ldots, \theta_n$ and let θ_{-i} denote the vector with the element θ_i removed. Then $\pi_i(\theta_i|\theta_{-i}, x)$ for $i = 1, \ldots, n$ are called the full conditional distributions of $\pi(\theta|x)$.

The Gibbs algorithm samples from the joint posterior distribution using the full conditional distributions, by sampling each of the θ_i in turn. Thus the Gibbs algorithm is

Algorithm 1.2: Generic Gibbs update for parameter vector θ.

1 For $\theta^{(t)}$;
2 Generate $\theta_1^{(t+1)}$ from $\pi(\theta_1|\theta_{-1}^{(t)}, x)$;
3 Generate $\theta_2^{(t+1)}$ from $\pi(\theta_2|\theta_{-2}^{(t)}, x)$;
4 $\vdots$;
5 Generate $\theta_n^{(t+1)}$ from $\pi(\theta_n|\theta_{-n}^{(t)}, x)$;

The Gibbs update is a special case of the Metropolis-Hastings with acceptance probability one. The proposal distribution is the full conditional distribution, i.e. $q(\theta_i|\theta_{-1}) = \pi_i(\theta_i|\theta_{-i})$. Let $\theta^{(t)} = (\theta_1^t, \ldots, \theta_n^t)$ be the current state and $\phi = (\theta_1^{t+1}, \ldots, \theta_n^t)$ be the proposed state when updating θ_1. Using the fact that

$$\pi_1(\theta_1|\theta_2, \ldots, \theta_n) = \frac{\pi(\theta_1, \theta_2, \ldots, \theta_n)}{\pi(\theta_2, \ldots, \theta_n)},$$

the acceptance probability of such an update is

$$\begin{aligned}
\alpha(\theta^{(t)}, \phi) &= \min\left\{1\,,\ \frac{\pi(\phi)q(\theta^{(t)}|\phi)}{\pi(\theta^{(t)})q(\phi|\theta^{(t)})}\right\} \\
&= \min\left\{1\,,\ \frac{\pi(\theta_1^{t+1}, \theta_2^t, \ldots, \theta_n^t)\pi_1(\theta_1^t|\theta_2^t, \ldots, \theta_n^t)}{\pi(\theta_1^t, \theta_2^t, \ldots, \theta_n^t)\pi_1(\theta_1^{t+1}|\theta_2^t, \ldots, \theta_n^t)}\right\} \\
&= \min\left\{1\,,\ \frac{\pi(\theta_1^{t+1}, \theta_2^t, \ldots, \theta_n^t)\pi_1(\theta_1^t, \theta_2^t, \ldots, \theta_n^t)\pi(\theta_2^t, \ldots, \theta_n^t)}{\pi(\theta_1^t, \theta_2^t, \ldots, \theta_n^t)\pi_1(\theta_1^{t+1}, \theta_2^t, \ldots, \theta_n^t)\pi(\theta_2^t, \ldots, \theta_n^t)}\right\} \\
&= 1.
\end{aligned}$$

The benefit of Gibbs updates is in no longer needing to calculate and tune an acceptance probability, since all proposed samples are used. This can result in much shorter runs to

obtain a suitable sample. It is possible to combine the Metropolis-Hastings and Gibbs updates into the so called Metropolis within Gibbs, as well as other update methods; see Gilks et al. (1996) for more details.

1.3.2.3 Proposal Distributions

The Gibbs algorithm is a special case of the Metropolis-Hastings using an appropriately chosen proposal distribution and performing sequential updates on each parameter in turn.

For the standard MH algorithm we are free to choose an arbitrary proposal distribution provided it is reversible, i.e. $q(\theta^{(t)}|\phi) \neq 0$.

The choice of proposal distribution directly affects the rate of convergence of the chain. Often the proposal will have tunable parameters that are determined before the MCMC algorithm is run. These tunable parameters, also known as scaling factors are vital to optimal performance. The following are a selection of common proposals, for details on convergence and performance see for example Sherlock et al. (2009)

Independence Sampler The simplest choice for a proposal distribution is one that is independent of the current state,

$$q(\phi|\theta^{(t)}) = q(\phi).$$

This is called an independence sampler, as each proposal is independent of the chain. Independence samplers will perform poorly if they do not cover the regions of the state space with large posterior density, though as with many possible options in Bayesian inference, there are situations in which it performs better. The acceptance probability

for an independence sampler is

$$\alpha(\theta^{(t)}, \phi) = \min\left\{1\,,\, \frac{\pi(\phi)q(\theta^{(t)})}{\pi(\theta^{(t)})q(\phi)}\right\}.$$

Symmetric Random Walk Metropolis (RWM) The random walk is a popular choice, with the symmetric case being common due to a simpler acceptance probability. In this case the proposal distribution is

$$q(\phi|\theta^{(t)}) = q(|\phi - \theta^{(t)}|).$$

The acceptance probability is then simply the ratio of the likelihoods. Thus a proposal with a higher likelihood is always accepted, which can cause problems with multi-modal posteriors.

$$\alpha(\theta^{(t)}, \phi) = \min\left\{1\,,\, \frac{\pi(\phi)}{\pi(\theta^{(t)})}\right\}.$$

Commonly, the chosen symmetric density is a normal distribution centred at the current state, with the variance σ^2 as a tunable parameter, i.e. $\phi - \theta^{(t)} \sim \mathrm{N}(0, \sigma^2)$. This is the original case proposed by Metropolis et al. (1953). For vectors of parameters, we can generalise to the multivariate normal distribution letting $\phi - \theta^{(t)} \sim \mathrm{N}_n(0, \Sigma)$, where Σ is the n dimensional covariance matrix.

Multiplicative RWM For non-negative parameters, the additive nature of the random walk requires special attention at the boundary. Instead, let the proposal be a random multiple of the current state, $\phi = \theta^{(t)} \exp(U)$ where $U \sim \mathrm{N}(0, \sigma^2)$. This avoids the complication of checking for negative candidate values. The acceptance probability

of a multiplicative RWM is

$$\alpha(\theta^{(t)}, \phi) = \min\left\{1\,,\ \frac{\pi(\phi)\phi}{\pi(\theta^{(t)})\theta^{(t)}}\right\}.$$

1.3.2.4 Convergence, Burn In And Thinning

The theoretical justification of MCMC states that the samples are approximately drawn from the posterior density once the chain has reached its stationary distribution. That is $\theta^{(t)}$ as $t \to \infty$, is a sample from $\pi(\theta|x)$.

The speed at which this happens, known as the rate of convergence of the chain, may not be possible to describe analytically. The proposal distribution will have a large affect on the rate of convergence.

For example, consider a symmetric random walk metropolis with a normal distribution. If the scale factor σ^2 is too small the chain will accept many small jumps and it will take a long time to explore any tails of the distribution. Similarly, if the scale factor is too large, the chain will often reject large unlikely jumps and it will fail to move at all.

The acceptance probability determines how often the chain will move, and there has been much work on determining optimal scaling to give the optimal rate of convergence and mixing. Roberts et al. (1997) derived the often quoted value 0.234, that is the optimal scaling factor results in an acceptance rate of 0.234. The acceptance rate is the ratio of proposed candidates that are accepted. This result has received a great deal of attention, and its validity in a wide number of situations has been tested.

Without an analytic result for the convergence of a chain, it is necessary to adopt a more subjective measure. For a given component of θ, a trace plot shows the history

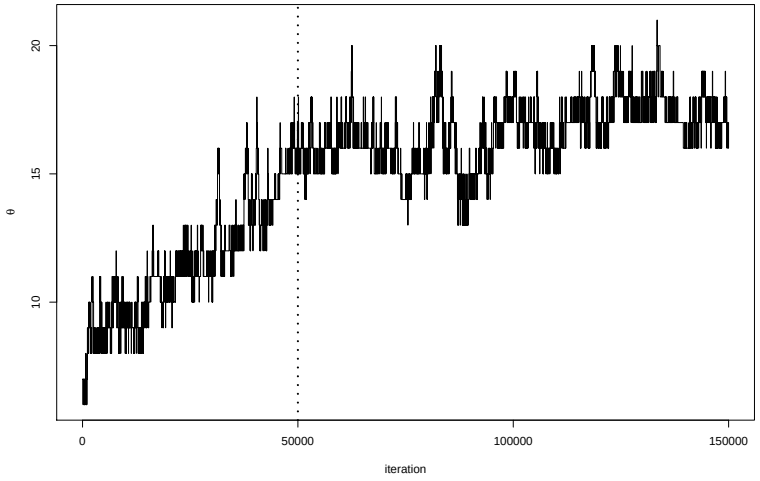


Figure 1.1: Example trace plot, the dotted line indicated the end of the burn in period. There after each iteration is a usable sample from the posterior density.

of the parameter for each iteration of the chain. An example is shown in Figure 1.1, there is a clear initial period where the parameter θ changes dramatically from its initial value (an arbitrarily chosen point in the parameter space). Once this period has finished, the plot shows a more stable variation and the chain has reached its stationary distribution as desired.

A burn in period is the number of the initial iterations that are not used as samples to estimate the posterior density. The number of iterations to remove is generally determined from looking at a trace plot of the parameters. Beyond the dotted line in Figure 1.1, we claim the chain has converged. Iterations past this point are thus approximate samples from the posterior density of interest.

We require independent samples from the posterior in order to generate a reasonable approximation to the density. Since each iteration is dependent upon the previous one, the chain does not draw truly independent samples. To overcome this, a process of thinning is applied to the observations after convergence has been reached. An example

of thinning would be to take only every fifth sample. The amount of thinning is based on how dependent consecutive samples are (a result of the proposal distribution) and the total number of available samples (less those removed in the burn in period).

We can inspect the independence of samples using the Auto Correlation Function (ACF), a measure of the correlation of a process. Let $\{X_t : t \geq 0\}$ be a process, then the ACF between two time points, s and t is given by

$$R(s,t) = \frac{\mathrm{E}\left[(X_t - \mu_t)(X_s - \mu_s)\right]}{\sigma_s \, \sigma_t},$$

where X_t has mean μ_t and variance σ_t^2.

If the process is second-order stationary, then the mean and variance are time independent and we may approximate the ACF for a discrete finite process. Note, the computed ACF for a chain will vary depending on the amount of thinning and the form of the proposal distributions.

Adaptive MCMC Commonly there will be no indication as to how to set the tunable parameters or how to choose the initial value of the chain to minimise the burn in period. An initial MCMC run can be performed to obtain a better estimate, but for complicated models this can be impractical.

To overcome this issue Haario et al. (2001) developed an adaptive proposal scheme, the theory of which is developed by Roberts and Rosenthal (2007). The proposal density is adapted for optimal scaling as the chain converges.

Adaptive MCMC can perform poorly if the chain adapts to a region of the state space away from the posterior mode. The chain can become 'stuck', that is the chain remains in a subset of the state space of low posterior density, so several runs are still required

with varying initial conditions.

The adaptive proposals discussed are RWM using a normal distribution, for non-normal proposals it would be necessary to check that the resulting adapted proposal conserved the ergodicity of the Markov chain.

1.3.3 Non-Centred Parameterisations

To make inference possible for the epidemic models we shall impute some missing data, otherwise the likelihood is intractable for the models we wish to consider. Thus we augment the observed data set X with an imputed data set Y. Then the likelihood for the parameters of interest is $\pi(\theta|x, y)$ and the posterior is the joint posterior of the parameters of interest and the imputed data, i.e. $\pi(\theta, y|x)$.

The imputed data $Y = y$ will naturally be dependent upon the parameters θ, and the likelihood of the observed data $X = x$ is in turn dependent upon the imputed data. This natural framework is called Centred Parameterisation, as the imputed data y is centred between the parameters and the observed data. This dependence can cause poor mixing of the chain and very slow rates of convergence.

Non-centred parameterisations and partially non-centred parameterisations attempt to make a new data set y' that is independent of θ. The imputed data y is then some function of y' and θ. This can greatly improve the mixing properties of the Markov chain.

It is not always possible to re-parameterise the augmented data, nor is the benefit guaranteed in all situations. For more details and a summary of non-centred methods see Papaspiliopoulos et al. (2003), Neal and Roberts (2005), Kypraios (2007) and Jewell

et al. (2009).

1.3.4 Approximate Bayesian Computation

The standard Bayesian approach requires the likelihood of the data x given a realisation of the parameters θ, if the likelihood is intractable we may augment by incorporating missing data y as additional parameters. The form of $\pi(x|\theta)$ or $\pi(x|y,\theta)$ is key to the efficiency of the algorithm. However, if the likelihood cannot be augmented or is still intractable then another approach is necessary.

Beaumont et al. (2002) proposed a likelihood free method named Approximate Bayesian Computation (ABC). Originally applied to population genetics, the method was expanded and investigated further by Marjoram et al. (2003), Plagnol and Tavare (2004) and Blum (2009). Interest in ABC is growing due to its simplicity over MCMC, in terms of coding and complexity, see Toni et al. (2009) for examples applications.

To implement likelihood free methods, the underlying stochastic process must be easy to simulate relative to the cost of computing the likelihood. Thus, for a given parameter set θ we can generate an observation x'. As in MCMC, a proposal distribution is used to generate a candidate parameter ϕ, we then simulate a realisation of the stochastic process using the candidate parameter and update $\theta = \phi$ if the outcome matches the observed data, i.e. if $x' = x$.

Algorithm 1.3: Approximate Bayesian Computation (ABC) using exact match

```
Draw φ ~ π(θ);
Simulate x' from process with parameter φ;
if x' = x then
    accept φ
else
    reject φ
```

Using an exact match between the observed data x and the simulated data is only viable if the probability of the observed outcome is sufficiently large, i.e. $\mathrm{P}[x|\theta] > \alpha$, where α is equivalent to the acceptance probability for a Metropolis-Hastings update.

Thus if the process is highly variable or of high dimensionality, the simulated data will rarely equal the observed. In this case we may consider not only exact matches but those that are close as well. We define a metric ρ between two outcomes of the stochastic process, and a tunable distance parameter ϵ. The modified algorithm is:

Algorithm 1.4: Approximate Bayesian Computation (ABC) using distance metric

```
Draw φ ~ π(θ);
Simulate x' from process with parameter φ;
if ρ(x, x') < ε then
    accept φ
else
    reject φ
```

Using ABC we obtain approximate samples from the posterior conditional on the distance. As $\epsilon \to \infty$ the samples are drawn from the proposal (to simplify the algorithm an independence sampler is commonly used, the samples are then drawn from the prior on θ). As $\epsilon \to 0$ we obtain samples from $\pi(\theta|\rho(x, x') < \epsilon)$. The choice of ϵ is a balance between accuracy and acceptance.

An appropriate distance metric may not be immediately obvious, instead it is common to use summary statistics, $S(x)$, defining a metric in terms of them, i.e. $\rho(S(x), S(x'))$. Using sufficient statistics is related to the accuracy of the approximation, see Sousa et al. (2009) for an example on a discussion on summary statistics for ABC.

1.4 Previous Literature On Epidemic Models And Inference

The literature on mathematical modelling of epidemics is extensive, see Section 1.2 for examples on the topics discussed. For a summary text, the books by Andersson and Britton (2000) and Diekmann and Heesterbeek (2000) give background and a detailed introduction to the area. The former discusses stochastic modelling and their statistical analysis while the latter is focused on deterministic models.

Inference for stochastic epidemics, in particular Bayesian inference, has developed rapidly in recent decades due to the increase in available computer power. An introduction and summary of progress in the use of Markov Chain Monte Carlo (MCMC) methods applied to SIR epidemic models is given by O'Neill (2002). In particular, the work by Gibson and Renshaw (1998) and O'Neill and Roberts (1999) demonstrate the first use of MCMC methods for epidemic inference. This was later expanded to non-Markovian infectious periods by O'Neill et al. (2000) and to incorporate two-level mixing models by Demiris and O'Neill (2005b).

Interest is not only restricted to the Markovian continuous time SIR model (and its derivatives). For example, O'Neill and Becker (2001) return to a previously analysed final size outbreak to consider varying susceptibility. O'Neill (2003) considers inference for the discrete-time Reed-Frost model and Streftaris and Gibson (2004) consider inference for continuous time epidemic where infectious periods follow a Weibull distribution using MCMC.

Modelling and inference of partially observed epidemics has also developed. Panaretos (2007) considers a partially observed branching processes to model the early stages of an epidemic, focusing on the probabilities of a minor or major outbreak conditioned

on the observed process. Britton and O'Neill (2002) consider the contact structure (or social network) underlying the epidemic to be a random graph with Bernoulli random edges using MCMC. In addition to extending inference to multi-level mixing models, Hayakawa et al. (2003) allowed for multi-type models where differing types have their own infection rates. They also consider the case where the number of susceptibles is unobserved, i.e. a partially observed epidemic.

Though MCMC methods are common in inference for epidemic models, there are a variety of specific algorithms for specific problems. For example, the Approximate Bayesian Computation (ABC) technique has been used by Blum and Tran (2008) to make inference on the spread of HIV and Clancy and O'Neill (2007) use rejection sampling algorithms instead of MCMC to obtain exact Bayesian inference and perform model selection.

1.5 Thesis Outline

We begin in Chapter 2 studying a stochastic epidemic process using a representation of the epidemic as a directed random graph. Properties of the representation are investigated and various approaches to its analysis are considered and compared. The representation removes the need for temporal analysis.

Using the directed random graph representation, an MCMC algorithm is constructed in Chapter 3. We proceed to analyse a well known data set and compare results to previous work for a one-type one-level model. The algorithm is adapted to include extensions discussed in Section 1.2.5 and 1.2.6. Section 3.3 incorporates partially observed one-type one-level models, which are extended to two-level mixing models. In Section 3.5 we extend the algorithm to accommodate an arbitrary number of levels and types,

this general framework covers multi-type multi-level models. Section 3.6 applies the algorithm to a previous study by Demiris and O'Neill (2005a). Finally, in Section 3.7 we discuss practical issues of implementing the algorithm and techniques used to overcome them.

Chapter 4 is a case study of an outbreak of Equine Influenza (H3N8) at Newmarket in 2003. The algorithms developed in Chapter 3 are implemented and we consider approaches to overcome the practical difficulties, in particular the length of the MCMC runs required.

Finally, Chapter 5 gives a summary of the results of the thesis, the methods and algorithms developed and their application to the case study data sets.

Chapter 2

Conditioned Epidemic Processes

2.1 Introduction And Motivation

When investigating the behaviour of a disease, it is common to have many unobserved events. Normally the exact infection times of individuals are unobserved, typically only their removal times are known. Given the relative timescale for some diseases the removal times recorded may be inadequate in detail for temporal inference methods.

Instead of considering the temporal information, we can infer something about the infection rates from the final size alone, i.e. the number of initially susceptible individuals that are infected by the end of the epidemic. By considering only the end point of the epidemic we are potentially losing a lot of information, indeed if more complete temporal data is available or if analysis of temporal effects such as interventions are desired then alternative methods are required.

The non-temporal representation of the epidemic process we consider is a directed random graph. In this chapter we shall present the relationship between the final size of an epidemic and a digraph, then investigate the properties of the directed random graphs. In particular, we consider a random graph conditioned on having a certain connectedness property, which corresponds to conditioning the epidemic on a particular final size. Being able to sample from the set of such conditioned graphs leads in turn to Equation (2.2). In general it is also of interest to understand the distribution of

conditioned random graphs, since such information can be used to design better Markov Chain Monte Carlo (MCMC) algorithms.

Section 2.2 defines a directed random graph and its relationship to a stochastic epidemic process. The correspondence between connectedness and final size is explained and notation for the following sections is presented.

We begin by studying small graphs with less than twenty nodes, which correspond to small populations. The graphs are first characterised by their edges, Section 2.3 considers two approaches to calculate the probability of a given connectedness in terms of edges. Each approach is presented as a counting procedure, with the two formulae being derived in Sections 2.3.2 and 2.3.3. The graph is then characterised by rank in Section 2.4, with comparison to the edge characterisation and counting methods. The correspondence with epidemic processes with varying infectious periods is made and a general framework for an arbitrary infectious period is presented in Section 2.4.4. This allows for a specified distribution for the infectious period, provided each individual's infectious period is independent and the specified expectation exists.

Numerical results are presented for the edge and generation representations, however computational limits are reached for fairly small populations. Section 2.5 considers discrete-time branching processes conditioned on their total progeny. These processes can be used to approximate the connectedness of a directed random graph under certain limiting conditions. Exact coupling of the two processes is not derived, instead, a numerical investigation is used to demonstrate the limiting behaviour.

2.2 Directed Random Graphs

2.2.1 Definition Of A Directed Random Graph And C-Connectedness

A directed random graph is a mathematical structure from Graph Theory, which has its own standard notation and definitions. In this section we shall restrict our discussion to the directed graph only. In Section 2.2.2 corresponding terms for an epidemic process as defined in Section 1.2 will also be used. For the remainder of this chapter several epidemic and graph notations will be used interchangeably. For Chapters 3 and 4 the epidemic definitions will be preferred.

The definitions and theory presented in this section are sufficient for this thesis. For a more detailed investigation of Graph Theory, including concepts not related to epidemic modelling, see Ore (1967) and Bollobás (1998). Also Harary et al. (1965) discusses the theory and applications of directed graphs to structural models in the social sciences, though not with regard to epidemic modelling.

Standard Graph Theory is concerned with fixed graphs, however we are interested in those of varying characteristics, so called Random Graphs. Again we refer the reader to more specific literature, Bollobás (1985) for example, to explore random graphs in more detail. There are two types of graph, directed and undirected, the correspondence to an epidemic requires the concept of one individual infecting another, i.e. a direction for the infection to occur, hence we consider the directed random graphs.

Define a directed random graph, G, as a collection of N labelled vertices, $1, \ldots, N$ (for finite N). Set a subset of the vertices as roots, let there be R roots ($1 \leq R \leq N$). Without loss of generality we may assign root vertices the labels $1, \ldots, R$ and non-root vertices the labels $R+1, \ldots, N$. For each ordered pair of distinct vertices (i, j), where

$1 \leq i, j \leq N$, $i \neq j$, a directed edge from i to j occurs with a probability $p_{i,j}$. If the edge (i, j) exists, we say there is a path from i to j. There are $N(N-1)$ possible directed edges between all pairs of distinct points, we do not consider parallel edges or loops beginning and ending at the same vertex. Thus a directed random graph G is a collection of N vertices and probabilities for the directed edges between all vertices.

The edge probabilities can take many forms. We shall initially consider independent edges, i.e. letting $\mathrm{P}[(i, j)]$ be the probability of the edge (i, j) being present, then $\mathrm{P}[(i, j), (i, k)] = \mathrm{P}[(i, j)]\mathrm{P}[(i, k)]$ for all $1 \leq i, j, k \leq N$ and $i \neq j \neq k$. More generally, random graphs are defined in terms of the out-degree distribution of each vertex. Let V_i be the out-degree distribution for vertex i. For independent edges, the out-degree distribution is multinomial, though we shall initially consider the simpler case of a binomial with parameters $N-1$ and p, $V_i \sim \mathrm{bin}(N-1, p)$.

A directed path is a sequence of directed edges, $(v_1, v_2), (v_2, v_3), \ldots, (v_{n-1}, v_n)$. A non-root vertex i is said to be directionally connected from the root vertices if there exists a directed path from at least one root vertex to i, i.e. $v_n = i$ and $1 \leq v_1 \leq R$. The graph G, is said to be directionally connected if each non-root vertex is (directionally) connected to the root vertices. A random graph is said to be C-connected if exactly C non-root vertices are directionally connected to the root vertices. If $C = N - R$ the graph is directionally connected.

The distance of vertex i to vertex j is equal to the number of edges in the shortest directed path from i to j. Let d_{ij} denote the distance of i to j. By convention, $d_{ii} = 0$ and if there is no directed path from i to j then $d_{ij} = \infty$.

Definition 2.1

The rank of an individual is its minimal distance from a root vertex, i.e. rank$(i) = \min\{d_{ji} : \textit{vertex } j \textit{ is a root}\}$.

If vertex i is not connected to the root vertices, then it has an infinite distance from all of them, hence an infinite rank. For a given random directed graph, we can summarise the ranks of all vertices into a rank chain. The rank chain counts the number of vertices of a given rank. Define X_i to be the number of vertices of rank i in the digraph for $0 \leq i \leq N - R + 1$. The zeroth rank contains the root vertices, i.e. $X_0 = R$ for all digraphs, and hence the maximum finite rank for a vertex is $N - R$, i.e. one vertex of each rank. Thus we terminate the rank chain at rank $N - R + 1$ so that $X_{N-R+1} = 0$ for all digraphs. Note that, $X_\infty = N - R - C$, i.e. the number of vertices that are not connected to the roots and are at infinite distance.

For $t = 0, 1, \ldots$ let $Y_t = \sum_{i=0}^{t} X_i$, a cumulative total of the number of vertices and define $Z_t = (X_t, Y_t)$. Then the rank chain for a digraph can be expressed as the vector $Z = (Z_0, Z_1, \ldots, Z_{N-R+1})$.

The connectivity of a digraph can be written in terms of the ranks of its vertices, all vertices of finite non-zero rank are connected to the root vertices. The rank chain also encapsulates the connectivity of the digraph it corresponds to. For a given digraph $G = g$ with the corresponding rank chain $Z = z$, its connectivity is

$$C = \sum_{i=1}^{N} \mathbb{I}_{\{0<\text{rank}(i)<\infty\}} = \sum_{t=1}^{N-R+1} |\{i : i \in g, \text{rank}(i) = t\}| = \sum_{t=1}^{N-R} x_t = y_{N-R} - y_0.$$

Where $\mathbb{I}_{\{E\}}$ is the indicator function, equal to one if the condition E is true and zero otherwise.

We shall condition the random graph on its connectedness property in order to investigate its behaviour in comparison to the unconditioned structure.

2.2.2 Epidemic Model And Its Relation To G

Recall the definition of the standard SIR (susceptible→infective→removed) stochastic epidemic model from Chapter 1. Consider a population of N individuals, of which R are initially (i.e. at $t = 0$) infective and $N - R$ are susceptible. An infective individual remains so for a period of time T_I, the infectious period, a non-negative random variable. The infectious periods of different individuals are independent. For this chapter we shall initially let T_I be a point mass distribution, i.e. $T_I = c$ for some $c > 0$. While infectious an individual has potential contacts with other individuals within the population at times given by the points of a Poisson process of rate $\frac{\lambda}{N} > 0$. Each such contact with another infective has no effect, while a contact with a susceptible individual immediately makes the susceptible an infective. At the end of its infectious period an infective no longer makes any contacts and is said to be removed, it is no longer involved in the epidemic. Let S_t and I_t be the number of susceptibles and infectives at time $t \geq 0$, respectively. The epidemic continues until there are no more infectives remaining, so $I_\tau = 0$ where τ is the stopping time of the epidemic, i.e. $\tau = \inf\{t \geq 0 : I_t = 0\}$. The final size of the epidemic is the number of susceptibles who became infected, $S_0 - S_\tau = N - R - S_\tau$.

The relation between the directed random graph G defined in Section 2.2.1 and the epidemic model is as follows (see, for example, Andersson and Britton (2000), chapter 7). The R root vertices correspond to the initial infective individuals, and the remaining vertices correspond to the initially susceptible individuals. By settting $p = 1 - \exp(-\frac{\lambda}{N}T_I)$, an edge represents an infectious contact, since the probability of a susceptible avoiding infection from a single infective is $\exp(-\frac{\lambda}{N}T_I)$. The (random) set of vertices that are directionally connected to the root vertices has the same distribution as the set of individuals who become infected in the epidemic. Thus the number of directionally non-root connected vertices has the same distribution as the final size

of the epidemic.

The result is based on Ludwig (1974) (presented separately in Ludwig (1975)), which state that for every epidemic process there is a corresponding Markov Chain which has the same final size distribution. We construct such a Markov chain, which determines the size of the next generation based only on current generation, this Markov chain allows us to construct the directed random graph and study its connectedness.

2.3 Random Directed Graphs Characterised By Edges

Denote by $\mathcal{G}_N^R$ the set of all random directed graphs on N labelled vertices of which R are roots as defined in Section 2.2.1 with out-degree distribution V. Let g be a specific directed graph from the set of all possible graphs. There are $2^{N(N-1)}$ directed graphs on N labelled vertices if we characterise the graphs by edges, since each edge is either present or not. Let $\chi_N^R(C)$ be the subset of $\mathcal{G}_N^R$ containing graphs that are C-connected.

We now consider the problem of calculating $\mathrm{P}[G = g \,|\, G \in \chi_N^R(C)]$, where $g \in \mathcal{G}_N^R$ and $0 \leq C \leq N - R$. In other words, we are interested in the distribution of G conditioned upon its being C-connected. It is not immediately obvious how best to describe an arbitrary graph. A natural approach is to characterise G by the number of edges it contains.

Henceforth we shall restrict attention to random digraphs with independent edges and set $p_{i,j} = p$ for all $1 \leq i, j \leq N$, $i \neq j$. Then for two specific digraphs g and g' both with l edges ($0 \leq l \leq N(N-1)$), we have

$$\mathrm{P}[G = g] = \mathrm{P}[G = g'] = p^l(1-p)^{N(N-1)-l} \quad \text{for } 0 \leq l \leq N(N-1). \tag{2.1}$$

The probability of a given digraph is a function of its number of edges and total number of vertices, and all digraphs with l independent edges on N vertices are equally likely. In particular, this means that a natural way to calculate $\mathrm{P}[G = g|G \in \chi_N^R(C)]$ is to evaluate $\mathrm{P}[L(G) = l|G \in \chi_N^R(C)]$, where $L(G)$ denotes the number of edges in G. In the sequel we will usually write L instead of $L(G)$ for simplicity. Note, that Equation (2.1) is only true if each edge is independent and occurs with probability p. It is simple to generalise to vertex-dependent edge probabilities if all edges are still independent. Specifically, if the probability of an edge emanating from vertex i is p_i, and if $l_1, \ldots, l_N$ are the number of edges emanating from vertices $1, 2, \ldots, N$ in g respectively, then

$$\mathrm{P}[G = g] = \prod_{i=1}^{N} p_i^{l_i}(1 - p_i)^{N(N-1)-l_i}. \tag{2.2}$$

We shall not consider different edge probabilities here, since our results in this section can not easily be extended to this situation. Theorems 2.10 and 2.12 are derived assuming interchangeable edges, reducing two isomorphic digraphs to the same case. If the edge probabilities were different per vertex the digraphs may no longer be isomorphic in general. In epidemic terms, different edge probabilities correspond to different types of individuals. We shall consider multi-type epidemics from an inference view point in Chapter 3, extending the results of Section 2.4.

Returning to the single edge probability, since all random digraphs with l edges are equally probable, we shall need to count the number of digraphs that satisfy the C-connectedness property of interest. For a given N and l there are $\binom{N(N-1)}{l}$ digraphs. However not all will be in the set $\chi_N^R(C)$ for a given $0 \leq C \leq N - R$.

Definition 2.2

Let $A_N^R(C, l)$ be the number of digraphs on N labelled vertices of which R are roots, with l edges and that are C-connected.

A digraph $g \in \mathcal{G}_N^R$ consists of N labelled vertices and a set of edges each denoted (i, j), $1 \leq i, j \leq N$. A subset of the vertices are roots. Given the set of root vertices and the distance between two vertices we have defined the rank of each vertex in Section 2.2.1. We now introduce edge types depending upon the rank or connectivity of each vertex at the end of the directed edge. For $g \in \chi_N^R(C)$, define $g + (i, j)$ to be the digraph g with the edge (i, j) added, for $1 \leq i, j \leq N$. Similarly $g - (i, j)$ is the digraph g with the edge (i, j) removed.

Definition 2.3

An edge (i, j) is called Free if adding the edge to the digraph $g \in \chi_N^R(C)$ does not change its connectivity, i.e. $g + (i, j) \in \chi_N^R(C)$.

Definition 2.4

An edge (i, j) is called Backward if the rank of vertex i is greater than or equal to the rank of vertex j, i.e. rank$(i) \geq$ rank(j).

All Backward edges are Free, though not all free edges are backward. If rank$(i) < \infty$, then both vertices are connected and adding an edge between them will not affect the connectivity of the digraph (it will also have no effect on the rank of either vertex). If rank$(i) = \infty$, then vertex i is not connected to the root vertices. Since the edge is directed, from vertex i to vertex j, adding it will not connect vertex i nor will it affect the rank of vertex j (the edge adds another path from the root vertices to vertex j of distance infinity). Strictly Backward edges are those where rank$(i) >$ rank(j) and Equal Backward edges are between vertices of the same rank, i.e. rank$(i) =$ rank(j).

Definition 2.5

An edge (i, j) is called Forward if the the rank of vertex j is greater than the rank of vertex i and rank$(i) < \infty$.

Definition 2.6

An edge (i, j) is called Required if it is a Forward edge and removing that edge will increase the rank of vertex j, i.e. rank$(j) >$ rank$(i) + 1$ in the digraph $g - (i, j)$.

By the definition of rank, once a forward edge is added from vertex i to vertex j, the rank of vertex j is exactly greater than the rank of vertex i by one, i.e. $\text{rank}(j) = \text{rank}(i)+1$ in the digraph $g + (i, j)$. With a digraph g, each edge is either forward or backward.

These definitions will be helpful in explaining the results of Sections 2.3.2 and 2.3.3. In these sections we shall give two methods for calculating $A_N^R(C, l)$. Section 2.3.2 describes how to decompose the possible digraphs into simpler basis digraphs, while Section 2.3.3 defines a recursive algorithm. The methods have many similar aspects, the latter is simpler to implement practically and the former gives more information about the underlying structure of the problem. Example output is presented in Section 2.3.4 for both approaches.

2.3.1 Digraph Connectedness Probability Mass Function On The Number Of Edges

We now consider combining the calculation of $A_N^R(C, l)$, the number of digraphs on N vertices of which R are roots such that C are connected using l edges, with the probability of a given digraph as given in Equation (2.1).

Since the values of $A_N^R(C, l)$ for varying C and l are counts of the number of digraphs, and we know the total number of possible digraphs, the counts correspond to a partition of digraphs into equivalent classes defined by their number of edges and connectedness. Specifically, for fixed N, $0 \leq R \leq N$ and $0 \leq l \leq N(N-1)$,

$$\sum_{C=0}^{N-R} A_N^R(C, l) = \binom{N(N-1)}{l}, \tag{2.3}$$

and,

$$\sum_{l=0}^{N(N-1)} \sum_{C=0}^{N-R} A_N^R(C,l) = \sum_{l=0}^{N(N-1)} \binom{N(N-1)}{l} = 2^{N(N-1)}, \tag{2.4}$$

using the identity

$$\sum_{b=0}^{a} \binom{a}{b} = 2^a.$$

We will use Equations (2.3) and (2.4) to provide a practical check of our results in Section 2.3.4.

There are $2^{N(N-1)}$ possible directed graphs on N labelled vertices, each of the $N(N-1)$ possible edges is either present or not. We can partition these into sets specified by their connectedness and number of edges. Extending the notation of Section 2.3, denote the set of digraphs on N vertices of which R are roots with C connected (not including the roots) and with l edges as $\chi_N^R(C,l)$. Then,

$$|\chi_N^R(C,l)| = A_N^R(C,l) \quad \text{for} \quad \begin{aligned} &N \in \mathbb{N}, \\ &0 \leq R \leq N, \\ &0 \leq C \leq N-R, \\ &0 \leq l \leq N(N-1), \end{aligned}$$

and

$$|\chi_N^R(C)| = \sum_{l=0}^{N(N-1)} A_N^R(C,l).$$

Following from Section 2.3, we assume each edge is present independently with probability p. Thus the number of edges present in a random digraph is distributed bi-

nomially, i.e. $L \sim \text{Bin}(N(N-1), p)$. For a digraph $G \in \mathcal{G}_N^R$ we are interested in $\text{P}[L(G) = l | G \in \chi_N^R(C) \subset \mathcal{G}_N^R]$.

Given $L = l$, all digraphs are equally likely, so

$$\text{P}[G = g \in \chi_N^R(C) | L = l] = \frac{A_N^R(C,l)}{\binom{N(N-1)}{l}},$$

and

$$\text{P}[L = l] = \binom{N(N-1)}{l} p^l (1-p)^{N(N-1)-l},$$

whence

$$\text{P}[G \in \chi_N^R(C)] = \sum_{l=0}^{N(N-1)} A_N^R(C,l) p^l (1-p)^{N(N-1)-l}. \tag{2.5}$$

Using Bayes' Theorem we have

$$\begin{aligned}
&\text{P}[L = l | G \in \chi_N^R(C) \subset G_N^R] = \\
&\quad = \frac{\text{P}[G \in \chi_N^R(C) | L = l] \text{P}[L = l]}{\text{P}[G \in \chi_N^R(C)]} \\
&\quad = \frac{\text{P}[G \in \chi_N^R(C) | L = l] \text{P}[L = l]}{\sum_{k=0}^{N(N-1)} \text{P}[\in \chi_N^R(C) | L = k] \text{P}[L = k]}, \\
&\quad = \begin{cases} \frac{A_N^R(C,l)\, p^l\, (1-p)^{N(N-1)-l}}{\sum_{k=0}^{N(N-1)} A_N^R(C,k)\, p^k\, (1-p)^{N(N-1)-k}} & C \leq l \leq N(N-1)-(C+R)(N-(C+R)), \\ 0 & \text{otherwise.} \end{cases}
\end{aligned} \tag{2.6}$$

The range of l given in Equation (2.6) is derived in Theorem 2.12 in Section 2.3.3.

In the special case when $p = 1/2$, Equation (2.6) reduces to a ratio of digraph counts,

$$\mathrm{P}[L = l | g \in \chi_N^R(C), p = 1/2] = \frac{A_N^R(C,l)}{\sum_{k=0}^{N(N-1)} A_N^R(C,k)}.$$

2.3.2 Counting C-Connected Digraphs Using Basis Digraphs

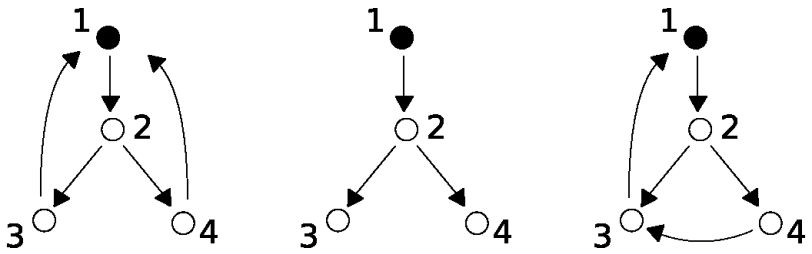
In order to partition the set of all possible digraphs according to their connectedness, we shall use the underlying structure of such a partition. Each partition can be reduced to a set of minimal digraphs which form the basis of all other digraphs in the partition.

If $g \in \chi_N^R(C)$, then there may exist other digraphs $g' \in \chi_N^R(C)$ which comprise of all the edges of g with additional free edges added, i.e. $g' = g + (i_1, j_1) + \cdots + (i_n, j_n)$ for some n and vertex pairs i, j where (i, j) is a free edge. Note that both g and g' are both C-connected as the additional edges are free.

Definition 2.7

A digraph $g \in \chi_N^R(C)$ is a basis if there are no backward edges in g, i.e. there are only forward edges. It is a minimal basis if there are no free edges and a maximal basis if all free edges are present.

Consider the three digraphs below. the middle digraph can be seen as a minimal basis for the ones either side. The two outer digraphs consist of the middle digraph with two additional backward edges.



In fact it is a minimal basis for three-connected digraphs on four vertices of which one if a root.

The digraph g has labelled vertices, however, since all edge probabilities are equal and we are characterising the digraph by its number of edges, we can consider two digraphs g and g' to be isomorphic if one is obtained by permuting the vertex labelling of the other.

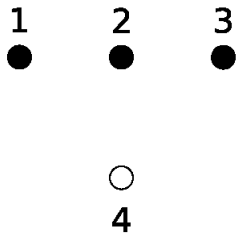
To count the number of C-connected digraphs with l edges, i.e. $A_N^R(C,l)$, we will find the set of C-connected bases. Then find the number of digraphs that have l edges in total on each basis. This amounts to decomposing the $A_N^R(C,l)$ into a sum with binomial coefficients, one per basis and accounting for the number of isomorphic digraphs.

We shall consider a simple example first, then a more complex example and finally present a general theorem for decomposing all possible digraphs characterised by their connectedness and edges into bases.

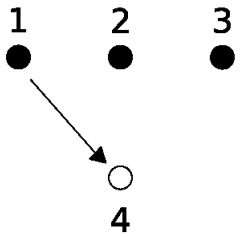
Example $(N,R) = (4,3)$

Let us consider the case where $N = 4$ and $R = 3$, i.e. four vertices of which three are roots. There are $2^{4(4-1)} = 2^{12} = 4096$ possible digraphs in this case. There are two possible classes of digraph, 0-connected or 1-connected, since $0 \leq C \leq N - R = 4 - 3 = 1$.

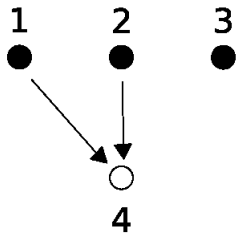
Figure 2.1 shows all the basis digraphs (for zero and one connected digraphs) up to isomorphism, since we arbitrarily label the vertices. For example, consider Figure 2.1(b), by relabelling the root vertices from $1, 2$ and 3 (reading from left to right) to $1, 3$ and 2, the two digraphs are the same in terms of their edge structure.



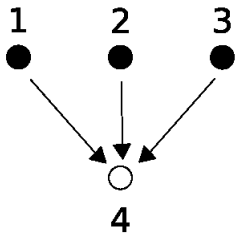
(a) Only Basis for $C = 0$



(b) First Basis for $C = 1$ with one forward edge



(c) Second Basis for $C = 1$ with two forward edges



(d) Third Basis for $C = 1$ with three forward edges

Figure 2.1: All basis digraphs up to isomorphism of four vertices with three roots, covering both $C = 0$ and $C = 1$ (connectedness). The four vertices are labelled, however this is not all possible digraphs.

Let us consider $A_4^3(0, l)$, the number of digraphs with l edges that are 0-connected. For such graphs there can be no edge from a root vertex to the single non-root vertex. This discounts 3 edges, i.e. $(1, 4)$, $(2, 4)$ and $(3, 4)$, where (i, j) denotes an edge from vertex i to vertex j, from the total number of possible edges, $4(4 - 1) = 12$. In this case the remaining 9 edges cannot affect the connectivity of the digraph. This is easy to check by looking at Figure 2.1(a), since adding any edge apart from the three excluded means vertex 4 cannot become connected. These 9 edges are backward, they emanate from a vertex of rank greater than or equal to the one they go to and thus do not affect connectivity of the digraph.

If we specify that there are l edges, then $l \leq 9$ otherwise the digraph will not be 0-connected. We have nine possible edge locations from which to choose the l specified,

i.e. choosing l from among the 9 backward edges. Thus,

$$A_4^3(0,l) = \binom{9}{l} \quad \text{for } 0 \leq l \leq 12, \tag{2.7}$$

where we use the convention that for integers $a \geq 0, b$,

$$\binom{a}{b} = \begin{cases} 0 & \text{if } b > a \text{ or } b < 0, \\ \frac{a!}{(a-b)!b!} & \text{otherwise.} \end{cases} \tag{2.8}$$

Moving on to $A_4^3(1,l)$, i.e. the number of digraphs with l edges that are 1-connected, we must have at least one edge from a root vertex to vertex 4. However, we must take care to account for all such digraphs. The three edges $(1,4)$, $(2,4)$ and $(3,4)$ are of special importance, they are forward edges. From these we can deduce the three basis digraphs shown in Figures 2.1(b), 2.1(c) and 2.1(d). Each of these three basis digraphs has one required edge, to connect vertex 4 to the root vertices.

The first, Figure 2.1(b), has only one forward edge which is required. Thus the first digraph is the minimal 1-connected basis. The second, Figure 2.1(c), has two forward edges either of which can be removed without effecting the connectivity, i.e. one of the two edges is free. Figure 2.1(d) has three forward edges between rank zero and rank one, any two of which are free. All three of these bases have the same number of potential backward edges, they differ in the number of forward edges that are used.

For a given basis graph we again wish to find the binomial coefficient formed of the number of potential backward edges and the number of these to choose, which is l less the number used as forward edges in the basis. Finally we must find the number of digraphs isomorphic to a given basis, which is a product of binomial coefficients we shall collectively call the isomorphism coefficient.

C \ L	0	1	2	3	4	5	6	7	8	9	10	11	12
0	1	9	36	84	126	126	84	36	9	1			
1		3	30	136	369	666	840	756	486	219	66	12	1
Total	1	12	66	220	495	792	924	792	495	220	66	12	1

Table 2.1: Values of $A_4^3(C, l)$ using basis decomposition. Zero values are left blank for clarity.

In this case we obtain,

$$A_4^3(1,l) = 3\binom{9}{l-1} + 3\binom{9}{l-2} + \binom{9}{l-3} \quad \text{for } 0 \leq l \leq 12. \tag{2.9}$$

Each of the terms corresponds to a basis digraph in Figure 2.1. Consider the first term, $3\binom{9}{l-1}$ (Figure 2.1(b)). The factor of three accounts for the three sets of isomorphic digraphs. The nine backward edges that cannot affect connectivity are the same as for the 0-connected case, but for this basis we need one forward edge to be assigned. This forward edge is in addition to the nine backward edges, so we obtain the term $l-1$ in the binomial coefficient, i.e. we can assign one to ten edges in total, one forward and nine backward. A similar argument yields the terms in (2.9) relating to the other two basis digraphs.

Equations (2.7) and (2.9) are expanded fully in Table 2.1. The bases decompose the digraphs into isomorphic digraphs with forward and backward edges, which can then be combined as binomial coefficients representing the choice of l less the forward edges from among the backward edges. For $l > 9$ there are not enough backward edges on the 0-connected basis, so no such digraphs exist. Similarly, if $l = 0$ then there are not enough edges to attain a 1-connected digraph.

It is simple to check Equation (2.3) for the decomposition using the relation

$$\binom{a}{b} = \binom{a-1}{b-1} + \binom{a-1}{b}.$$

Applying the relation iteratively,

$$\begin{aligned}\sum_{C=0}^{1} A_4^3(C,l) &= \binom{9}{l} + 3\binom{9}{l-1} + 3\binom{9}{l-2} + \binom{9}{l-3} \\ &= \left[\binom{9}{l} + \binom{9}{l-1}\right] + 2\left[\binom{9}{l-1} + \binom{9}{l-2}\right] + \left[\binom{9}{l-2} + \binom{9}{l-3}\right] \\ &= \binom{10}{l} + 2\binom{10}{l-1} + \binom{10}{l-2} \\ &= \binom{11}{l} + \binom{11}{l-1} = \binom{12}{l},\end{aligned}$$

satisfying Equation (2.3).

Example $(N, R) = (4, 1)$

Now consider a single root vertex and three non-root vertices, so that the possible values of connectedness, C are $0, 1, 2$ and 3. We shall use the terms forward and backward to describe edges as before, as well as using a concept of distance between vertices to define rank.

For the 0-connected basis, we proceed as before. From Figure 2.2(a), there are no forward edges and nine backward edges. Recall, all backward edges are free, i.e. it can be added or removed and not affect the connectivity and is to a vertex of less than or equal rank than from where it emanated. Thus,

$$A_4^1(0,l) = \binom{9}{l} \quad \text{for } 0 \leq l \leq 12. \tag{2.10}$$

Now suppose we add the edge $(1, 2)$ to Figure 2.2(a), giving a 1-connected basis. Clearly there is only one such basis up to to isomorphism. We can select $(1, 3)$ or $(1, 4)$ instead, hence the isomorphism coefficient is three.

(a) Only Basis for $C = 0$

(b) Basis with maximum distance for $C = 2$

Figure 2.2: Two basis digraphs up to isomorphism of four vertices with one root. Only one of the two basis digraphs for $C = 2$ is shown.

In this case there are only seven backward edges. Assume without loss of generality $(1, 2)$ is the forward edge, other choices are accounted for up to isomorphism. Then vertices 3 and 4 are not connected and have infinite rank. Hence, from vertex 2 there is only one backward edge, i.e. $(2, 1)$. There are three backward edges from both vertices 3 and 4, giving the total of seven backward edges. The edges $(2, 3)$ and $(2, 4)$ would be forward edges if added, they would alter the connectivity of the digraph making it a different basis for a higher connectedness. Thus for the 1-connected basis,

$$A_4^1(1, l) = 3\binom{7}{l-1} \quad \text{for } 0 \leq l \leq 12. \tag{2.11}$$

For 2-connectedness we must take care to define the bases correctly. In Section 2.2.1 the rank chain was defined, this sequence counts the number of vertices of a given finite rank. The total number of vertices of non-zero finite rank is the number of connected vertices. The bases can be summarised by a rank chain, which characterises the forward edges used.

To be 2-connected, both vertices can have rank one, or a single vertex of rank one then a single vertex of rank two. No other rank chains are possible, for example if both vertices are rank two, there must exist a rank one vertex which is a contradiction since

the other vertices are rank zero and infinity respectively.

The latter case is shown in Figure 2.2(b), with the rank chain $Z = (1, 1, 1, 0)$. The isomorphism coefficient is the product of binomial coefficients for choosing individuals to be of each rank. The rank one vertex has $\binom{3}{1}$ choices and the rank two vertex has $\binom{2}{1}$, giving $\binom{3}{1}\binom{2}{1} = 6$. For this basis, the number of backward edges is six: $(4, 1)$, $(4, 2)$, $(4, 3)$, $(3, 1)$, $(3, 2)$ and $(2, 1)$.

For the other 2-connected basis, with rank chain $Z = (1, 2, 0)$, we can augment Figure 2.2(a) by adding the edges $(1, 2)$ and $(1, 3)$. There are three such isomorphic digraphs having seven backward edges. We combine the possible digraphs by adding the totals from each basis. Hence,

$$A_4^1(2, l) = 6\binom{6}{l-2} + 3\binom{7}{l-2} \quad \text{for } 0 \leq l \leq 12. \tag{2.12}$$

Note in both 2-connected bases we had two forward edges, both of which were required making these minimal bases. In order to have l edges, two must be assigned as the forward edges leaving $l - 2$ edges to be chosen from among the backward edges.

Finally we consider 3-connectedness, there are five basis digraphs as shown in Figure 2.3. Figures 2.3(a), 2.3(b) and 2.3(c) correspond to the rank chains $(1, 3, 0)$, $(1, 1, 1, 1, 0)$ and $(1, 1, 2, 0)$ respectively, the basis shown in each figure is minimal and so consist entirely of required edges.

Bases 2.3(d) and 2.3(e) have the same rank chain, i.e. $(1, 2, 1, 0)$. They differ in the number of forward edges between the first and second rank. Both are basis digraphs, as they contain no backward edges. Figure 2.3(d) is the minimal basis for the rank chain $(1, 2, 1, 0)$ whereas Figure 2.3(e) is the maximal basis, it contains every potential forward edge.

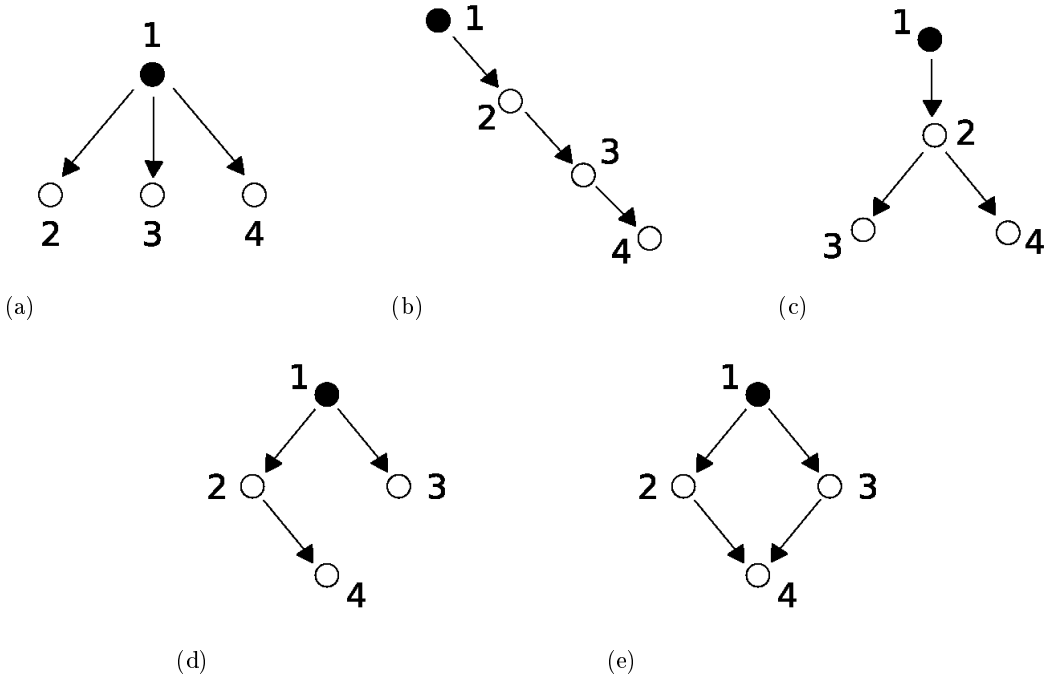


Figure 2.3: All basis digraphs up to isomorphism of four vertices with one root which are three-connected.

Given the five basis digraphs, we can count the number of backward edges on each, for the choice of forward edges write down the isomorphism coefficient and combine all the terms. For the bases 2.3(a)– 2.3(e) respectively,

$$A_4^1(3,l) = \binom{3}{3}\binom{9}{l-3} + \binom{3}{1}\binom{2}{1}\binom{1}{1}\binom{6}{l-3} + \binom{3}{1}\binom{2}{2}\binom{7}{l-3} + \binom{3}{2}\binom{2}{1}\binom{1}{1}\binom{7}{l-3} + \binom{3}{2}\binom{1}{1}\binom{7}{l-4}$$

collecting terms gives,

$$= \binom{9}{l-3} + 6\binom{6}{l-3} + 9\binom{7}{l-3} + 3\binom{7}{l-4} \quad \text{for } 0 \leq l \leq 12. \tag{2.13}$$

Again we can check Equation (2.3), that the sum for a given l is as expected,

$$\sum_{c=0}^{3} A_4^1(c,l) = \binom{12}{l} \quad \text{for } 0 \leq l \leq 12.$$

General Basis Decomposition

We now derive a general form for the decomposition of a digraph into component bases that will yield $A_N^R(C,L)$. We begin by stating several lemmas that will be required for Theorem 2.10.

For c-connectedness, excluding non-connected vertices the largest attainable rank is c. This corresponds to the rank chain $Z = (R, x_1, \ldots, x_C, 0) = (R, 1, \ldots, 1, 0)$, where $x_i = 1$ for $1 \leq i \leq c$. If we remove the zeroth rank and the terminating rank, $c+1$, then we have a sequence of integers of length k that sum to $c \in \mathbb{Z}_+$. This is a partition of c into k parts.

Lemma 2.8

For $c \in \mathbb{N}$, there are $\binom{c-1}{k-1}$ partitions of c into $1 \leq k \leq c$ parts and 2^{c-1} partitions in total.

A basis is characterised by its rank chain and the number of forward edges. A forward edge, (i,j) for vertices i,j in g, is such that vertex j has rank one greater than vertex i. By the definition of a forward edge, vertex j has a greater rank than vertex i, however by the definition of rank, if the edge (i,j) is present then the rank of vertex j can be at most one greater than that of vertex i. From the rank chain we can find the number of vertices of two consecutive ranks, then all bases on this rank chain will assign edges between the minimal and maximal number possible. The Inclusion-Exclusion Principle can be used to find the number of possible forward edges.

Lemma 2.9

Let u and x denote two consecutive ranks in a rank chain. The number of potential forward edges is ux. The number of ways to assign e edges from among the potential forward edges such that all x vertices are connected is

$$\binom{ux}{e} - \sum_{j=1}^{x} (-1)^{j-1} \binom{x}{j} \binom{u(x-j)}{e},$$

which is zero if $e > ux$ or $e < x$ (since at least x edges are needed to connect x vertices).

Proof

Denote by $\mathcal{R}$ the set of all assignments of e edges from u vertices to x vertices. Then clearly

$$|\mathcal{R}| = \binom{ux}{e}.$$

$\mathcal{R}$ may include assignments that do not connect all x vertices. Label the x vertices $1, 2, \ldots, x$ and let $\mathcal{R}_i$ denote the set of assignments where vertex i is not connected, for $1 \leq i \leq x$. The number of valid assignments, such that all x vertices are connected is,

$$|\mathcal{R}| - |\mathcal{R}_1 \cup \cdots \cup \mathcal{R}_x|.$$

Using the Inclusion-Exclusion Principle we can express the size of the union of sets as a sum of sizes of intersections,

$$|\mathcal{R}_1 \cup \cdots \cup \mathcal{R}_x| = \sum_{1 \leq i_1 \leq x} |\mathcal{R}_{i_1}| - \sum_{1 \leq i_1 < i_2 \leq x} |\mathcal{R}_{i_1} \cap \mathcal{R}_{i_2}| + \cdots + (-1)^{x-1} |\mathcal{R}_1 \cap \cdots \cap \mathcal{R}_x|.$$

It is possible to express these terms in closed form. The set $\mathcal{R}_i$ is all assignments where vertex i is not connected, which is equivalent to all the possible assignments without vertex i being present, i.e. $|\mathcal{R}_i| = \binom{u(x-1)}{e}$. The summation is equivalent to selecting

one vertex from the x to remove. Thus

$$\sum_{1\leq i_1\leq x} |\mathcal{R}_{i_1}| = \binom{x}{1}\binom{u(x-1)}{e}.$$

Similarly, for $|\mathcal{R}_{i_1} \cap \mathcal{R}_{i_2}|$ we can consider all assignments where vertices i_1 and i_2 are removed. The summation, over all i_1 and i_2 such that $1 \leq i_1 < i_2 \leq x$ is equivalent to choosing two vertices to remove from the x. Thus,

$$\sum_{1\leq i_1<i_2\leq x} |\mathcal{R}_{i_1} \cap \mathcal{R}_{i_2}| = \binom{x}{2}\binom{u(x-2)}{e}.$$

Continuing for all terms of the expansion, we obtain the following,

$$|\mathcal{R}| - |\mathcal{R}_1 \cup \cdots \cup \mathcal{R}_x| = \binom{ux}{e} - \left(\sum_{l=1}^{x}(-1)^{l-1}\binom{x}{l}\binom{u(x-l)}{e}\right).$$

For the final term in the summation, $l = x$, the assignment binomial coefficient becomes $\binom{u(x-x)}{e} = \binom{0}{e}$, which using the definition of (2.8) is zero if $e > 0$. This is consistent with our definition, as $R_1 \cap \cdots \cap R_r$ is the set of all ways to assign e edges from u vertices to no vertices (all excluded) which is zero if $e > 0$, so we can reduce the range of l to, $1 \leq l \leq x - 1$. □

Figure 2.4 shows an example of Lemma 2.9 where $u = 2$, $x = 3$ and $e = 3$. The grey boxes denote removed vertices for the inclusion-exclusion calculation. In full we have the total number of valid assignments as,

$$\begin{aligned}&\binom{2(3)}{3} - \left(\binom{3}{1}\binom{2(3-1)}{3} - \binom{3}{2}\binom{2(3-2)}{3}\right)\\ =&20 - (3(4) - 3(0)) = 8.\end{aligned}$$

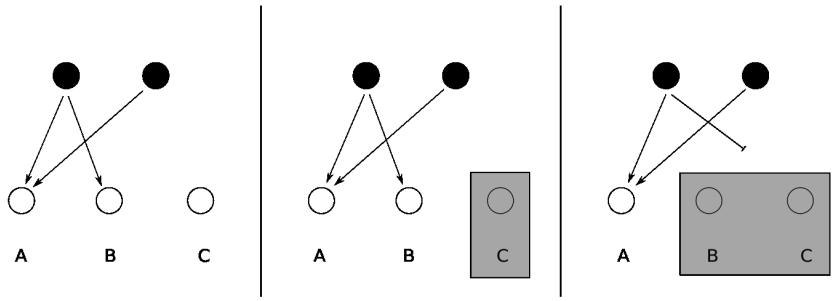


Figure 2.4: Example of Lemma 2.9 using the Inclusion-Exclusion Principle to determine the number of valid edge assignments when $u = 2$, $x = 3$ and $e = 3$. The first figure shows the case of interest, assigning edges that fail to connect the three target vertices, the second shows the number of ways to do this by excluding one of the target vertices, and the final figure excludes two vertices.

The three terms above are illustrated in Figure 2.4, firstly the unrestricted assignment, then having removed one vertex and finally removing two. It is impossible to assign three edges in the third case since there are only two potential, hence the zero for the final term. It is easy to verify the eight possible assignments explicitly.

We can now derive a general expression decomposing any C-connectedness on $g \in \mathcal{G}_N^R$ into components corresponding to basis digraphs.

Theorem 2.10 (Digraph Bases Formula)

Let $A_N^R(C,l)$ be the number of digraphs on N vertices of which R are roots with l edges such that C non-root vertices are directionally connected to at least one root vertex. Then,

$$A_N^R(C,l) = \sum_{\{Z^{(m)}:1\leq m\leq 2^{C-1}\}} \left[\prod_{i=1}^{k}\binom{N-y_{i-1}}{x_i}\right] \times$$
$$\times \sum_{e_1=x_1}^{x_0x_1} \cdots \sum_{e_k=x_k}^{x_{k-1}x_k} \binom{\sum_{j=1}^{k} x_j(y_j-1)}{l-\sum_{j=1}^{k} e_j} \left[\prod_{i=1}^{k}\binom{x_{i-1}x_i}{e_i} - \sum_{j=1}^{x_i}(-1)^{j-1}\binom{x_i}{j}\binom{x_{i-1}(x_i-j)}{e_i}\right]$$

$$0 \leq C \leq N-R \text{ and } 0 \leq l \leq N(N-1). \tag{2.14}$$

Here $Z^{(m)}$ is the mth partition of C into parts with the zeroth position set to R and a running total, i.e. $Z^{(m)} = (R, Z_1, \ldots, Z_k)$ for some $1 \leq k \leq C$, $Z_i = (x_i, y_i)$ and $y_i = \sum_{t=1}^{i} x_t$.

Proof

Firstly, we notice that all bases can be classified depending upon their rank chain, Z and the number of forward edges. Thus the first summation is over all possible rank chains. By Lemma 2.8 there are 2^{C-1} possible partitions which correspond to rank chains with suitably amended vectors (adding $x_0 = R$ and $x_{k+1} = 0$ for partitions of length k). The sum is indexed by the mth partition.

Given a rank chain, we must assign the labelled vertices to each rank. This comprises part of the isomorphism coefficient, relating to the choice of vertices. The number of ways to assign the vertices is

$$\prod_{i=1}^{k} \binom{N - y_{i-1}}{x_i},$$

where k is the length of the current partition and $y_0 = x_0 = R$, since for each rank we must choose x_i vertices from those that remain, i.e. $N - y_{i-1}$.

For a given partition, corresponding to a rank chain, bases are characterised by the number of forward edges connecting each rank with the next. Let e_i denote the number of edges assigned to connect the x_i vertices of rank i from rank $i-1$ consisting of x_{i-1} vertices. The minimal number of edges, i.e. the number of required edges is x_i. The maximal number of edges is $x_{i-1}x_i$, every possible edge from rank $i-1$ to rank i. The k summations for each rank determine the number of forward edges to assign, having the form

$$\sum_{e_1=x_1}^{x_0 x_1} \cdots \sum_{e_k=x_k}^{x_{k-1}x_k}$$

for a partition of length k. Note the amended vector, including $x_0 = R$ is required.

For the current partition, once the total number of forward edges is chosen, i.e. $\sum_{i=1}^{k} e_i$, we can write the binomial coefficient corresponding to the number of ways to assign the l less the number of forward edges among the potential backward edges. The number of backward edges from rank i is all the edges among vertices of rank i and to vertices of lower rank. There are $x_i(x_i - 1)$ edges among the rank i vertices and $x_i(y_{i-1})$ edges to vertices of lower ranks. Hence the binomial coefficient for the bases with partition Z and $e_1, e_2, \ldots, e_k$ forward edges is

$$\binom{\sum_i x_i y_{i-1} + x_i(x_i - 1)}{l - \sum_i e_i} = \binom{\sum_i x_i(y_i - 1)}{l - \sum_i e_i},$$

since by definition, $y_i = x_i + y_{i-1}$.

Lastly, the other component of the isomorphism coefficient accounts for isomorphic assignments of the forward edges. In this case, the number of ways to assign the e_i edges between each rank. Using Lemma 2.9 and taking the product over all the ranks completes the proof. □

2.3.3 Counting C-Connected Digraphs Using A Recursive Approach

Theorem 2.10 provides a constructive approach to calculating $A_N^R(C, l)$, the number of digraphs that have N labelled vertices of which R are roots and have L edges resulting in being C-connected. However, though mathematically acceptable the approach is not simple to implement as the decomposition must consider the entire problem at once. Specifically, the partitions become very large for moderate N, making explicit basis decompositions difficult to express.

Instead of determining the basis digraphs we can consider a recursive approach. Roughly speaking, by adding edges we then reduce the problem to one on a sub-digraph with fewer edges and vertices to consider. This continues until we reach a trivial assignment of edges. At each step we consider vertices of the next rank until C vertices have been connected.

Of interest is the number of digraphs characterised by the number of edges, so the number of edges l, and the connectedness C are fixed constants in the recursive formula (as well as N and R). For each rank, we track the number of edges assigned from the initial l and the number of vertices connected so far. The edges assigned at each rank will be either backward or forward in type.

For Theorem 2.10 we considered the total number of forward and backward edges to obtain the binomial coefficients, instead we shall now choose these at each rank. The two approaches are different formulations of the same problem.

For a given rank t for $0 \leq t \leq C+1$, we write k to denote the number of vertices of rank t, n to denote the number of already connected vertices (not including the k vertices of rank t) and m to denote the number of unconnected vertices.

We can relate these variables to the rank chain of the digraph, i.e. $k = x_t$, $n = y_{t-1}$ and $m = N - y_t$. The recursive approach is a way to count the number of digraphs without having to consider the whole rank chain at once, thus we shall avoid using the notation of rank chains but still use the definition of rank.

For each rank the total number of vertices is clearly constant, i.e. $k + m + n = N$. At rank t, a number of edges will be assigned (both forward and backward) from the fixed total l, leaving $\tilde{l}$ unassigned edges. When $\tilde{l} = 0$ we want to have connected C vertices to the roots, leaving $m = (N - R) - C$ unconnected.

At each rank, we calculate the number of digraphs that have k root vertices and m non-root vertices that connect $C-n-k$ of the m vertices using l edges. The l edges must emanate from the current k roots and must connect the required number of vertices from the m non-roots. The l edges may also link with the n already connected vertices.

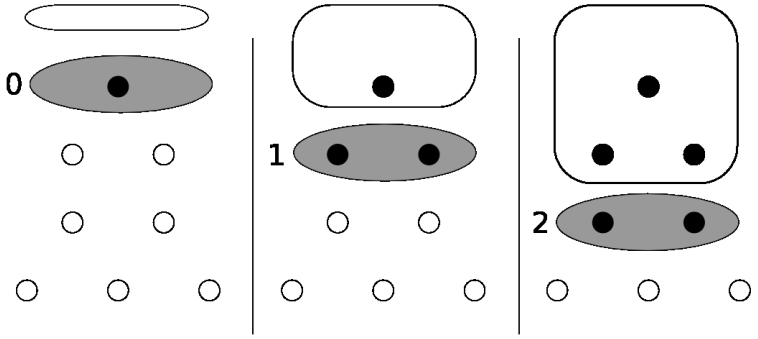
Definition 2.11

Let $B_C[k,m,n,l]$ be number of ways of assigning l edges from the k roots such that $C-n-k$ of the m non-root vertices are connected.

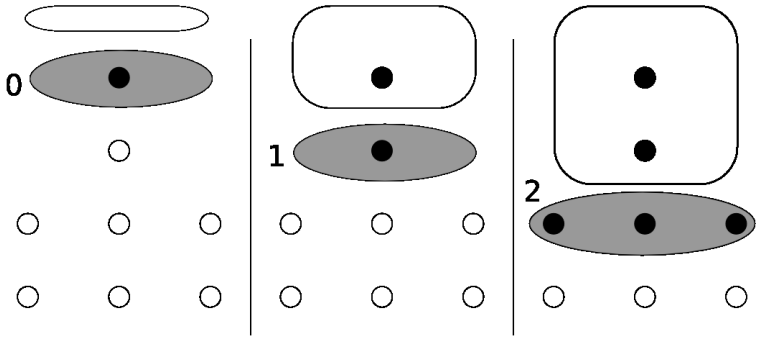
Then $A_N^R(C,l)$ is the sum of all the possibles sub-digraphs from rank 1, each of which is a sum of possible sub-digraphs from rank 2 and so on recursively, ending at a trivial sub-digraph. A trivial sub-digraph is one for which we have a closed form for $B_C[k,m,n,l]$.

To illustrate the recursive algorithm, consider Figure 2.5(a) where $N=8$ and $R=1$, with the aim to calculate $A_8^1(4,l)$ for $0 \leq l \leq N(N-1)$. The following presents the concept of the counting method without explaining the edge assignment in detail. We begin by considering the initial root vertices, i.e. rank 0. There are no previously connected vertices, so $(k,m,n)=(1,7,0)$.

For example, select two vertices to be of rank 1. Assign a number of forward and backward edges, j to connect the two vertices leaving l unassigned edges of the fixed initial number l, i.e. $\tilde{l}=l-j$. Then, we consider rank 1 to be the roots of a sub-digraph, with fewer vertices and fewer edges to assign. In this example, the rank 1 sub-digraph has two root vertices, five non-root vertices and one already connected, i.e. $(k,m,n)=(2,5,1)$. The problem is now reduced to a smaller digraph, and we continue by setting the number of rank 2 vertices as two say. These then become roots of a new sub-digraph at rank 2 where $(k,m,n)=(2,3,3)$. At this point the recursion stops, we have connected 4 vertices as specified.



(a) Example (k, m, n) Sequence: $(1, 7, 0)$, $(2, 5, 1)$, $(2, 3, 3)$



(b) Example (k, m, n) Sequence: $(1, 7, 0)$, $(1, 6, 1)$, $(3, 3, 2)$

Figure 2.5: Example recursive steps to calculate $A_8^1(4, l)$, showing the root vertices of successive sub-digraphs. The shaded group of vertices are the current root set, the unshaded group are those vertices that have already been connected.

We have not yet discussed the details of choosing the number of vertices in the next rank or the number of edges to assign, they will be discussed in the proof of Theorem 2.12.

The rank of a vertex is always defined with respect to the original digraph, thus the roots of a sub-digraph will have a non-zero rank.

Returning to our example, choose a single vertex to be of rank 1 and three vertices for rank 2 as shown in Figure 2.5(b). The diagrams for choosing one or two vertices at the rank 2 are not shown, in fact there are eight possible paths resulting in different

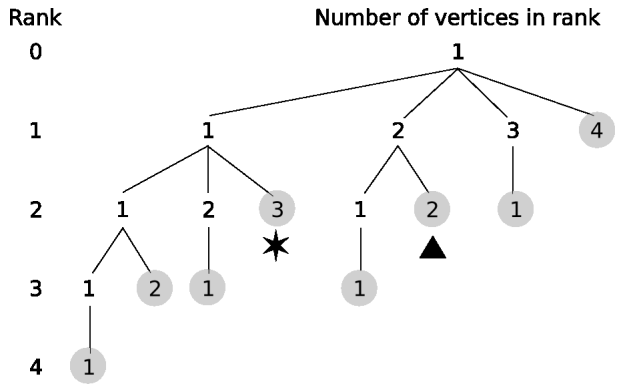


Figure 2.6: Tree diagram showing all choices of rank for calculating $A_8^1(4, L)$. The grey circles denote trivial sub-digraphs and are the end points for the recursion. The black triangle and star mark the sequences shown in Figures 2.5(a) and 2.5(b) respectively.

choices at each rank.

The example is to calculate $A_8^1(4, l)$, from Lemma 2.8 there are $2^{4-1} = 8$ corresponding rank chains. Unlike the basis approach, we do not consider all chains at once, but search them recursively by choosing the number of vertices in each rank successively. This search is shown in Figure 2.6. The grey circles denote trivial sub-digraphs, where the desired connectivity has been reached leaving only backward edges to be assigned. The choices at each rank for Figures 2.5(a) and 2.5(b) can be traced in Figure 2.6, along with all the other possibilities.

Theorem 2.12 (Digraph Recursive Formula)

Let $A_N^R(C,\tilde{l})$ be the number of digraphs on N vertices of which R are roots with $\tilde{l}$ edges such that C non-root vertices are directionally connected to at least one root vertex. Then,

$$A_N^R(C,\tilde{l}) = \begin{cases} B_C[R,N-R,0,\tilde{l}] & \text{for } \begin{array}{l} 0 \le C \le N-R, \\ C \le \tilde{l} \le N(N-1)-(C+R)((N-R)-C), \end{array} \\ 0 & \text{otherwise.} \end{cases} \tag{2.15}$$

If $m = (N-R)-C$, then

$$B_C[k,m,n,l] = \binom{m^2+(k+m)(k+n-1)}{l} \quad \text{for } \begin{array}{l} 1 \le k \le C, \\ 0 \le m \le N-R, \\ 0 \le n \le N, \\ 0 \le l \le N(N-1), \end{array} \tag{2.16}$$

otherwise,

$$B_C[k,m,n,l] = \sum_{r=1}^{m-((N-R)-C)} \binom{m}{r} \left\{ \sum_{j=\max(r,l-m(k+m+n-1))}^{\min(l,k(k+n-1+r))} B_C[r,m-r,k+n,l-j]\ \alpha(k,n,r,j) \right\}, \tag{2.17}$$

where

$$\alpha(k,n,r,j) = \sum_{w=\max(r,j-k(k+n-1))}^{\min(j,kr)} \binom{k(k-1+n)}{j-w} \left\{ \binom{kr}{w} - \sum_{l=1}^{r-1} (-1)^{l-1} \binom{r}{l} \binom{k(r-l)}{w} \right\}. \tag{2.18}$$

Proof

We obtain Equation (2.15) from the definition of $B_C[k, m, n, l]$, the sub-digraph count. The initial digraph has R roots giving $k = R$; there are $N - R$ non-root vertices which are initially unconnected, so $m = N - R$; there are no previously connected vertices initially, so $n = 0$ and we wish to assign $\tilde{l}$ edges, $l = \tilde{l}$.

The count is zero if $\tilde{l} < C$, i.e. if there are not enough edges to connect C vertices. Likewise, conditioned on C-connectedness, there are edges that cannot be assigned. In particular, there cannot be any edges from the $R+C$ vertices that are to be connected to the $(N-R)-C$ which must remain unconnected, in total $(R+C)((N-R)-C)$. Thus if the number of edges to assign is greater than the total number less these forbidden edges, i.e. $\tilde{l} > N(N-1) - (R+C)((N-R)-C)$, then some of these edges must be assigned, connecting at least one of the vertices that is to be left unconnected, violating the C-connectedness.

The stopping conditions of the recursion are the trivial sub-digraphs, which are reached once we have connected the required C vertices, i.e. when $m = (N-R) - C$. Also $l = 0$ is a stopping condition, but if this occurs before connecting C vertices the valid sub-digraph count would be zero (in fact, we shall select the sub-digraphs such that this does not occur).

When $m = N - (C + R)$ and we are conditioning on C-connectedness we do not want to connect any more vertices. Thus we must assign all remaining edges as backward edges, i.e. those that connect to a vertex of equal or lower rank. The number of ways to do this is a binomial coefficient, choosing the l edges left to assign from among all possible backward edges from the rank that triggered the stopping of the recursion.

The backward edges are the following: $m(m-1)$ edges between the m unconnected vertices; $k(k-1)$ between the k vertices of the stopping rank; $m(k+n)$ edges from

the m unconnected to the $k+n$ connected vertices (since these are directed edges they will not connect any of the m vertices to the roots) and finally the kn edges from the k vertices of the stopping rank to the previously connected vertices. Adding the totals for the four types of backward edge together gives

$$m(m-1)+k(k-1)+m(k+n)+k(n)=m^2+(k+m)(k+n-1)$$

potential backward edges, from which the l remaining edges must be assigned, giving Equation (2.16).

For $m>(N-R)-C$ there is no simple closed form for the number of sub-digraphs. However, given a (sub-)digraph with k root vertices, m unconnected vertices and n previously connected, then $B_C[k,m,n,l]$ denotes the number of such (sub-)digraphs that have l edges assigned resulting in the original digraph being C-connected. By choosing the number of vertices of the next rank we can reduce to this sub-digraph (denoted by prime) and count the number of possible digraphs of that form, i.e. $B'_C[k',m',n',l']$. Hence $B_C[k,m,n,l]$ is the sum over all possible sub-digraph counts. Denote by B and B' the digraph and a particular sub-digraph.

We shall now explain the details of Equation (2.17) (Figures 2.5 and 2.6 show an example that we shall refer to). Firstly, select the number of vertices of the next rank, k', for clarity we shall relabel this as r. If we are not at a stopping point then $r>0$ and $r\leq m-((N-R)-C))$ by the C-connectedness condition, otherwise we would connect too few or too many vertices. There are $\binom{m}{r}$ ways to choose the r vertices from the m unconnected (this is the number of unconnected vertices in the digraph B, we select the new root vertices for the sub-digraph B' from the m unconnected vertices in B).

Given the set of vertices in the next rank, it remains to select the number of edges to

assign from the current vertices to those in the next rank. These j edges will connect the k root vertices of B to the $k' = r$ root vertices of B', and may also include backward edges. If we assign j edges to link B to B' (i.e. connect the root vertices of B' to B's roots) plus backward edges, then we carry forward $l - j$ edges to the sub-digraph B', i.e. $l' = l - j$. Denote by $\alpha(k, n, r, j)$ the number of ways to assign the j edges among the potential forward and backward edges.

We have now reduced to a smaller sub-digraph by choosing $k' = r$ and $l' = l - j$, the k roots of B become part of the set of previously connected vertices thus $n' = n + k$ and since the total number of vertices is constant $m' = m - r$. Without considering the limits of indices, we have Equation (2.17) roughly as,

$$\begin{aligned} B_C[k, m, n, l] &= \sum_{k'} \binom{m}{r} \sum_{j} \alpha(k, n, k', j)\ B'_C[k', m', n', l'] \\ &= \sum_{r} \binom{m}{r} \sum_{j} \alpha(k, n, r, j)\ B_C[r, m - r, n + k, l - j]. \end{aligned}$$

The number of digraphs of the form of B is the sum of all possible sub-digraphs B' for all choices of r and j, with coefficients $\binom{m}{r}$ and $\alpha(k, n, r, j)$ accounting for the number of ways to choose the new roots and assign the edges respectively.

The range of root vertices for B', given that the sub-digraph B was not a stopping point of the recursion, i.e. Equation (2.16), is $1 \leq r \leq m - ((N - R) - C)$. The range for j is more complicated, since it must account for the number of edges required to achieve C-connectedness and the number of potential edges in subsequent sub-digraphs. It is not strictly necessary, since any specification which cannot achieve C-connectedness would have a count of zero, however specifying the range on the indices exactly leads to much faster implementation on a computer.

In the sub-digraph B' the total number of potential edges of any type is,

$$k'(k'-1)+k'(m'+n')+m'(m'-1)+m'(k'+n')=(k'+m')(k'+m'+n'-1),$$

ignoring any connectedness condition (the total edges from the k' roots and m' unconnected vertices to all other vertices, not allowing any loops). Thus l' must be less than this, otherwise there would be too many edges to assign and not enough possible places. Hence,

$$l' \leq (k'+m')(k'+m'+n'-1),$$

and substituting $l'=l-j$, $k'=r$, $m'=m-r$ and $n'=n+k$,

$$\begin{aligned} l-j &\leq (r+m-r)(r+m-r+k+n-1) \\ j &\geq l-m(k+m+n-1). \end{aligned}$$

Also, j must be at least r to connect the r vertices to the roots of B. Combining these two bounds, we have

$$j \geq \max(r, l-m(k+m+n-1)).$$

For the upper bound, j is trivially bounded above by l, the total number of edges available. However, there is also a limit to the number of edges that can be added to link B to B'. Specifically, $k(k-1)$ within the roots of B, there are $k(r)$ forward edges from the roots of B to the roots of B' and finally $k(n)$ possible edges form the roots of

B to previously connected vertices. Again, combining these two bounds we have

$$j \leq \min(l, k(k+n-1+r)),$$

which completes the derivation of Equation (2.17).

Finally we consider the coefficient $\alpha(k,n,r,j)$, the number of ways to assign the j edges linking digraph B to B'. The j edges must connect the chosen r root vertices of the next rank and they must not connect any of the other $m-r$ vertices (otherwise it would violate the choice of r).

The number of ways to assign the j edges must first determine the number of those edges that are to be forward edges, connecting the k root vertices of B to the $k'=r$ root vertices of B'. Let w be the number of forward edges assigned, leaving $j-w$ to be chosen from among the potential backward edges.

The number of potential backward edges is $k(k-1)+kn$, from the k root vertices to the k root vertices of B and to the previously connected vertices. Thus the number of ways to assign the $j-w$ is $\binom{k(k-1+n)}{j-w}$.

The forward edges are those that ensure the r vertices are connected. They are chosen from all the edges between the k and r vertices, i.e. there are kr possible edges of which we choose w. Using Lemma 2.9, the number of valid assignments is

$$\binom{kr}{w} - \left(\sum_{l=1}^{r} (-1)^{l-1} \binom{r}{l} \binom{k(r-l)}{w} \right).$$

Hence $\alpha(k,n,r,j)$ is the sum of the number of ways to assign the $j-w$ backward edges and w forward edges over all values of w.

It remains only to place limits on the range of w. The upper limit is the smaller of either the number of edges to assign on B, i.e. j and the number of potential edges, kr, giving

$$w \leq \min(j, kr).$$

The lower limits is the greater of r (since there must be at least r edges to connect the r vertices) and $j-k(k-1+n)$. The latter limit is so there are not more backward edges, $j-w$, left to assign than potential backward edges, of which there are $k(k-1+n)$, i.e. $j-w < k(k-1+n)$. Thus

$$w \geq \max(r, j-k(k+n-1)),$$

completing the derivation of Equation (2.18). □

Corollary 2.13

If $C = N - R$, that is the digraph is totally connected, then

$$A_N^R(C,l) = A_N^R(N-R,l) = \binom{N(N-1)}{l} \quad \textit{for } l > (N-1)^2. \tag{2.19}$$

Proof

There are $N(N-1)$ total potential edges on a digraph with N vertices. For a vertex to be unconnected, assuming every other vertex is connected, we must not assign any of the $N-1$ edges from a connected vertex. Hence to have a single unconnected vertex, there are $N(N-1)-(N-1) = (N-1)^2$ potential edges. If $l > (N-1)^2$ then the vertex will be connected, hence $C = N - R$. So $A_N^R(C,l) = 0$ for $C < N - R$ if $l > (N-1)^2$.

C \ L	0	1	2	3	4	5	6	7	8	9	10	11	12
0	1	9	36	84	126	126	84	36	9	1			
1		3	21	63	105	105	63	21	3				
2			9	57	153	225	195	99	27	3			
3				16	111	336	582	636	456	216	66	12	1
Sum	1	12	66	220	495	792	924	792	495	220	66	12	1

Table 2.2: Generated $A_4^1(C, L)$ values using Theorem 2.12. Zero values are left blank for clarity.

Then,

$$A_N^R(N-R, l) = \sum_{c=0}^{N-R} A_N^R(c, l) = \binom{N(N-1)}{l} \qquad \square$$

2.3.4 Numerical Examples

Theorems 2.10 and 2.12 give two different methods to compute $A_N^R(C, L)$, the number of digraphs on N vertices of which R are roots that are C-connected with L edges. The bases give an insight into the underlying structure of the partitions, as a sum of isomorphic minimal digraphs, whereas the recursive theorem provided bounds on the number of edges.

For computation, an algorithm derived from Theorem 2.12 is easier to implement, since the recursive function is simple to implement.

Table 2.2 gives the number of digraphs in each subset $\chi_N^R(c, l)$ of $\mathcal{G}_N^R$ for $N = 4$ and $R = 1$, listing all values of $A_4^1(C, L)$ for $0 \leq C \leq 3$ and $0 \leq L \leq 12$. These were computed using both methods, firstly using the basis decompositions given by Equations (2.10), (2.11), (2.12) and (2.13). Secondly the recursive algorithm was implemented, the two methods give identical results.

Table 2.2 gives the values of $A_4^1(C, L)$ for all appropriate C and L as given in Theorem

C \ L	0	1	2	3	4	5	6
0	1	4	6	4	1		
1		2	6	6	2		
2			3	10	12	6	1
Total	1	6	15	20	15	6	1

Table 2.3: Generated $A_3^1(C, L)$ values using Theorem 2.12. Zero values are left blank for clarity.

2.12. The number of subsets $\chi_N^R(C, L)$ is of order $O(N^3)$, accounting for the $N(N-1)$ edges choices for each of the $N - R$ possible connectedness properties. This does not fully account for the number of recursive iterations that are required, for larger connectedness the computation will be longer.

The bases decomposition of Theorem 2.10 also becomes computationally expensive for large N. For all connected values there are 2^{C-1} possible minimal bases, in total there are $2^{N-R+1}-1$ bases to consider (there is a single minimal basis, up to isomorphism, for each rank chain). So the computation is of order $O(2^N)$, accounting for the recursion along each rank chain.

Tables 2.3 and 2.2 demonstrate Corollary 2.13 and the bounds for l derived in Theorem 2.12. For $(N, R) = (4, 1)$ in Table 2.2 we see that for $l > (N-1)^2 = 9$ all counts are zero except for $C = 3$ (i.e. total connectedness). The connectedness that is exceeded with minimal edges is $c_f = \frac{N-2R}{2} = 1$ where $f(c_f) = f(1) = 8$, i.e. if $l > 8$ the digraph cannot be 1-connected. Similarly, the connectedness that is impossible without sufficient edges is $c_h = f(c_h) = 3$. Hence for $3 \leq l \leq 8$ all values of connectedness are possible.

The recursive algorithm was implemented in a program that computed the size of all the subsets of $\mathcal{G}_N^R$. Using the exact summation limits derived in Theorem 2.12 prevents searching zero count iterations. Also, by storing the sub-digraph counts, $B_C[k, n, m, j]$ they will be reused in the recursion. For example, consider the partition $(N, R, C, L) = (4, 1, 3, 6)$ which from $B[1, 3, 0, 6]$ will consider the sub-digraphs

R\C	0	1	2	3	4	5	6	7
1	5s	9s	19s	55s	2m51	10m14	31m33	1h41
2	8s	24s	56s	1m41	3m08	10m37	50m06	-
3	8s	50s	2m25	6m38	17m37	45m06	-	-

Table 2.4: Computation times for $A_8^R(C,\cdot)$ values using Theorem 2.12, where m and s denote minutes and seconds respectively.

$B[2,1,1,3] \to B[1,0,3,2]$ on one recursion and $B[2,1,1,4] \to B[1,0,3,2]$ on another, thus by storing $B[1,0,3,2]$ on the first pass we will not have to calculate it again. This example is rather trivial as the stored step is also a stopping condition, for larger N and C this will result in a greater improvement by storing intermediate sub-digraphs. Using these two improvements, the recursive algorithm was able to calculate the complete table of subset counts for $N < 15$ within a period of hours. For $15 \leq N \leq 18$ the algorithm required several days, larger N were not considered.

Table 2.4 lists the times to compute $A_8^R(C,\cdot)$ for pairs (R,C), calculating the counts for all $0 \leq l \leq N(N-1)$. Obviously if there are more root vertices, the maximum connectedness C is reduced (i.e. the maximum value of C, total connectedness is when all vertices are connected, $C = N - R$). The relationship between computation time and (N,R,C) is not linear, and hidden in Table 2.4 is the dependence on L for the individual calculations (whose range of valid values is also influenced by the triplet (N,R,C)).

2.4 Random Digraphs Characterised By Rank Chains

In the previous section we derived a method for computing the probability of a particular directed graph, conditioned on it being C-connected, by enumerating all possible locations of edges and counting those that were C-connected. We wish to investigate conditioned random directed graphs as a method of data imputation for inference on

epidemic processes with final size data.

The methods of Section 2.3 consider the directed edges of the graph. We have studied their properties, conditioned on a connectedness property of the digraph which will correspond to the final size of an epidemic process. If we impute the edges, we hope to make the inference of the epidemic process tractable.

During the derivation of Theorems 2.10 and 2.12 we used the concept of a rank chain to encode some information about the digraph. The rank chain includes the connectedness of the digraph, but omits details of the edges within the digraph. In this section we shall investigate the rank chain as an alternative way to encode the digraph, with the aim of using this information in the data imputation step of our inference.

Primarily, the difference is the amount of information about the digraph being stored. Section 2.3 investigates the information the edges give about the digraph. We now consider whether the rank chain provides sufficient information for our purpose.

The theorems given in Section 2.3 are valid for independent random edges, which corresponds to epidemics with fixed infectious periods. In Section 2.4.3 we will express the rank chain equivalent for fixed infectious periods, and then consider different infectious periods and rank chain models. We will then simulate rank chain representations of digraphs, which are representations of an epidemic process.

2.4.1 Rank Chain/Path Notation And Definition

In Section 2.3.3 we counted the number of C-connected digraphs by using a recursive method, which involved reducing the desired digraph into sub-digraphs with fewer vertices and edges. We will consider stepping along the rank chain in a similar recursive

manner.

The term rank is used in Ludwig (1975) to couple the final size of the epidemic process to a Markov chain, the term generation will also be used as an equivalent name. However, they are not the same in an epidemiological sense, see Pellis et al. (2008) for a discussion of the two definitions. It is possible for two individuals to be of the same rank, but different generations, if the timing of the infectious contacts is considered. Since the actual times of infectious contacts are the missing data we are attempting to avoid the need to impute, rank and generation shall be equivalent in our case.

The edge information is no longer important when considering the rank chain, there is no explicit information about the edges being recorded. We consider instead knowing only the rank of each connected vertex. It is possible to construct another recursive method to calculate the probability of a digraph being C-connected, omitting edges there are fewer variables to account for and hence fewer recursive steps. The results characterising the number of edges cannot be directly related to the rank methods, as the latter does not store the necessary information about the edges required to reconstruct the exact digraph. The rank method cannot produce output as in Table 2.2, but it can compute the rank chain probabilities.

Recall, a single rank chain corresponds to several bases in Theorem 2.10. Covering all the possibilities from the minimal basis to the maximal basis, in terms of the number of forward edges assigned at each rank. Under the rank chain method we no longer track the edges, thus we have reduced to considering the rank chain as all bases at once.

Briefly we restate the definitions given in Section 2.2.1. Let r and s denote the initial number of root and non-root vertices, with a total of $n = r + s$ vertices. Let $\mathrm{P}_{r,s}[E]$ be the probability of event E given r roots and s non-roots. Denote the rank chain as the vector $Z = (Z_1, Z_2, \dots)$ where $Z_t = (X_t, Y_t)$. The number of vertices of rank t is

X_t and the total number connected including rank t is Y_t, i.e. $Y_t = \sum_{k=0}^{t} X_t$. Since the cumulative totals, Y_t are a function of the size of each rank, we shall often write the rank chain as $Z = (X_0, X_1, X_2, \dots)$ for clarity. The number of vertices is finite and for the chain to continue there must be at least one vertex in each rank, it is sufficient to consider only ranks $0 \leq t \leq n - r + 1$. As $X_{n-r+1} = 0$, then $X_t = 0$ for all $n - r + 1 < t < \infty$. We shall use the term rank t to denote a vector Z_t or X_t, and (rank) chain to denote Z.

We will condition on the digraph being D-connected (to emphasise the difference between the edge and rank methods we shall use D instead of C). So $D = d$ corresponds to the final component of Z being $Y_{n-r+1} = r + d$ for $0 \leq r \leq n$ and $0 \leq d \leq n - r$. Let τ denote the length of each chain such that $\tau = \min\{t : X_{t+1} = 0\}$, i.e. τ is the last rank of non-zero size.

For example, the two diagrams in Figure 2.5 (p63) show two possible rank chains that connect four vertices, $d = 4$ from among seven non-root vertices, $s = 7$ with a single root vertex, $r = 1$. By Lemma 2.8 there are $2^{d-1} = 2^3 = 8$ rank chains. Figures 2.5(a) and 2.5(b) show the rank chains $Z = (1, 2, 2, 0)$ and $Z = (1, 1, 3, 0)$ respectively, both having $\tau = 2$. The remaining six rank chains can be deduced from Figure 2.6.

Relating the above to an Susceptible-Infective-Removed (SIR) epidemic process as defined in Section 1.2, the initial number of susceptibles and infectives are $S_0 = s$ and $I_0 = r$ respectively in a fixed population of size $n = S_t + I_t + R_t$ (where t denotes continuous time). The final size of an epidemic is the number of initial susceptibles that ultimately become infected, corresponding to the connectedness of the digraph, i.e. $S_0 - S_\infty = d$.

The space of all possible chains Z is a subset of $\mathbb{Z}_+^{n-r+1}$. To differentiate an epidemic process from a digraph we shall call such rank chains paths, though both rank chains

and paths are interchangeable due to the equivalence stated in Section 2.2.2. The path Z for the epidemic gives the sequence of infected individuals, terminating when there are no individuals subsequently infected. The following method will calculate the probability of a given path.

Each path consists of elements $Z_t = (x_t, y_t)$, the number of individuals infected in rank t and the total number infected so far, in the following sections we shall calculate the probability of moving from one such state to another, which we shall term the step probability.

For a given final size d, it is possible to calculate the number of possible paths. By definition $Z_0 = (a, a)$ for all paths, and the d individuals are assigned to the generations such that there are $y_\tau = a + d$. Then by Lemma 2.8, there are 2^{d-1} possible paths, which is the sum of the number of paths for each length, $1, \ldots, d$.

Note that we use τ as the length of the path, not the stopping time of the epidemic as is typical, since we are not interested in temporal data. Though not identical, there is a relation between the stopping time of an epidemic and the length of the corresponding rank chain, the latter can be used to give an approximate scale of the former.

2.4.2 Conditioned Path Probabilities

We now present an analogous set of results for paths as given for edge characterised digraphs in Section 2.3.1, noting that the two are not directly comparable.

The set of conditioned paths can be listed as the partitions of the connectedness into

integer components. Hence the probability of a path being D-connected is,

$$\mathrm{P}[D = d] = \sum_{\{z^{(m)}:1\leq m\leq 2^{d-1}\}} \mathrm{P}_{r,s}[Z = z^{(m)}],$$

where m indexes the integer partitions.

Instead of considering the entire path at once, we apply the recursive approach of Theorem 2.12. The probability of a path can be considered as the product of the probability of each rank. The probability of a rank being of a given size depends only on the size of the previous rank, corresponding to the potential forward edges. For a random digraph we know the probability of an edge, p, which we assume are all independent for the moment. Thus for a given rank, only those edges emanating from the previous rank determine the probabilities. This reduces the path into independent steps,

$$\mathrm{P}_{r,s}[Z = (z_0, z_1, \ldots, z_d)] = \prod_{t=1}^{d} \mathrm{P}_{r,s}[Z_{t+1} = z_{t+1}|Z_t = z_t]. \tag{2.20}$$

For the recursive approach we must search valid paths given the condition of being D-connected. Let $\mathcal{Z}_{+1}(u, v)$ denote the set of valid (i.e. having a non-zero probability) states at rank $t + 1$ from a given origin state $Z_t = (u, v)$, we shall call these states targets and $\mathcal{Z}_{+1}$ the target set. Then

$$\mathcal{Z}_{+1}(u, v) = \{(x, y) : 0 \leq x \leq n - v, y = v + x\}.$$

Similarly, define the origins of $Z_t = (x, y)$ to be valid $t - 1$ ranks,

$$\mathcal{Z}_{-1}(x, y) = \{(u, v) : v = y - x, 1 \leq u \leq y - x\}.$$

In general $\mathcal{Z}_i(u,v)$ is the set of states (x,y) that are i ranks further along the path than (u,v) such that $\mathrm{P}_{r,s}[Z_{t+i}=(x,y)|Z_t=(u,v)]>0$. It is not possible to express a general form for the set $\mathcal{Z}_i$ for $|i|>1$, hence we will require a recursive method to search the paths.

Since $(0,y)\in\mathcal{Z}_{+1}(x,y)$ for all x and y, in particular for $y<r+d$, i.e. the path terminates before achieving the desired connectedness, we cannot consider all individual steps without care. Instead we consider only paths that result in D-connectedness and weight the probabilities accordingly of individual steps.

If we condition on $D=d$ $(0\leq d\leq s)$, then for a general rank t $(0\leq t\leq\tau=d+1)$ we derive the probability of rank $t+1$ conditioned on the path being d-connected. For $0<t\leq\tau$: $0\leq u\leq s$, $r\leq v\leq r+s$, $0\leq x\leq s$ and $y=v+x$,

$$\begin{aligned}
&\mathrm{P}_{r,s}[Z_{t+1}=(x,y)|Z_t=(u,v),D=d]\\
&=\mathrm{P}_{r,s}[Z_{t+1}=(x,y)|Z_t=(u,v),Z_{d+1}=(0,d+r)]\\
&=\frac{\mathrm{P}_{r,s}[Z_{t+1}=(x,y)|Z_t=(u,v)]\mathrm{P}_{r,s}[Z_{d+1}=(0,d+r)|Z_{t+1}=(x,y),Z_t=(u,v)]}{\mathrm{P}_{r,s}[Z_{d+1}=(0,d+r)|Z_t=(u,v)]}\\
&=\frac{\mathrm{P}_{r,s}[Z_{t+1}=(x,y)|Z_t=(u,v)]\mathrm{P}_{x,r+s-y}[Z_{d+r-y+1}=(0,d+r-y+x)]}{\mathrm{P}_{u,r+s-v}[Z_{d+r-v+1}=(0,d+r-v+u)]}.
\end{aligned}\tag{2.21}$$

The final rearrangement is a similar idea to the recursive method of reducing to a smaller sub-digraph, i.e. given a rank t and desired d-connectedness, we can reduce to a sub-digraph on fewer vertices with $n-y_t$ vertices of which x_t are roots and consider paths of the sub-digraph. Specifically,

$$\begin{aligned}
\mathrm{P}_{r,s}[Z_{d+1}=(0,d+r)|Z_t=(x,y)]&=\mathrm{P}_{x,s-(y-r)}[D=d-(y-r)]\\
&=\mathrm{P}_{x,r+s-y}[Z_{d+r-y+1}=(0,d+r-y+x)].
\end{aligned}$$

For $t > \tau$, there are no vertices of rank $\tau + 1$ by the definition of τ, then $\mathcal{Z}_{+i}(0, y_\tau) = \{(0, y_\tau)\}$ for all $i \in \mathbb{Z}^+$. Thus it is sufficient to consider paths of length $\tau \leq d + 1$, as the longest possible path with $D = d$ has $x_t = 1$ and $y_t = t$ for $1 \leq t \leq d$.

For expression (2.21) we need $\mathrm{P}_{r,s}[Z_{d+1} = (0, d+r)]$, which is calculated using the total probability relation

$$\begin{aligned}
\mathrm{P}_{r,s}[Z_{t+1} = (x, y)] &= \\
&= \sum_{(u,v)\in\mathcal{Z}_{-1}(x,y)} \mathrm{P}_{r,s}[Z_t = (u, v)]\mathrm{P}_{r,s}[Z_{t+1} = (x, y)|Z_t = (u, v)] \\
&= \sum_{0\leq u\leq(y-x)} \mathrm{P}_{r,s}[Z_t = (u, y - x)]\mathrm{P}_{r,s}[Z_{t+1} = (x, y)|Z_t = (u, y - x)],
\end{aligned}$$

which must be applied recursively until the origin set consists of the initial conditions, i.e. $\mathcal{Z}_{-1} = \{(r, r)\}$. This backward search from rank $t+1$ can be approached in reverse, instead perform a forward search from the initial conditions and consider the paths contained in the sequence of target sets, $\mathcal{Z}_1(r, r), \mathcal{Z}_2(r, r), \ldots$ such that $(0, r + d) \in \mathcal{Z}_{d+1}(r, r)$. We shall consider methods of performing this search in Section 2.4.7.

2.4.3 Fixed Infectious Period

The equations derived in the Section 2.4.2 are independent of the form of $\mathrm{P}_{r,s}[Z_{t+1} = (x, y)|Z_t = (u, v)]$, which is the basic component required to calculated the chain probabilities. We describe the exact form of this component.

We begin in the simple case where all edges in the random digraph are independent, which corresponds to the class of epidemics with fixed infectious periods. Recall that T_I is the infectious period distribution for an infected individual. We consider the special case when $T_I = c$, a constant. We now derive the dynamics of the path Z, i.e. step

probabilities.

Denote the probability of an edge occurring as p. In terms of an epidemic, an edge corresponds to an infectious contact which occur at times given by the points of a Poisson process of rate λ/n during the infectious period T_I. Hence in the special case of fixed infectious periods, the probability of a contact is one minus that of avoiding infection, i.e. $p = 1 - \exp(-\frac{\lambda}{n}c)$.

We have that for $0 \leq t \leq \tau$

$$\begin{aligned} \mathrm{P}_{r,s}[Z_{t+1} = (x,y)|Z_t = (u,v)] = \binom{(r+s)-v}{x}(1-(1-p)^u)^x((1-p)^u)^{(r+s-v)-x} \qquad (2.22) \\ r > 0, s \geq 0 \\ 0 \leq u \leq \max(r,s), r \leq v \leq r+s \\ 0 \leq x \leq r+s-v, \max(r,v) \geq y = u+x, \end{aligned}$$

which follows from the fact that the size of the next rank is a binomial distribution given independent edges,

$$(X_{t+1}|X_t, Y_t) \sim \mathrm{Bin}(n - Y_t, 1 - (1-p)^{X_t}).$$

In other words, X_{t+1} lies between zero and the number of susceptibles remaining, $n - Y_t$, and the probability of each such susceptible being infected is one minus the probability that they avoid infection from all the infectives in rank t, which is $(1-p)^{X_t}$ by the independence.

2.4.4 General T_I Distributions

In general for arbitrary infectious period distributions, T_I, the edges of the digraph from a given vertex will not be independent. That is for a vertex i, the probability of the edges (i, j) and (i, k) $(j \neq k)$ will be dependent.

As mentioned in Section 2.4.3, the probability of a path (either conditioned or not) can be expressed in terms of the basic component

$$\mathrm{P}^{T}_{r,s}[Z_{t+1} = (x, y)|Z_t = (u, v)].$$

Here each individual has an infectious period T_I drawn from a common distribution T with (vector) parameter ϑ, $T_I \sim T$.

In general we have the following,

$$\begin{aligned}
&\mathrm{P}_{r,s}[Z_{t+1} = (x, y)|Z_t = (u, v)] \\
&\quad = \mathrm{E}_T\left[\mathrm{P}_{r,s}[Z_{t+1} = (x, y)|Z_t = (u, v), T_{i_1} + \cdots + T_{i_u} = T_t]\right]
\end{aligned}$$

where T_{i_j} is the infectious period of individual j in rank t, i_j being the individuals label.

$$\begin{aligned}
&= \mathrm{E}\left[\binom{r+s-v}{x}(1 - \exp(-\frac{\lambda}{n}T_t))^x(\exp(-\frac{\lambda}{n}T_t))^{r+s-(v+x)}\right] \\
&= \binom{r+s-v}{x}\mathrm{E}\left[\sum_{k=0}^{x}(-1)^{x-k}\binom{x}{k}\exp(-\frac{\lambda}{n}T_t(r+s-v-k))\right] \\
&= \binom{r+s-v}{x}\sum_{k=0}^{x}(-1)^{x-k}\binom{x}{k}\mathrm{E}\left[\exp(-\frac{\lambda}{n}T_t(r+s-v-k))\right]
\end{aligned}$$

by the independence of the infectious periods

$$= \binom{r+s-v}{x} \sum_{k=0}^{x} (-1)^{x-k} \binom{x}{k} \mathrm{E}\left[\exp(-\frac{\lambda}{n} T_I(r+s-v-k))\right]^u. \qquad (2.23)$$

Equation (2.23) can in principle be evaluated numerically for any T_I which has a moment generating function. In some cases a closed form solution is available, and we now briefly describe two such choices of T_I.

In the constant infectious period case, if $T_I = c$ for an individual, then for a generation of u such individuals we have,

$$\mathrm{P}_{r,s}[Z_{t+1} = (x,y) | Z_t = (u,v)] = \binom{(r+s)-v}{x} \sum_{k=0}^{x} (-1)^{x-k} \binom{x}{k} \exp\left(-\frac{\lambda}{n} c(r+s-v-k)\right)^u. \qquad (2.24)$$

It can be easily checked that Equation (2.24) is equivalent to Equation (2.22) by setting $p = 1 - \exp(-\frac{\lambda}{n} c)$.

Similarly for an exponential infectious period with rate γ, i.e. $T_I \sim \mathrm{Exp}(\gamma)$, after some manipulation we obtain,

$$\mathrm{P}_{r,s}[Z_{t+1} = (x,y) | Z_t = (u,v)] = \binom{(r+s)-v}{x} \sum_{k=0}^{x} (-1)^{x-k} \binom{x}{k} \left(\frac{\gamma}{\frac{\lambda}{n}(r+s-v-k)+\gamma}\right)^u. \qquad (2.25)$$

It is common to consider Gamma infectious periods with shape parameter α and rate β, i.e. if $T_I \sim \Gamma(\alpha, \beta)$ the T_I has probability density function

$$f_{T_I}(x) = \frac{\beta^\alpha}{\Gamma(\alpha)} x^{\alpha-1} e^{-\beta x}, \qquad x > 0,$$

where

$$\Gamma(z) = \int_{t=0}^{\infty} t^{z-1} e^{-t} \, \mathrm{d}t, \qquad z > 0.$$

For positive integer z, $\Gamma(z) = (z-1)!$. Thus, for a Gamma infectious period,

$$\mathrm{P}_{r,s}[Z_{t+1} = (x,y) | Z_t = (u,v)] = \binom{(r+s)-v}{x} \sum_{k=0}^{x} (-1)^{x-k} \binom{x}{k} \left(\frac{\beta}{\frac{\lambda}{n}(r+s-v-k)+\beta} \right)^{\alpha u}. \quad (2.26)$$

The edge counting method assumes independent edges, therefore there is no analogous result in Section 2.3.1 for Equation (2.25). It is impossible to construct such a result using only the number of edges present within a digraph due to the dependence between edges from the same vertex.

2.4.5 Step Distributions

For paths considered so far we have considered a given infectious period distribution, T_I, to determine the probability of rank $t+1$ given rank t. We have defined this to be the step probability,

$$\mathrm{P}_{r,s}[Z_{t+h} = (x,y) | Z_t = (u,v)] \quad \text{for} \quad \begin{array}{l} r,s,t,h,u,v \in \mathcal{Z}^+, \\ 0 \leq x \leq r+s-v-h-1, v \leq y \leq r+s. \end{array}$$

Since each step is independent, using the rank representation we can express the probability of a given path as the product of 1-steps.

For a given choice of infectious distribution we have a step distribution from each state

$Z_t = (u, v)$, denote this distribution by H. Specifically, define

$$H^h_{r,s,u,v,t}(x) = \mathrm{P}_{r,s}[Z_{t+h} = (x, y)|Z_t = (u, v)] \quad \text{for } 0 \leq x \leq r + s - v - h - 1,$$

so that

$$\sum_{x=0}^{r+s-v-h-1} H^h_{r,s,u,v,t}(x) = 1,$$

where the distribution depends on the initial conditions r and s, the rank t of the current state $Z_t = (u, v)$, and the step length h. The distribution H is discrete on the finite set of integers, $0, 1, \ldots, r+s-v$. Step lengths greater than one will not generally be considered, since the expressions for longer steps are more complicated and they are combinations of 1-steps; hence we shall set $H = H^1$.

Instead of focusing on the infectious period T_I, it is also possible to consider a step distribution directly. For example,

$$H_{r,s,u,v,t}(x) = \begin{cases} \frac{1}{r+s-v+1} & \text{for } 0 \leq x \leq r + s - v, \\ 0 & \text{otherwise.} \end{cases}$$

This uniform distribution on all the possible target states from $Z_t = (u, v)$ is clearly different from Equations (2.24) and (2.25), though it can be used to calculate path probabilities using the expressions in Section 2.4.2.

However, though an arbitrary step distribution is mathematically valid it is difficult to relate to an epidemic process. For the uniform example given it is not possible to find an infectious period distribution T_I such that Equation (2.23) will yield the distribution H.

Given the complex form of Equation (2.23) for an infectious period distribution where a closed form is not available, it may be advantageous to approximate using a simpler step distribution H. Depending upon the numerical properties of the full step distribution, an approximation may be more appropriate. Estimating the error between the approximate and true distributions is not easy, as the recursive method will compound any error, hence we do not consider this idea any further.

2.4.6 Summary Of The Path

Since $y_t = \sum_{k=0}^{t} x_k$, is a function of x_i, $0 \leq i \leq t$, it is sufficient to store a path as the vector $X = (X_0, \ldots, X_{s+1})$. The paths are high dimensional vectors and are difficult to adequately summarise. We shall consider the path of the average, the component wise expectation of each generation, as a summary. That is

$$\mathrm{E}[Z|D=d] = \big((r,r), (\mathrm{E}[Z_1|D=d], \mathrm{E}[Z_2|D=d]), \ldots, (\mathrm{E}[Z_d|D=d]), (0, r+d) \big),$$

where $Z_0 = (r,r)$ and $Z_{d+1} = (0, r+d)$ by definition. We treat each rank individually to obtain the following,

$$\begin{aligned} &\mathrm{E}[Z_{t+1}|Z_{n-r} = (0, d+r)] \\ &\quad = \mathrm{E}\big[\, \mathrm{E}\left[Z_{t+1}|Z_t, Z_{n-r} = (0, d+r)\right] \big] \end{aligned} \tag{2.27}$$

taking expectation conditioned on rank t

$$= \mathrm{E}\left[\sum_{(x,y) \in \mathcal{Z}_{+1}(Z_t)} (x,y)\, \mathrm{P}\left[Z_{t+1} = (x,y) | Z_t, Z_{n-r} = (0, d+r)\right] \right]$$

then taking expectation to remove the rank t conditioning

$$= \sum_{(u,v)\in\mathcal{Z}_{+t}(r,r)} \sum_{(x,y)\in\mathcal{Z}_{+1}(u,v)} (x,y) \times$$
$$\times \mathrm{P}[Z_{t+1} = (x,y) | Z_t = (u,v), Z_{n-r} = (0, d+r)] \mathrm{P}[Z_t = (u,v) | Z_{n-r} = (0, d+r)], \tag{2.28}$$

which contains terms we can calculate using the expressions in Section 2.4.2.

The summations in Equation (2.28) are not easily evaluated, the outer summation is over all states $(u,v) \in \mathcal{Z}_{+t}(r,r)$, i.e. all the states that are t ranks from rank zero ($Z_0 = (r,r)$ uniquely, if $i > 0$ then $x_i < y_i$). As previously mentioned, there is no simple description of this set, the most efficient way to obtain it is by a forward search from rank zero, recursively considering all possible paths. However, since this search is also required to compute the conditioned probabilities in Equation (2.28) there is no additional cost.

We can similarly calculate the variance of each rank as

$$\mathrm{Var}(Z_t) = \mathrm{E}[Z_t' Z_t] - \mathrm{E}[Z_t'] \mathrm{E}[Z_t],$$

where Z_t' is the transpose of Z_t. The form is as in Equation (2.28), so we do not give the expression in full.

The covariance between two ranks can also be computed in the standard way,

$$\mathrm{Cov}(Z_i, Z_j) = \mathrm{E}[Z_i' Z_j] - \mathrm{E}[Z_i'] \mathrm{E}[Z_j],$$

for ranks i and j.

2.4.7 Simulated And Exact Conditioned Path Probabilities

To obtain exact conditioned path probabilities, in terms of conditioned steps, we must first calculate the probability of the path being d-connected, where d is the connectedness we wish to condition on, i.e. $\mathrm{P}_{r,s}[D = d] = \mathrm{P}_{r,s}[Z_{d+1} = (0, r + d)]$.

In this section we discuss several methods to compute the exact conditioned step probabilities and an approximation using rejection sampling. We initially derive exact algebraic expressions by hand for small digraphs, then three exact numerical methods, either a brute force listing of all possible paths, a stepwise forward search or applying the Forward-Backward Algorithm.

2.4.7.1 Exact Algebraic

Using Section 2.4.2 it is possible to form algebraic expressions for the conditioned step probabilities. For example, if $(r, s, d) = (1, 2, 2)$ with a constant infectious period $T_I = c$ (i.e. $q = \exp(-\frac{\lambda}{r+s}c)$, the probability of avoiding an infectious contact), thus the probability of an edge is $p = 1 - q$. Using Equation (2.21) for the first rank

$$\begin{aligned}
\mathrm{P}_{1,2}[Z_1 = (0, 1)|Z_0 = (1, 1)] &= q^2 \\
\mathrm{P}_{1,2}[Z_1 = (1, 2)|Z_0 = (1, 1)] &= \binom{2}{1}[q - q^2] \\
\mathrm{P}_{1,2}[Z_1 = (2, 3)|Z_0 = (1, 1)] &= \binom{2}{2}[1 - 2q + q^2],
\end{aligned}$$

and for the subsequent ranks

$$\begin{aligned}
&\mathrm{P}_{1,2}[Z_2 = (0,2)|Z_1 = (1,2)] = q \\
&\mathrm{P}_{1,2}[Z_2 = (1,3)|Z_1 = (1,2)] = \binom{1}{1}[1-q] \\
&\mathrm{P}_{1,2}[Z_2 = (0,3)|Z_1 = (2,3)] = 1 \\
&\mathrm{P}_{1,2}[Z_3 = (0,3)|Z_2 = (1,3)] = 1.
\end{aligned}$$

These expressions combine to give,

$$\begin{aligned}
\mathrm{P}_{1,2}[Z_2 = (0,2)] &= \mathrm{P}_{1,2}[Z_2 = (0,2)|Z_1 = (1,2)] \,.\, \mathrm{P}_{1,2}[Z_1 = (1,2)|Z_0 = (1,1)] \\
&= q \,.\, \binom{2}{1}[q - q^2],
\end{aligned}$$

and

$$\begin{aligned}
&\mathrm{P}_{1,2}[Z_3 = (0,3)] \\
&= \mathrm{P}_{1,2}[Z_3 = (0,3)|Z_2 = (1,3)] \,.\, \mathrm{P}_{1,2}[Z_2 = (1,3)|Z_1 = (1,2)] \,.\, \mathrm{P}_{1,2}[Z_1 = (1,2)|Z_0 = (1,1)] \\
&\quad + \mathrm{P}_{1,2}[Z_3 = (0,3)|Z_2 = (0,3)] \,.\, \mathrm{P}_{1,2}[Z_2 = (0,3)|Z_1 = (2,3)] \,.\, \mathrm{P}_{1,2}[Z_1 = (2,3)|Z_0 = (1,1)] \\
&= 1 \,.\, (1-q) \,.\, 2q(1-q) + 1 \,.\, 1 \,.\, (1-q)^2 = (1-q)^2(2q+1).
\end{aligned}$$

The d-connectedness probabilities derived so far are not sufficient to apply Equation (2.21). The approach includes reducing to a smaller sub-digraph and considering its connectivity. Thus we also needed to compute the following,

$$\begin{aligned}
\mathrm{P}_{1,1}[D = 0] &= \mathrm{P}[Z_1 = (0,1)|Z_0 = (1,1)] = p \\
\mathrm{P}_{1,1}[D = 1] &= \mathrm{P}[Z_1 = (1,2)|Z_0 = (1,1)] = 1 - p.
\end{aligned}$$

Given the probabilities of being d-connected on the original and sub-digraphs, we can use Equation (2.21) to obtain the conditioned step probabilities. For example,

$$\begin{aligned}\mathrm{P}_{1,2}[Z_1 = (1,2)|Z_0 = (1,1), D = 2] &= \mathrm{P}_{1,2}[Z_1 = (1,2)|Z_0 = (1,1)]\,\frac{\mathrm{P}_{1,1}[D=1]}{\mathrm{P}_{1,2}[D=2]} \\ &= 2q(1-q)\,\frac{1-q}{(1-q)^2(2q+1)} = \frac{2q}{2q+1} = \frac{2(1-p)}{3-2p},\end{aligned}$$

similarly

$$\begin{aligned}&\mathrm{P}_{1,2}[Z_2 = (1,3)|Z_1 = (1,2), D = 2] = 1, \\ &\mathrm{P}_{1,2}[Z_1 = (2,3)|Z_0 = (1,1), D = 2] = \frac{1}{3-2p}, \\ &\mathrm{P}_{1,2}[Z_2 = (0,3)|Z_1 = (2,3), D = 2] = 1.\end{aligned}$$

2.4.7.2 Numerical Brute Force

Since we can enumerate all such paths as the integer partitions of d, it is possible to calculate this by brute force,

$$\mathrm{P}_{r,s}[D = d] = \sum_{\{z^{(m)}:1\le m\le 2^{d-1}\}} \mathrm{P}_{r,s}[Z = z^{(m)}].$$

Alternatively, as discussed in Section 2.4.2, we can consider an algorithm to search all possible paths that are d-connected. This avoids having to enumerate the partitions explicitly, this equivalent method is more intuitive to program as an algorithm.

Deriving the exact algebraic expressions by hand becomes difficult and the expressions become unwieldy for $d > 5$. For example, Figure 2.6 shows the eight paths (ending with grey circles) that are required to calculate $\mathrm{P}_{1,s}[D = 4]$, becoming impractical by algebraic methods.

To achieve results for any n of interest it is possible to compute these step probabilities numerically using a given value of p. Unfortunately to compare for a different edge probability p' would require recalculating all the step probabilities. For large n this repeated computation can become excessive, the balance between finding the exact algebraic expressions and computing for each p will depend upon the situation.

Using target sets $\mathcal{Z}_i(x, y)$, i.e. the possible states reachable from (x, y) in i steps, we can program an algorithm to search from the initial state, $Z_0 = (r, r)$ to compute all path probabilities.

Considering the path in steps allows some optimisation techniques that are discussed in Section 2.4.8.

2.4.7.3 Forward-Backward Algorithm For Hidden Markov Model

The Forward-Backward Algorithm (FBA) was derived by Baum as a technique to solve optimisation problems of functions of a Markov processes. The algorithm is defined in Baum et al. (1970) and Baum (1972) and applied to biological examples. The original descriptions consider probabilistic functions of a Markov process, the term Hidden Markov Model (HMM) was defined later. Hidden Markov Models have many applications and have been researched considerably with the increase of computing power. For an introduction to modern HMMs see MacDonald and Zucchini (1997). The tutorial by Rabiner (1989), intended for an audience interested in speech recognition, shall form the basis of our notation in this section.

For $t = 0, 1, \ldots$ let O_t and Q_t be the observed and hidden state at time t. We define an HMM such that there is a hidden Markov process $\{Q_t : t \geq 0\}$ and for each time t the observed state O_t follows a distribution defined by the hidden state.

Returning to the epidemic process, the steps along the path z are a Markov process on the enumerated states $S_i = (x, y)$, where x is the size of the rank and y is the running total, indexed by i. The state space is finite, as $0 \leq x \leq s$ and $r \leq y \leq r + s$. The transitions are the step probabilities between two states, where many transition probabilities will be zero. If we assume these states are unobserved, instead we observe whether the connectedness is d or not, then we can consider the epidemic process as a Hidden Markov Model and appeal to the Forward-Backward Algorithm to find the probability of observing d-connectedness.

Define L, E and G to be the events that y_t is less than $r+d$, equal to $r+d$ and greater than $r + d$ respectively. Assume the actual steps along the path are hidden, at each step we only observe one of the three events: L, E or G. In a probabilistic HMM the events would have a probability distribution dependent upon the hidden state S_i, in our case each distribution is a point mass at the appropriate event.

We can relate this to evaluating the probability of a final size as follows. If we observe the sequence, $O = (L, L, L, E, E)$, this corresponds to observing a d-connectedness path of length four, but we do not observe the exact path. The two consecutive observations of the event E, i.e. that $y_3 = d + r$ and $y_4 = d + r$, imply the path has terminated since this can only occur if $x_3 = 0$. Thus we can rephrase the probability of all paths of length four that are d connected as the probability of observing the sequence $O = (L, L, L, E, E)$.

The Forward-Backward procedure is defined as follows. Let O_i be the ith observed event and q_i be the ith hidden state, with O and Q being the complete sequences respectively. Define the set of transition probabilities from a hidden state i to another hidden state j to be the matrix $A = (a_{ij})$ and the probability of an observation given the hidden state i as $b_i(O)$ in the matrix B (accounting for all possible observed and hidden

state combinations), finally define the initial state distribution as π; let $\lambda = (A, B, \pi)$ for convenience.

For the epidemic process, the observation probabilities are point masses on the events L, E or G and the initial state will be a point mass corresponding to (r, r).

We wish to compute

$$\mathrm{P}[O] = \sum_{\text{all } Q} \mathrm{P}[O|Q]\mathrm{P}[Q|\lambda],$$

but since $\mathrm{P}[O|Q]$ is an indicator function, the hidden process will either match the observation or not, this reduces to the brute force approach in Section 2.4.7.2. The Forward-Backward Algorithm makes this calculation more efficient. Define the forward variable

$$\alpha_t(i) = \mathrm{P}[O_1, O_2, O_3, \ldots, O_t, q_t = S_i|\lambda],$$

that is, the probability of the observed partial sequence from 1 to t and the t^{th} hidden state being S_i given the parameters λ. We can calculate $\alpha_i(t)$ recursively.

We initialise as,

$$\alpha_1(i) = \pi_i b_i(O_1), \quad 1 \leq i \leq N.$$

Here there are N possible hidden states, i.e. an ordered list of the pairs (x, y) corresponding to states of the epidemic process, the initial state distribution is $\pi = (\pi_1, \ldots, \pi_N)$, which will be a point mass at the state corresponding to (r, r) in the list. Finally, $b_i(O)$ is the probability of observing O if the hidden process is in state i, which is an indicator function for state i as $S_i = (x, y)$ with conditioned d-connectedness,

then

$$b_i(L) = \begin{cases} 1 & \text{if } y < r+d \\ 0 & \text{otherwise} \end{cases} \quad b_i(E) = \begin{cases} 1 & \text{if } y = r+d \\ 0 & \text{otherwise} \end{cases} \quad b_i(G) = \begin{cases} 1 & \text{if } y > r+d \\ 0 & \text{otherwise} \end{cases}.$$

Induction on each step is performed as,

$$\alpha_{t+1}(j) = \left(\sum_{i=1}^{N} \alpha_t(i) a_{ij} \right) b_j(O_{t+1}) \qquad \begin{array}{l} 1 \leq t \leq T-1 \\ 1 \leq j \leq N, \end{array}$$

where a_{ij} is the transition probability from state i to state j from the matrix A and T is the length of the observed sequence.

The induction terminates with the final step,

$$\mathrm{P}[O|\lambda] = \sum_{i=1}^{N} \alpha_T(i).$$

By the definition of the forward variable α, we have at each step t found the probability of observing the partial sequence up to step t and being in the state S_i. The terminating step sums over all the forward variables to give the probability of the observed sequence.

The Forward procedure as stated here is not a great improvement on the brute force method, since the observable states are indicator functions and the transition matrix A contains many zeroes. In cases where the number of observable states is larger and not point mass distributions the algorithm is a great improvement over the standard forward search. If there are N hidden states and T observed states, the brute force search is of order $O(TN^T)$ where as the forward procedure is of order $O(N^2T)$.

This is only the forward part of the algorithm, the backward part can be defined in a

similar way, using a the probability

$$\beta_t(i) = \mathrm{P}[O_{t+1}, O_{t+2}, \ldots, O_T | q_t = S_i, \lambda],$$

which is the probability of the hidden state being S_i at time t given the observed states from time $t+1$ until the end of the process. This can be used to make inference about the parameters of λ, which is the problem we are motivated to solve. The techniques are described in Rabiner (1989) and MacDonald and Zucchini (1997) are Expectation Maximisation (EM). We shall consider a Bayesian approach in Chapter 3, namely MCMC.

2.4.7.4 Monte Carlo Approximations

Finally we consider a method to calculate an approximation to the conditioned step probabilities. The method of Rejection Sampling will be outlined and a comparison made with the exact methods in Section 2.4.7.5.

Given the initial condition $Z_0 = (r, r)$, corresponding to r initial infectives and using Equation (2.21) to calculate the probabilities of the next rank for all possible sizes, we sample from that distribution, then repeating for subsequent ranks, we can simulate an entire path until $x_t = 0$ for some rank t. Once a path ends its connectedness can be checked and if it matches, then it is kept, otherwise discarded, i.e. we reject samples that do not conform to the required condition.

This process will be extremely quick, since we can construct an unconditioned tree, exactly as in Figure 2.6, for each branching point we can enumerate all the branches. We need only calculate branch probabilities as they are required. Also, we can terminate samples that will fail to be d-connected before they achieve the stopping condition, i.e. if

$y_t > r + d$ for any t we do not need to continue as this path will be rejected.

However, unconditioned simulation and then checking connectedness will become inefficient for large n or for p which do not match the desired connectedness. By match we mean the probability of observing the d-connectedness given edge probability p is very low.

2.4.7.5 Comparison Of Exact And Approximate Methods

If we consider rejection sampling as another method to compute the conditioned path probabilities, then considering when it is appropriate to use each method would require further investigation. We omit such considerations since we are primarily interested in inferring the value of p given the d-connectedness condition, whereas the methods presented in Sections 2.4.7.1, 2.4.7.3 and 2.4.7.4 calculate the path probabilities given p. We are investigating the behaviour of these path probabilities to gain insight into the Markov Chain Monte Carlo (MCMC) algorithms presented in Chapters 3 and 4.

Two programs were implemented in the C programming language, the first initially performs a complete forward search from rank zero of all possible paths and computes the conditioned step probabilities which depend upon the infectious period specified; from this paths could be simulated if desired but the exact probabilities are available (for the examples considered the forward step search was used, though the brute force method and Forward-Backward Algorithm give identical results). The second program simulated paths unconditioned, using the step distribution specified, from which only those achieving the desired connectedness were stored. The rejection sampling is based on a fixed step distribution, hence there is no inference made directly for a given set of parameters.

For $(r, s) = (1, 20)$ the first program took two days to compute the probabilities of all possible paths for all degrees of connectedness. Clearly, the number of paths for $d = 1$ is far fewer $d = 10$, hence the time to compute all paths for large and small connectedness is relatively quick. Over half the time was spent computing paths of length eight to twelve. Part of the issue is numerical accuracy, the probability of any given path decreases for large numbers of vertices. So for larger digraphs, there are more paths to consider and each path's probability takes longer to compute to sufficient accuracy (see the discussion of GNU MPFR in Section 3.7.2). Exact results, such that the sum of all non-zero probability paths equalled one, required accuracy to over thirty decimal places; reducing the precision dramatically reduces computation time at the cost of the total sum of probabilities not being one. For $(r, s) = (1, 9)$ computing the probabilities was completed in fifteen minutes. Note that, as a consequence of the complete forward search we have the probabilities of any connectedness $0 \leq d \leq s$.

Table 2.5 compared the path of the average using the exact probabilities and the rejection sampling method for $(r, s, d) = (1, 2, 2)$. The expected exact path was calculated using the expressions derived in Section 2.4.7.1. The simulated values, $\hat{X}$ are the average of the accepted runs for each rank. For each simulation K runs are performed and K_α are accepted, the kth simulated path is denoted $Z^{(k)}$. Then

$$\mathrm{E}[\hat{X}_t] = \frac{\sum_{k=1}^{K} Z_t^{(k)} \, \mathbb{I}_{\{y_\tau^{(k)} = d+r\}}}{\sum_{k=1}^{K} \mathbb{I}_{\{y_\tau^{(k)} = d+r\}}} == \frac{\sum_{k=1}^{K} Z_t^{(k)} \, \mathbb{I}_{\{y_\tau^{(k)} = d+r\}}}{K_\alpha}.$$

Since the zeroth rank is fixed, $\hat{X}_0 = X_0$ and given the $D = d$ condition, $\hat{Y}_{d+1} = Y_{d+1}$ for all simulated paths that are accepted, hence in the table $\hat{Y}_2 = 3$ exactly.

Table 2.5 demonstrates the issue with rejection sampling if the event observed is very unlikely. Even for this small example, the number of runs required to obtain estimates

	Theoretical				Simulated				
p	X_0	$\mathrm{E}[X_1]$	$\mathrm{E}[X_2]$	Y_2	$\mathrm{E}[\hat{X}_1]$	$\mathrm{E}[\hat{X}_2]$	$\hat{Y}_2$	K	K_α/K
1.0	1	2.0	0.0	3	2.0	0.0	3	10^4	1.000
0.6	1	$1.\overline{5}$	$0.\overline{4}$	3	1.55848	0.44152	3	10^4	0.6315
0.4	1	$1.\overline{45}$	$0.\overline{54}$	3	1.4583	0.5417	3	10^5	0.3502
0.01	1	$1.3355\ldots$	$0.6644\ldots$	3	1.3348	0.6652	3	10^7	2.978e−4

Table 2.5: Comparison of the path of the average for $(r, s, d) = (1, 2, 2)$ between the exact probabilities and rejection sampling for various edge probabilities given independent edges.

that are close to the exact values is growing in orders of magnitude for increasingly unlikely edge probabilities. For the example presented, the simulations were completed in under a minute, we use this example only to illustrate the problem of the acceptance rate K_α/K.

2.4.8 Algorithm Implementation And Optimisation

The recursive method presented in Section 2.4.2 considers each rank of the path by reducing the problem to small paths. The states that form each path Z are used to calculate step probabilities, which combine to give the path probability (each rank is independent, thus the path probability is the product of step probabilities).

The calculation of the path probabilities can be greatly optimised by noting two observations. First, the step probability distribution may be independent of rank. This is true for the examples given so far: constant and exponential infectious periods, though this does not have to be the case in general. Specifically,

$$\mathrm{P}_{r,s}\left[Z_{t+1} = (x, y) | Z_t = (u, v)\right] = \mathrm{P}_{r,s}\left[Z_1 = (x, y) | Z_0 = (u, v)\right] \quad \forall t \geq 0. \tag{2.29}$$

Thus storing the computed step probabilities will reduce the number of calculations required.

Secondly, storing intermediate probabilities for the probability of a given rank being a given size will reduce duplicate calculations. Denote these probabilities by $q_t(x, y)$, where

$$q_t(x,y) = \mathrm{P}_{r,s}[Z_t = (x,y) | Z_{d+1} = (0,d), Z_0 = (r,r)] \qquad \begin{array}{l} 0 \le t \le d+1, \\ 0 \le x \le d+1-t, \\ t \le y \le d. \end{array}$$

When calculating the probability of a path, first check if the probability of reaching any part of the path has already been calculated. Then,

$$\begin{aligned} &\mathrm{P}[Z_0 = z_0, Z_1 = z_1, \ldots, Z_t = z_t, Z_{t+1} = z_{t+1}, \ldots, Z_s = z_s] \\ &\quad = q_t(x,y)\mathrm{P}[Z_{t+1} = z_{t+1}, \ldots, Z_s = z_s]. \end{aligned}$$

If using a rejection simulation approach, these intermediate probabilities will be approximations to the true values after K simulations, i.e. $\hat{q}_t^{(K)}(x, y)$, which can be used to estimate the summary path. In the limit,

$$\hat{q}_t^{(K)}(x,y) \to q_t(x,y) \quad \text{as } K \to \infty.$$

The intermediate probabilities, $q_t(x, y)$ are comparable to the forward and backward variables of the Forward-Backward Algorithm described in Section 2.4.7.3.

Finally, there are practical issues relating to the implementation of the algorithms. In particular, for large d the number of paths is large and the probability of an individual path becomes very small. As the probability become small, it will cause buffer underflow on a computer, i.e. the numerical value will be smaller than the minimum the computer can store. There are two approaches to solve this problem, the first is to use

high precision code. We shall discuss this in more depth in Section 3.7.2, though the principle is to increase the accuracy by storing more decimal places in the calculations.

The second approach is to restructure the problem to make computation easier, the optimisations discussed can be considered part of this restructuring. Though mathematically identical, re-writing a problem can greatly affect computation. It may reduce the number of calculations required, as the Forward-Backward Algorithm does or it may make the calculations for numerically stable, see Section 3.7.1 for more details. For the Forward-Backward Algorithm a technique of scaling the probabilities can be used, see Devijver (1985), which attempts to prevent underflow by scaling the probabilities up such that the normalising constant will cancel.

It will be more beneficial to restructure a problem than to simply increase the computational precision, though in some cases this may be the only option. Also, restructuring for numerical stability may involve more calculations, so there is a trade off between higher precision and complexity.

2.4.9 Dependence Of The Number Of Additional Non-root Vertices On Conditioned Probabilities

We have considered several step distributions in previous sections, either derived from a given infectious distribution (determining edge probabilities) or from a specified distribution (without any comparable infectious distribution). For all the step distributions we can condition on a connectedness to derive a new distribution, $H|D = d$, using the expressions in Section 2.4.2 (H is defined in Section 2.4.5).

In the special case when edges from a vertex are independent of every other edge, the digraph can be reduced to a sub-digraph consisting of only the vertices that are to be

connected. Since the edges are independent, every connected vertex will independently avoid connecting with the additional non-connected vertices. These two independent events that can be treated separately.

Following from Section 2.4.7, again letting $p = 1 - \exp(-\frac{\lambda c}{n})$ we can express the probability of a path that we will condition to be d-connected on a digraph with r roots and $d + s$ non-root vertices in terms of a smaller digraph (note we use s to represent the number of additional non-root vertices, i.e. $s \geq 0$). The sub-digraph consisting of only those vertices that are connected, i.e. r roots and d non-roots.

$$\mathrm{P}_{r,d+s}[Z = z] = \mathrm{P}[r + d \text{ vertices do not connect to the } s \text{ vertices}]\mathrm{P}_{r,d}[Z = z]$$
$$\mathrm{P}_{r,d+s}[Z = z] = \binom{d+s}{s}(p^{r+d})^s \mathrm{P}_{r,d}[Z = z].$$

The binomial coefficient accounts for selecting which of the non-root vertices are to be connected. The next term accounts for all the edges from the $d + r$ connected vertices that must not connect to the s vertices. Finally the probability of the sub-digraph. This is only possible for independent edges (and in this case all edges are independent and identically Bernoulli trials).

Since the path is composed of independent steps, we can expand the expressions for

the digraph and sub-digraph and equate terms as follows,

$$\begin{aligned}
\mathrm{P}_{r,d+s}[Z=z] &= \prod_{t=0}^{d} \mathrm{P}_{r,d+s}[Z_{t+1}=z_{t+1}|Z_t=z_t] \\
&= \prod_{t=0}^{d} \binom{r+d+s-y_t}{x_{t+1}} (p^{x_t})^{x_{t+1}} (1-p^{x_t})^{r+d+s-y_t-x_{t+1}} \qquad (2.30) \\
\mathrm{P}_{r,d}[Z=z] &= \prod_{t=0}^{d} \mathrm{P}_{r,d}[Z_{t+1}=z_{t+1}|Z_t=z_t] \\
&= \prod_{t=0}^{d} \binom{r+d-y_t}{x_{t+1}} (p^{x_t})^{x_{t+1}} (1-p^{x_t})^{r+d-y_t-x_{t+1}}. \qquad (2.31)
\end{aligned}$$

Recall that the generations are related by,

$$y_{t+1} = y_t + x_{t+1} \qquad \text{and} \quad \sum_{i=0}^{d} x_i = y_d = r+d,$$

then equating Equations (2.30) and (2.31) we have,

$$\mathrm{P}_{r,d+s}[Z_{t+1}|Z_t] = (p^{x_t})^s \, \frac{r+d-y_t+1}{r+d-y_{t+1}+1} \cdots \frac{r+d-y_t+s}{r+d-y_{t+1}+s} \, \mathrm{P}_{r,d}[Z_{t+1}|Z_t],$$

for $s > 0$, which combine to give the probability of a path Z as

$$\begin{aligned}
\mathrm{P}_{r,d+s}[Z] &= (p^{r+d})^s \, \prod \frac{r+d-y_t+1}{r+d-y_{t+1}+1} \cdots \frac{r+d-y_t+s}{r+d-y_{t+1}+s} \, \mathrm{P}_{r,d}[Z_{t+1}|Z_t]. \\
&= (p^{r+d})^s \, \frac{r+d-y_0+1}{r+d-y_{d+1}+1} \cdots \frac{r+d-y_0+s}{r+d-y_{d+1}+s} \, \prod \mathrm{P}_{r,d}[Z_{t+1}|Z_t],
\end{aligned}$$

and since the sub-digraph is to be totally connected, i.e. $y_0 = r$ and $y_{d+1} = d + r$,

$$\begin{aligned}
\mathrm{P}_{r,d+s}[Z = z] &= (p^{r+d})^s \, \frac{d+1}{1} \ldots \frac{d+s}{s} \prod \mathrm{P}_{r,d}[Z_{t+1} = z_{t+1} | Z_t = z_t] \\
&= (p^{r+d})^s \binom{d+s}{s} \prod \mathrm{P}_{r,d}[Z_{t+1} = z_{t+1} | Z_t = z_t] \\
&= (p^{r+d})^s \binom{d+s}{s} \prod \binom{r+d-y_t}{x_{t+1}} (p^{x_t})^{x_{t+1}} (1-p^{x_t})^{r+d-y_t-x_{t+1}}.
\end{aligned}$$

This is the relationship between the path probabilities on the digraph and sub-digraph. The product of binomial coefficients can be combined,

$$\mathrm{P}_{r,d+s}[Z = z] = (p^{r+d})^s \binom{d+s}{s} \binom{d}{x_1, x_2, \ldots, x_d} \prod (p^{x_t})^{x_{t+1}} (1-p^{x_t})^{r+d-y_t-x_{t+1}}, \tag{2.32}$$

where the multinomial coefficient is defined as,

$$\binom{n}{k_1, k_2, \ldots, k_m} = \frac{n!}{k_1!\, k_2! \ldots k_m!} \quad \text{where} \quad \sum_{i=1}^{m} k_i = n.$$

Apart from the implicit dependence of λ on the number of non-root vertices, Equation (2.32) shows that for edges that are independent Bernoulli trials with probability p of an edge being absent, then the number of non-root vertices can be reduced to only those that are to be connected and the probabilities adjusted for any additional non-root vertices.

In terms of the conditioned probabilities, the number of additional non-root vertices is conditioned out (except the implicit dependence). Since,

$$\mathrm{P}_{r,d+s}[Z = z] = \binom{d+s}{s} (p^{r+d})^s \mathrm{P}_{r,d}[Z = z]$$

and

$$\mathrm{P}_{r,d+s}[Z_{d+1} = (0, r+d)] = \sum_{\{z : y_d = y_{d+1} = r+d\}} \mathrm{P}_{r,d+s}[Z = z].$$

Then applying these relations to the conditional step expression,

$$\begin{aligned}
&\mathrm{P}_{r,d+s}[Z_{t+1} = (x,y) | Z_t = (u,v), Z_{d+1} = (0, d+r)] = \\
&\quad = \frac{\mathrm{P}_{r,d+s}[Z_{t+1} = (x,y) | Z_t = (u,v)] \mathrm{P}_{x,r+d+s-y}[Z_{d+r-y+1} = (0, d+r-y+x)]}{\mathrm{P}_{u,r+d+s-v}[Z_{d+r-v+1} = (0, d+r-v+u)]} \\
&\quad = (p^u)^s \frac{(r+d-v+1)\dots(r+d-v+s)}{(r+d-y+1)\dots(r+d-y+s)} \mathrm{P}_{r,d}[Z_{t+1} = (x,y) | Z_t = (u,v)] \times \\
&\qquad \times \frac{\sum \binom{r+d-y+s}{s} (p^{x+r+d-y})^s \mathrm{P}_{x,r+d-y}[Z]}{\sum \binom{r+d-v+s}{s} (p^{u+r+d-v})^s \mathrm{P}_{u,r+d-v}[Z]} \\
&\quad = (p^u)^s \frac{(r+d-v+1)\dots(r+d-v+s)}{(r+d-y+1)\dots(r+d-y+s)} \frac{\binom{r+d-y+s}{s} (p^{x+r+d-y})^s}{\binom{r+d-v+s}{s} (p^{u+r+d-v})^s} \times \\
&\qquad \times \mathrm{P}_{r,d}[Z_{t+1} = (x,y) | Z_t = (u,v), Z_{d+1} = (0, d+r)] \\
&\quad = \mathrm{P}_{r,d}[Z_{t+1} = (x,y) | Z_t = (u,v), Z_{d+1} = (0, d+r)].
\end{aligned}$$

Cancelling the binomial coefficients and recalling that $x + v = y$, we see that the additional non-root vertices, $s \geq 0$, have no effect on the conditioned step probabilities.

This is not true if the edges are not independent and identically distributed, for example an exponential infectious period or the uniform step distribution.

2.5 Branching Process Conditioned On Total Progeny

As descibed in Ball (1983), the early stages of an epidemic process can be approximated by a suitable constructed branching process, defined in Section 2.5.2. For minor outbreaks, the entire process may be approximated by a branching process. This motivates us to investigate branching processes conditioned on their total progeny, i.e. the total number of offspring not including the initial ancestors. For a thorough background on branching processes see Mode (1971) and Jagers (1975), both derive many key results for continuous and discrete processes.

In Section 2.5.1 we present the Galton-Watson Process and define notation analogous to the random digraph. Using an expression for the probability of a given progeny in Section 2.5.3 we then derive four example step distributions in Section 2.5.4.

The example distributions are investigated numerically in Section 2.5.6. Finally, the approximation is related to the epidemic and digraph.

2.5.1 Branching Process

We shall consider only the case of discrete time generations, commonly referred to as a Galton-Watson process. The theorems of Ball (1983) require continuous time branching processes, see Jagers (1975) for further details.

A Galton-Watson Process is a stochastic process, $\{X_t : t \geq 0\}$ which obeys the recurrence relation,

$$X_0 = 1 \qquad \text{and} \qquad X_{t+1} = \sum_{j=1}^{X_t} \xi_j^{(t)},$$

where $\left(\xi_j^{(t)}\right)$ is a sequence of independent and identically distributed random variables on $\mathbb{Z}_+$.

The branching process will be used as an approximation to the rank chains on a digraph as presented in Section 2.4. Thus we define an analogous set of notation, let $W = (W_0, W_1, \dots)$ denote a sequence of generation sizes of a branching process. Each individual, i in generation, t, has a number of offspring according to a given distribution; denote by $\xi_t^{(i)}$ the random variable of the number of offspring. Let a be the number of initial ancestors of the branching process, which is equivalent to a independent copies of the branching process. As for the random digraph, let $W_t = (x_t, y_t)$, where x_t is the number of individuals in generation t and $y_t = \sum_{i \le t} x_i$.

Similarly, let $\mathrm{P}_a[\mathrm{E}]$ be the probability of an event E in a branching process with a initial ancestors. Let T denote the total progeny of the branching process, i.e. the total number of individuals ever born, excluding the initial ancestors, when the process becomes extinct. We will be interested in the behaviour of the branching process conditioned upon $T = k$, for some fixed k.

For unconditioned branching processes, the extinction probability is defined as,

$$P_{\text{ext}} = \lim_{n \to \infty} \mathrm{P}[X_n = 0].$$

If $\mathrm{E}[\xi_1] \le 1$ then $P_{\text{ext}} = 1$, otherwise if $\mathrm{E}[\xi_1] > 1$ then $0 \le \theta < 1$.

Conditioning on a specific finite total progeny requires the process to become extinct once that progeny has been reached, i.e. $P_{\text{ext}} = 1$ regardless of the offspring distribution.

The sequence of generation sizes is directly analogous to the rank chains of the random digraph, we shall call both paths. As for the digraph we consider step probabilities to

describe the branching process. Each step is from one generation to the next, each of which is independent given the initial size of the starting generation.

We derive a set of expressions as in Section 2.4.2,

$$\begin{aligned}
&\mathrm{P}_a[W_{t+1} = (x, y)|W_t = (u, v), T = k] \\
=&\frac{\mathrm{P}_a[W_{t+1} = (x, y)|W_t = (u, v)] \quad \mathrm{P}_a[T = k|W_{t+1} = (x, y), W_t = (u, v)]}{\mathrm{P}_a[T = k|W_t = (u, v)]} \\
=&\mathrm{P}_a[W_{t+1} = (x, y)|W_t = (u, v)]\frac{\mathrm{P}_x[T = k + a - y]}{\mathrm{P}_u[T = k + a - v]} \quad \text{for } a, k, u, v, x, y, t \in \mathbb{Z}_+.
\end{aligned} \tag{2.33}$$

Recall the total progeny does not include the initial ancestors (as the connectedness does not include the initial root vertices). Hence the event $\{T = k\}$ is equivalent to $W_{k+1} = (0, k+a)$, by the same reasoning as for the digraph. This reduces to considering the step probabilities,

$$\mathrm{P}_a[W_{t+1} = (x, y)|W_t = (u, v)].$$

2.5.2 Epidemic Model And Its Branching Process Approximation

We appeal to the results of Ball (1983) and Ball and Donnelly (1995). We shall briefly explain the approximation of a closed population stochastic epidemic by a branching process described in these papers, omitting the proofs of any results.

We match an epidemic to a branching process as described in Ball (1983), namely the initial a ancestors of the branching process are matched to the r initial infectives of the epidemic. The epidemic process is as defined in Section 1.2.2, denoted $Z_N(\omega)$ and

a continuous time branching process, denoted $Z(\omega)$, with a given offspring distribution (ω is a member Ω from a suitably constructed probability space). Each new offspring is matched to a labelled individual in the population. If the individual was already infected, then the offspring is a ghost, it and all of its subsequent offspring are ignored. We shall now present several theorems that, given this construction, prove the correspondence between the two processes. Let $\tilde{Z}_N(\omega)$ be the epidemic process relabelled and $Z_N(\omega, t)$, $\tilde{Z}_N(\omega, t)$ and $Z(\omega, t)$ be the epidemic process, relabelled process and branching process restricted to the interval $[0, t]$.

Theorem 2.14 (Ball (1983) Theorem 3)

For any fixed $t > 0$ and any metric on the space of sample paths $\{Z(\omega, t), \omega \in \Omega\}$,

$$\tilde{Z}_N(\omega, t) \xrightarrow{\text{a.s.}} Z(\omega, t) \qquad \text{as } N \to \infty.$$

Thus the epidemic process, as the population size tends to infinity, converges to the corresponding branching process. This result is also known as Kendall's approximation and was stated in Kendall (1956).

Let $T_N(\omega)$ and $T(\omega)$ be the number of individuals born in $Z_N(\omega)$ and $Z(\omega)$ respectively, for all $\omega \in \Omega$ (as defined in Ball (1983)). Then,

Theorem 2.15 (Ball (1983) Theorem 4)

$$T_N(\omega) \xrightarrow{\text{a.s.}} T(\omega) \qquad \text{as } N \to \infty.$$

The final size of the epidemic converges to the total progeny of the branching process,

note that $T(\omega)$ may be infinite.

Theorem 2.14 is presented again in Ball and Donnelly (1995) using similar notation and further details of the approximation.

Theorem 2.16 (Ball and Donnelly (1995) Theorem 2.1)

There is a probability space $(\Omega, \mathcal{F}, \mathbb{P})$ on which are defined a sequence of epidemic models indexed by N (the initial number of susceptibles) and the approximating branching process, with the following properties.

Denote by A the set on which the branching process $Z(\cdot)$ becomes extinct, $A = \{\omega \in \Omega : \lim_{t\to\infty} Z(\omega, t) = 0\}$. Then, as $N \to \infty$,

$$\sup_{0\le t<\infty} |Z_N(t) - Z(t)| \to 0 \qquad \text{for } \mathbb{P}-\text{almost all } \omega \in A.$$

Further, for any $c_1 < (2\alpha)^{-1}$ and $c_2 > (2\alpha)^{-1}$, as $N \to \infty$,

$$\sup_{0\le t\le c_1 \log N} |Z_N(t) - Z(t)| \to 0 \quad \text{and} \quad \sup_{0\le t\le c_2 \log N} |Z_N(t) - Z(t)| \to \infty,$$

for $\mathbb{P}-$almost all $\omega \in \Omega\backslash A$.

Thus for a minor epidemic, for sufficiently large N, the process behaves like a branching process. For major outbreaks, where the corresponding branching process does not go extinct, the epidemic grows like a branching process until about $\sqrt{N}$ individuals have been infected.

We are interested in the total progeny approximation of the final size, which for sufficiently large population size N approximates the final size of the corresponding epidemic by Theorem 2.15. From Ball (1986), if the infectious period is a constant, i.e. $T_i = c$ then a branching process with a Poisson offspring distribution approximates such an epidemic. Since during an individual's infectious period they have a contacts at

the times of a Poisson process. All contacts result in an infection under the branching process approximation, thus the number of offspring must be the count of the Poisson process, i.e. a Poisson distribution of rate λc. For a Negative binomial offspring distribution, the corresponding epidemic process has gamma infectious periods, with integer value shape parameter.

We shall consider these two cases in Section 2.5.4, giving total progeny probabilities as in Ball et al. (2002). We also consider two other offspring distributions that do not match an explicit infectious period.

2.5.3 Conditioned Probabilities Of An Entire Path

As for the digraph, there are 2^{k-1} possible sequences of generations that result in a total progeny of k. So to calculate the conditioned probabilities we must again either calculate them exactly or use rejection sampling to generate approximate solutions.

For the digraph, to obtain an exact probability required all the possible 2^{d-1} paths, in terms of the branching process then

$$\mathrm{P}_a[T=k] \;=\; \sum_{\{w|T=k\}} \mathrm{P}[W=w] \;=\; \sum_{\{w|T=k\}} \prod_{0\le t\le \tau} \mathrm{P}[W_{t+1}=w_{t+1}|W_t=w_t].$$

However, for a Galton-Watson Process we can use Theorem 2.17, presented by Dwass (1969) in terms of branching processes based on Good (1949).

Theorem 2.17 (Dwass (1969))

For a simple Galton-Watson Process we have the following relation,

$$\mathrm{P}_a[T=k] = \frac{a}{k+a}\mathrm{P}[\xi_1+\xi_2+\cdots+\xi_{k+a}=k] \quad (k=0,1,\dots) \tag{2.34}$$

where $\sum_{k\geq 0} \mathrm{P}_a[T=k] = 1$.

Instead of calculating the total probability of all 2^{k-1} paths, we now need only calculate a single probability, namely $\mathrm{P}[\xi_1+\xi_2+\cdots+\xi_{k+a}=k]$. Theorem 2.17 is of particular use when the right hand side can be evaluated analytically for a given offspring distribution, ξ.

In the following section we present four example offspring distributions that yield closed forms for Equation (2.34), these will be related to their corresponding epidemic process in Section 2.5.6.

2.5.4 Example Offspring Distributions With Algebraic Conditioned Probabilities

Poisson Distribution

Each individual has a random number of offspring with a Poisson distribution, i.e. $\xi \sim \mathrm{Pois}(\lambda)$. Let $S_n = \xi_1 + \cdots + \xi_n$, then $S_n \sim \mathrm{Pois}(\lambda n)$. Hence using Theorem 2.17 we have,

$$\begin{aligned}
\mathrm{P}_a[T=k] &= \frac{a}{k+a}\mathrm{P}[\xi_1+\cdots+\xi_{k+a}=k] \\
&= \frac{a}{k+a}\mathrm{P}[S_{k+a}=k] \\
&= \frac{a}{k+a}\frac{(\lambda(k+a))^k}{k!}\exp(-\lambda(k+a)) \qquad a>0, k\geq 0.
\end{aligned}$$

Given Poisson offspring, the step probabilities are given by

$$\mathrm{P}_a[W_{i+1}=(x,y)|W_i=(u,v)] = \mathrm{P}[S_u=x] = \frac{(\lambda u)^x}{x!}\exp(-\lambda u).$$

Combining these expressions as in Equation (2.33) gives,

$$\begin{aligned}
&\mathrm{P}_a\left[W_{i+1}=(x,y)|W_i=(u,v),T=k\right]= \\
=&\mathrm{P}_a[W_{i+1}=(x,y)|W_i=(u,v)]\frac{\mathrm{P}_x[T=k+a-y]}{\mathrm{P}_u[T=k+a-v]} \\
=&\frac{(\lambda u)^x}{x!}\exp(-\lambda u)\frac{\frac{x}{k+a-y+x}\frac{(\lambda(k+a-y+x))^{k+a-y}}{(k+a-y)!}\exp(-\lambda(k+a-y+x))}{\frac{u}{k+a-v+u}\frac{(\lambda(k+a-v+u))^{k+a-v}}{(k+a-v)!}\exp(-\lambda(k+a-v+u))} \\
=&\left[\binom{x}{u}\left(\frac{k+a+u-v}{k+a+x-y}\right)\right]\binom{k+a-v}{x}\left(\frac{u}{k+a+x-y}\right)^x\left(\frac{k+a+x-y}{k+a+u-v}\right)^{k+a-v}.
\end{aligned}$$

It is interesting to note that the rate of the Poisson offspring distribution, λ, does not appear in the conditioned step probability for any step. We will return to this observation in Section 2.5.5.

Geometric Distribution

For a geometric distribution, $\xi \sim \mathrm{Geo}(p)$, we note that the sum of n geometric random variables is a negative binomial, i.e. $S_n \sim \mathrm{NegBin}(n,p)$.

$$\mathrm{P}[\xi=x]=p(1-p)^x \quad \text{and} \quad \mathrm{P}[\xi_1+\cdots+\xi_n=S_n=k]=\binom{k+n-1}{k}p^n(1-p)^k.$$

Hence the progeny distribution is

$$\mathrm{P}_a[T=k]=\frac{a}{k+a}\binom{k+(k+a)-1}{(k+a)-1}p^{(k+a)}(1-p)^k \qquad k=0,1,2,\ldots,$$

giving the step probabilities,

$$\begin{aligned}&\mathrm{P}_a\left[W_{i+1}=(x,y)|W_i=(u,v),T=k\right]\\&=\left[\binom{x}{u}\binom{k+a+u-v}{k+a+x-y}\right]\binom{x+u-1}{u-1}\left(\frac{\binom{(k+a-y)+(k+a-y+x)-1}{(k+a-y+x)-1}}{\binom{(k+a-v)+(k+a-v+u)-1}{(k+a-v+u)-1}}\right).\end{aligned}$$

As for the Poisson case, the conditioned step probabilities are independent of the geometric offspring parameter, p.

Binomial Distribution

For a binomial distribution, $\xi\sim\mathrm{Bin}(m,p)$, and the sum of independent and identically distributed (i.i.d.) binomials is a binomial, i.e. $S_n\sim\mathrm{Bin}(nm,p)$.

$$\mathrm{P}[\xi=x]=\binom{m}{x}p^x(1-p)^{m-x}\quad\text{and}\quad\mathrm{P}[\xi_1+\cdots+\xi_n=S_n=k]=\binom{nm}{k}p^k(1-p)^{nm-k}.$$

Hence the progeny distribution is

$$\mathrm{P}_a[T=k]=\frac{a}{k+a}\binom{(k+a)m}{k}p^k(1-p)^{(k+a)m-k}\qquad k=0,1,2,\ldots,$$

giving the step probabilities

$$\begin{aligned}&\mathrm{P}_a\left[W_{i+1}=(x,y)|W_i=(u,v),T=k\right]\\&\qquad=\left[\binom{x}{u}\binom{k+a-v+u}{k+a-y+x}\right]\binom{um}{x}\left(\frac{\binom{(k+a-y+x)m}{k+a-y}}{\binom{(k+a-v+u)m}{k+a-v}}\right).\end{aligned}$$

Unlike the Poisson and geometric cases, the conditioned step probabilities for a binomial offspring distribution are dependent on the parameters, though only on m, the maximum number of offspring an individual may have. This is not that surprising, since the m limits the number of valid paths, which may now be less than 2^{k-1}. For the first step, $x_0 = y_0 = a$,

$$\mathrm{P}_a\left[W_1 = (x, y) | W_0 = (a, a), T = k\right] = \left[\binom{x}{a}\binom{k+a}{k}\right]\binom{am}{x}\left(\frac{\binom{km}{k-x}}{\binom{(k+a)m}{k}}\right),$$

which is zero for $x > am$. So if $k > am$ the path $W = (a, k, 0)$ is not valid. In the most extreme case $m = 1$, each individual has either zero or one offspring and if $a = 1$ there is only one valid path, i.e. $x_0 = x_1 = \cdots = x_k = 1$ and $x_{k+1} = 0$.

Despite the dependence on m, the shape parameter, p, does not appear in the conditioned step probability.

Uniform Distribution

The final example is included as a comparison to the uniform step distribution on the random digraph in Section 2.4.4. Obviously, for a uniform to be valid there we need to induce an upper limit, the range of the uniform is a parameter, as for a branching process there is no limit to the population as for the finite digraph. For the digraph equivalent, each step was limited by the number of unconnected vertices remaining, so there was no explicit parameter.

We need the following lemma to obtain a closed form for Theorem 2.17, for a proof see Uspensky (1937, p23–24).

Lemma 2.18 (Uspensky (1937))

The probability of obtaining a total of p on n s-sided dice requires the number of ways to achieve this total. This is the coefficient of x^p in $(x+x^2+\cdots+x^s)^n$, divided by the total number of outcomes s^n. Thus

$$\mathrm{P}[p,n,s] = \frac{1}{s^n}\sum_{l=0}^{\lfloor (p-n)/s \rfloor}(-1)^l\binom{n}{l}\binom{p-sl-1}{n-1}.$$

where $\lfloor x \rfloor$ is the floor function, giving the integer part of x (rounding down).

Let $\xi \sim \mathrm{Uni}[0,m]$, then we can express the sum of n such random variables as,

$$\mathrm{P}[\xi_1+\cdots+\xi_n = S_n = k] = \frac{1}{(m+1)^n}\sum_{l=0}^{\lfloor \frac{k}{m+1} \rfloor}(-1)^l\binom{n}{l}\binom{k+n+(m+1)l-1}{n-1}$$

$$k = 0,1,\ldots,$$

Which follows from Lemma 2.18, consider the dice as the number of offspring $(0,\ldots,m)$ offset by one $(1,\ldots,m+1)$ and requiring a total of p plus the n offset (one per dice), i.e. $s = m+1$ and $p = k+n$.

As before we can then write

$$\mathrm{P}[T=k] = \frac{a}{k+a}\left(\frac{1}{m+1}\right)^{k+a}\sum_{l=0}^{\lfloor \frac{k}{m+1} \rfloor}(-1)^l\binom{k+a}{l}\binom{k+(m+1)l+k+a-1}{k+a-1}$$

$$k = 0,1,\ldots,$$

giving the step probabilities

$$\mathrm{P}_a\left[W_{i+1}=(x,y)|W_i=(u,v),T=k\right] = \left[\binom{x}{u}\left(\frac{k+a+u-v}{k+a+x-y}\right)\right] \left(\frac{\sum_{l=0}^{\lfloor\frac{x}{m+1}\rfloor}(-1)^l\binom{u}{l}\binom{x+u-1-(m+1)l}{u-1}\sum_{j=0}^{\lfloor\frac{k+a-y}{m+1}\rfloor}(-1)^j\binom{k+a+x-y}{j}\binom{2(k+a-y)+x-1-(m+1)j}{k+a+x-y-1}}{\sum_{n=0}^{\lfloor\frac{k+a-v}{m+1}\rfloor}(-1)^n\binom{k+a+u-v}{n}\binom{2(k+a-v)+u-1-(m+1)n}{k+a+u-v-1}}\right).$$

Though this step probability is not elegant, it is a closed expression for a conditioned step probability. For the equivalent digraph step, a recursive search of all valid paths is required to find a conditioned step probability. For similar reasons to the binomial offspring distribution, the uniform parameter m remains in the conditioned expression.

2.5.5 Parameter Invariance Of Conditioned Step Probabilities

In the previous section we observed that for the Poisson and Geometric offspring distribution, the conditioned step probabilities are invariant of the respective parameter (i.e. the Poisson rate λ or geometric probability p). We now explain the invariance and relate the example distributions.

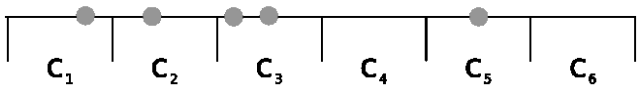
First consider the Poisson case. We have conditioned the branching process to a fixed number of offspring, k. We use the following standard result of Theorem 2.19, the distribution of waiting times for a Poisson process conditioned on the number of points in a fixed time are uniformly distributed (see p139–141 Parzen, 1964).

Theorem 2.19 (Parzen (1964))

Let $\{N(t):t\geq 0\}$ be a Poisson process with intensity ν. Under the condition $N(T)=k$, the k times $\tau_1<\tau_2<\cdots<\tau_k$ in the interval 0 to T that the events occur have the same distribution as the order statistics corresponding to k independent uniform random

variables on the interval $[0, T]$.

Using Theorem 2.17, conditioned on k offspring we can consider all the lifetimes together. That is we conditioned on there being k points in a Poisson Process of length $(k+a)c$ with rate $\lambda(k+a)$. By Theorem 2.17, these k points are distributed uniformly at random on $[0, k+a]$, and in particular their distribution is independent of λ. For example, in the diagram below we have one initial ancestor, $a = 1$, and condition on a total progeny of five, $k = 5$. So there are six intervals corresponding to all the individuals and the grey circles are points of the Poisson Process.



The k points will occur in an interval corresponding to a specific individual. There is the issue of assembling the individuals and their offspring counts (number of points in their interval) into a valid Galton-Watson Process, though this is combinatoric in nature and does not affect the current discussion.

In the example given, individuals 4 and 6 cannot be the initial ancestor (as they have no offspring). A valid path would be $W = (1, 2, 2, 1)$ where $W_0 = \{c_3\}$, $W_1 = \{c_1, c_5\}$, $W_2 = \{c_2, c_6\}$ and $W_3 = \{c_4\}$. There are many other valid Galton-Watson processes that can be constructed from the diagram, though they only depend on the counts in each interval.

By Theorem 2.19, each point is uniformly distributed over an interval, the interval counts do not depend on λ. Hence, the conditioned process is a combinatoric problem on the interval counts that are independent of the rate. Finally, the conditioned step

probabilities are a function of the entire conditioned path, which we have shown is λ invariant.

For the Geometric offspring, the sum of $k+a$ i.i.d. geometric random variables is a negative binomial distribution, $\text{NegBin}(k+a,p)$. The geometric is a discrete analogue to the exponential, so the negative binomial is related to the Poisson process.

Theorem 2.20

Let X be a negative binomial random variable, $X \sim \text{NegBin}(r,p)$. Under the condition $X=T$, the r successes at the points $t_1 < t_2 < \cdots < t_r$, where $t_i \in \{1,2,\ldots,T+r\}$ for each success i, $t_r = T+r$ and $t_i \neq t_j$ for $i \neq j$, have the same distribution as the order statistics corresponding to k uniform draws from the set $\{1,2,\ldots,T+r-1\}$ without replacement.

Proof

The proof follows that of Theorem 2.19 in Parzen (1964).

Denote the times of the $r-1$ successes (the rth success always occurs at $t_r = T+r$ by the definition of the negative binomial) as U_i and the ordered version as $U_{(i)}$. Let f denote the joint density of the $r-1$ draws. Then

$$f(u_{(1)},u_{(2)},\ldots,u_{(r-1)}) = \frac{1}{T+r-1}\frac{1}{T+r-2}\cdots\frac{1}{T} = \frac{T!}{(T+r-1)!}.$$

The probability of successes at $t_1,\ldots,t_r$ and failures everywhere else given T failures is

$$\text{P}[(t_1,\ldots,t_r)|X=T] = \frac{p^r(1-p)^T}{\binom{T+r-1}{r-1}p^r(1-p)^T} = \frac{T!(r-1)!}{(T+r-1)!}.$$

There are $(r-1)!$ permutations of the $r-1$ unordered uniform draws, hence

$$f(u_{(1)},\ldots,u_{(r)}) = (r-1)!\,f(u_1,\ldots,u_r) = f(t_1,\ldots,t_r|X=T). \qquad \square$$

By a similar reasoning to the Poisson case, using Theorem 2.20 the Geometric conditioned step probabilities are invariant of the success probability p. The binomial step probability is equivalent to a conditioned negative binomial except for the limiting parameter n, the maximum number of offspring. Under suitable conditions,

$$\text{Bin}(n, p) \to \text{Pois}(1/p) \text{ as } n \to \infty,$$

so without formal justification, we expect a similar p-invariance as observed, but there to be a dependence on n.

2.5.6 Numerical Results

We present some numerical results relating to the expressions derived in the previous section. As established in Section 2.3, we display the results graphically by considering the path of the average for each generation.

The averages are over a number of simulated paths, typically 10^5, each generation average can then be calculated as well as an approximate of the 95% interval. The interval approximation is based on ordering the generation sizes and defining the range covering 95% of the sample paths.

Under the branching process approximation to the random digraph there is no recursive component. Neither is there any implicit dependence on the population size, as discussed for random digraphs in Section 2.4.9 in terms of vertices. For small total progeny the time to compute the probabilities is the same as for the equivalent epidemic process. However for large final sizes the digraph approach is unfeasible (the recursive path searching becomes too costly), which is not the case for the branching process method.

Recall the total number of valid paths of total progeny is 2^{k-1}, so even though we can calculate a conditioned path probability in one pass, to calculate exact expectations for large k is still too computationally intensive. For example, if $k = 30$ it would take several decades to compute all the possible paths if we could evaluate one per second. Since this is so impractical, we use the exact conditioned step probabilities to estimate the expected generation size using simulated paths. We do not reject any samples, as we did in Section 2.4.7.5, as all paths are simulated from the exact conditioned probabilities.

As with the digraph calculations, we can store intermediate steps to speed our calculations. Which is a greater saving for the branching process step probabilities that are parameter invariant.

Poisson Offspring

We start by considering the Poisson offspring distribution derived in Section 2.5.4. Since the step probabilities are invariant of the rate parameter λ, these results are valid for all Poisson offspring distributions.

Estimating by simulation the path of the average, as defined in Section 2.4.6, we can compare across different numbers of ancestors, a and the total progeny we condition on, k.

Figure 2.7 shows the path of the average generation size for $a = 1$ and varying k. The total progenies shown are: 10, 20, 40, 60, 80, 100, 200, 500 and 1000. To meaningfully compare the plots, in Figure 2.7(a) the generation and average size of the generation have been scaled by k, the conditioned total progeny, i.e. plot t/k against $\mathrm{E}[X_t]/k$.

Since $x_0 = a$ for all paths, on the scaled plot the expected size of the zeroth generation will tend to zero. There seems to be a limiting behaviour when scaled by the conditioned total progeny.

Figure 2.7(b) shows the same average paths as Figure 2.7(a), except the generation and expected generation size are scaled by the square root of the conditioned total progeny, i.e. plot $t/\sqrt{k}$ against $\mathrm{E}[X_t]/\sqrt{k}$. This rescaled process hints at a limiting distribution as the conditioned total progeny tends to infinity.

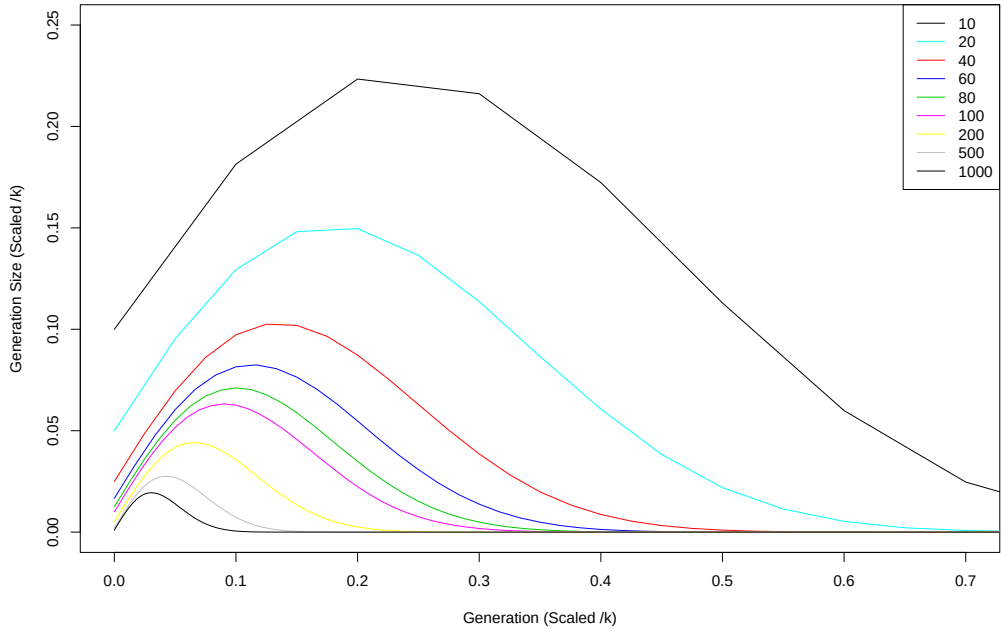
This $\sqrt{k}$ scaling is related to results in Drmota and Gittenberger (1997) and Gittenberger (1998); specifically let $(L_n(t), t \geq 0)$ be a Galton-Watson process conditioned on a total progeny of n, where $L_n(t)$ is the size of the t-th generation. For a sequence of positive numbers, $(c_n, n \geq 0)$ such that $c_n \to \infty$ and $c_n = o(\sqrt{n})$, then the scaled process,

$$l_n = \frac{1}{c_n} L_n(c_n t), \quad t \geq 0,$$

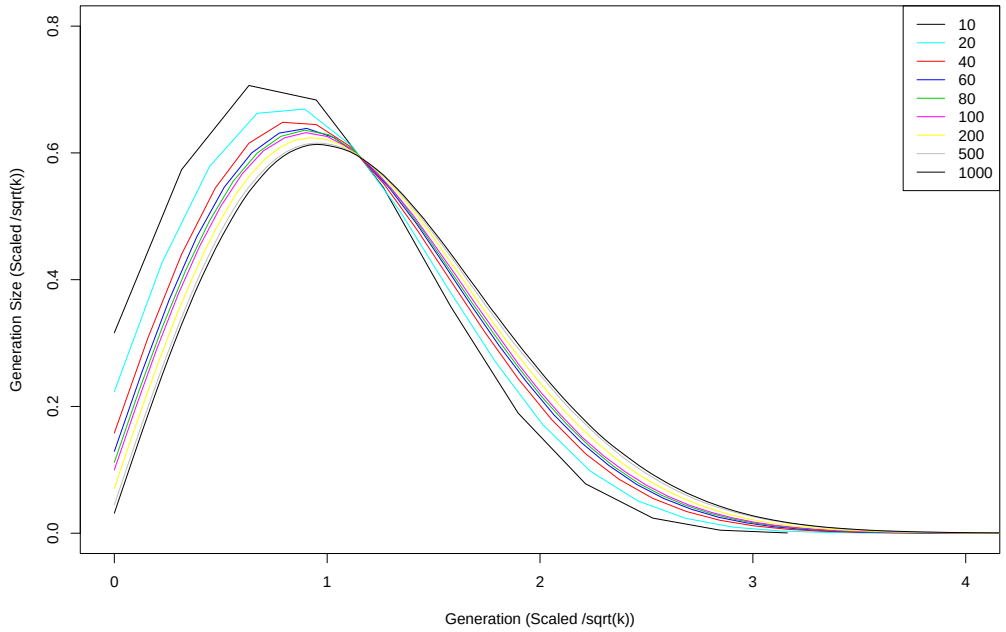
weakly converges to the local time of a three-dimensional Bessel process. If the scale factor is $c_n = \sqrt{n}$, the limit process obtained is Brownian excursion local time.

From Gittenberger (1998), the average extinction time of a branching process conditioned on the total progeny n is proportional to $\sqrt{n}$.

This provides a limiting result for the conditioned Galton-Watson processes considered. Further, the maximum generation size also has a limiting distribution. For unconditioned branching processes results for the maximum are given by Lindvall (1976) and Weiner (1984), these results are extended to conditioned branching processes under finiteness constraints on the offspring distribution by Kerbashev (1999) and finally the expectation of the maximum generation size conditioned on total progeny is derived



(a) Scaled by k



(b) Scaled by $\sqrt{k}$

Figure 2.7: Scaled path of the average for a Poisson offspring distribution with one initial ancestor, $a = 1$ and varying conditioned total progeny, k. The generation and estimated expected generation size are normalised by k and $\sqrt{k}$ to facilitate comparison

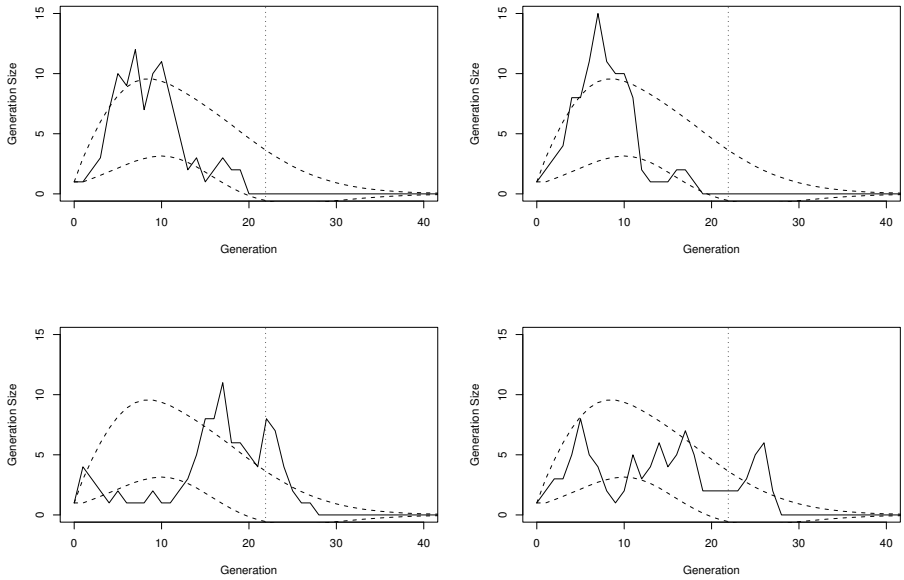


Figure 2.8: Four selected sample paths from Poisson offspring distribution $a = 1$ and $k = 100$ chosen from 10 simulated paths. Normal approximation 95% intervals are shown as dotted curves and the expected number of generations is shown as a dotted vertical line.

by Bondarenko and Topchiĭ (2001).

Figure 2.7 shows the estimated expected size of each generation, which we have called the average path. Despite the smooth nature of the average path, there is a great deal of variability in specific realisations of a path.

This variability in sample paths of the conditioned branching process is illustrated in Figure 2.8, which shows four realisations in the Poisson offspring case. The paths were selected from 10 simulated runs and chosen to display the variability not captured in the path of the average graphs, i.e. they have been chosen to look different. The dotted lines are an approximate 95% interval over the generation sizes. The vertical dotted line corresponds to the expected number of generations, i.e. $\mathrm{E}[\tau]$ where $\tau = \min\{t : x_{t+1} = 0\}$.

The interval is approximated assuming the generation sizes are normal distributed, this is clearly inaccuracte near the boundaries as the interval will not be symmetric and the assumption of normality is not theoretically derived. As a first approximation the interval for each generation, i is given by $\bar{X}_i \pm 1.96\sqrt{\text{Var}(X_i)}$. These are the mean and variance of samples of the i-th generation, assuming the average size of a generation is normally distributed with mean $\text{E}[X_i]$ and variance $\text{Var}(X_i)$. This leads to intervals with negative lower bounds and upper bounds that exceed the maximum attainable generation size, i.e. k in a branching process conditioned on having total progeny equal to k. Alternatively, we can consider the interval of the empirical quantiles as a measure of the variability of the average path, i.e. ordering the samples for the i-th generation and taking the 2.5% and 97.5% quantiles.

Figure 2.9 compares the average generation sizes, normal approximated confidence and quantile interval obtained from 6×10^4 sample paths with a Poisson offspring distribution conditioned on $k = 100$ with a single initial ancestor. The normal approximation intervals are shown as dotted lines, clearly becoming negative for some generations. The thicker solid line is the average path, included as a reference to Figure 2.7. Finally, the empirical quantiles are shown as solid lines. Approximating the distribution of the expected generation size as normal seems fairly adequate, setting any negative lower bound to zero.

As noted in Figure 2.7(a), considering the limiting behaviour of the average path as $k \to \infty$ for a fixed number of initial ancestors will cause the zeroth generation to tend to zero, $x_0 = \frac{a}{k} \to 0$ as $k \to \infty$. Figure 2.10 shows the average path when the ratio $\frac{a}{k}$ is kept constant, the generation size is scaled by k to ensure the zeroth generations coincide, though this may not be the optimal scaling. The lines are for conditioned progeny's of 10, 50, 100 and 200 with the ratio $\frac{a}{k} = 0.1$. The average paths are calculated from 10^5 simulated conditioned paths.

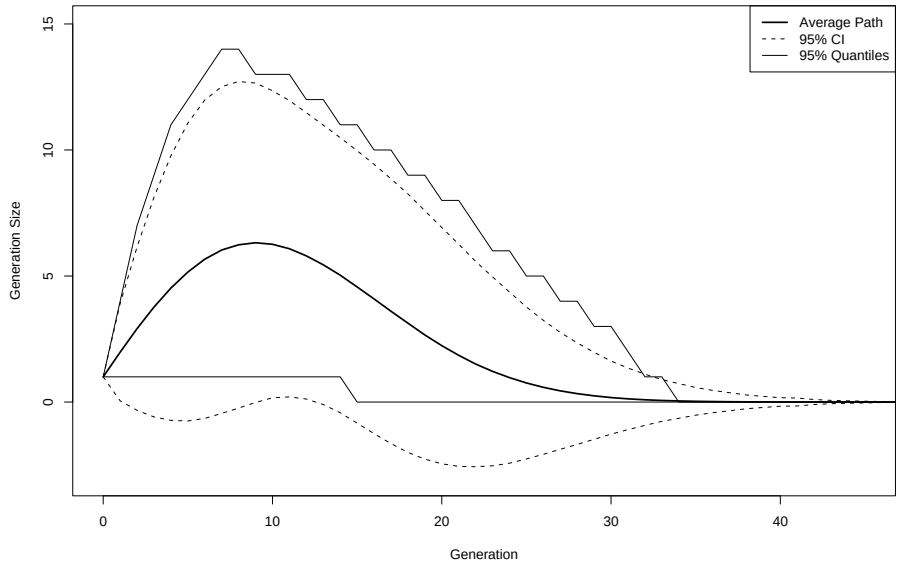


Figure 2.9: Comparison of generation sizes obtained from a normal approximation and empirical quantiles for a branching process with one ancestor conditioned on a total progeny of a hundred with Poisson offspring, i.e. $a = 1$ and $k = 100$ paths. The normal approximation and empirical 95% intervals are shown as dotted and solid lines respectively. The average path is a thick solid line.

Figure 2.11 shows the approximate and empirical intervals for the corresponding fixed ratio when there are twenty initial ancestors and a total progeny of two hundred. The approximate intervals still closely match the empirical.

Finally we consider the variance of each generation size, i.e. $\mathrm{Var}(X_i)$ for $0 \leq i \leq k+1$. Since $x_0 = a$ and $x_{k+1} = 0$ for all paths, clearly $\mathrm{Var}(X_0) = \mathrm{Var}(X_{k+1}) = 0$. Figure 2.12 shows the estimated variances for each generation conditioned on various total progenies. The generation is scaled by the square root of the conditioned total progeny, i.e. $\frac{i}{\sqrt{k}}$ and the variance is scaled by the progeny, i.e. $\frac{\mathrm{Var(X_i)}}{k}$. It is clear from Figure 2.12 that there is a limiting behaviour being observed.

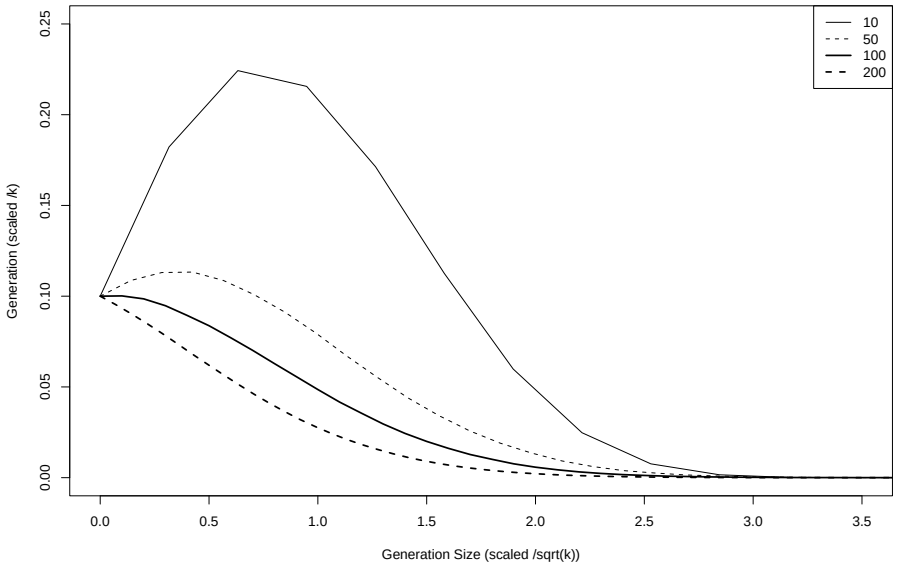


Figure 2.10: Path of the average for a Poisson offspring distribution for values of k, maintaining the ratio of $a/k = 0.1$ Both the generation number and generation size are normalised by $\sqrt{k}$ to facilitate comparison

Alternate Offspring Distributions

So far we have considered only the Poisson offspring case, this corresponds to an epidemic with fixed infectious periods. For fixed periods, in the directed random graph all edges are independent which greatly simplifies calculations.

Though we consider three alternative offspring distributions, we expect them to behave in a similar manner. A negative binomial distribution with parameters r and $p = r/\lambda + r$ tends to a Poisson distribution with parameter λ as r tends to infinity. For sufficiently large r we may approximate using the Poisson case. For a gamma infectious period, $\Gamma(a, b)$ with integer shape parameter, the corresponding approximate offspring distribution is a negative binomial, $\text{NegBin}(a, b)$. We considered the case where $a = 1$, i.e. an exponential giving a geometric offspring. Since our conditioned geometric step probabilities do not depend on the probability, we can make the approximation to the

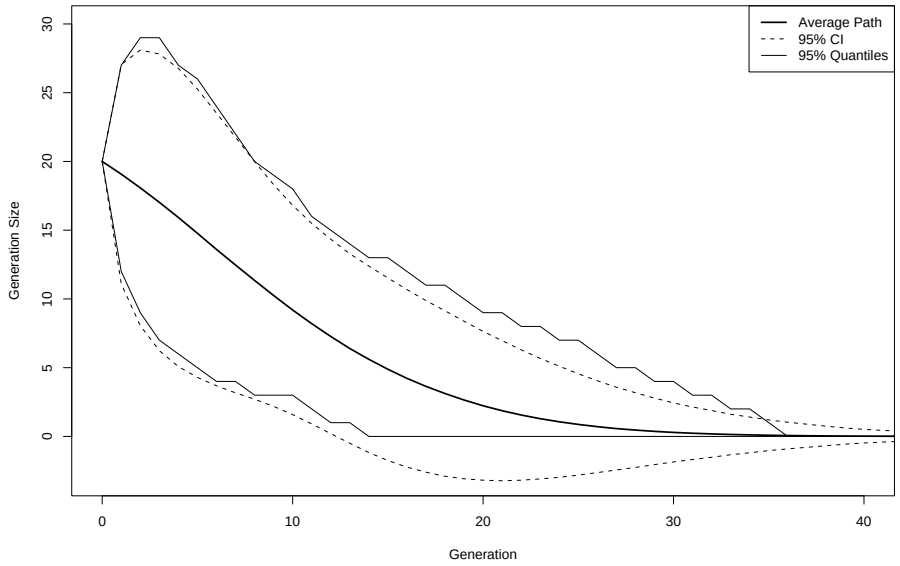


Figure 2.11: Comparison of generation intervals for a Poisson offspring branching process with $a = 20$ and $k = 200$. The normal approximation and empirical 95% intervals are shown as dotted and solid lines respectively. The average path is a thick solid line.

Poisson arbitrarily good. Hence we expect the negative binomial and Poisson offspring cases to be similar.

Similarly, the binomial distribution converges to the Poisson distribution as the number of trials goes to infinity while the product np remains fixed. Since p may be arbitrary, we need only consider the number of trials. If $n > 20$ with sufficiently small p, we may approximate the Binomial by a Poisson distribution with parameter np. For sufficiently large n, we expect the binomial and Poisson offspring cases to be similar.

The uniform case does not tend to a Poisson, so we expect the average path to be different.

We compare the average path for four branching processes conditioned on a total progeny of one hundred with a single initial ancestor. We set the parameters of each

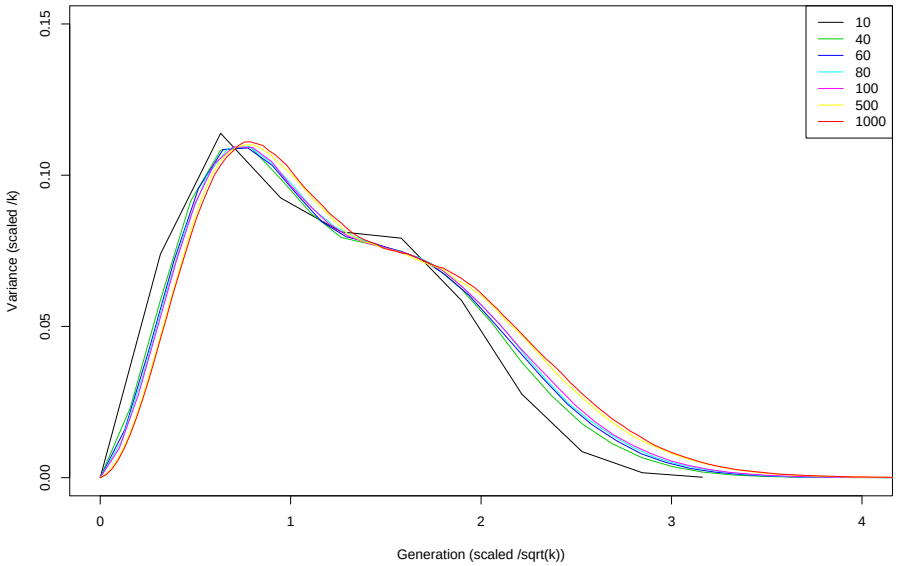


Figure 2.12: Comparison of generation variances for Poisson offspring branching process conditioned on various total progenies. The generation is scaled by $\sqrt{k}$ and the variance by k. All have a single ancestor

distribution to be equivalent in a sense, so that the comparison is meaningful. For the Poisson offspring we do not need to specify a rate, since in the conditioned process it is an invariant parameter. We consider the negative binomial with ten success events, though we only derived the result for a single success, i.e. a geometric, the result is simple to generalise. For the binomial and (discrete) uniform we set $n = 10$, the maximum offspring from each individual.

Figure 2.13 shows the average paths estimated from 10^6 simulated conditioned branching processes for the four offspring distributions derived in Section 2.5.4. As expected, the Poisson, binomial and negative binomial appear very similar and the uniform exhibits a different behaviour. The parameters of each distribution have been chosen to have similar characteristics.

Finally we consider the variance of the path, the separate generation variances esti-

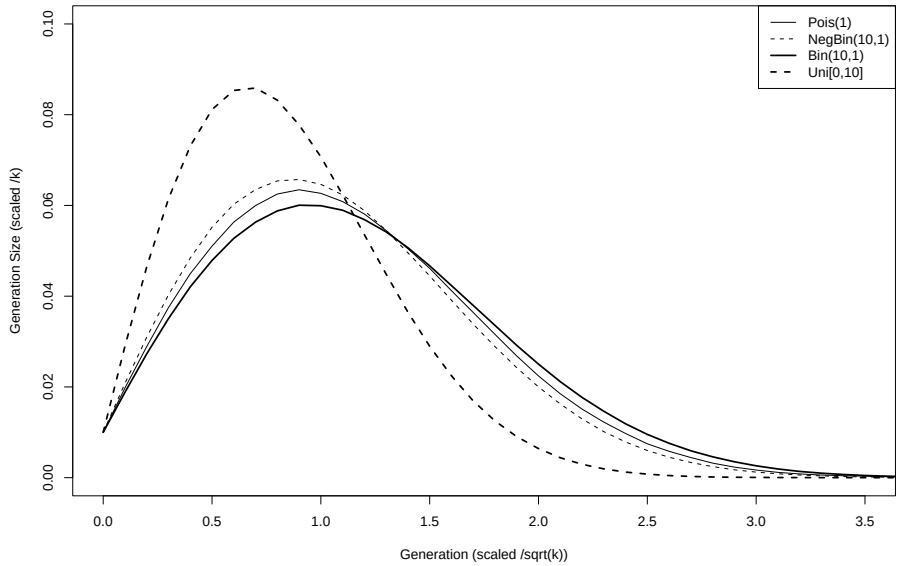


Figure 2.13: Comparing scaled offspring distributions, with $a = 1$ and $k = 100$, of the empirical expected size of each generation for the distributions: Pois(1), NegBin(10,1), Bin(10,1) and Uni(10)

mated from simulations. Figure 2.14 compare the four offspring distributions of Figure 2.13, plotting the variance for each generation. The generation and variance are scaled as in Figure 2.13 to facilitate comparison. The behaviour is similar for the Poisson, Negative Binomial and Binomial. The Uniform is clearly distinct, as expected.

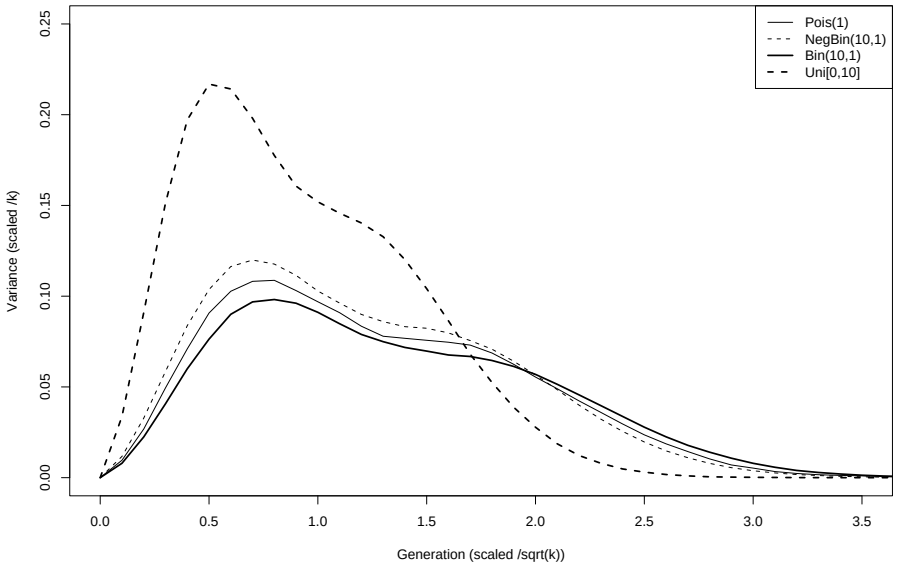


Figure 2.14: Comparing scaled offspring distributions, with $a = 1$ and $k = 100$, of the empirical variance in each generation for the distributions: Pois(1), NegBin(10,1), Bin(10,1) and Uni(10)

2.5.7 Branching Process Approximation To Finite Random Digraph

In Section 2.5.5 we showed the Poisson offspring branching process is invariant to the rate of the Poisson distribution. As described in Section 2.5.2, a branching process with Poisson offspring can be used to approximate an epidemic with fixed infectious periods in the early stages. We now investigate this approximation using the random digraph representation to give a relationship between the rank chain of the digraph and path of the branching process.

The random digraph conditioned on connectedness and the branching process conditioned on its total progeny are both tools to consider the final size of an epidemic. Since the branching process approximation assumes a large population, it would seem that increasing s, the number of susceptibles in the finite graph, should affect how close the two processes are. In particular, the random digraph is λ-dependent whereas the branching process is not.

For the random digraph, consider the constant infectious period case, let $c = 1$ without loss of generality. Figure 2.15 shows the effect of varying λ at various values of s, given one root node and a conditioned connectedness of five, on the expected size of the first generation, X_1. The graph was produced using the expression derived in Section 2.4.4 for the digraph simulations.

In Figure 2.15 a horizontal line has been added to represent the equivalent branching process, with a single ancestor and conditioned total progeny of five, i.e. $a = 1$ and $k = 5$. The line is horizontal as the path probabilities, and hence the expectations, are invariant to λ. The line corresponds to the digraph letting the population be infinite in size, i.e. letting $s \to \infty$.

It is interesting to note that the lowest non-horizontal line in Figure 2.15 corresponds to

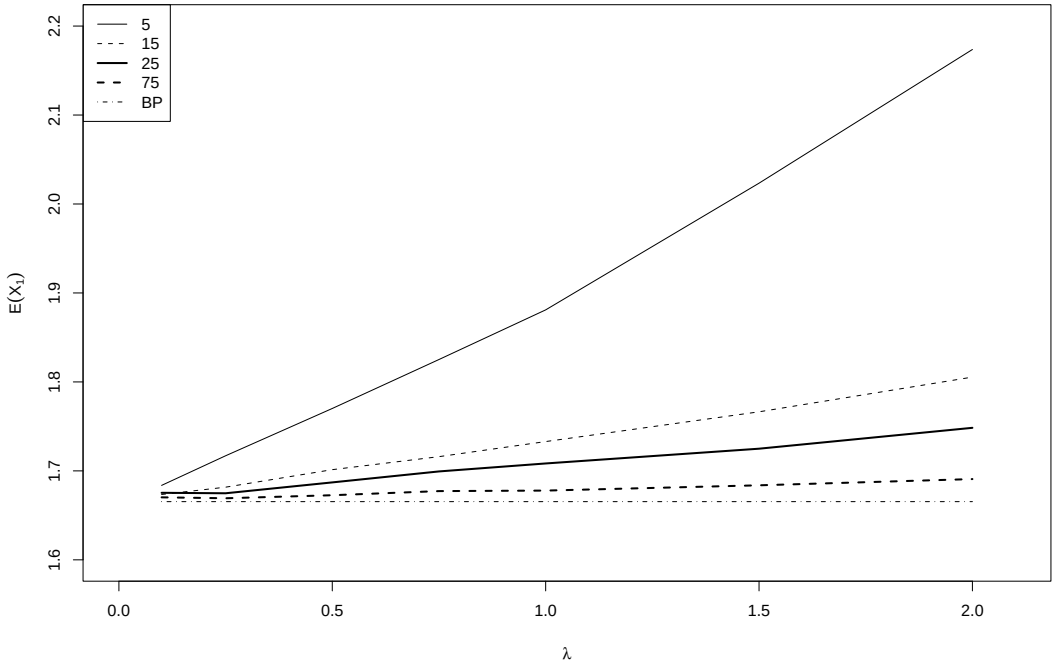


Figure 2.15: Comparing the expected size of the first generation in a conditioned random digraph and a conditioned Poisson branching process. With a single root and ancestor, $r = a = 1$ and conditioned on $d = k = 5$ while varying λ. The digraphs have different number of initial susceptibles s.

seventy five susceptibles, $s = 75$ which is not particularly large. The approximation is fairly good even for small populations. Conversely, for the smallest population shown, $s = 5$, the approximation is very poor.

Finally, the limiting behaviour illustrated in Figure 2.15 can be shown algebraically with the following example. The step probabilities of a conditioned random digraph are s-invariant for suitable λ, see Section 2.4.9. Though the number of susceptibles, s is implicit in the definition of λ. Using the example derived in Section 2.4.7.1, for

$(r, s, d) = (1, s, 2)$ and letting $s \to \infty$ we have,

$$\begin{aligned} \mathrm{P}_{1,s}[Z_1 = (1,2) | Z_0 = (1,1), D = 2] &= \frac{2e^{-\frac{\lambda c}{r+s}}}{1 + 2e^{-\frac{\lambda c}{r+s}}} \\ &\to \frac{2}{3} \quad \text{as} \quad s \to \infty \\ &= \mathrm{P}_1[W_1 = (1,2) | W_0 = (1,1), T = 2]. \end{aligned}$$

That is, the probability tends to the conditioned step probability of the equivalent conditioned branching process, with a Poisson offspring corresponding to the fixed infectious period.

Since the digraph step probabilities cannot be expressed in an easily obtainable algebraic form, we cannot generalise this for any such step probability. Though, by the definition of the branching process approximation, we expect this to be true.

CHAPTER 3

Inference For Final Size Data Using Markov Chain Monte Carlo Methods

3.1 Introduction And Motivation

We are motivated by the need to analyse epidemic data, specifically final size data, to gain insight from previous outbreaks. As discussed in Sections 1.2.6 and 2.1, observations of epidemics are often incomplete in regard to each individual as well as only covering a subset of the population.

In this chapter we consider the following problem. Given final size data and a stochastic epidemic model, what can be inferred about the parameters of this model from the data, what insight can be gained?

The final size data will consist only of counts of the number of susceptibles infected at the end of the epidemic. We shall use the stochastic Susceptible-Infective-Removed (SIR) epidemic model defined in Section 1.2.2, using the directed random graph representation investigated in Chapter 2. To make statistical inference about parameters of the model given the data we use Markov Chain Monte Carlo (MCMC) methods as outlined in Section 1.3.2, we shall present update algorithms specific to the final size data.

For final size data, the likelihood under the simple SIR model is intractable, since the only information from the data is the state of the population at the end and beginning of the epidemic. There is no explicit information about the start of the epidemic, specifically the number of initial infectives is unknown and at first we shall assume a single initial infective, later this will be considered another unknown parameter. To proceed we must augment the likelihood with sufficient information about the course of the epidemic to obtain a tractable expression. The imputed course of the epidemic will be the representations investigated in Chapter 2, firstly we shall consider the edge representation and then the generation representation.

Using MCMC we will make inference for the infection rates of the SIR model. Without more detailed temporal information, it is not possible to make inference about the infectious period directly. For the fixed infectious period case, it is impossible to separate the infection rate and infectious period, the two parameters are indistinguishable. For this chapter we shall only consider a fixed infectious period, that will be considered a known constant of the model.

In Section 3.2 we consider the simple SIR epidemic model, with a single type of individual with the course of the epidemic as missing data that we impute. Imputation of edges has been investigated by Demiris and O'Neill (2005a), we present this approach and the generation representation developed in Chapter 2. Both methods are compared using sample data sets.

We define an epidemic with missing data to include all types of data that are observations of a process omitting some detail, e.g. the exact infection and removal times. Final size data can then be viewed as an example of missing data, where the infection times, removal times and which individuals infect each other are unknown. Partially observed epidemics are defined to be those where the data only represent a subset of

the population, i.e. a specified fraction of the total population. In Section 3.3 we extend the algorithm to enable the analysis of such data.

Including unobserved individuals naturally leads to incorporating multiple types of individuals, some of which may be unobserved. We briefly expand the algorithm in Section 3.4, though we present a more complete general framework in Section 3.5, allowing individuals to have multiple levels of mixing. Thus, the general framework allows for arbitrary types of individuals and an arbitrary number of levels of mixing, together with a general form for the rates of contacts among individuals. There are limits to the type of model that can be fitted, namely no temporal effects can be included, e.g. weekday-weekend cycles. Also, the data may be too sparse to implement the complicated general model, which may lead to overfitting or poorly converging Markov Chain Monte Carlo algorithms.

The multi-type multi-level algorithm is applied to the household data presented in Longini et al. (1988), and comparisons are made to the edge imputation methods by O'Neill (2009) on the same data set.

Finally, in Section 3.7 we consider practical considerations of implementing the MCMC algorithms. In particular the use of parallel computing using GNU OpenMP and the need for arbitrary precision using GNU MPFR. To implement the MCMC algorithm in the C programming language we have also used the GNU Scientific Library (GSL), see Galassi et al. (2003) for further details.

3.2 MCMC Algorithms For Simple SIR Epidemic Model

In this section we consider an SIR model, as defined in Section 1.2.2, with homogeneously mixing and homogeneous population of N individuals, which we shall denote a one-type one-level (1t1l) model, with a fixed infectious period, i.e. $I = c$ for some constant $c \geq 0$ and $\iota = \mathrm{E}[I] = c$. An infectious individual has infectious contacts with another given individual at the points of a Poisson process with rate $\frac{\lambda}{N}$ over some interval I, where N is the size of the population of which n are initially susceptible and a are initially infective. Initially we set the number of initial infectives to be one, i.e. $a = 1$, without any explicit information it seems a reasonable assumption that the epidemic is initiated by a single external infection to an individual at random. All individuals are labelled by an index i from the set $\{1, 2, \ldots, N\}$. Let κ denote the set of indicies of the initial infectives, for a single initial infective we abuse the notation and let κ denote the index of the single initial infective. Since the population is homogeneous and we are restricting attention to the case $a = 1$, we may set $\kappa = \{1\}$ without loss of generality.

We wish to make inference for λ, the infection rate, given the final size of an outbreak in the population. Let d be the number of initial susceptibles that are ultimately infected, not including the initial a infectives, out of the initial n susceptibles. Let $N = a + n$ and $D = a + d$ be the total population size and total number of individuals who are ever infective respectively. The data may be summarised as the vectors

$$\theta = (a, n, d) \qquad \text{for} \quad \begin{array}{l} a \in \mathbb{Z}_+ \\ n \in \mathbb{Z}_+ \\ 0 \leq d \leq n \end{array} \tag{3.1}$$

or

$$\psi = (N, D) \qquad \text{for} \quad \begin{matrix} N \in \mathbb{Z}_+ \\ 0 \leq D \leq N \end{matrix}, \tag{3.2}$$

depending on the assumption on the initial number of infectives. We shall initially consider the case where $a = 1$ and use the θ notation. Thus we wish to find the posterior density $\pi(\lambda|\theta)$, using Bayes' Theorem we have

$$\pi(\lambda|\theta) \propto \mathrm{L}(\theta|\lambda)\pi(\lambda).$$

The likelihood can be derived for any population size, specifically Ball (1986) derive a general expression to compute the distribution of the final size given the rate λ. However, these equations to compute the likelihood of θ given only λ become numerically intractable for large populations, even with the assumptions of a fixed infectious period and single initial infective we cannot derive an expression for the likelihood that is efficiently computable. To obtain such an expression, dependent only upon the infection rate λ, we must integrate out all other dependencies. In particular, it is difficult to integrate over all possible paths to achieve a final size of d, the approach of Ball (1986) using a set of recursive triangular equations. It is not impossible though, to obtain a numerical result for the likelihood using arbitrary precision computing as by Demiris (2004), we shall return to this in Section 3.7.2. However, MCMC methods rely on repeated iterations of the chain that require evaluation of the likelihood, if the computational cost (usually time) to evaluate the likelihood is too large then the method will be infeasible.

The likelihood of λ given all the information from the data and the model (e.g. fixed

infectious period and a single initial infective)

$$\pi(\lambda|\theta, I, \kappa) \propto \mathrm{L}(\theta|\lambda, I, \kappa)\pi(\lambda),$$

is still intractable, despite additional parameters being fixed in the model. To apply MCMC methods we require a computable form for $\mathrm{L}(\theta|\cdot, \lambda)$. To achieve this we augment the likelihood with additional data, that we consider to be a new parameter, giving a joint posterior density of λ and the imputed data.

In Section 3.2.1 we reproduce the results of Demiris and O'Neill (2005a), wherein the likelihood is augmented with a random digraph characterised by its edges, following Section 2.2.2 we use the relationship of the digraph to the final size of the epidemic to compute the likelihood of the final size d given the imputed course of the epidemic and the infection rate.

Two edge methods are reviewed, both of which require detailed information on each individual in the population. In fact, it is possible to only consider those individuals who are ultimately infected, as discussed in Section 2.4.9, since there is no need to explicitly account for contacts between those who remain susceptible for the entire epidemic.

In Section 3.2.2 we augment the likelihood with a digraph characterised by generations, which contains less detail on individuals within the population. The update steps for an MCMC algorithm are explained in detail, defining notation that will be extended in subsequent sections.

The two augmentation methods are compared in Section 3.2.3, with particular attention to the convergence properties of the generation representation under various tunable parameters.

3.2.1 Imputing Edge Representation

Following Demiris and O'Neill (2005a), though using our notation defined in Sections 2.3 and 3.2, let G denote a random digraph on N vertices of which a are roots and edges are present with probability $p = 1 - \exp(-\frac{\lambda}{N}c)$, where c is the fixed infectious period and λ is the infection rate. Then, the connectedness of the digraph is equal in distribution to the final size of the matched SIR epidemic.

We shall consider two forms of representing the digraph G, either as a connectivity matrix or as contact lists, the latter yields a more efficient MCMC update step using a Gibbs update.

For a given digraph, G, the likelihood of $\theta = (a, n, d)$ is an indicator variable, either the digraph is d-connected or it is not. By Bayes' Theorem we have,

$$\begin{aligned}\pi(G, \lambda|\theta, I, \kappa) &\propto \mathrm{L}(\theta|G, \lambda, I, \kappa)\pi(G, \lambda|I, \kappa) \\ &\propto \mathrm{L}(\theta|G, \lambda, I, \kappa)\pi(G|\lambda, I, \kappa)\pi(\lambda|I, \kappa) \\ &\propto \mathrm{L}(\theta|G, \lambda, I, \kappa)\pi(G|\lambda, I, \kappa)\pi(\lambda),\end{aligned}$$

since we assume the infectious period, infection rate and seed infective index are independent a priori, then

$$\pi(G, \lambda|\theta, I, \kappa) \propto \mathbb{I}_{\{\theta|G,\kappa\}}\ \pi(G|\lambda, I, \kappa)\ \pi(\lambda).$$

Where $\mathbb{I}_{\{\theta|G,\kappa\}}$ is the indicator variable, one if the imputed digraph matches the final size data and seed infectives, otherwise zero. The digraph has edges with probability p as defined, thus $\pi(G|\lambda, I, \kappa)$ is the probability of the imputed digraph. Finally, $\pi(\lambda)$ is the prior density on λ, which is assumed independent of the other parameters.

The prior will be an exponential distribution with a hyperparameter of μ, i.e. $\pi(\lambda) = \mu \exp(-\mu\lambda) \propto \exp(-\mu\lambda)$ (note that the prior is on λ, not the scaled rate $\frac{\lambda}{N}$). Let μ be sufficiently small to induce a fairly flat non-informative proper prior, since the expectation of an exponential is μ^{-1}.

To implement an MCMC algorithm, we must establish an update step for the infection rate λ and the imputed digraph G, once the chain has converged it will draw samples from the joint density. We are primarily interested in λ, so we may compute the marginal density by considering all samples and ignoring G. This is only valid if the chain has converged and care must be taken to ensure the correlation between the infection rate and imputed digraph does not distort the marginal density. We shall update the parameters one at a time, effectively in two blocks, as discussed in Section 1.3.2.

Care must also be taken in selecting the initial state of the parameters. An initial short run of the MCMC algorithm may yield clues as to suitable initial values for the infection rate λ, however the digraph G is a high dimensional object that must be summarised. The appropriate summary is considered, it must be sufficient to determine convergence, but also minimal to reduce the quantity of output of the algorithm. Hence, there may not be an obvious 'good' initial digraph, or a method to construct it.

The update steps for the digraph and infection rate depend on the form of G, we present the intuitive form first using a connectivity matrix and symmetric Random Walk Metropolis. However, a more efficient update is possible using a Poisson representation of contact lists and a Gibbs update.

3.2.1.1 Connectivity Matrix

The following method is as implemented by Demiris and O'Neill (2005a), we present the approach in detail for comparison with the generation method that we develop in Section 3.2.2. A digraph G consists of a set of vertices and edges, we may represent G as a matrix indicating the presence of edges on the graph. Label the vertices $1, \ldots, a, a+1, \ldots, a+d, a+d+1, \ldots, N$, such that the first a vertices are the roots and the next d vertices are those that are ultimately infected. Let G be an $N \times N$ matrix such that $g_{ij} = 1$ if there is an edge from vertex i to vertex j, for $i \neq j$. Since this is a directed graph, the matrix G need not be symmetric, i.e. $g_{ij} \neq g_{ji}$ in general. The correspondence to the epidemic is as before.

Define the update for λ to be a symmetric Random Walk Metropolis (RWM) as defined in Section 1.3.2.3, the candidate value is $\lambda' = \lambda + l$, where $l \sim \mathrm{N}(0, \sigma_l^2)$. The candidate must be non-negative, thus if $\lambda' < 0$ we reject the proposal immediately and do not have to calculate the acceptance probability, since $\pi(\lambda) = 0$ for $\lambda < 0$. The variance, σ_l^2 is a tunable hyperparameter that must be specified beforehand, commonly a trial MCMC run will be performed to tune the hyperparameters. Following Section 1.3.2.3, the acceptance probability for the symmetric proposal $q(\cdot|\lambda)$ is

$$\begin{aligned}\alpha(\lambda, \lambda') &= \min\left\{1\,,\ \frac{\pi(G, \lambda'|\theta, I, \kappa)q(\lambda|\lambda')}{\pi(G, \lambda|\theta, I, \kappa)q(\lambda'|\lambda)}\right\} \\ &= \min\left\{1\,,\ \frac{\pi(G|\lambda', I, \kappa)\pi(\lambda')}{\pi(G|\lambda, I, \kappa)\pi(\lambda)}\right\}.\end{aligned}$$

Since the proposal is constructed to be symmetric in that the proposal probabilities are equal, i.e. $q(\lambda'|\lambda) = q(\lambda|\lambda')$, they cancel from the acceptance probability. Also, for the λ update the digraph G is unchanged for the candidate, the term $\pi(\theta|G, \lambda, I, \kappa)$ is independent of the value of λ, i.e. $\pi(\theta|G, \lambda, I, \kappa) = \pi(\theta|G, \lambda', I, \kappa)$.

Though we desire a non-informative prior on the infection rate λ, the chosen proper prior is not truly non-informative. If it were, then all values for λ would be equally likely and the ratio of the candidate to current prior would be one. However, since the exponential is being used, the prior density contributes to the acceptance probability. Namely, since $\pi(\lambda) = \mu \exp(-\mu\lambda)$, then

$$\frac{\pi(\lambda')}{\pi(\lambda)} = \exp\left(-\mu(\lambda' - \lambda)\right). \tag{3.3}$$

The digraph G is a parameter in the model, we must search the space of digraphs using an update step to obtain the joint density of λ and G. The vertices and their labels are fixed, without loss of generality we have set $\kappa = \{1, \ldots, a\}$ and we shall assume $a = 1$ for the present discussion. Thus, the space of digraphs is concerned with the random edges, each of which is independent of all other edges and present with probability $p = 1 - \exp(-\frac{\lambda}{N}c)$, where c is fixed and λ is a constant during the update of G as we are updating the parameters independently.

First, we must define a proposal distribution to generate a candidate digraph. We can either add or remove an edge from G to generate a candidate digraph G'. The simplest scheme would be to select an element of the matrix G at random and invert the entry, if the edge is present remove it or if it is absent add it, i.e. $g'_{ij} = g_{ij} + 1 \mod 2$. There are $N^2 - N$ possible edges, self edges are excluded, thus we choose all edges equally. The acceptance probability is then

$$\alpha(G, G') = \min\left\{1\,,\, \frac{\mathbb{I}_{\{\theta|G',\kappa\}}\, \pi(G'|\lambda, I, \kappa)}{\mathbb{I}_{\{\theta|G,\kappa\}}\, \pi(G|\lambda, I, \kappa)}\right\}.$$

The chain should begin in a valid state, thus there should be no need to check the term $\mathbb{I}_{\{\theta|G,\kappa\}}$ as the current state should always be a valid digraph. However, it is necessary to check $\mathbb{I}_{\{\theta|G',\kappa\}}$ for both additions and removals.

It is possible to reduce the digraph G to only those individuals who ultimately become infected, this invariance is shown in Section 2.4.9. If reduce to the sub-digraph on D vertices, then is it no longer necessary to check validity after adding an edge, since any additional edges cannot increase the connectivity. It is important to still account for the remaining $N - D$ vertices, as they will have a great affect on the likelihood of different infection rates.

To check whether the digraph is valid, a recursive search can be performed beginning at the root vertex. Let v be an N length vector, and initialise it such that $v_i = 1$ for $i \in \kappa$ and zero otherwise. Beginning at the root vertices, travel along each edge away from the roots, to the set of vertices comprising the first generation. For each visited vertex i, set $v_i = 1$. Then visit all connected vertices from the first generation, i.e. the second generation, during the recursive search set the i^{th} component to one if vertex i is visited. It is possible to make this search more efficient, if a search meets a vertex that has already been visited, then that specific recursive search can be terminated. Then the digraph is valid if the required number components of v are one, i.e. $\sum_{i=1}^{N} v_i = a + d$. There seems no more efficient method to check connectivity. For small populations this recursive search is sufficient. However, as D increases the search becomes more costly and the amount of information stored grows by order N^2, which means the MCMC algorithm must move about the large space of G and check connectivity for each iteration that requires $\mathbb{I}_{\{\theta|G,\kappa\}}$.

Since each edge is present independently with probability p, where the probability is constant for both current and candidate digraphs, the probability of a digraph G is

$$\pi(G|\lambda, I, \kappa) = p^{|G|}(1-p)^{N(N-1)-|G|},$$

where $|G| = \sum_i \sum_j g_{ij}$, i.e. the number of edges in the digraph G.

Finally, the form of the starting digraph must be specified. For simplicity, and to guarantee a valid initial digraph, let $g_{1j} = 1$ for $2 \leq j \leq a + d$, $g_{ij} = 0$ for $i, j > a + d$ and $i = j$. Thus there are no self edges, the required d individuals are connected to an initial infected and there are no contacts to the remaining susceptibles.

3.2.1.2 Poisson Representation

The following representation by O'Neill (2009) demonstrates a key issue in MCMC, that an appropriate form of the likelihood can lead to a more efficient algorithm. By considering the digraph as a set of contact lists, it is possible to form a Gibbs update step for both the infection rate and digraph, though it is still necessary to check that the digraph is compatible with the observed final size data.

Let X be a $D = a + d$ length vector, $X = (x_1, x_2, \ldots, x_D)$, where x_i is the number of contacts individual i makes during its infectious period, including repeat contacts. Let C_i be a vector of the individuals i contacts, i.e. c_{i1} is the first individual contacted by individual i, where the length of C_i is x_i. As mentioned, we restrict to the sub-digraph consisting of only those individuals that are ultimately infected.

Recall, an infectious individual makes contacts with a given individual, uniformly selected from the population, at the points of a Poisson process of rate $\frac{\lambda}{N}$, over a period of length c. Thus x_i is the count of the corresponding Poisson processes. Note, although we restrict attention to the sub-digraph on D individuals, the infection rate is still normalised by the total population size N. The contacts are made uniformly with the population, since we are considering a homogeneously mixing population, thus each individual is equally likely, i.e. $\mathrm{P}(c_{ij} = k) = 1/N$ for all $k \in \{1, \ldots, N\}$. Define the digraph G to be the vector of Poisson process counts and the collection of contact lists for each individual, i.e. $G = \{X_1, C_1, \ldots, X_D, C_D\}$. Giving the probability of a digraph

G as,

$$\begin{aligned}\pi(G|\lambda, I, \kappa) &= \prod_{i=1}^{D} \left(\frac{\lambda}{N}c\right)^{x_i} \frac{\exp\left(-\frac{\lambda}{N}c\right)}{x_i!} \left(\frac{1}{N}\right)^{x_i} \\ &= \left(\frac{\lambda}{N}c\right)^{\sum_i x_i} \exp(-\frac{\lambda}{N}cD) \left(\frac{1}{N}\right)^{\sum_i x_i} \frac{1}{\prod_i x_i!}. \end{aligned} \tag{3.4}$$

Then the joint posterior density is,

$$\begin{aligned}\pi(G, \lambda|\theta, I, \kappa) &\propto \pi(\theta|G, \lambda, I, \kappa)\pi(G|\lambda, I, \kappa)\pi(\lambda) \\ &\propto \mathbb{I}_{\{\theta|G,\kappa\}} \ (\lambda c)^{\sum_i x_i} \ \exp(-\lambda cD) \ \exp(-\mu\lambda) \\ &\propto \mathbb{I}_{\{\theta|G,\kappa\}} \ (\lambda)^{\sum_i x_i} \ \exp\left(-\lambda(cD+\mu)\right), \end{aligned} \tag{3.5}$$

up to proportionality, and ignoring constant factors that will cancel out in the acceptance probability.

Following Section 1.3.2.2, we select the proposal distribution for the infection rate as the full conditional distribution, i.e. $\pi(\lambda|G, \theta, I, \kappa)$. From Equation (3.5), if the proposal is a gamma distribution with shape parameter $1 + \sum_{i=1}^{D} x_i$ and rate parameter $Dc + \mu$, then such a proposal is the full conditional distribution, i.e. $q(\lambda'|\lambda) \sim \Gamma\left(1 + \sum_{i=1}^{D} x_i, Dc + \mu\right)$. It is simple to check the resulting update is a Gibbs step and has an acceptance probability of one.

To update the digraph G, we see from Equation (3.4) and by the construction of the digraph, that the number of contacts for each individual has a Poisson distribution, independent of all other individuals. Thus, choosing an individual at random (from among those that are ultimately infected), proposing a new number of contacts X_i' according to a Poisson distribution of mean $\frac{\lambda}{N}c$ and uniformly assigning these contacts among the population to propose a candidate, C_i', is the full conditional distribution,

$\pi(G|\lambda, \theta, I, \kappa)$. It follows immediately that the acceptance probability is one, provided the candidate digraph G' is compatible with the observed data θ.

For the seed digraph, to ensure a valid starting configuration, set $x_i = 1$ for $1 \leq i \leq D-1$ and zero otherwise, with $c_{i1} = i+1$. This is a minimal tree, with a single branch including all the vertices. That is, each generation consists of a single individual who has a single contact during their infectious period. This contact is with a susceptible, who becomes the single infective in the next generation.

Hence using the Poisson representation of the digraph it is possible to form Gibbs updates for the infection rate and for the imputed digraph is also a Gibbs step, provided the candidate digraph is valid with respect to the final size data.

The algorithms presented, using the connectivity matrix or Poisson representation, both require a method to check the validity of a candidate digraph. A simple recursive approach was presented in Section 3.2.1.1, which is applicable to any population size. Using the Gibbs updates results in a more efficient algorithm, in terms of computation time, since all proposals are accepted removing the computational cost of evaluating an acceptance probability.

3.2.2 Imputing Generation Representation

In Chapter 2 we began by investigating directed random graphs characterised by their edges and then proceeded to consider representing the digraph in terms of its generations. The generations approach reduces the amount of information recorded about the digraph, specifically the details of each individuals contacts are no longer known. However, in terms of our MCMC approach, this loss of information is about the imputed data. Our aim is to augment the parameter space with the minimal information

necessary to form a likelihood, thus we now consider the generations representation as a sufficient and more efficient augmentation.

Another benefit of the generation representation is the simplification of checking the imputed digraph corresponds to the observed data. For the edge representations a recursive search technique was necessary to check connectivity. Thus there is a saving in computation for the generations approach, however there is no proposal distribution giving rise to a Gibbs update, which is a benefit of the Poisson representation. There is a balance between these benefits for different situations.

3.2.2.1 Notation And Definitions

The generation representation is as described in Section 2.4, there are N individuals of which a are initial infectives and n are initial susceptibles, $a + n = N$. Initial infectives are members of the zeroth generation, those they directly infect are the first generation and so on for each successive generation. Recall, we use the term generation as equivalent to rank, though this is not the case for the temporal definitions. Similarly, we shall use the term path instead of rank chain to emphasis the application to epidemics. Denote the observed data, i.e. the population and final size, as either Expression (3.1) using the vector $\theta = (a, n, d)$ or Expression (3.2) using the vector $\psi = (N, D)$.

Let Z be the random variable denoting the path of an epidemic, Z is a vector of two dimensional vectors (effectively a matrix), i.e. $Z = (Z_0, Z_1, \ldots, Z_d, Z_{d+1})$ consisting of the vectors $Z_t = (X_t, Y_t)$ for $0 \leq t \leq d + 1$. The random variable X_t is the number of individuals of rank t, and $Y_t = \sum_{i=0}^{t} X_i$ is the cumulative sum. It is sufficient to specify only the size of each generation, the cumulative sum is used to simplify expressions and in a practical sense can be stored to aid computation and as a rapid checking tool.

Since we restrict attention to digraphs that are valid and d connected, the largest attainable rank is d. For a given path $Z = z$, denote by τ the last non-zero generation, i.e. $\tau = \max\{t : X_t > 0\}$.

From Section 2.4.4, using the notation $P_\theta[E]$ to denote the probability of an event E given $\theta = (a, n, d)$, i.e. a initial infectives, n initial susceptibles and a final outcome of d. From Equation (2.20), the probability of a given digraph is

$$\mathrm{P}_\theta[Z = z] = \prod_{t=0}^{\tau} \mathrm{P}_\theta[Z_{t+1} = (x_{t+1}, y_{t+1}) | Z_t = (x_t, y_t)], \tag{3.6}$$

and by Equation (2.24) for a fixed infectious period $I = c$,

$$\begin{aligned}
&\mathrm{P}_\theta[Z_{t+1} = (x, y) | Z_t = (u, v)] \\
&\quad = \binom{(r+s)-v}{x} \left(1 - \exp(-\frac{\lambda}{N}c)^u\right)^x \left(\exp(-\frac{\lambda}{N}c)^u\right)^{r+s-y} \\
&\quad = \binom{(r+s)-v}{x} \sum_{k=0}^{x} (-1)^{x-k} \binom{x}{k} \exp\left(-\frac{\lambda}{N}c(r+s-v-k)\right)^u .
\end{aligned} \tag{3.7}$$

Using the path Z as a representation of the course of the epidemic, we can augment the likelihood as before, to obtain the joint posterior density of the path and infection rate as the product of an indicator function, the likelihood of a given path and the prior.

$$\begin{aligned}
\pi(z, \lambda | \theta, I, \kappa) &\propto \pi(\theta | z, \lambda, I, \kappa) \pi(z, \lambda | I, \kappa) \\
&\propto \pi(\theta | z, \lambda, I, \kappa) \pi(z | \lambda, I, \kappa) \pi(\lambda | I, \kappa) \\
&\propto \mathbb{I}_{\{\theta | z, \kappa\}} \; \pi(z | \lambda, I, \kappa) \; \pi(\lambda).
\end{aligned} \tag{3.8}$$

This is the density we explore using our MCMC algorithm, drawing approximate samples to estimate the marginal posterior density of the infection rate λ, the parameter of interest. In the following two sections we discuss the proposal distributions for the

two parameters, λ and z, describing their update algorithms in detail.

The data used for parameter inference is only the length two vector $\psi = (N, D)$, if we consider a an unknown parameter, or the length three vector $\theta = (a, n, d)$, from which we are attempting to estimate the joint density of λ, κ and z (which will include the imputed value of a if this is treated as an unknown). Attempting to make inference on so many correlated parameters from two or three numbers is clearly rather optimistic.

3.2.2.2 λ Update Steps

For the infection rate λ we use a proper prior, an exponential with rate parameter μ and a symmetric Random Walk Metropolis proposal for new values using a normal distribution, i.e. $\lambda' \sim \mathrm{N}(\lambda, \sigma^2)$; rejecting any negative proposals to ensure the candidate is non-negative, i.e. $\pi(\lambda) = 0$ for $\lambda < 0$. The proposal variance σ^2 is a tunable hyperparameter.

The acceptance probability for the candidate λ' is the minimum of one and the ratio of the likelihoods and probability of the candidate and current state under the proposal distribution, as defined in Section 1.3.2.1, thus

$$\begin{aligned}\alpha(\lambda, \lambda') &= \min\left\{1\,,\, \frac{\pi(z, \lambda'|\theta, I, \kappa)q(\lambda|\lambda')}{\pi(z, \lambda|\theta, I, \kappa)q(\lambda'|\lambda)}\right\} \\ &= \min\left\{1\,,\, \frac{\pi(z|\lambda', I, \kappa)\pi(\lambda')}{\pi(z|\lambda, I, \kappa)\pi(\lambda)}\right\}.\end{aligned}$$

Where $\pi(\lambda)$ is the prior distribution, an exponential with rate μ and $\pi(z|\lambda, I, \kappa)$ is the likelihood of the current path give the infection rate, given by Equations (3.6) and (3.7). The full infection rate update is shown in Algorithm 3.1.

For a suitable seed value for the infection rate, it is common to perform a short trial run

Algorithm 3.1: λ-update for one-type one-level model

```
Propose λ' ~ N(λ, σ²);
if λ' < 0 then
    reject
Calculate acceptance probability α(λ, λ');
Draw A ~ U(0, 1);
if α < A then
    reject λ'
else
    accept λ'
```

of the MCMC algorithm to obtain an estimate if there is no specific guidance otherwise, though this is generally unnecessary.

3.2.2.3 Z Update Steps

Using the edge representation of Section 3.2.1 there was a natural method to update the digraph, adding and removing edges. For the generation representation, we no longer retain specific information on an individual, only the generation in which it appears. Hence, the natural update is to alter the generation an individual belongs to. As stated in Section 2.4.1, there are 2^{d-1} possible valid paths of final size d, and it is not feasible to integrate out the path parameter, Z, to obtain the posterior density of the infection rate alone, hence we are using an augmented MCMC algorithm to estimate the joint posterior density.

The structure of Z, an epidemic path conditioned on a final size of d was investigated in Section 2.4, we shall use that information to form the initial seed path and to motivate the update techniques that follow. Our motivation for imputing the path over edges is use the minimal information necessary, in particular specific details of each individual are no longer recorded. Thus, we no longer label the individuals explicitly and only know the size of each generation, not which individuals comprise it.

m	ABC	z	m	ABC	z
0	000	(4)	4	100	(1,3)
1	001	(3,1)	5	101	(1,2,1)
2	010	(2,2)	6	110	(1,1,2)
3	011	(2,1,1)	7	111	(1,1,1,1)

Table 3.1: Example correspondence between path index and path using binary representation

Since the cumulative totals are a function of the generation sizes, we can consider the a path as a vector denoted $(x_0|x_1, \ldots, x_\tau, x_{\tau+1})$, where $x_0 = a$ and $x_{\tau+1} = 0$ by definition. The bar (|) is used to separate the zeroth generation to emphasis that it is fixed. We wish to propose a new candidate path, z', in such a way that we can explore the space of all possible paths and consider candidates that are in some sense close to the current path.

Independence Sampler An obvious proposal would be an independence sampler. Specifically, we can enumerate the set of all possible paths, $\{z^{(m)} : 0 \leq m \leq 2^{d-1} - 1\}$, then select a new path uniformly. To derive the m^{th} path for $0 \leq m \leq 2^{d-1} - 1$ we can convert from a binary representation of the index to a path. To illustrate the correspondence between a path index m and $z^{(m)}$, consider the following example. Let $d = 4$, so there are 8 possible paths, then consider the four individuals in a row with the three spaces between them. Label these spaces A, B and C. Then we relate the binary representation of the index to the presence of dividing lines in these spaces. Scan from left to right along the line of objects, we move to the next generation when we meet a dividing line. Such a correspondence is shown in Table 3.1 for the case $d = 4$. The proposal is simple to implement and the resulting acceptance probability is reduced to the ratio of the densities, as the proposal is a uniform distribution, i.e. $q(z'|z) = q(z|z') = \frac{1}{2^{d-1}}$. However for moderate d, the space of valid paths is large but only a small subset have a high (marginal) posterior density. Thus many proposed candidates will be rejected and the chain will mix poorly.

Using the path index as described, it is difficult to consider paths that are 'close', in order to achieve a higher acceptance rate. For example, if $d = 8$ then $z^{(64)} = (1, 7)$ and paths that are 'close' in terms of index are $z^{(63)} = (2, 1, 1, 1, 1, 1, 1)$ and $z^{(65)} = (1, 6, 1)$, which have very different likelihoods for a given infection rate.

We have not yet properly defined when two paths are 'close'. If we consider all paths as vectors of length d, then we can define the Euclidean norm as the distance Δ, between two paths $z^j = (x_1, \ldots, x_d)$ as

$$\Delta(z^1, z^2) = ||z^1 - z^2||_2 = \sqrt{\sum_{i=1}^{d} (x_i^1 - x_i^2)^2}.$$

Where $||\cdot||_2$ is the L2-norm. From the index of a path it is not immediately possible to determine its distance from another path. When the space of paths is large there is no direct method to obtain the set of paths within a given distance of the current path in terms of their index, i.e. for the path indexed by m and a distance ϵ, the set of indicies $\{i : \Delta(z^{(m)}, z^{(i)}) \leq \epsilon\}$. It would be necessary to compute $\Delta(z^{(i)}, z^{(j)})$ for all $0 \leq i, j \leq 2^{d-1} - 1$ before running the MCMC algorithm. How to specify the distance ϵ is also uncertain, as the relationship of $\Delta(z, z')$ to the ratio of the marginal posterior densities $\frac{\pi(z'|\cdot)}{\pi(z|\cdot)}$, is complex. Given the shape of the state space, consisting of all valid paths, $\{z^{(m)} : 0 \leq m \leq 2^{d-1} - 1\} \subset \mathbb{Z}_+^d$, it is not clear that such a distance is well suited to selecting candidate paths, in fact the state space is an integer simplex, since $\sum_i x_i = d$ for all paths, and this structure should also be taken into account.

K-jump Proposal Instead we consider a candidate that differs from the current state by a single individual, who has been moved from its current generation to a different one. Such a candidate will always differ in two generations, thus the distance will be $\Delta(z, z') = \sqrt{2}$, which is the minimal distance any two distinct valid paths can

differ by.

We shall term our proposal a K-jump, where K determines the number of generations an individual is moved. Given $K = k$, we determine the generations that contain individuals that can be moved k generations, this is done to ensure the candidate is a valid path. Alternatively, it would be possible to select the length of jump, move an individual and then check if the path was still valid. This alternative procedure is sufficient for the one-type SIR model, but not for the two-level mixing model we shall discuss in Section 3.5. In particular, the rejection rate due to proposing an invalid path becomes prohibitively high. Thus we outline the more complex method for the simple one-type one-level model first, to introduce the notation and approach.

The update is performed as follows. We determine the range for K such that there is at least one individual who can move for each value. A specific $K = k$ is then chosen uniformly from this range. The current path is scanned to find generations with individuals that can be moved k, the total number of possible moves is counted and denoted by $\mathcal{J}_k$. One such move is chosen uniformly from among the $\mathcal{J}_k$ and an individual is moved, forming the candidate path, z'. The proposal distribution is a product of uniform random variables determined by the origin path. The proposal is guaranteed to be reversible, since the individual moved can be returned with a jump of the same length.

For the K-jump we introduce the hyperparameter $K_{\max} \geq 1$, this limits the range of K, that is the furthest an individual can be moved in a single K-jump. For the one-type one-level model, the value of $K_{\max}$ primarily effects the length of the burn in period. It is introduced as a tunable hyperparameter to increase the acceptance rate in the two-type model.

We assume the initial number of infectives a is fixed, so we must take care with the

shortest valid path $z^{(0)} = (a|d, 0)$. For this path, the length is $\tau^{(0)} = 1$ and this is the unique path of length one. The only possible jump is of length one, moving an individual from the first to the second generation. Thus for $\tau = 1$ we have $K \sim \text{Uni}[1, 1]$, i.e. there is only one choice.

For paths of length greater than one, $z = (a|x_1, \ldots, x_\tau, 0)$, then an individual can always be moved from τ^{th} to the 1^{st} generation, a jump of $k = \tau - 1$ generations. This may result in the length of the candidate being different to the current path, i.e. if $x_\tau = 1$ then moving the single individual will result in a shorter path. Conversely, if $x_1 > 1$ then it is possible to move an individual from the 1^{st} to the $(\tau+1)^{\text{th}}$ generation, a jump of $k = \tau$, which will result in a longer candidate path. It is important to note that the jump of length τ only results in a valid candidate path if the first generation has more than one individual, otherwise the move will result in $z' = (a|0, x_2, \ldots, x_\tau, 1, 0)$ which is an invalid path.

Thus to determine the range of possible jumps we must consider the length of the current path and the size of the first generation, combining the above we have $1 \leq k \leq \mathcal{K} = \min\{(\tau - 1 + \mathbb{I}_{\{x_1>1\}}), K_{\max}\}$. There is no reason to prefer any jump length, thus we propose the length from a discrete uniform, i.e. $K \sim \text{Uni}[1; \mathcal{K}]$.

It is important to note that, so far we have not determined how many possible jumps there are for $K = k$. Only that there is at least one such jump resulting in a valid candidate path, since if an individual can be moved k generations then is could be moved $k - 1$ generations; if $x_1 > 1$ and for $1 \leq k \leq \mathcal{K}$, then an individual can always be moved from the first generation forward k generations. Intuitively, there will be more valid moves for smaller k. Also, we have not yet determined the exact method to construct the candidate path.

Let $K = k$ and suppose that for a given path z in generation t there are x_t individuals.

We wish to move an individual to a new generation. This can either be to an earlier or later part of the path which we shall term backward and forward jumps respectively. For generation t, a backward k-jump is possible if the candidate path z', constructed such that $x'_t = x_t - 1$ and $x'_{t-k} = x_{t-k} + 1$, is a valid path. Similarly, a forward k-jump is possible if the candidate path z', constructed such that $x'_t = x_t - 1$ and $x'_{t+k} = x_{t+k} + 1$, is a valid path.

There are criteria to determine if a backward or forward k-jump is possible for each generation $1 \leq t \leq \tau$, and we define the function $J_k(x_t)$ as an indicator of this. Let $J_k(x_t)$ be 0, 1, -1 or 2 corresponding to none, only forward, only backward or both k-jumps are possible for generation t. Let $J_k(z) = (J_k(x_0)|J_k(x_1), \ldots, J_k(x_\tau), 0)$ be the function applied to the entire path. The criteria are:

$$J_k(x_t) = \begin{cases} -1 & \begin{cases} \text{if } x_t > 1 \text{ and } k < t < \tau, \\ \text{if } t = \tau \text{ and } k < \tau, \end{cases} \\ 1 & \text{if } x_t > 1 \text{ and } t + k \leq \tau + 1, \\ 2 & \text{if both,} \\ 0 & \text{otherwise,} \end{cases} \tag{3.9}$$

where $1 \leq t \leq \tau$ and $1 \leq k \leq \mathcal{K} = \min\{(\tau - 1 + \mathbb{I}_{\{x_1 > 1\}}), K_{\max}\}$. Recall that the possible ranges of t and k have been derived earlier. The criteria for a forward k-jump require that the target generation $t + k$ not be beyond $\tau + 1$, otherwise the path will not be valid. In addition, the origin generation, t, must have more than one individual, otherwise moving them will result in a zero generation part way through the path.

Similarly, for the backward k-jump, we cannot move to a generation before the first (we assume the zeroth generation is fixed), thus $k < t < \tau$ and again the origin generation must have more than one individual, i.e. $x_t > 1$. The special case when $t = \tau$ is needed

since we can always move an individual from the τ^{th} generation backwards, which may result in a shorter path, though we must still check that $k < t = \tau$ since the range of valid k can include τ (in the case when a forward jump from the 1^{st} to $(\tau+1)^{\text{th}}$ generation is valid).

Equation (3.9) will be implemented in our MCMC algorithm to determine and construct candidate paths. We defined $\mathcal{J}_k$ to be the total number of possible k-jumps for the path z, hence

$$\mathcal{J}_k = \sum_{t=1}^{\tau} |J_k(x_t)|. \tag{3.10}$$

We shall select one of the possible k-jumps uniformly, so let g be the index of the chosen jump, $g \sim \text{Uni}[1; \mathcal{J}_k]$. Once a jump g is selected, it is necessary to determine the corresponding generation and direction. This is done by scanning the vector $J_k(z)$ and determining the g^{th} entry. The origin generation t_O is defined by,

$$t_O = \min\left\{ t : g \leq \sum_{i=1}^{t} |J_k(x_i)| \right\}.$$

The direction δ of the move is either backwards ($\delta = -1$) or forwards ($\delta = 1$), care must be taken to account for the generations where both backward and forward moves are valid.

$$\delta = \begin{cases} J_k(x_{t_O}) & \text{if } J_k(x_{t_O}) = \pm 1 \\ 1 & \text{if } J_k(x_{t_O}) = 2 \text{ and } g - \sum_{i=1}^{t_O} J_k(x_i) = 0 \\ -1 & \text{if } J_k(x_{t_O}) = 2 \text{ and } g - \sum_{i=1}^{t_O} J_k(x_i) = -1 \end{cases}.$$

The definitions of the origin generation, t_O and the direction δ, are in terms of the counting function J_k and presented in an algorithmic form. This obscures the simple principle behind the K-jump update, thus we shall present an example shortly.

As mentioned above, the procedure is overly complicated for the one-type one-level model we are considering. In particular the need to scan the path twice, first to determine the number of valid k-jumps, $\mathcal{J}_k$, and then to determine the origin generation t_O, and direction δ, requires many additional calculations. The procedure is designed to construct valid candidate paths, since in contrast, proposing arbitrary k-jumps and then checking whether the path is valid becomes less efficient for the extensions to the model in Section 3.3, in particular see Section 3.3.6.3.

Finally, we construct the candidate path z' as,

$$x'_{t_O} = x_{t_O} - 1$$
$$x'_{t_O+\delta k} = x_{t_O+\delta k} + 1.$$

For a given k, there are $\mathcal{J}_k$ possible candidate paths which are all unique. In total there are $\mathcal{J} = \sum \mathcal{J}_k$, where the sum is over the range of valid k. All such paths are unique and consist of all the paths whose distance from the current path is $\sqrt{2}$, i.e. they differ from the current path by moving a single individual.

Not all $\mathcal{J}$ candidate paths are proposed with equal probability, larger jumps are more likely to be proposed. We determine the length of jump k before considering the number of possible candidate paths, though by the construction of the range of k there exists at least one such valid candidate. Clearly, $\mathcal{J}_1 \geq \mathcal{J}_2 \geq \cdots \geq \mathcal{J}_\mathcal{K}$ where $\mathcal{K} = \min\{(\tau - 1 + \mathbb{I}_{\{x_1>1\}}), K_{\max}\}$, since the criteria reduce the number of potential generations where a k-jump is possible to $k < t \leq \tau$ and if a generation is k-jumpable then it is $(k-1)$-jumpable, i.e. $J_k(x_t) \leq J_{k-1}(x_t)$.

Assume we propose z' from z using a K-jump, so there is a unique $1 \leq k \leq \mathcal{K}$ and a

unique $1 \leq g \leq \mathcal{J}_k$ corresponding to the proposal, thus the proposal probability is

$$q(z'|z) = \frac{1}{\mathcal{K}} \frac{1}{\mathcal{J}_k}.$$

The proposal distribution is not in general symmetric between candidate and current paths, i.e. $q(z'|z) \neq q(z|z')$. Importantly, $\mathcal{J}_k$ depends on the k chosen so candidates of different lengths have different probabilities of being proposed.

The acceptance probability can only be calculated after the first stage of the proposal is determined, i.e. when the length k is chosen.

$$\begin{aligned}
\alpha(z, z') &= \min\left\{1\,,\, \frac{\pi(z', \lambda|\theta, I, \kappa) q(z|z')}{\pi(z, \lambda|\theta, I, \kappa) q(z'|z)}\right\} \\
&= \min\left\{1\,,\, \frac{\mathbb{I}_{\{\theta|z',\kappa\}} \pi(z'|\lambda, I, \kappa) \frac{1}{\mathcal{K}'} \frac{1}{\mathcal{J}'_k}}{\mathbb{I}_{\{\theta|z,\kappa\}} \pi(z|\lambda, I, \kappa) \frac{1}{\mathcal{K}} \frac{1}{\mathcal{J}_k}}\right\} \\
&= \min\left\{1\,,\, \frac{\pi(z'|\lambda, I, \kappa)\ \mathcal{K}\ \mathcal{J}_k}{\pi(z|\lambda, I, \kappa)\ \mathcal{K}'\ \mathcal{J}'_k}\right\}.
\end{aligned}$$

By construction, the candidate is always a valid path and so $\mathbb{I}_{\{\theta|z',\kappa\}} = \mathbb{I}_{\{\theta|z,\kappa\}} = 1$. If the origin and target generation do not include the first or last generation, then $\mathcal{K}' = \mathcal{K}$ since the length of the candidate and current path are the same. However, even for this case, in general $\mathcal{J}'_k \neq \mathcal{J}_k$. The complete K-jump update is summarised in Algorithm 3.2.

Example Of K-jump We present the following example to clarify the K-jump update, the proposal distribution and the construction of a candidate path. Using an example of $\theta = (a, n, d) = (1, n, 5)$, that is a single initial infective and a final size of five. The size of the population need not be specified since it is not required for the proposal and construction. The population size is accounted for in the likelihood of the path z, where the number of initial susceptibles that escape infection, $n - d$.

Algorithm 3.2: Z-update using a K-jump for one-type one-level model

Let $\mathcal{K} = \min\{(\tau - 1 + \mathbb{I}_{\{x_1>1\}}), K_{\max}\}$;
Sample $k \sim \text{Uni}[1, \mathcal{K}]$;
Calculate the vector $J_k(z) = (J_k(x_0)|J_k(x_1), \ldots, J_k(x_\tau))$;
Calculate $\mathcal{J}_k = \sum_{i=1}^{\tau} |J_k(x_i)|$;
Sample $g \sim \text{Uni}[1, \mathcal{J}_k]$;
Determine the origin t_O, and direction δ corresponding to g;
Construct the candidate path z';
Calculate acceptance probability α;
Draw $A \sim \text{U}(0, 1)$;
if $\alpha < A$ **then**
    reject z'
else
    accept z'

Let the current path be $z = (1|2, 2, 1, 0)$ where $\tau = 3$. For clarity, we shall re-write the path as the transpose of this row vector, include the length τ as a subscript, and exclude the $(\tau + 1)^{\text{th}}$ generation. Then the path z is expressed as the column vector,

$$z = \begin{bmatrix} 1 \\ \hline 2 \\ 2 \\ 1 \end{bmatrix}_3 .$$

First we determine the valid range of k. Let $K_{\max} = \infty$ for this example, since the final size is so small and the number of candidate paths are easy to manage. Since larger jumps are more likely to be proposed, the tunable parameter $K_{\max}$ can be used to ensure that the probability of small jumps is not too small, we shall return to this in Section 3.3. Using the path z we have

$$\mathcal{K} = \min\{(\tau - 1 + \mathbb{I}_{\{x_1>1\}}), K_{\max}\} = \min\{(3 - 1 + 1), \infty\} = 3.$$

Instead of selecting a specific $K = k$, $1 \leq k \leq 3$, we shall construct all the possible

candidate paths and their probabilities. Using the transpose notation for $J_k(z)$ we have,

$$z = \begin{bmatrix} 1 \\ \hline 2 \\ 2 \\ 1 \end{bmatrix}_3 \begin{cases} \text{if } k=1,\ J_1(z) = \begin{bmatrix} 0 \\ \hline 1 \\ 2 \\ -1 \end{bmatrix}, \mathcal{J}_1 = 4 \\ \text{if } k=2,\ J_2(z) = \begin{bmatrix} 0 \\ \hline 1 \\ 1 \\ -1 \end{bmatrix}, \mathcal{J}_2 = 3 \\ \text{if } k=3,\ J_3(z) = \begin{bmatrix} 0 \\ \hline 1 \\ 0 \\ 0 \end{bmatrix}, \mathcal{J}_3 = 1 \end{cases}$$

Thus there are eight possible candidate paths, $\mathcal{J} = \sum_k \mathcal{J}_k = 8$. Each is uniquely indexed by the pair (k, g), where k is the length of jump and $1 \le g \le \mathcal{J}_k$. Below are shown all eight possible paths,

$$(1,1) = \begin{bmatrix} 1 \\ \hline 1 \\ 3 \\ 1 \end{bmatrix}_3 \quad (1,2) = \begin{bmatrix} 1 \\ \hline 3 \\ 1 \\ 1 \end{bmatrix}_3 \quad (1,3) = \begin{bmatrix} 1 \\ \hline 2 \\ 1 \\ 2 \end{bmatrix}_3 \quad (1,4) = \begin{bmatrix} 1 \\ \hline 2 \\ 3 \end{bmatrix}_2$$

$$(2,1) = \begin{bmatrix} 1 \\ \hline 1 \\ 2 \\ 2 \end{bmatrix}_3 \quad (2,2) = \begin{bmatrix} 1 \\ \hline 2 \\ 1 \\ 1 \\ 1 \end{bmatrix}_4 \quad (2,3) = \begin{bmatrix} 1 \\ \hline 3 \\ 2 \end{bmatrix}_2 \quad (3,1) = \begin{bmatrix} 1 \\ \hline 1 \\ 2 \\ 1 \\ 1 \end{bmatrix}_4 .$$

The probability of proposing a specific 1, 2 or 3 jump is $\frac{1}{3}\frac{1}{4} = \frac{1}{12}$, $\frac{1}{3}\frac{1}{3} = \frac{1}{9}$ and $\frac{1}{3}\frac{1}{1} = \frac{1}{3}$ respectively.

The unequal candidate probabilities at first seems a problem, given our concept of two paths being 'close', such an imbalance would seem to move around the state space in an odd manner, especially compared to the independence sampler. However, our distance metric does not account for the epidemic process that the path represents.

For example, moving an individual from the last to the first generation has the following effect. The second generation is infected by one more individual, then it and every subsequent generation must fail to infect one less individual. Since originally the moved individual was in the last generation, it avoided infection by all but the penultimate generation of infectives. This will cause a great effect on the likelihood for the candidate. If however, the individual was moved only a single generation the effect is much less. Hence, longer jumps are proposed more often but may have a lower acceptance probability, whereas shorter jumps are proposed less often but are more likely to be accepted. Together the likelihood and unequal proposal probabilities counteract each other to a certain degree. To influence this balance we introduce the tunable parameter $K_{\max}$.

3.2.2.4 Algorithm

The generation representation is used to augment the likelihood, we have presented the update steps for the infection rate parameter λ using a symmetric Random Walk Metropolis and the imputed path Z using a specified proposal distribution, in order to obtain an estimate for the joint posterior density $\pi(Z, \lambda|\theta, I, \kappa)$.

For the generation approach there are two tunable hyperparameters, namely the vari-

ance of the proposal distribution for λ and the maximum jump length of the K-jump update of Z. Both updates require the calculation of an acceptance probability, compared to the Gibbs updates of the Poisson representation.

Each parameter is updated independently, though we expect there to be a correlation between the infection rate and the path in the joint posterior. This may affect the mixing of the algorithm and we investigate this in the following section.

Since the space of all possible paths is large, it is reasonable to perform multiple Z-updates between λ-updates. This is a common strategy to aid mixing in MCMC algorithms, particular for imputed data since we are only interested in the marginal posterior density for λ. Hence for each iteration we obtain a single approximate sample from $\pi(\lambda|\cdot)$.

3.2.3 Results And Comparison To Estimates In The Literature

We shall consider two data sets, the first of which is commonly cited in the epidemic literature. The methods used to estimate the infection rate vary, and care must be taken to make direct comparison between various methods. Secondly, we consider the example data sets in Demiris and O'Neill (2005a), specifically the application to single-type homogeneously mixing data. Demiris and O'Neill (2005a) use a random directed graph to augment the likelihood in their MCMC algorithm, specifically the edge representation in Section 3.2.1.1. It is then a fair comparison between the estimates from the edge and generation methods.

	Gaussian Method	Generation Method
mean	1.177	1.183
median	1.165	1.171
s.d.	0.211	0.217

Table 3.2: Comparison of estimates for the infection rate λ, reported as $R_0 = 4.1\lambda$, between the Gaussian method of Demiris (2004) and the generation method of Section 3.2.2. On $\theta_1 = (1, 119, 29)$ using a fixed infectious period of 4.1 days.

3.2.3.1 Comparison To Classical Data And Gaussian Method

The first data set we consider consists of a total population of $N = 120$, in which we observe a total of $D = a + d = 30$ or $D = 60$ individuals who were infected. Following our assumptions, let $a = 1$ and hence $\theta_1 = (a, n, d) = (1, 119, 29)$ and $\theta_2 = (1, 119, 59)$. The infectious period is a constant, $I = c$, that must be specified prior to the MCMC. To compare to results in the literature we let $c = 4.1$, to give an infectious period of 4.1 days. Though we make inference for the infection rate λ, we report the reproductive number R_0. For the one-type one-level model, $R_0 = \mathrm{E}[I]\lambda = c\lambda$, given a fixed infectious period of length c. Recall that R is a threshold, such that the final size in an infinite population is finite almost surely for $R \leq 1$.

For the data augmentation approach, either using edges or generations, we do not condition on $R > 1$, i.e. we do not assume the epidemic is above threshold. For many classical inference results such an assumption is necessary to derive the estimators for λ, for example the martingale approach derived in Becker (1989). Demiris (2004) (see also Demiris and O'Neill (2005b)) use a final size approximation, using the Gaussian final size result in Section 1.2.4, in an MCMC algorithm. The estimates are taken from Demiris (2004), and compared to the generation algorithm in Tables 3.2 and 3.3

	Gaussian Method	Generation Method
mean	1.424	1.429
median	1.413	1.421
s.d.	0.182	0.186

Table 3.3: Comparison of estimates for the infection rate λ, reported as $R_0 = 4.1\lambda$, between the Gaussian method of Demiris (2004) and the generation method of Section 3.2.2. On $\theta_2 = (1, 119, 59)$ using a fixed infectious period of 4.1 days.

corresponding to the data θ_1 and θ_2 respectively. The mean, median and standard deviation are calculated from the marginal posterior of λ.

The estimates agree for both data sets, the differences are expected from using the MCMC approximation and the different posterior densities. The generation method draws samples from the full posterior, $\pi(\lambda, z|\theta, I, \kappa)$, from which we obtain the marginal posterior density $\pi(\lambda|z, \theta, I, \kappa)$; where as the Gaussian method estimates the posterior density $\pi(\lambda|\theta, I)$.

As mentioned, many classical inference techniques assume $R_0 > 1$, for example Becker (1989) (p.153) estimate $R_0 = 1.10$ using a martingale approach. The restriction to an epidemic above threshold can cause artifacts in the estimates, in particular, it is common for confidence intervals to have their lower bound below one, despite the method conditioning on $R > 1$.

3.2.3.2 Comparison To Edge Representation

Demiris and O'Neill (2005a) consider three sample data sets, each with a population of $N = 100$ and D as 25, 50 and 75. We express these as three vectors, while assuming $a = 1$, i.e. $\theta_3 = (1, 99, 24)$, $\theta_4 = (1, 99, 49)$ and $\theta_5 = (1, 99, 74)$. They consider a

θ	Edge Method		Generation Method	
	Mean	(Standard Deviation)	Mean	(Standard Deviation)
$(1, 99, 24)$	1.16	(0.23)	1.17	(0.24)
$(1, 99, 49)$	1.42	(0.21)	1.42	(0.21)
$(1, 99, 74)$	1.86	(0.21)	1.88	(0.23)

Table 3.4: Comparison of estimates for the infection rate λ, reported as $R_0 = \iota\lambda$, between the Poisson method of Demiris and O'Neill (2005a) and the generation method of Section 3.2.2. On $\theta_3 = (1, 99, 24)$, $\theta_4 = (1, 99, 49)$ and $\theta_5 = (1, 99, 74)$ using a fixed infectious period of 1 day.

fixed infectious period with $c = 1$, since the infection rate and infectious period are indistinguishable it is an arbitrary decision. Setting an infectious period gives a scale to the epidemic, in terms of the temporal behaviour, thus the observed final size data θ can not be used for inference on this scale.

The reproductive number, $R_0 = \iota\lambda$, where $\iota = \mathrm{E}[I] = 1$, accounts for the infectious period, thus it is consistent between the two examples, despite the previous case using an infectious period of 4.1 days. Since the case of θ_1 and θ_3 are approximately equivalent, we expect similar results. From Tables 3.2 and 3.4, the estimated mean reproductive numbers are 1.18 and 1.17 respectively.

The results of Demiris and O'Neill (2005a) are compared to the generation algorithm in Table 3.4. As expected, for larger final sizes the estimates for the reproductive number increase. The generation method produces estimates consistent with the Poisson method. The results are directly comparable, both use an exponential prior on the infection rate, $\pi(\lambda)$, with rate $\mu = 10^{-6}$ and the models are identical. In particular the infectious period is constant and assumed to be one, $c = 1$ and the infection rate is normalised by the size of the population, i.e. $\frac{\lambda}{N}$. Care must be taken when comparing models to ensure they are equivalent, since normalising the infection rate is not done for

all models. In fact in Section 3.5, where we introduce multi-level models, the infection rates may be scaled in different ways.

3.2.3.3 Effect Of Seed And Tuning Hyperparameters On Burn In, Convergence And Mixing

The generation algorithm used to produce Tables 3.2, 3.3 and 3.4 was implemented in the C programming language. Each chain was run for 10^6 iterations and completed in 5 minutes. As discussed in Section 1.3.2.4, for the samples to be valid estimates of the joint posterior density the chain must converge to its stationary distribution.

The path z is a vector of length d, and it is difficult to determine suitable criteria in terms of z to test if the Markov Chain Monte Carlo has converged. For the infection rate λ we can qualitatively determine a burn in period from the trace plot.

One quick summary of a path z is its length, τ. Since there is no exact mapping between the number of generations and the length of the epidemic in real time, there is no specific interpretation to τ. However, as we showed in Section 2.5.6 using a discrete branching processes as an approximation to the generation representation, for a given final size d there is a limiting form to the average path as the population tends to infinity. Hence, there is a limiting value for the expected number of generations as $N \to \infty$, assuming such a result is valid then the expected value of τ should converge to this limit. Hence we shall use a trace plot of τ to determine if the chain has converged. For the one-type one-level model, the length of the path may be sufficient to summarise z, but it cannot convey the form of z. We shall return to this in Section 4.6.3, to define further summaries of the path in the general multi-type multi-level model.

The seed path, i.e. where the MCMC algorithm is initialised, will have no effect on

the final estimate of the joint posterior density, as it is estimated from the samples after the chain has converged. However, if the chain is started at a position of low posterior density it may take a long time to escape the region and converge. Thus, the seed will have an effect on the burn in period, i.e. the number of iterations that are ignored up to the point the chain has (approximately) converged. From Section 2.5.6, we expect the length of the path to be approximately $2\sqrt{d}$, using Figure 2.7 and the 'kink' in the variance shown in Figure 2.12. To demonstrate the effect of the initial seed we shall consider the two extreme paths, the unique path of maximal and minimal length, i.e. $z_{\max} = (1|1, 1, 1, \ldots, 1, 0)$ and $z_{\min} = (1|d, 0)$ where $\tau_{z_{\max}} = d$ and $\tau_{z_{\min}} = 1$ respectively (in terms of the enumeration of all possible paths, using the binary representation of each path, the extreme paths are $z^{(2^{29-1}-1)}$ and $z^{(0)}$).

Using the path length, τ, as our indicator for convergence of the Markov chain we have the trace plots for four runs in Figures 3.1 and 3.2. All four runs consider the same data, $\theta_1 = (1, 119, 29)$, using a fixed infectious period of 4.1 days and an exponential prior on λ with rate $\mu = 10^{-6}$. The λ-update is a symmetric Random Walk Metropolis as described in Section 3.2.2.2, the seed infection rate it set at 0.1. In each figure the two extreme seeds are used, with differing hyperparameter $K_{\max}$, the largest K-jump allowed.

The simplest update is to move an individual a single generation, this would propose very likely candidate paths, and would seem a reasonable update. Thus, letting $K_{\max} = 1$, we run an MCMC chain using the two extreme seeds and show the trace plots of τ in Figure 3.1. We clearly see the slow convergence of the length to the estimate, $\mathrm{E}[\tau] = 10.30$ (estimated from the second pair of runs using samples after convergence). The two chains in Figure 3.1 have not yet converged, since neither have crossed as of the 3000^{th} iteration. In particular, the slow rate of convergence from the maximal path is evident.

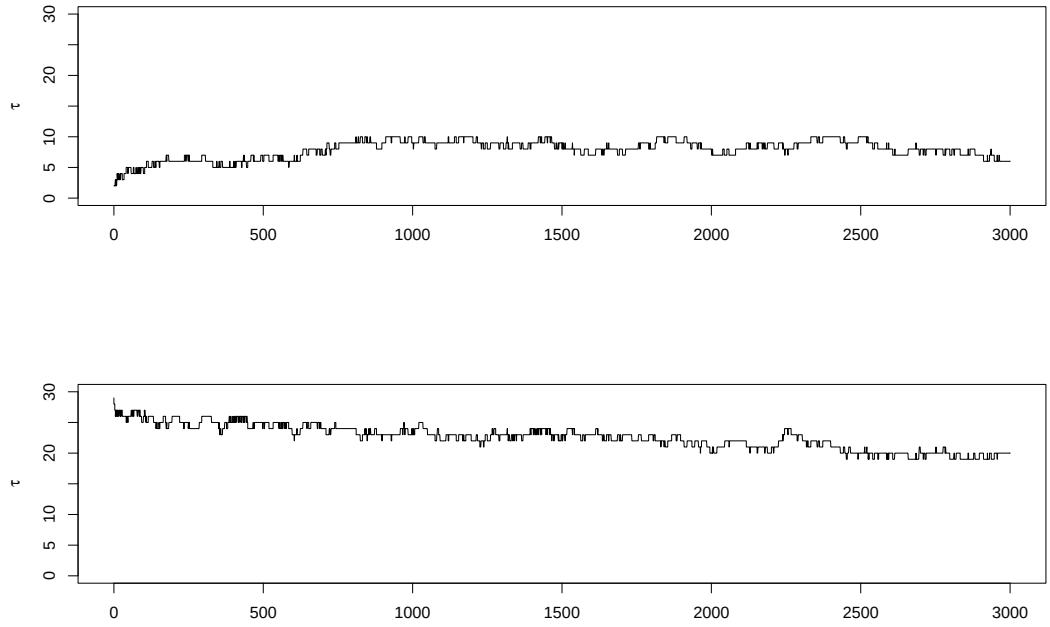


Figure 3.1: Comparison of burn in period between two seed paths with respect to the length of the path, τ, for the case $\theta_1 = (1, 119, 29)$ with a constant infectious period of 4.1 days. The K-jump tunable parameter is set to one, i.e. $K_{\max} = 1$

In comparison, Figure 3.2 uses the same seed paths, but with the maximum K-jump set to $K_{\max} = 15$. This has been chosen to be approximately $3\sqrt{d} = 3\sqrt{29} = 16.2$. The rate of convergence is much faster, from either seed path we reach the target region within 1000 iterations.

Using the chains from Figure 3.2 we can plot the marginal posterior density of τ, as shown in Figure 3.3. The MCMC algorithm, after convergence at the 1000^{th} iteration, draws approximate samples from $\pi(z, \lambda|\theta, I, \kappa)$. Since the path length is a function of z, we can compute its density from the samples, considering all samples drawn regardless of the value of λ gives the marginal posterior density.

The estimated mean is from the branching process approximation, which is independent

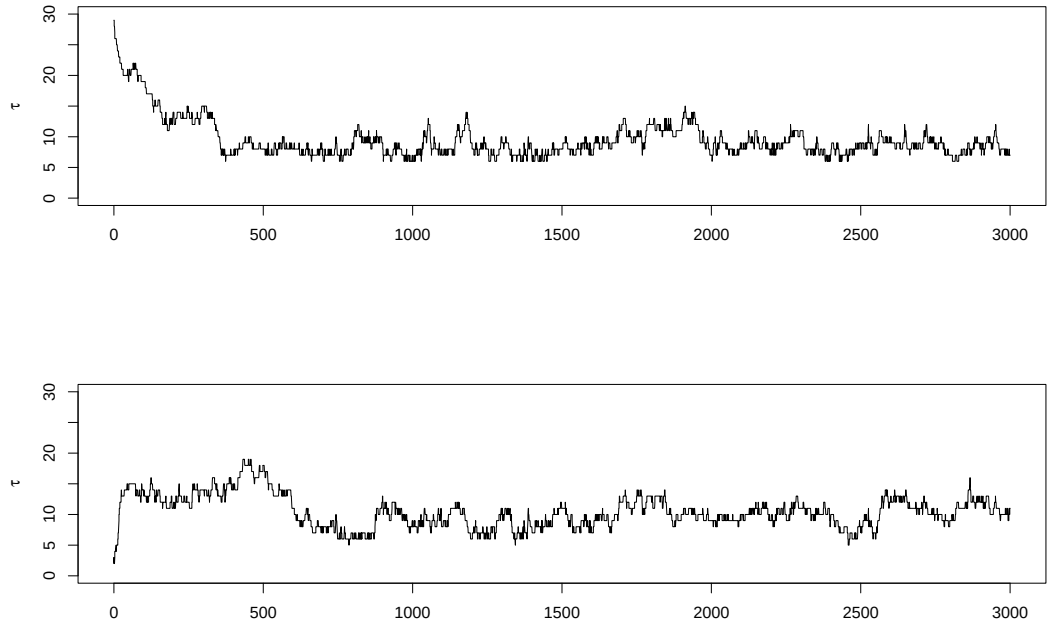


Figure 3.2: Comparison of burn in period between two seed paths with respect to the length of the path, τ, for the case $\theta_1 = (1, 119, 29)$ with a constant infectious period of 4.1 days. The K-jump tunable parameter is set to fifteen, i.e. $K_{\max} = 15$

of the infection rate λ and valid for large populations. The estimate of $2\sqrt{d} = 10.78$ gives a guide to suitable seed paths.

Instead of using the minimal or maximal path, we can use the estimate of the mean length to construct a seed path as follows. Let the generations be equal in size to $\lceil \frac{1}{2}\sqrt{d} \rceil$, where $\lceil\ \rceil$ denotes the ceiling function, until the d individuals have been assigned to a generation. The benefit of such a construction is to generate a seed path near the region of convergence.

The reason the latter chains converge quickly is in regard to how often the candidate path is of a different length to the current path. Recall, for a given k all jumps are equally likely. Thus if we limit to jumps of length one, i.e. $K_{\max} = 1$, then for the

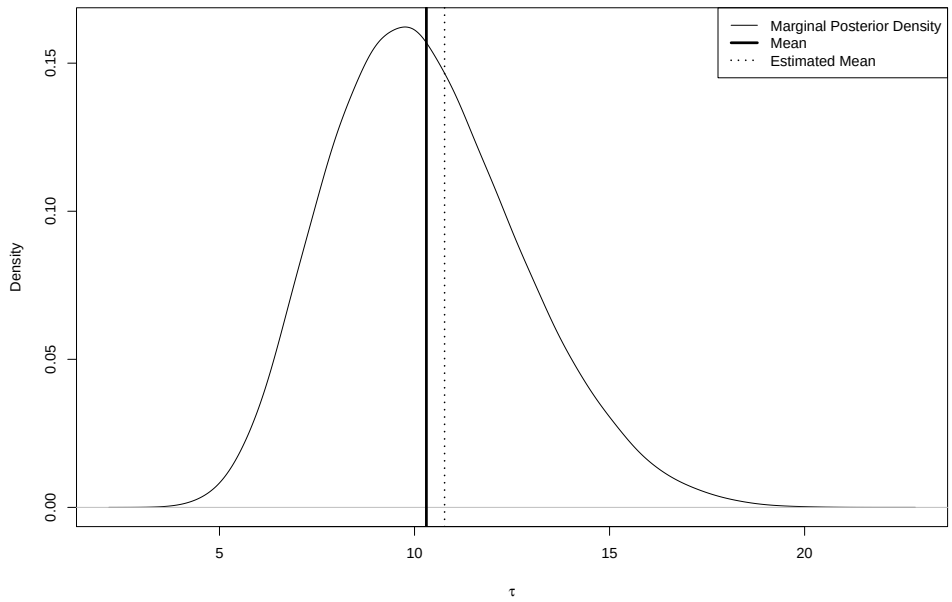


Figure 3.3: Marginal posterior density of the path length τ, for the case $\theta_1 = (1, 119, 29)$ with a constant infectious period of 4.1 days. The posterior mean is shown, as well as the estimate of $2\sqrt{d}$.

path to reduce in length there is only one possibility. The final generation must be of size one, then the probability of proposing a shorter path is $\mathcal{J}_1^{-1}$. However, if we let $K_{\max} = 2$ (assuming the path is at least length three so that a backward 2-jump is valid from the last generation) and $x_\tau = 1$, then the probability of proposing a shorter path is

$$\begin{aligned}\frac{1}{2}\frac{1}{\mathcal{J}_2} + \frac{1}{2}\frac{1}{\mathcal{J}_1} &\geq \frac{1}{2}\frac{1}{\mathcal{J}_1} + \frac{1}{2}\frac{1}{\mathcal{J}_1} \\ &= \frac{1}{\mathcal{J}_1}.\end{aligned}$$

Hence we are more likely to propose the candidate path that is shorter, meaning the length of the path has the potential to change more rapidly. A similar argument applies to increasing the length of the path, a larger $K_{\max}$ means there are more potential individuals that can be moved to the $(\tau+1)^{\text{th}}$ generation.

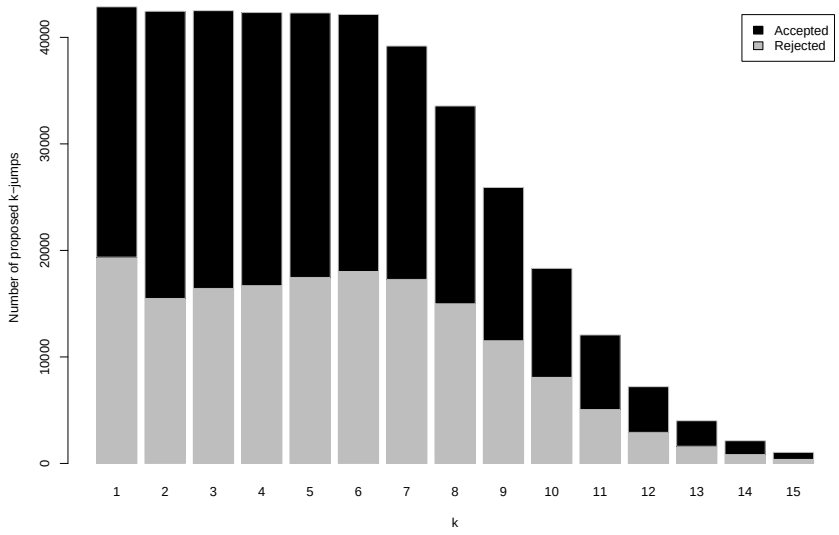


Figure 3.4: Plot of the counts of proposed k value for all k-jump update steps and the number accepted, for the case $\theta_1 = (1, 119, 29)$ with constant infectious period of 4.1 days and $K_{\max} = 15$.

For the one-type one-level model, using the branching process approximation in Figure 2.12, the range of path lengths seems to vary between $\sqrt{d}$ and $3\sqrt{d}$. Hence, setting $K_{\max} = 3\sqrt{d}$ is a logical choice for the hyperparameter. For the models in the following sections, we must revert to the method used to tune the variance hyperparameter for the λ-update proposal distribution. Namely, a small trial run is performed to better gauge the mixing of the chain.

Setting $K_{\max} = 15$ and running our MCMC algorithm we obtain Figure 3.4, the counts of proposals of length k separated into those that are accepted and rejected. The chain was run for 2×10^5 iterations, within each iteration there were two updates to the path z and one update to the infection rate λ, of which the initial 10^3 iterations where classed as the burn in period.

Comparing the marginal posterior density of τ in Figure 3.3 to the proposal counts for k-jumps in Figure 3.4, we clearly see that jumps of length 7 or greater are proposed less

often. This results from the derivation of the range of k, depending on the length of the current path. Thus, since the posterior mean length is 10.3, we expect the number of 15-jumps to be far fewer than 5-jumps for example. It would seem tempting to reduce the upper limit, however the longer length jumps enable the chain to rapidly move around the space of paths. Also, from the bar plot we see than the proposed candidate is accepted approximately 50% of the time, independent of the length of jump. In fact, longer jumps are accepted slightly more often as the likelihood drives the chain towards the highest posterior density of the shorter paths.

The non-centred methods discussed in Section 1.3.3 are of interest if the relationship between the imputed data and the parameters of interest causes high rejection rates, adversely affect the mixing of the Markov chain. In Figure 3.5 we plot the joint posterior density of the path length τ and the infection rate λ using a kernel density estimate from the samples drawn. The λ and z updates make perpendicular moves on this plot, which may conflict with the shape of the posterior density. In fact, the posterior is unimodal and regular in shape giving no cause to consider more involved techniques.

3.2.4 Extending The Generation Representation

The model described by Demiris and O'Neill (2005a) has two aspects not yet considered for the generation representation, namely non-constant infectious period distributions and varying the individual chosen as the initial infective. We shall also consider removing the assumption of a single initial infective.

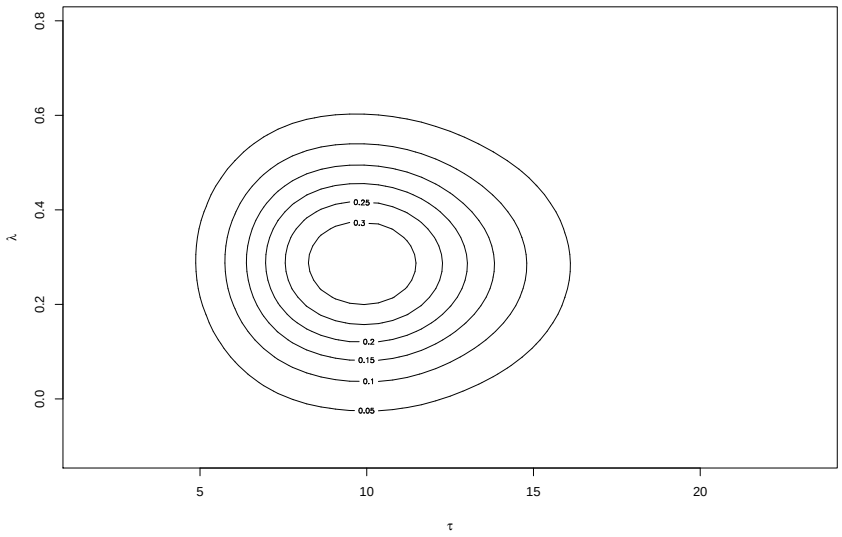


Figure 3.5: Joint posterior density of the infection rate λ and the path length τ for the case $\theta_1 = (1, 119, 29)$ with constant infectious period of 4.1 days and $K_{\max} = 15$.

3.2.4.1 Updating Label Of Initial Infective

The population is fixed to be N individuals, of which a are initially infective and n are initially susceptible, such that $a+n = N$. All individuals are labelled, i, for $1 \leq i \leq N$, and the set κ consists of the labels of individuals who are initially infective. Assuming $a = 1$, we have let $\kappa = \{1\}$ in the generation algorithm, a fixed parameter of the model. Demiris and O'Neill (2005a) consider allowing the index of the initial susceptible to be an unknown parameter. Thus the posterior is now the joint posterior of the imputed path, z, the infection rate, λ, and the label of the initial infective κ. Therefore the joint posterior is

$$\begin{aligned}\pi(z, \lambda, \kappa|\theta, I) &\propto \pi(\theta|z, \lambda, \kappa, I)\pi(z, \lambda, \kappa|I) \\ &\propto \pi(\theta|z, \lambda, \kappa, I)\pi(z|\lambda, \kappa, I)\pi(\lambda, \kappa|I) \\ &\propto \mathbb{I}_{\{\theta|z,\kappa\}}\ \pi(z|\lambda, I, \kappa)\ \pi(\lambda)\ \pi(\kappa). \qquad (3.11)\end{aligned}$$

Here $\pi(\kappa)$ is the prior density on the label of the initial susceptible (for the current discussion we assume $a = 1$) and the infectious period and set of initial infectives are assumed independent a priori. It seems reasonable to assume a proper uniform prior for κ, i.e. $\pi(\kappa) = 1/N$.

For the edge presentation, each individual is labelled and their edges recorded in order to determine if the digraph is consistent with θ. However, for the generation presentation we do not note which individuals are in each generation. Thus the labelling is arbitrary, and during a K-jump there is no accounting for which individual is moved where. This reduction in information is the motivation for the generation method, reducing to the minimal information required to augment the likelihood. In a homogeneous population, it is reasonable to remove the accounting of which individual is in a given generation since all are equally likely.

Hence, updating κ for the edge method will result in a new root set, which can then be checked for consistency with θ and the likelihood calculated for the candidate. For the generation method, updating κ properly is impossible. It would require knowing the generation of each individual and who infected who, this is necessary to construct the candidate path.

For example, consider the case $\theta = (1, n, 4)$ with the current path as $z = (1|2, 2, 0)$, where $x_1 = 2$ consisting of individuals $\{2, 3\}$, $x_2 = 2$ consisting of individuals $\{4, 5\}$ and $\kappa = \{1\}$ (implying x_0 is the individual labelled 1). If we propose $\kappa' = \{2\}$, it is impossible to construct the resulting candidate path z' from the information given. Since we do not know who from generation one infected who in generation two, the possibilities for the candidate first generation are: $\{4, 5\}$, $\{4\}$, $\{5\}$ or $\emptyset$; without knowing the edges we cannot determine which is correct (note the final option results in a path that is no longer valid). More obviously, to which generation does individual 1 belong in

the candidate path? Without detailed edge information, we cannot begin to determine into which generation the individual should be placed, or even if they are connected to another generation (using terminology from Section 2.3, if there were no backward edges to the current root vertex, then it could not be directionally connected to the candidate root vertex).

In fact, the updating of κ in the generation representation is equivalent to proposing a new path by a mechanism we cannot describe, i.e. we cannot construct the candidate path, z'. Viewing a κ-update as simply proposing a new path means we can say that the update is implicitly performed as a finite sequence of 1-jumps (since the number of initial infectives remains constant, any path can be transformed into another by repeatedly moving a single individual).

So, a κ-update is another type of Z-update. An algorithm that performed both κ-updates and K-updates would potentially mix better and may converge quicker (allowing for occasional large jumps in the space of paths). Not performing any κ-updates does not then affect the marginal posterior density of λ or z. Since κ is an arbitrary label, there is no benefit in knowing the marginal posterior density of κ, thus we can safely ignore the parameter in homogeneous populations.

3.2.4.2 Updating Number Of Initial Infectives

Given an outbreak within a closed population, it is reasonable to assume the epidemic was initiated by a single outside infection. This is the reasoning on assuming there is always a single initial infective, i.e. $a = 1$.

We have introduced the notation $\theta = (a, n, d)$ to denote the final size data. However, in reality we are given the total population size, N, and the total number of individuals

who were infective at any time during the epidemic, including the initial infectives, i.e. $a + d$. We have used the notation $D = a + d$, thus the actual data are usually reported as the vector $\psi = (N, D)$, where $a \in \mathbb{Z}_+$ is unknown. Define the function $\psi(a)$ as the map $\psi \to \theta$ by $\psi(a) = (a, N - a, D - a)$.

Considering a as an unknown parameter gives the joint posterior conditioned on ψ as,

$$\begin{aligned}
\pi(z, \lambda, \kappa, a|\psi, I) &\propto \pi(\psi|z, \lambda, \kappa, a, I)\pi(z, \lambda, \kappa, a|I) \\
&\propto \pi(\psi|z, \lambda, \kappa, a, I)\pi(z|\lambda, \kappa, a, I)\pi(\lambda, \kappa, a|I) \\
&\propto \mathbb{I}_{\{\theta|z,\kappa\}}\ \pi(z|\lambda, I, \kappa)\ \pi(\lambda|\kappa, a, I)\ \pi(\kappa, a|I) \\
&\propto \mathbb{I}_{\{\theta|z,\kappa\}}\ \pi(z|\lambda, I, \kappa)\ \pi(\lambda)\ \pi(\kappa|a)\ \pi(a),
\end{aligned}$$

where $\pi(\kappa, a)$ is the prior density on the number of initial susceptibles and the set of individuals, obviously κ depends on a, thus we let $\pi(a)$ be the prior density on the number of initial infectives and $\pi(\kappa|a)$ be the density of the set κ given its size a.

There are several ways to perform a-updates, we shall consider two. The first is to adapt the K-jump as defined in Section 3.2.2.3, allowing jumps from and to the zeroth generation. The algorithm already checks to ensure the origin generation is not reduced to zero individuals, thus the only adjustment is to the range of valid k. The maximum jump length including the zeroth generation is

$$\mathcal{K}_a = \min\{(\tau + \mathbb{I}_{\{x_0>1\}}), K_{\max}\},$$

as the last generation can always be τ-jumped back to the zeroth and if $x_0 > 1$, then an individual can be $(\tau + 1)$-jumped from the the 0^{th} to the $(\tau + 1)^{\text{th}}$ generation.

Alternatively, we can update by adding or removing an individual from the zeroth generation, this requires removing or adding an individual at random from the path

respectively. We shall give details of this method in Section 3.3.6 and 3.3.8, in particular we shall present d-updates and n-updates that can be adapted to perform a-updates.

Care must be taken with the a-update, as the prior density on a will have a great effect on the joint posterior density of z, λ and a. The issue is the strong dependence between the parameters, i.e. the number of initial infectives will determine the form of z which in turn will determine λ. The difficulty in estimating λ and a at the same time is due to non-identifiability within the model, as outlined above.

For example, the degenerate case when $a = D$, i.e. $z = (D|0)$, implies that $\lambda = 0$ is a valid value for the parameter. In fact, if $a = D$ then $\pi(\lambda|\theta) = \pi(\lambda)$, since the data contain no information to update λ. A uniform prior on a would be an unreasonable choice, as it would give a significant weight to the degenerate case in the joint posterior density. A better choice would be to restrict the range of a, or a prior with the majority of probability near $a = 1$. For this reason the MCMC mixing benefits from an informative prior on the a parameter.

A small study of a-update steps, using a full non-informative prior and restricted uniform prior were performed. Ultimately, the marginal posterior density of a had sufficient density at $a = 1$ to question the need for the added complexity. For the remainder we shall assume $a = 1$ to simplify the inference.

3.2.4.3 Alternative Infectious Period Distributions

Finally, Demiris and O'Neill (2005a) compare the fixed infectious period to an exponential and gamma infectious period. We consider two approaches to extend the generation method to enable varying infectious period distributions.

Thus far, for a point mass distribution, the infectious period of each individual has been equal, i.e. $I^i = c$ for each individual i, $1 \leq i \leq N$, where I^i is the random variable denoting the infectious period of the i^{th} individual. If we consider alternate infectious periods, for example a gamma distribution with shape hyperparameter α_I and rate hyperparameter β_I, i.e. $I \sim \Gamma(\alpha_I, \beta_I)$, then each individual has a random infectious period denoted by I^i, which are independent and identically distributed (i.i.d.) copies of I. We write $I^i \overset{d}{=} I$ to indicate that I^i is equal in distribution to I. Finally, each I^i will take a specific value denoted ζ^i.

Integrate Out The Infectious Period Our first approach is to integrate out the infectious period from the likelihood. Recall from Section 2.4.4,

$$\mathrm{L}(z|\lambda, I, \kappa) = \mathrm{P}_\theta[Z = z] = \prod_{t=0}^{\tau} \mathrm{P}_\theta[Z_{t+1} = z_{t+1}|Z_t = z_t].$$

In order to integrate out the infectious period we can take the expectation with respect to I,

$$\mathrm{P}_\theta[Z_{t+1} = z_{t+1}|Z_t = z_t] = \mathrm{E}_I\left[\mathrm{P}_\theta[z_{t+1} = (x, y)|z_t = (u, v)]\middle| I_t = (\zeta^{(1)}, \ldots, \zeta^{(u)})\right],$$

where $I^{(j)} = \zeta^{(j)}$ is the realisation of the j^{th} individual's infectious period and $I_t = (\zeta^{(1)}, \ldots, \zeta^{(u)})$ is the vector of infectious periods of individuals in the t^{th} generation. Note, for the vector I_t the individuals are indexed by j for $1 \leq j \leq x_t$, i.e. by the number of individuals in generation t. Each individual has a unique label i, for $1 \leq i \leq N$, however it is not necessary to know the exact labels of individuals within a generation, only the number of them when integrating out the infectious periods.

We shall restate Equation (2.23) from Section 2.4.4 in order to highlight taking expectations for the simple one-type one-level model. Assuming all infectious periods are

independent and identically distributed, i.e. $I^i \stackrel{d}{=} I$ for all individuals i in the population. More specifically, $I^{(j)} \stackrel{d}{=} I$, for all individuals j in the generation. Then, the probability of a single step (all steps being independent), is as follows

$$\begin{aligned}
&\mathrm{P}_\theta^I[z_{t+1} = (x,y)|z_t = (u,v)] \\
&= \mathrm{E}_I\left[\mathrm{P}_\theta[z_{t+1} = (x,y)|z_t = (u,v)]\middle| I_t = (\zeta^{(1)},\ldots,\zeta^{(u)})\right] \\
&= \mathrm{E}_I\Bigg[\binom{a+n-v}{x}\left(1-\exp\left(-\frac{\lambda}{N}\sum_{j=1}^{u}\zeta^{(j)}\right)\right)^x \\
&\qquad\qquad \left(\exp\left(-\frac{\lambda}{N}\sum_{j=1}^{u}\zeta^{(j)}\right)\right)^{a+n-(v+x)}\Bigg]. \qquad (3.12)
\end{aligned}$$

We can rewrite the product in order to extract the expectation with respect to I as,

$$\begin{aligned}
&= \binom{N-v}{x}\mathrm{E}_I\left[\sum_{k=0}^{x}(-1)^{x-k}\binom{x}{k}\exp\left(-\frac{\lambda}{N}\sum_{j=1}^{u}\zeta^{(j)}\right)^{(N-v-k)}\right] \\
&= \binom{N-v}{x}\sum_{k=0}^{x}(-1)^{x-k}\binom{x}{k}\mathrm{E}_I\left[\exp\left(-\frac{\lambda}{N}(N-v-k)\sum_{j=1}^{u}\zeta^{(j)}\right)\right] \\
&= \binom{N-v}{x}\sum_{k=0}^{x}(-1)^{x-k}\binom{x}{k}\mathrm{E}_I\left[\prod_{j=1}^{u}\exp\left(-\frac{\lambda}{N}(N-v-k)\,\zeta^{(j)}\right)\right],
\end{aligned}$$

since the infectious periods are i.i.d. we have,

$$\begin{aligned}
&= \binom{N-v}{x}\sum_{k=0}^{x}(-1)^{x-k}\binom{x}{k}\prod_{j=1}^{u}\mathrm{E}_I\left[\exp\left(-\frac{\lambda}{N}(N-v-k)\,\zeta^{(j)}\right)\right] \\
&= \binom{N-v}{x}\sum_{k=0}^{x}(-1)^{x-k}\binom{x}{k}\mathrm{E}_I\left[\exp\left(-\frac{\lambda}{N}(N-v-k)\,\zeta\right)\right]^u. \qquad (3.13)
\end{aligned}$$

For the final step, there is no superscript on ζ, since all individual's infectious periods are i.i.d.. By taking the expectation, the infectious period is integrated out from the

expression.

Hence, Equation (3.13) is the likelihood of a single step and the likelihood of a path z is the product of steps along the generations. Having integrated the infectious period out of the likelihood, it does not appear as a parameter in joint posterior likelihood nor is there a prior. We condition on knowing the infectious period distribution for each individual and are assuming they are independent and identically distributed.

Unfortunately, the alternating sum in Equation (3.13) is numerically unstable when the generation sizes are large. Also, if the expectation

$$\mathrm{E}_I\left[\exp\left(-\frac{\lambda}{N}\left(N-v-k\right)\zeta\right)\right]$$

is costly to compute then the calculation of the likelihood, which is needed for z-updates and λ-updates, will be slow and cause the MCMC algorithm to take longer to generate a sufficient number of samples.

Let I Be An Additional Parameter The alternative method, used by Demiris and O'Neill (2005a), is to consider the vector of infectious periods for all individuals as a parameter, i.e. $I = (\zeta^1, \zeta^2, \ldots, \zeta^N)$. It is necessary to know the labels of individuals that comprise each generation, as was the case for κ-updates. However, it is possible to use this approach with the generation representation, whereas κ-updates are impossible.

For each generations t, there are x_t individuals in that generation and we require the set $\mathcal{X}_t$ which contains the labels of the x_t individuals, i.e. $\mathcal{X}_t \subset \{1, \ldots, N\}$. Only D of the N individuals are ever included in the digraph at anytime, and since the population is homogeneous it suffices to reduce I to a D-length vector, the permutation of individuals is accounted for in the binomial coefficient in Equation (3.12). Hence we may reduce

to $\mathcal{X}_t \subset \{1, \ldots, D\}$. To maintain a valid path the following must be true, $\mathcal{X}_s \cap \mathcal{X}_t = \emptyset$ for $0 \leq s \neq t \leq \tau$ and $\cup_{t=0}^{\tau} \mathcal{X}_t = \{1, \ldots, a, a+1, \ldots, a+d\} = \{1, \ldots, D\}$.

Then the posterior density becomes (ignoring κ and reverting to the case of a fixed initial number of infectives for clarity),

$$\begin{aligned} \pi(z, \lambda, \zeta|\theta) &\propto \pi(\theta|z, \lambda, \zeta)\pi(z, \lambda|\zeta) \\ &\propto \pi(\theta|z, \lambda, \zeta)\pi(z|\lambda, \zeta)\pi(\lambda, \zeta) \\ &\propto \pi(\theta|z, \lambda, \zeta)\pi(z|\lambda, \zeta)\pi(\lambda)\pi(\zeta), \end{aligned} \tag{3.14}$$

where ζ is the D-length vector of infectious periods for all individuals and $\pi(\zeta)$ is the prior density of that vector, assuming a priori that the infectious period and infection rate are independent.

Thus, for the likelihood component of Equation (3.14) we can use the probability of a step in the path as Expression (3.12). Since ζ is a vector of constants under the the likelihood, the expectation in Equation (3.12) is ignored and we only require the sum of infectious periods of individuals in generation t, hence why we must now specify the sets of labels, $\mathcal{X}_t$.

It is necessary to then include an I-update step in the MCMC algorithm. The prior will be a vector of D independent copies of the infectious period under consideration. Since we are making no inference for I, as it follows the preset distribution according to the prior, we will only consider the marginal posterior density $\pi(z, \lambda|\cdot)$. Let the proposal distribution be equal to the infectious period distribution, i.e. $q(I'^i|I^i) = q(I'^i) \sim I$, for each infectious period independently. Let $I_{(-t)}$ be the vector of infectious periods I, less the individuals in generation t, i.e. $I_t = I \backslash I_{(-t)}$. Then the acceptance probability

for proposing a new vector of infectious periods for generation t is,

$$\begin{aligned}\alpha(I_t, I_t') &= \min\left\{1\,,\, \frac{\pi(z,\lambda,I_{(-t)},I_t'|\theta)q(I_t|I_t')}{\pi(z,\lambda,I_{(-t)},I_t|\theta)q(I_t'|I_t)}\right\}\\ &= \min\left\{1\,,\, \frac{\pi(\theta|z,\lambda,I_{(-t)},I_t')\pi(z|\lambda,I_{(-t)},I_t')\pi(\lambda)\pi(I_t')q(I_t)}{\pi(\theta|z,\lambda,I_{(-t)},I_t)\pi(z|\lambda,I_{(-t)},I_t)\pi(\lambda)\pi(I_t)q(I_t')}\right\}\\ &= \min\left\{1\,,\, \frac{\pi(z|\lambda,I_{(-t)},I_t')}{\pi(z|\lambda,I_{(-t)},I_t)}\right\}.\end{aligned}$$

Where $q(I_t)$ is set to be equal to $\pi(I_t) = \prod_{j=1}^{x_t} \mathrm{P}[I^{(j)} = \zeta^{(j)}]$. Since the likelihood of the path is a product of independent steps between generations, there will be further cancellations possible in calculating α.

For the edge representation, it is a trivial matter to assign a variable for each individual's infectious period. There is no need to amend any of the update steps. However, for the generation method we must amend the K-jump (and any other updates that affect the path z). In this case, when an origin generation is selected, a specific individual within that generation must be chosen and moved and the corresponding origin and target label sets, $\mathcal{X}_O$ and $\mathcal{X}_{O+\delta k}$, updated.

For large final sizes, updating all the infectious periods at once would lead to low acceptance probabilities. Thus it is more efficient to update them in blocks. The generation representation provides a natural partition, and this is how the acceptance probability is stated above, i.e. perform an I-update on each $\mathcal{X}_t$ in turn. However, any blocking structure could be used, for example by type, level or class (see Section 3.5.1.2 for these defintions).

The I-update and modified K-jump are presented in Algorithms 3.3 and 3.4 respectively. The λ-update is unaffected with respect to its method, only requiring the use

Algorithm 3.3: I-update of the infectious period vector $I = (\zeta^1, \ldots, \zeta^D)$ for one-type one-level model

Sample $t \sim \text{Uni}[0, \tau]$;
Sample $I'_i \sim I$ for each $i \in \mathcal{X}_t$;
Calculate acceptance probability α;
Draw $A \sim \text{U}(0, 1)$;
if $\alpha < A$ **then**
    reject I'_t
else
    accept I'_t

of the modified likelihood including the vector of infectious periods. For the K-jump, the proposal distribution also needs to be amended, since we now select a specific individual to be moved from among those in the origin generation. Then the proposal is in three parts,

$$q(z'|z) = \frac{1}{\mathcal{K}} \frac{1}{\mathcal{J}_k} \frac{1}{x_{t_O}}.$$

Where $\mathcal{J}_k$ depends on the k value selected and x_{t_O} is the size of the origin generation (from the point of view of the current path). Conversely,

$$q(z|z') = \frac{1}{\mathcal{K}'} \frac{1}{\mathcal{J}'_k} \frac{1}{x'_{t_O}},$$

where by construction, $x'_{t_O} = x_{t_O+\delta k} + 1$.

Choosing To Integrate Or Incorporate I The choice between integrating out the infectious periods or incorporating them as additional parameters is a balance between computational cost and adequate mixing in the parameter space.

Equation (3.13) is numerically unstable, it requires a high degree of precision to accurately calculate the likelihood. We shall discuss how we overcame this problem using

Algorithm 3.4: Z-update using K-jump with infectious period vector $I = (\zeta^1, \ldots, \zeta^D)$ for one-type one-level model

Let $\mathcal{K} = \min\{(\tau - 1 + \mathbb{I}_{\{x_1>1\}}), K_{\max}\}$;
Sample $k \sim \text{Uni}[1, \mathcal{K}]$;
Calculate the vector $J_k(z) = (J_k(z_0)|J_k(z_1), \ldots, J_k(z_\tau))$;
Calculate $\mathcal{J}_k = \sum_{i=1}^{\tau} |J_k(z_i)|$;
Sample $g \sim \text{Uni}[1, \mathcal{J}_k]$;
Determine the origin t_O, and direction δ corresponding to g;
Select an individual i at random from $\mathcal{X}_{t_O}$;
Construct the candidate path z';
Update the label sets;
Calculate acceptance probability α;
Draw $A \sim \text{U}(0, 1)$;
if $\alpha < A$ **then**
    reject z'
else
    accept z'

GNU MPFR in Section 3.7.2, though essentially we increase the number of bytes used to represent each number in memory. This dramatically increases the time to compute the likelihood, and thus slows down the MCMC algorithm.

Conversely, letting the infectious periods be additional parameters allows quick evaluation of the likelihood at the cost of having to explicitly explore the space of infectious periods. Care must be taken to ensure the marginal posterior density, $\pi(z, \lambda|\cdot)$, adequately covers the space of I for all D individuals.

Results And Comparison Returning to the data sets analysed by Demiris and O'Neill (2005a) for the one-type one-level model, e.g. $\theta_3 = (1, 99, 24)$, ignoring κ for the reasons discussed in Section 3.2.4.1 and considering the case of a fixed single initial infective, $a = 1$, we compare the infectious distributions considered by Demiris and O'Neill (2005a) on the same sample data. Namely, a fixed period, exponential and gamma all with mean one, $\iota = \text{E}[I] = 1$, and with variances 0, 1 and 10 respectively,

θ	I	Edge Method Mean	(sd)	Generation Method Mean	(sd)
$(1, 99, 24)$	$I = c$	1.16	(0.23)	1.17	(0.24)
	$I \sim \text{Exp}(1)$	1.27	(0.37)	1.28	(0.40)
	$I \sim \Gamma(0.1, 0.1)$	1.37	(0.50)	1.36	(0.53)
$(1, 99, 49)$	$I = c$	1.42	(0.21)	1.42	(0.21)
	$I \sim \text{Exp}(1)$	1.49	(0.27)	1.49	(0.28)
	$I \sim \Gamma(0.1, 0.1)$	1.55	(0.36)	1.51	(0.33)
$(1, 99, 74)$	$I = c$	1.86	(0.21)	1.88	(0.23)
	$I \sim \text{Exp}(1)$	1.96	(0.27)	1.99	(0.28)
	$I \sim \Gamma(0.1, 0.1)$	2.04	(0.37)	1.99	(0.35)

Table 3.5: Comparison of estimates for the infection rate λ, reported as $R_0 = \iota\lambda$, between the edge method of Demiris and O'Neill (2005a) and the generation method of Section 3.2.2 incorporating the infectious periods I for three distributions. On $\theta_3 = (1, 99, 24)$, $\theta_4 = (1, 99, 49)$ and $\theta_5 = (1, 99, 74)$.

i.e. $I = 1$, $I \sim \text{Exp}(1)$ and $I \sim \Gamma(0.1, 0.1)$.

In Table 3.5 we reproduce the values from Demiris and O'Neill (2005a) and present our results for the generation representation using the method of including the infectious periods as parameters, the estimates are in general agreement with respect to the standard deviations.

3.3 Partially Observed Epidemics

3.3.1 Definition And Notation

As discussed in Section 1.2.6, we differentiated between missing data and a partially observed epidemic. The former accounts for incomplete information about individuals (perhaps of varying degree), e.g. unknown time of infection or unknown time of removal.

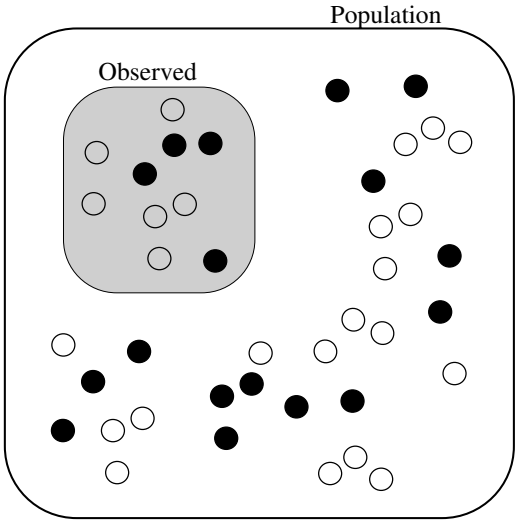


Figure 3.6: Diagram of an example partially observed setting. Black circles indicate individuals who became infected during the course of the epidemic. The shaded region denotes the observed subset of the population. In this example, the complete data can be summarised as $\psi = (N, D) = (41, 17)$ of which we only observe $\psi_{\text{ob}} = (10, 4)$.

The latter accounts for when only a subset of the population are observed, i.e. for some individuals no information is known.

For example, consider Figure 3.6, assuming a closed population, an epidemic has occurred within the population. However, we only observe a subset of the population (shaded grey on the diagram) and thus we have no information about the unobserved component. Let N denote the total population size including all individuals, with N_{ob} and N_{un} being the number of individuals observed and unobserved respectively, i.e. $N = N_{\text{ob}} + N_{\text{un}}$. Similarly, define D_{ob} and D_{un} to be the total number of infectives in the observed and unobserved components respectively (we shall consider the number of initial infectives, a, shortly).

There are two types of partially observed epidemic we shall consider, models with unknown number of infectives and unknown number of susceptibles . In both cases there is an observed component, $\psi_{\text{ob}} = (N_{\text{ob}}, D_{\text{ob}})$, which is the constant data in our

model. The two types differ in the additional information that is known, both shall be considered and update algorithms presented.

For an unknown number of infectives, the size of the unobserved component is known, i.e. N_{un}. Such data are usually described by saying that the total population size, N, is known and a proportion, η, is observed. Then, $\eta N = N_{\text{ob}}$, and only the number of unobserved infectives, D_{un}, is uncertain. Alternatively, with an unknown number of susceptibles we assume there are no unobserved infectives, i.e. $D_{\text{un}} = 0$, and only the number of unobserved susceptibles is uncertain, N_{un}.

Unknown N_{un} is common when there may not be a closed population, or there is a potential for part of the population to be immune, i.e. they are not susceptible to the disease and should not be included. The case of unknown D_{un} occurs when the population is large and only a small part can be (randomly) sampled.

For this section we shall again consider the one-type one-level model, though even with such a model it is important to consider how the subset was selected. For multi-type multi-level models, a bias in the subset observed would affect the inference. For the unknown D_{un}, we assume individuals are observed at random in the one-level model, implying no bias in the subset and the proportion of infected individuals in the observed component is similar to the population as a whole. For the two-level model, including households, then we would assume households are observed at random. Observing individuals at random in a two-level model is a very different assumption. In Bayesian analysis, it is possible to alter these assumptions by placing informative priors on the parameter of interest, and we shall consider this for the case of unknown N_{un}.

3.3.2 Previous Literature

Demiris (2004) (see also Demiris and O'Neill (2005b)) consider the case of partially observed populations with unknown D_{un} using an augmented psuedolikelihood based on the total severity of the epidemic to estimate the posterior using MCMC . Their approach assumes the epidemic is above threshold, as do many classical approximations and limiting results. The generation method gives an exact likelihood to generate the posterior density and makes no assumption on the epidemic being above threshold.

Hayakawa et al. (2003) consider the case of unknown N_{un}, i.e. the number of susceptible individuals is uncertain. They again use MCMC, imputing the times of infection with known removal times (a missing data problem) using the approach outlined in O'Neill and Roberts (1999).

Both Hayakawa et al. (2003) and Demiris and O'Neill (2005b) investigate the case $\psi_{\text{ob}} = (120, 30)$, a standard example in the epidemics literature. The full data set consists of temporal information, thus it has been considered by many authors, using many inference techniques, for example Bailey (1975), Becker (1989), O'Neill and Roberts (1999), Hayakawa et al. (2003), Demiris (2004) and O'Neill (2009).

3.3.3 Outline For Partially Observed Model

First, we wish to make inference for the case of unknown D_{un}, i.e. given the total population size N and the observed subset $\psi_{\text{ob}} = (N_{\text{ob}}, D_{\text{ob}})$, where $N_{\text{un}} = N - N_{\text{ob}}$ and D_{un} is unknown. We shall denote by η the ratio of the number of observed individuals to the total population, i.e. $\eta = \frac{N_{\text{ob}}}{N}$. Thus $\eta = 1$ reduces to the case considered in Section 3.2 and as $\eta \to 0$ the subset observed is a negligible fraction of the total population.

In Section 3.3.5 we shall discuss adapting the edge method to account for a partially observed population. The algorithm requires only a few amendments, whereas in Section 3.3.6 we adapt the generation representation and introduce a new update that is required for the MCMC algorithm. The difference between the two approaches, edge or generation, is in the amount of information needed to obtain the posterior density. The edge representation is required to store a much larger imputed data set. As the observed proportion, η, grows small this becomes a prohibitive issue, in terms of algorithm run-time.

The methods for unknown N_{un} using the generation representation require only minor modification to the methods in Section 3.2.2. The outline for the edge method is discussed in Section 3.3.7 and details for the generation method in Section 3.3.8.

We do not implement the edge methods, instead citing results from papers using equivalent or comparable methods in order to verify the generation method. In Section 3.3.9 we apply the generation approach to several test data sets. First, with a fixed observed component and varying η, i.e. an increasing total population size N. Second, we consider a fixed total population and observe a varying sized subset. In each case we use a fixed infectious period of 4.1 days and report the reproductive number R_0.

3.3.4 Posterior Density

As discussed in Section 3.2.4.2, we shall assume there is a single initial infective. However, we must consider if the initial infective is in the observed component or not, since

$a = a_{\text{ob}} + a_{\text{un}} = 1$. For simplicity, assume the initial infective is observed, which implies

$$\theta_{\text{ob}} = \psi_{\text{ob}}(1) = (1, N_{\text{ob}} - 1, D_{\text{ob}} - 1)$$
$$\theta_{\text{un}} = \psi_{\text{un}}(0) = (0, N_{\text{un}}, D_{\text{un}}).$$

For the case of an unknown number of infectives, we wish to make inference on the joint posterior density of λ and z (note that D_{un} is encoded in z) given the data $\psi_{\text{ob}} = (N_{\text{ob}}, D_{\text{ob}})$ and N. Then the posterior density is

$$\begin{aligned}\pi(z, \lambda|\theta_{\text{ob}}, N, I) &\propto \pi(\theta_{\text{ob}}|z, \lambda, N, I)\pi(z, \lambda|N, I) \\ &\propto \pi(\theta_{\text{ob}}|z, \lambda, N, I)\pi(z|\lambda, N, I)\pi(\lambda) \\ &\propto \mathbb{I}_{\{\theta_{\text{ob}}|z\}}\pi(z|\lambda, N, I)\pi(\lambda). \qquad (3.15)\end{aligned}$$

For the case of an unknown number of susceptibles, we wish to make inference on the joint posterior density of z, λ and N_{un} given the data $\psi_{\text{ob}} = (N_{\text{ob}}, D_{\text{ob}})$ and D_{un}. Note that N_{un} must be explicitly included, since it is not encoded in z. Then the posterior density is

$$\begin{aligned}\pi(z, \lambda, N_{\text{un}}|\theta_{\text{ob}}, D_{\text{un}}, I) &\propto \pi(\theta_{\text{ob}}, D_{\text{un}}|z, \lambda, N_{\text{un}}, I)\pi(z, \lambda, N_{\text{un}}|I) \\ &\propto \mathbb{I}_{\{\theta_{\text{ob}}, D_{\text{un}}|z\}}\pi(z|\lambda, N_{\text{un}}, I)\pi(\lambda)\pi(N_{\text{un}}), \qquad (3.16)\end{aligned}$$

where $\pi(N_{\text{un}})$ is the prior on the number of unobserved susceptibles. Assuming the infection rate and number of unobserved susceptibles are independent a priori. The indicator term ensures the imputed path z is consistent with the observed data. Commonly, $D_{\text{un}} = 0$ and $N_{\text{ob}} = D_{\text{ob}}$, i.e. there are no unobserved infections and there are sufficient individuals to support the observed data. It is not clear what prior to

use for the unknown number of susceptibles, N_{un}, where the parameter is unbounded, i.e. $N_{\text{un}} \geq 0$.

3.3.5 Edge Representation For Unknown Number Of Infectives

The edge method described in Section 3.2.1 is unchanged for the partially observed data. For each individual in the population we must record its edges, we cannot restrict our attention to a smaller sub-digraph in this case. Using the edges of all individuals, it is possible to perform a recursive connectivity check as before to ensure the imputed digraph is consistent with the observed data, i.e. $\mathbb{I}_{\{\theta_{\text{ob}}|z\}}$. The check will also calculate the number of individuals connected in the unobserved component, D_{un}.

Hence, when adding or removing an edge from the digraph we may increase or decrease D_{un}. The effect of altering the number of unobserved infectives on the acceptance probability is two fold, firstly in the likelihood of the digraph and secondly in the prior density on D_{un}.

The digraph is defined on N vertices and the MCMC updates are as in Section 3.2.1, namely a symmetric Random Walk Metropolis for the λ-updates and adding or removing edges from G for the G-updates. The connectivity check is required to determine if the digraph G is consistent with ψ_{ob} and to determine D_{un}.

It is necessary to store all the edges for all the individuals in the population, that is the size of the stored digraph is of order N^2. Whereas, for the generation method the path is a vector with size of order N. Thus, if the number of individuals is increased the edge methods suffer in terms of computation costs. For example, returning to Figure 3.6, consider the number of edges that need to be imputed. Care must also be taken to ensure the chain is properly mixing in the space of all imputed unobserved components.

3.3.6 Generation Representation For Unknown Number Of Infectives

We can extend the one-type one-level representation of Section 3.2.2 to account for having observed and unobserved individuals. The population is still homogeneous and homogeneously mixing, with a single infection rate parameter λ we wish to make inference given $\psi_{\text{ob}} = (N_{\text{ob}}, D_{\text{ob}})$ and N (implying $N_{\text{un}} = N - N_{\text{ob}}$).

For simplicity, assume there is a single observed initial infective and there are no unobserved initial infectives, we shall return to this assumption shortly. Also, assume the infectious period is a fixed time c. Hence, we do not need to consider integrating out or including a vector of the infectious periods. Importantly, we do not need to account for which individuals constitute each generation.

3.3.6.1 Extending Generation Notation

The epidemic process can still be represented as a path, consisting of a sequence of generations, where each generation may have observed and unobserved individuals. Therefore, we may naturally extend the notation and definitions of Section 3.2.2.1.

Let $x_{\text{ob},t}$ and $x_{\text{un},t}$ be the number of observed and unobserved individuals in generation t respectively. Then, the total number in generation t is $x_t = x_{\text{ob},t} + x_{\text{un},t}$. Similarly, define the cumulative totals $y_{\text{ob},t}$, and the combined cumulative total $y_t = y_{\text{ob},t} + y_{\text{un},t}$.

The definition of the length of a path z remains unchanged, i.e. $\tau = \max\{t : x_t > 0\}$. In addition, we can deduce the generation where the first observed and unobserved individual occurs. Let $r_{\text{ob}} = \min\{t : x_t > 0\}$ and $s_{\text{ob}} = \max\{t : x_t > 0\}$, similarly for r_{un} and s_{un}. By assumption, $r_{\text{ob}} = 0$, since $x_{\text{ob},0} = a_{\text{ob}} = 1$ and by definition $\tau = \max\{s_{\text{ob}}, s_{\text{un}}\}$.

We may represent the path, z, as the matrix consisting of the sizes of each generation split into observed and unobserved counts. Recall, we need only specify x_t for clarity, since the cumulative totals y_t are a function of the generation totals.

Also for clarity, we use the column vector notation introduced in Section 3.2.2.3. For example, consider the case where $d_{\text{ob}} = 4$ and $d_{\text{un}} = 2$, then an example path can be expressed as,

$$z = \left[\begin{array}{cc} 1 & 0 \\ \hline 2 & 0 \\ 1 & 1 \\ 1 & 0 \\ 0 & 1 \end{array}\right]_4 .$$

Where the left column is the observed component and the right is the unobserved.

Note, each column need not be a valid epidemic path when taken on its own. In the example presented, the unobserved path (using the row vector notation) is $z_{\text{un}} = (0|0, 1, 0, 1, 0)$, which is clearly not a valid path since there are zero generations before the desired total number of infectives is reached.

The density of interest is the joint posterior of λ, z and D_{un} as shown in Equation (3.15). Thus we must consider how to perform updates for all three parameters for the MCMC algorithm.

3.3.6.2 λ-update

The λ-update can be performed exactly as described by Algorithm 3.1 in Section 3.2.2.2. The proposal remains a symmetric Random Walk Metropolis and the acceptance prob-

ability is unchanged, using the generation totals. For clarity, given the example path above we can explicitly state the generation totals as,

$$z = \left[\begin{array}{cc|c} \text{ob} & \text{un} & \text{Total} \\ 1 & 0 & 1 \\ \hline 2 & 0 & 2 \\ 1 & 1 & 2 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \end{array}\right]_4 .$$

3.3.6.3 Z-update Using K-jump

For the path z, we are conditioning on the total number of infectives in each column (using the column notation above). Hence, an individual is restricted to a given column, but may be moved to any (valid) generation within the path. Therefore we may apply Algorithm 3.2, the K-jump, as the Z-update for fixed d_{un}.

Define $J_k(z)$ to apply the Equation (3.9) to the matrix z instead of the vector z. Calculate the range of valid k using the path length τ. The total number of k jumps is the sum over all observed and unobserved generations. Note, the check to see if a move is valid still uses $x_t > 1$, i.e. the generation total across both observed and unobserved.

For example, given the path z above, it has length $\tau = 4$. Assuming a fixed number of initial infectives, the range of valid k is

$$\begin{aligned} 1 \leq k \leq \mathcal{K} &= \min\{(\tau - 1 + \mathbb{I}_{\{x_1 > 1\}}), K_{\max}\} \\ &= \min\{(4 - 1 + 1), K_{\max}\} = \min\{4, K_{\max}\}. \end{aligned}$$

Applying the function $J_k(z)$ for each $1 \leq k \leq 4$ we have,

$$z = \left[\begin{array}{cc} 1 & 0 \\ \hline 2 & 0 \\ 1 & 1 \\ 1 & 0 \\ 0 & 1 \end{array}\right]_4 \quad J_1(z) = \left[\begin{array}{cc} 0 & 0 \\ \hline 1 & 0 \\ 2 & 2 \\ 0 & 0 \\ 0 & -1 \end{array}\right] \Rightarrow \mathcal{J}_1 = 6 \quad J_2(z) = \left[\begin{array}{cc} 0 & 0 \\ \hline 1 & 0 \\ 1 & 1 \\ 0 & 0 \\ 0 & -1 \end{array}\right] \Rightarrow \mathcal{J}_2 = 4$$

$$J_3(z) = \left[\begin{array}{cc} 0 & 0 \\ \hline 1 & 0 \\ 1 & 1 \\ 0 & 0 \\ 0 & -1 \end{array}\right] \Rightarrow \mathcal{J}_3 = 4 \quad J_4(z) = \left[\begin{array}{cc} 0 & 0 \\ \hline 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{array}\right] \Rightarrow \mathcal{J}_4 = 1.$$

Since $x_3 = 1$ and $\tau = 4$, i.e. the third generation is not the last generation and is only a single individual, then $J_k(z_t) = 0$ for $1 \leq k \leq 4$, where $z_t = (x_{\text{ob},t}, x_{\text{un},t})$. We assume a fixed number of initial infectives, i.e. $J_k(z_0) = (0, 0)$ for $1 \leq k \leq 4$.

Justifying The K-jump As an aside, we now justify the choice of the complex K-jump over the simple Z independence sampler discussed in Section 3.2.2.3.

Essentially, whereas it was simple to enumerate all possible paths for the one-type one-level model, for the partially observed case, this is not so. In particular, for $0 \leq m \leq 2^{d-1} - 1$, it was possible to construct the path $z^{(m)}$ using the binary decomposition.

For the partially observed case, it is not trivial to calculate the number of valid paths, which is necessary before we can enumerate them. To show this we first need Lemma 3.1.

Lemma 3.1

Let (n,k) denote the number of ways to place n objects into k boxes, $n,k \in \mathbb{Z}_+$. Then

$$(n,k) = \binom{n+k-1}{n} = \binom{n+k-1}{k-1}.$$

We place bounds on the number of valid paths consisting of an observed and unobserved component. It is possible to form a recursive algorithm to search all possible paths, but it must exhaustively search all such paths.

Lemma 3.2

Let M be the number of paths consisting of d_{ob} and d_{un} individuals. Then

$$2^{d-1} \leq M \leq \binom{d_{\text{ob}}+d-1}{d_{\text{ob}}}\binom{d_{\text{un}}+d-1}{d_{\text{un}}},$$

where $d = d_{\text{ob}} + d_{\text{un}}$.

Proof

The lower bound follows immediately from Lemma 2.8, there are d individuals in total and thus there are 2^{d-1} valid paths in terms of the generation totals, x_t.

However, there will be more possible paths since each generation total can be made by various combinations of observed and unobserved individuals. For example, if $x_t = 2$ it could be either $z_t = (2,0)$, $z_t = (1,1)$ or $z_t = (0,2)$ (assuming $d_{\text{ob}}, d_{\text{un}} \geq 2$).

As a crude estimate, the maximum length of a valid path is d. Thus, we assign the observed individuals to the d generations in a number of ways derived in Lemma 3.1. Similarly for the unobserved. Hence the crude upper bound of $\binom{d_{\text{ob}}+d-1}{d_{\text{ob}}}\binom{d_{\text{un}}+d-1}{d_{\text{un}}}$. □

From Lemma 3.2 it is clear we cannot easily enumerate the set of valid paths, thus it

is much harder to construct an efficient independence sampler in this case.

For the example path of this section, $d_{\text{ob}} = 4$ and $d_{\text{un}} = 2$, the possible number of valid paths lies between $32 \leq M \leq 2646$. This is a large range, the upper bound is an especially crude approximation. Consider a completely observed epidemic, i.e. $d_{\text{un}} = 0$, then the number of possible paths for $d = 6$ according to Lemma 3.2 is $32 \leq M \leq 462$; in this case equality holds with the lower bound and $M = 32$.

3.3.6.4 D_{un}-update

Finally, we must update the number of unobserved infectives. Clearly, since $0 \leq D_{\text{un}} \leq N_{\text{un}}$ there is a finite range and we can use proper priors for D_{un}. Also, by assumption $a_{\text{un}} = 0$ implying $d_{\text{un}} = D_{\text{un}}$.

The path z represents the sizes of each generation as a matrix, such that the column totals are the final sizes of the observed and unobserved components. Increasing or decreasing d_{un} is represented as adding or removing individuals from the unobserved column of z. Thus the d_{un}-update is another type of Z-update.

Additions may be made to any generation in the unobserved component up to the $(\tau + 1)^{\text{th}}$ generation.

For removals, we extend Equation (3.9) to include the case $k = 0$, representing whether an individual can be removed from generation t. Correspondingly, $\mathcal{J}_0$ is twice the total number of removals, this is done to be consistent with the definition of $\mathcal{J}_k$ for $k > 0$, see Equation (3.10). Define $x_{i,t}$ to be the number of individuals in component $i \in \{\text{ob}, \text{un}\}$

for generation t. Then,

$$J_0(x_{i,t}) = \begin{cases} 2 & \begin{cases} \text{if } x_{i,t} > 0 \text{ and } x_t > 1 \\ \text{if } x_{i,t} > 0 \text{ and } t = \tau \end{cases} \\ 0 & \text{otherwise} \end{cases} \quad \text{for} \quad \begin{array}{l} i \in \{\text{ob}, \text{un}\} \\ 1 \leq t \leq \tau. \end{array} \tag{3.17}$$

Note Equations (3.9) and (3.17) are only defined for $t > 0$, since we consider the zeroth generation as fixed.

As stated, we count twice the number of possible moves to maintain the relation, $\mathcal{J}_0 \geq \mathcal{J}_1 \geq \mathcal{J}_2 \geq \cdots \geq \mathcal{J}_{\mathcal{K}}$, assuming the entire population can be removed. For the partially observed case we apply J_0 only to z_{un}. Continuing with our example path, we can count the number of possible removals as,

$$z = \left[\begin{array}{cc} 1 & 0 \\ \hline 2 & 0 \\ 1 & 1 \\ 1 & 0 \\ 0 & 1 \end{array}\right]_4 \qquad J_0(z) = \left[\begin{array}{cc} 0 & 0 \\ \hline 2 & 0 \\ 2 & 2 \\ 0 & 0 \\ 0 & 2 \end{array}\right] \Rightarrow \mathcal{J}_0 = 8 \geq 6 = \mathcal{J}_1$$

$$J_0(z_{\text{un}}) = \left[\begin{array}{c} 0 \\ \hline 0 \\ 2 \\ 0 \\ 2 \end{array}\right] \Rightarrow \mathcal{J}_0(z_{\text{un}}) = 4.$$

Note, for the partially observed setting we do not wish to alter the observed component.

Thus we only wish to find the partial sum of

$$\mathcal{J}_0(z_{\text{un}}) = \sum_{i=1}^{\tau} J_0(x_{\text{un},i}).$$

The origin generation, t_O, corresponding to a specific choice of g in the range $1 \leq g \leq \frac{1}{2}\mathcal{J}_0(x_{\text{un},i})$, is defined as

$$t_O = \min\{t : g = \frac{1}{2}\sum_{i=1}^{t} J_0(x_{\text{un},i})\}.$$

Using an identical scanning technique as for the origin of the K-jump.

We propose adding or removing an individual with equal probability. For an addition, we propose the target generation at random from the entire length of the path plus one, so that

$$q(z'|z) = \frac{1}{2}\,\frac{1}{\tau+1}.$$

Conversely for removals, we must calculate $\mathcal{J}_0(z_{\text{un}})$, and select a generation at random to remove an individual from, so that

$$q(z'|z) = \frac{1}{2}\,\frac{1}{\frac{\mathcal{J}_0(z_{\text{un}})}{2}} \;=\; \frac{1}{\mathcal{J}_0(z_{\text{un}})}.$$

The acceptance probability of a d_{un}-update is

$$\begin{aligned}\alpha(d_{\text{un}}, d'_{\text{un}}) &= \min\left\{1\,,\ \frac{\pi(z', \lambda, |\theta_{\text{ob}}, N, I)q(z|z')}{\pi(z, \lambda, |\theta_{\text{ob}}, N, I)q(z'|z)}\right\} \\ &= \min\left\{1\,,\ \frac{\mathbb{I}_{\{\theta_{\text{ob}}|z\}}\pi(z'|\lambda, N, I)\pi(\lambda)\pi(d'_{\text{un}})q(z|z')}{\mathbb{I}_{\{\theta_{\text{ob}}|z\}}\pi(z|\lambda, N, I)\pi(\lambda)\pi(d_{\text{un}})q(z'|z)}\right\} \\ &= \min\left\{1\,,\ \frac{\pi(z'|\lambda, N, I)\pi(d'_{\text{un}})q(z|z')}{\pi(z|\lambda, N, I)\pi(d_{\text{un}})q(z'|z)}\right\}.\end{aligned}$$

Notice the proposal density is in terms of the path z, since updating d_{un} causes a new candidate path, z'. The update is summarised in Algorithm 3.5.

Algorithm 3.5: d_{un}-update within Z for partially observed one-type one-level model

Choose Add or Remove, P[Add] = P[Rem] = $\frac{1}{2}$;
if *Add* **then**
  Sample $g \sim \text{U}[1, \tau + 1]$;
  Construct z' by $x'_{\text{un},g} = x_{\text{un},g} + 1$;
  Let $q(z'|z) = \frac{1}{2}\frac{1}{\tau+1}$ and $q(z|z') = \frac{1}{2}\frac{1}{\tau'+1}$;
else
  Calculate the matrix $J_0(z)$ and $\mathcal{J}_0(x_{\text{un},i})$;
  Sample $g \sim \text{Uni}[1, \frac{\mathcal{J}_0(x_{\text{un},i})}{2}]$;
  Determine the origin generation t_O corresponding to g;
  Construct z' by $x'_{\text{un},t_O} = x_{\text{un},t_O} - 1$;
  Let $q(z'|z) = \frac{1}{\mathcal{J}_0(x_{\text{un},i})}$ and $q(z|z') = \frac{1}{\mathcal{J}'_0(x_{\text{un},i})}$;
Calculate acceptance probability α;
Draw $A \sim \text{U}(0, 1)$;
if $\alpha < A$ **then**
  reject z'
else
  accept z'

Unlike the K-jump update to Z, there are no tunable parameters for the d-update. We can adjust the frequency of d-updates in the MCMC algorithm. For a given value of d_{un}, it seems reasonable to give the chain a period to explore the distribution of λ and z. Hence we perform a d-update every fifth iteration of the chain, coupled with performing two K-jumps within each iteration we explore the path space adequately.

Adapting To An a-update The d_{un}-update described will only add or remove individuals from the unobserved component, excluding the zeroth generation. However, it is a simple matter to amend the algorithm into an a-update.

Once we have chosen the component in which we wish to update the number of initial infectives, then increasing or decreasing a requires removing or adding to d, since for an a-update we treat $D = a + d$ as a constant.

Thus we can use the function $J_0(z)$, defined in Equation (3.17), except alter the condition that $i = \text{un}$ to the component we wish to alter the number of initial infectives.

3.3.7 Edge Representation For Unknown Number Of Susceptibles

In contrast to the case of unknown number of infectives, the edge representation now requires a new type of update, we must add or remove vertices from the digraph as a whole.

Note, removing a vertex is not the same as deleting all edges into and out of it. Instead, the removed vertex has no part in the digraph. Select a vertex at random to remove, then check the candidate digraph is consistent with the observed data. Adding a vertex does not require a connectivity check. We omit further details, though such updates will follow a similar form of those in previous sections.

3.3.8 Generation Representation For Unknown Number Of Susceptibles

Hayakawa et al. (2003) address the case of an unknown number of susceptibles using a temporal approach and MCMC. Our generation approach cannot make inference for

the infectious periods as is done by the authors as has been discussed previously. Thus, we use a fixed infectious period of specified length.

As an interesting point, to adapt the edge representation for an unknown number of infectives was trivial, yet to adapt the generation representation required a new d-update to be developed. Conversely, for an unknown number of susceptibles, the edge method requires adding extra vertices to the digraph, whereas the generation method requires a trivial change. This difference is not explored further, however it suggests there may be some benefit to either method in certain situations.

The path representation, z, is unaffected by altering N_{un}. The number of unobserved susceptibles is accounted for in the likelihood of the path z given the infection rate λ, the observed component ψ_{ob} and the unobserved component ψ_{un}. The likelihood is presented as Equation 3.12, where there is dependence upon N, the total population size.

Unlike the case of an unknown number of infectives, for an unknown number of susceptibles there is no upper bound for the range of N_{un}. Hence, the posterior density may be driven mainly by the prior density and not the data; since the data provide no information directly about the number of unobserved susceptibles.

Since we assume $a_{\text{un}} = 0$, then $N_{\text{un}} = n_{\text{un}}$. If we choose to propose a candidate n_{un} according to the prior distribution, i.e. $q(n'_{\text{un}}|n_{\text{un}}) = \pi(n_{\text{un}})$, then the acceptance probability becomes the ratio of the likelihoods,

$$\begin{aligned}\alpha(n_{\text{un}}, n'_{\text{un}}) &= \min\left\{1\,,\ \frac{\pi(z,\lambda,n'_{\text{un}}|\theta_{\text{ob}},D_{\text{un}},I)q(n_{\text{un}}|n'_{\text{un}})}{\pi(z,\lambda,n_{\text{un}}|\theta_{\text{ob}},D_{\text{un}},I)q(n'_{\text{un}}|n_{\text{un}})}\right\}\\ &= \min\left\{1\,,\ \frac{\pi(z|\lambda,n'_{\text{un}},I)}{\pi(z|\lambda,n_{\text{un}},I)}\right\}.\end{aligned}$$

Algorithm 3.6 summarises the update when using the prior density as the proposal distribution. Though a valid algorithm, if the prior distribution is very sparse, then the proposals may have a very low acceptance probability since they will differ drastically from the current value, e.g. $n_{\text{un}} \sim \text{U}[0, 10^6]$ is an extreme example. This is an issue due to the high correlation of the infection rate λ to the number of susceptibles N.

Algorithm 3.6: n-update for partially observed one-type one-level model

```
Propose n'_un ~ π(n_un);
Calculate acceptance probability α(n_un, n'_un);
Draw A ~ U(0, 1);
if α < A then
    reject n'_un
else
    accept n'_un
```

3.3.9 Results

We have not implemented the edge or Poisson representations for the partially observed model. So we cannot make direct comparisons between properties of the MCMC, e.g. rate of convergence, mixing or parameter correlations. However, for the partially observed model using the Poisson representation with $\eta = 0.1$, i.e. observing 10% of the total population, using the data for an outbreak of influenze in Tecumseh, Michigan (see Table 3.9), O'Neill (2009) report run-times of several days. We have not performed an analysis on that data using the partially observed generation method. However, for the one-type one-level model, the run-times were of the order of several minutes to an hour. The reduction of the unobserved component to a single vector, as opposed to the entire digraph, gives a great improvement in computation time for the partially observed models.

We compare the generation representation to results from various authors, using dif-

ferent techniques and assumptions. The results are comparable, in their estimates and reported cost of computation. Following previous literature, we report the estimate for the infection rate as the basic reproductive number, $R_0 = \iota\lambda$, to enable comparison of various infectious periods. However, for the present case we only consider a fixed infectious period since it has the best convergence properties and minimal cost to compute.

Table 3.6 is for the case of an unknown number of infectives, with a fixed observed component and varying total population size. The point estimates for the reproductive number are comparable. The chains were run for 10^6 iterations, performing two K-jump Z-updates every iteration and a d_{un}-update every five iterations. Thus, for the larger populations the number of iterations may need to be increased to allow the chain to explore the larger path space.

The important observations are still evident despite the anomaly. Namely, for decreasing η the standard deviation of the estimate decreases as does the interval estimate for the epidemic being below threshold. As discussed in Demiris and O'Neill (2005b), the decreasing standard deviation seems counter intuitive, as we are observing a smaller proportion of the population we expect our uncertainty to increase. However, as the observed component is kept constant, we are actually assuming more about the unobserved component. Since the only information is contained in ψ_{ob}, hence we are implicitly assuming a major outbreak has occurred. This is illustrated by the reducing probability of the epidemic being below threshold.

As a more fair comparison, in Table 3.7 we assume the total population is fixed as $\psi = (1200, 300)$, then reduce η resulting in smaller observed components. We assume the observed component is exact, i.e. the proportion of individuals that are infected to those that avoid infection is the same. Contrasting $\eta = 1$ and $\eta = 0.01$, we see

that the reproductive number is higher and the probability of being below threshold is larger for the smaller observed component. Since we are inferring the behaviour of the entire population of 1200 individuals from a sample of 12, it is unsurprising that the standard deviation of the estimate is larger, as we are less sure about the point estimate. This is reflected in the interval estimate of R_0 being less than one, showing a greater uncertainty of the epidemic. It could be, the 12 individuals sampled by chance include the only infectives in the population, i.e. there was only a minor outbreak. Our method allows this possibility, and this is apparent in the probability of the epidemic being below threshold. Conversely the estimate for R_0 does increase for smaller observed components, since by the reverse reasoning the epidemic could be a major outbreak.

The uncertainty in the reproductive number is linked to the uncertainty of the final size, not knowing the total number of infectives. Table 3.8 shows the estimates and 95% intervals (as defined in Section 1.3.1.4) for D corresponding to the epidemics in Table 3.7. Again, contrasting $\eta = 1$ and $\eta = 0.01$, we see the interval in the latter case is very wide. This corresponds to the uncertainty on the outbreak within the total population, by chance we may under or over sample the number of infected individuals in such a small sub-sample, this is reflected in the wide interval for d_{un}. The correlation between $R_0 = \lambda c$ and d_{un} is estimated to be 0.711 and a plot is shown in Figure 3.7, as expected the parameters are clearly correlated.

3.4 Multi-type Epidemics

In the previous section we have introduced the concept of observed and unobserved individuals within the epidemic. To accommodate this in the generation representation an additional column was added to the representation of the path z. The unobserved

$(N_{\text{ob}}, D_{\text{ob}})$	η	N	R_0	(sd)	$\mathrm{P}[R_0 \leq 1]$
$(120, 30)$	1	120	1.1826	(0.2165)	0.2016
	0.5	240	1.1493	(0.1491)	0.1599
	0.2	600	1.1577	(0.1010)	0.0609
	0.1	1200	1.1603	(0.0749)	0.0205

Table 3.6: Estimates for λ, reported as $R_0 = \iota\lambda$ for a fixed infectious period of 4.1 days, observing a single initial infective with 119 initial susceptibles conditioned on $D_{\text{ob}} = 30$, with an unobserved component of the population of size $\frac{1-\eta}{\eta} N_{\text{ob}}$.

(N, D)	η	$(N_{\text{ob}}, D_{\text{ob}})$	R_0	(sd)	$\mathrm{P}[R_0 \leq 1]$
$(1200, 300)$	1	$(1200, 300)$	1.1574	(0.066969)	0.00801
	0.75	$(900, 225)$	1.1590	(0.067398)	0.00649
	0.5	$(600, 150)$	1.1605	(0.067590)	0.00618
	0.1	$(120, 30)$	1.1652	(0.071719)	0.00931
	0.01	$(12, 3)$	1.1680	(0.110924)	0.07427

Table 3.7: Estimates for λ, reported as $R_0 = \iota\lambda$ for a fixed infectious period of 4.1 days, observing a varying fraction of the total population, where $\psi = (N, D) = (1200, 300)$. We assume the sub-population has the same proportion of infected individuals as the total population and a single initial infective is is the observed component.

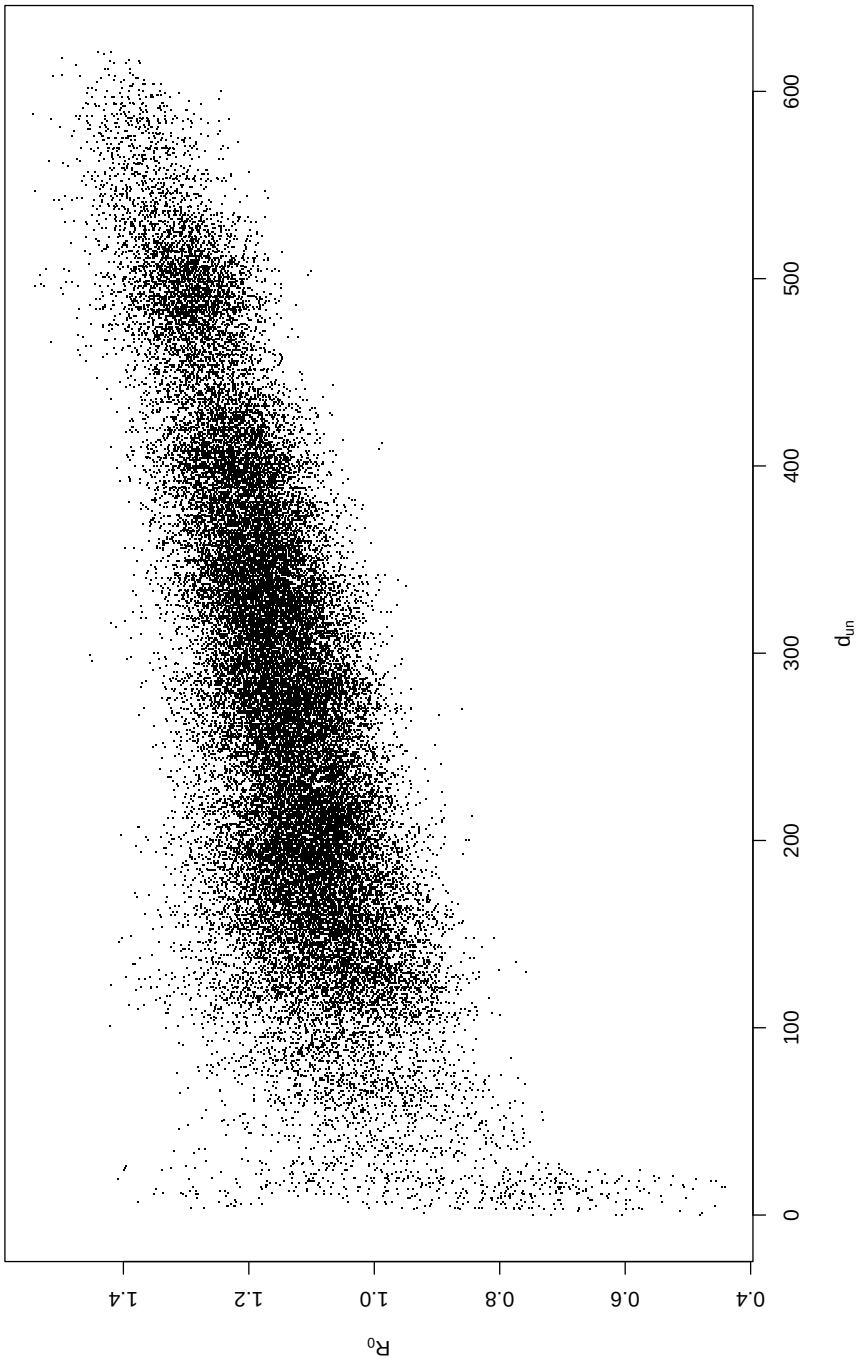


Figure 3.7: Plot of $R_0 = \lambda c$ against d_{un} for the case of a fixed population $\psi = (N, D) = (1200, 300)$ and $\eta = 0.01$, with a fixed infectious period of 4.1 days. The plot shows the high correlation of 0.711 between the two parameters, as well as the increasing uncertainty in λ for smaller d_{un}.

(N, D)	η	$(N_{\text{ob}}, D_{\text{ob}})$	d_{un}	D	95% highest posterior density region for D
$(1200, 300)$	1	$(1200, 300)$	0	300	(300,300)
	0.75	$(900, 225)$	75.287	300.287	(284,318)
	0.5	$(600, 150)$	151.312	301.312	(273,332)
	0.1	$(120, 30)$	281.904	311.905	(203,367)
	0.01	$(12, 3)$	282.280	285.280	(63,526)

Table 3.8: Estimates for d_{un} and $D = D_{\text{ob}} + d_{\text{un}}$ for the partially observed epidemics in Table 3.7, giving point estimate and highest posterior density region.

individuals were then subject to one of two new updates, either the d-update or the n-update for unknown numbers of infectives or susceptibles respectively; as well as the modified K-jump and λ-update.

Implicitly, we have two types of individual in the model, observed and unobserved. If we do not perform the d-update or n-update, then the two types are completely observed, instead we can adjust the model to include new infection rate parameters. Thus, we have two types of individual who can differ in infectivity within the model.

We shall now present the details of extending the generation model to include multiple types of individuals. Using the framework developed for partially observed models, the modification is primarily to the likelihood and λ-update. This section will be an outline of the amendments, as we shall combine multi-type and multi-level models together in Section 3.5.

In Section 3.4.1 we derive the path step probability for a fixed infectious period, the product of step probabilities along a path is the likelihood of z. Section 3.4.3 considers the form of Λ, the matrix of infection rates, as well as the MCMC update.

3.4.1 Multiple Infection Rates For Fixed Infectious Periods

Denote the type of an individuals by i, and for the moment consider a two-type model where $i \in \{1, 2\}$. In the most general case, we consider the infection rate between each combination of individuals that can occur, i.e. let Λ be the matrix of infection rates λ_{ij} from an individual of type i to type j. For the two-type case we have,

$$\Lambda = \begin{pmatrix} \lambda_{11} & \lambda_{12} \\ \lambda_{21} & \lambda_{22} \end{pmatrix}.$$

The infection rates need to be interpreted with care, they represent the rate of the Poisson process that determines the contacts an individual has during their infectious period.

Recall, for the one-type one-level model there were potential problems in attempting to make inference for many correlated parameters: the infection rate λ, the number of initial infectives a, the indicies of the initial infectives κ and the imputed path z, using only two or three numbers. We now introduce further, potentially highly correlated, parameters that may affect the inference. Care must be taken not to overfit the model if the data do not support the estimation of so many parameters.

Multi-type Likelihood

Generation t now consists of $x_{1,t}$ individuals of type 1 and $x_{2,t}$ individuals of type 2. A path z is still the product of independent step probabilities, where we must now modify

to consider the case of different types of individual. Therefore the step probability is,

$$\mathrm{P}_\theta^I\left(z_{t+1} = \begin{bmatrix} x_{1,t+1} & x_{2,t+1} \\ y_{1,t+1} & y_{2,t+1} \end{bmatrix} \middle| z_t = \begin{bmatrix} x_{1,t} & x_{2,t} \\ y_{1,t} & y_{2,t} \end{bmatrix}\right),$$

where we currently assume all individuals have independent and identically distributed infectious periods equal in distribution to I, i.e. for individual i, $I^i \stackrel{d}{=} I$. We assume for simplicity that the infectious period has the same distribution across multiple types of individual. This is reasonable to assume in many situation. Initially, we shall assume a fixed infectious period of all individuals, i.e. $I = c$. However, we shall return to this assumption and relax both the restriction to a fixed period and the need for all individuals to have an identical infectious period distribution in Chapter 4.

Hence, the step probability can be expressed in terms of the probability of avoiding infection from the previous generation, exactly as was the case for the one-type model. The probability of an individual of type 1 avoiding infection in generation $t+1$, given $x_{1,t}$ and $x_{2,t}$, is

$$\begin{aligned} &\exp\left(-\frac{\lambda_{11}}{N}\left(\zeta^{(1)} + \cdots + \zeta^{(x_{1,t})}\right)\right) \exp\left(-\frac{\lambda_{21}}{N}\left(\zeta^{(1)} + \cdots + \zeta^{(x_{2,t})}\right)\right) \\ &= \exp\left(-\sum_{i=1}^{2}\frac{\lambda_{i1}}{N}\sum_{k=1}^{x_{i,t}}\zeta^{(k)}\right) \\ &= \exp\left(-\sum_{i=1}^{2}\frac{\lambda_{i1}}{N}\, x_{i,t}\, c\right) \end{aligned}$$

That is, the product of avoiding an infectious contact with all infectives of type 1 in generation t and all infectives of type 2 in generation t. The probability of being infected is thus one minus this probability. Since we assume $I^i = c$, we can simplify the sum of infectious periods, $\zeta^{(k)}$, as shown.

In full, the step probability for a two-type model is

$$\begin{aligned}
\mathrm{P}_\theta^I &\left(z_{t+1} = \begin{bmatrix} x_{1,t+1} & x_{2,t+1} \\ y_{1,t+1} & y_{2,t+1} \end{bmatrix} \middle| z_t = \begin{bmatrix} x_{1,t} & x_{2,t} \\ y_{1,t} & y_{2,t} \end{bmatrix} \right) \\
&= \binom{N_1 - y_{1,t}}{x_{1,t+1}} \left(1 - \exp\left(-A_1\right)\right)^{x_{1,t+1}} \left(\exp\left(-A_1\right)\right)^{N_1 - y_{1,t+1}} \\
&\quad \times \binom{N_2 - y_{2,t}}{x_{2,t+1}} \left(1 - \exp\left(-A_2\right)\right)^{x_{2,t+1}} \left(\exp\left(-A_2\right)\right)^{N_2 - y_{2,t+1}} \\
&= \prod_{j=1}^{2} \binom{N_j - y_{j,t}}{x_{j,t+1}} \left(1 - \exp\left(-A_j\right)\right)^{x_{j,t+1}} \left(\exp\left(-A_j\right)\right)^{N_j - y_{j,t+1}}, \qquad (3.18)
\end{aligned}$$

where

$$A_j = \sum_{i=1}^{2} \frac{\lambda_{ij}}{N} x_{i,t}\, c, \qquad j = 1, 2.$$

From Equation (3.18), it is simple to extend to an arbitrary number of types of individuals. Each new type will contribute an additional term to the A_j sums and an additional term in the product.

3.4.2 Varying The Form Of The Infection Rate Matrix

The current model is for individuals mixing homogeneously within the population. The addition of the infection matrix, Λ, allows us to model varying infectivity within the population.

However, we can determine different forms of the infection matrix Λ, i.e. impose additional constraints on the entries in the matrix. Obviously we can choose to set certain rates to zero in the model, corresponding to a type of individual being unable to infect another. Such a model is uncommon for the multi-type setting, though will be more

relevant for the multi-level model. For example, in a three-type model we might suppose that only type 2 and 3 individuals can infect type 3, giving an infection matrix of the form,

$$\Lambda = \begin{pmatrix} \lambda_{11} & \lambda_{12} & 0 \\ \lambda_{21} & \lambda_{22} & \lambda_{23} \\ \lambda_{31} & \lambda_{32} & \lambda_{33} \end{pmatrix}.$$

For the most general model, consisting of H types of individual, there are H^2 infection rates to make inference for. The data will consist of $2H$ pieces of information, the total number N_i, and how many ultimately became infected D_i, for each type $1 \leq i \leq H$, during the epidemic (we currently consider a fully observed epidemic).

Thus, the model will over fit the data with a large number of types. Determining whether parameters are identifiable given final size data has been investigated by Britton (1998), who considers whether infectivity can be estimated given equal susceptibility. To account for this it is common to place further constraints upon the infection matrix Λ. These constraints will determine the model for which we are making inference. There are four common models for the infection matrix Λ in the literature. In the following we consider the multi-type one-level model with H distinct types of individual. These forms for Λ will be adapted for a multi-level model in Section 3.5.

General Model As mentioned, the general model has H^2 infection rates without any further constraints. That is,

$$\Lambda_{\text{gen}} = \begin{pmatrix} \lambda_{11} & \lambda_{12} & \cdots & \lambda_{1H} \\ \lambda_{21} & \lambda_{22} & & \\ \vdots & & \ddots & \\ \lambda_{H1} & & & \lambda_{HH} \end{pmatrix}. \tag{3.19}$$

The general model suffers from highly correlated parameters, partly due to over fitting the data. For final size data, the correlation adversely affects the mixing and convergence of the MCMC chains.

Product Model The product model, considered by Britton (1998) in terms of estimating parameters, is a reduction of the general model into only $2H$ infection rates, where Λ is the product for two H length vectors, i.e. $\Lambda = \beta\alpha^T$, for $\beta = (\beta_1, \ldots, \beta_H)$ and $\alpha = (\alpha_1, \ldots, \alpha_H)$. The infection rate between two types of individual is the product of their separate rates, i.e. $\lambda_{ij} = \beta_i\alpha_j$,

$$\Lambda_{\text{prod}} = \begin{pmatrix} \beta_1\alpha_1 & \beta_1\alpha_2 & \cdots & \beta_1\alpha_H \\ \beta_2\alpha_1 & \beta_2\alpha_2 & & \\ \vdots & & \ddots & \\ \beta_H\alpha_1 & & & \beta_H\alpha_H \end{pmatrix}. \tag{3.20}$$

Reducing to H parameters greatly reduces the correlation within Λ and is a more justifiable number of rates to make inference from. For the general model there are

more infection parameters than data points for $H > 2$. The constraint of symmetric infection rates may not be biologically accurate.

Susceptibility Model Again we reduce to only H parameters, one per type of individual, however the form of Λ is very different to the product model.

We consider the rate λ to encapsulate the susceptibility of an individual to being infected, instead of the infectivity. That is, the rate from type i to type j depends only on the target type j (the susceptible), i.e. $\lambda_{ij} = \lambda_j$, thus we consider λ_j to be the susceptibility of type j to infection.

$$\Lambda_{\text{sus}} = \begin{pmatrix} \lambda_1 & \lambda_2 & \cdots & \lambda_H \\ \lambda_1 & \lambda_2 & & \lambda_H \\ \vdots & & \vdots & \vdots \\ \\ \lambda_1 & & & \lambda_H \end{pmatrix}. \tag{3.21}$$

Infectivity Model The opposite to the susceptibility model is the infectivity model, where the rate between two types depends only on the origin type (the infector), i.e. $\lambda_{ij} = \lambda_i$.

$$\Lambda_{\text{inf}} = \begin{pmatrix} \lambda_1 & \lambda_1 & \cdots & \lambda_1 \\ \lambda_2 & \lambda_2 & \cdots & \lambda_2 \\ \vdots & & & \\ \\ \lambda_H & \lambda_H & \cdots & \lambda_H \end{pmatrix}. \tag{3.22}$$

It is important to note that the inferred parameters for the susceptibility and infectivity

model will be very different, despite both being a H length vector, i.e. $(\lambda_1, \cdots, \lambda_H)$. The notation used is slightly misleading, Λ_{sus} is not simply the transpose of Λ_{inf}.

3.4.3 Λ-update

The posterior density of interest now includes the matrix of infection rates Λ, which may itself be a function of sub-parameters, i.e. $\pi(\Lambda, z|\theta, I)$. Using the modified likelihood from the previous section, it is sufficient to replace the single infection rate λ by the matrix Λ in all expressions for the posterior density considered so far.

For the prior we may use independent exponentials of rate μ as before, with a sufficiently small rate to form a relatively flat prior. We may assume independent priors for each component of Λ. Thus

$$\pi(\Lambda) = \prod_{i=1}^{H} \prod_{j=1}^{H} \pi(\lambda_{ij}).$$

Where there are H types of individual under the general model for Λ. If we consider an alternate form of Λ, the prior will be H independent distributions,

$$\pi(\Lambda) = \prod_{i=1}^{H} \pi(\lambda_i).$$

We shall assume there are H infection rate parameters, i.e. Λ is of the form of Equation (3.20), (3.21) or (3.22).

For the proposal distribution, we use a multivariate normal with covariance matrix Σ. This allows us to propose correlated candidate parameters, which may lead to higher

acceptance rates. Let $\Lambda = (\lambda_1, \ldots, \lambda_H)$, the H length row vector of parameters, then

$$q(\Lambda'|\Lambda) \sim \mathrm{N}_H(\Lambda, \Sigma).$$

For a two-type model it is convenient to update both λ_1 and λ_2 simultaneously. However, for larger numbers of parameters it may be more efficient to use block updates as discussed in Section 1.3.2, or perhaps update each parameter individually.

The acceptance probability is similar to that derived in Section 3.2.2.2, however using the multi-type likelihood derived in Equation 3.18, the multivariate normal proposal distribution and independent priors for each parameter.

3.4.4 Z-update

Each type of individual will be a separate column in the representation of the path z. The total of each column is conditioned to be d_i for type $1 \leq i \leq H$.

Assuming a fully observed epidemic, then we need only perform K-jumps as modified for the partially observed setting in Section 3.3.6.3.

However, there is an issue concerning the initial infective. If we assume $a = 1$, where $a = a_1 + \cdots + a_H$, then the type of the initial infective could have an effect on the inference. Whereas for the partially observed one-type case we could assume a single observed initial infective without loss of generality, we must now allow for the type to vary.

To alter the type of the initial infective, the current path must be altered in two ways, first adapting the function $J_0(\cdot)$, we determine all generations that contain individuals

that can be moved, excluding the type of the current initial infective. To emphasis the different we denote this function $J_{(i)}(z)$, where i denotes the type of the current initial infective. Unlike the k-jump search, $\mathcal{J}_{(i)} = \sum_t \sum_j J_{(i)}(x_{j,t})$ can be zero, i.e. there may not be any free individuals.

$$J_{(i)}(x_{j,t}) = \begin{cases} 1 & \begin{cases} \text{if } j \neq i, x_{j,t} > 0 \text{ and } x_t > 1 \\ \text{if } j \neq i, x_{j,t} > 0 \text{ and } t = \tau \end{cases} \quad \text{for} \quad \begin{matrix} 1 \leq i, j \leq H \\ 1 \leq t \leq \tau. \end{matrix} \\ 0 & \text{otherwise} \end{cases}$$

Then, select an individual to move and place them in the zeroth generation. The current initial infective is then moved to a generation at random (within their own type).

The proposal distribution is composed of the product of the two independent steps, determining a free individual to become the new initial infective and removing the current initial infective.

$$q(z'|z) = \frac{1}{\mathcal{J}_{(i)}} \frac{1}{\tau + 1}.$$

The proposal is reversible, since the original initial infective must be free (it is either added to a valid generation or at the end of the path) and the candidate initial infective can be returned to its origin generation. The a-update is summarised in Algorithm 3.7.

3.4.5 Partially Observed Multi-type Model

The multi-type model was developed from the partially observed case having two types of individual. The unobserved individuals were special only in that they were subject

Algorithm 3.7: a-update within Z for multi-type model with fixed a

Determine the type of the current initial infective, i;
Calculate $J_{(i)}(z)$ and $\mathcal{J}_{(i)}$;
Sample $g \sim \text{Uni}[1, \mathcal{J}_{(i)}]$;
Determine the origin generation t_O and type i_O corresponding to g;
Sample $t \sim \text{U}[1, \tau + 1]$;
Construct z' by;
    $x'_{i,0} = x_{i,0} - 1 = 0$;
    $x'_{i,t} = x_{i,t} + 1$;
    $x'_{i_O,t_O} = x_{i_O,t_O} - 1$;
    $x'_{i_O,0} = x_{i_O,0} + 1 = 1$;
Calculate acceptance probability α;
Draw $A \sim \text{U}(0, 1)$;
if $\alpha < A$ **then**
    reject z'
else
    accept z'

to d-updates or n-updates.

For any of the Λ models described there are several additional constraints that can be commonly applied. The first, already mentioned, is to set a rate to zero prior to the MCMC run. It may be only a single entry in the infection matrix, or all locations of that parameter. Obviously, setting a parameter to zero in the product model means that type cannot be infected nor infect any other type, which is not sensible.

Secondly, we can set several parameters to be equal, thus considering the two types to be the same. Then, we can consider one of the equal types to by an unobserved component, performing d-updates or n-updates are desired (perhaps both on different types).

Combining everything together, we have developed a model to make inference for partially observed multi-type models with both unknown numbers of susceptibles and infectives of different types (we cannot have an unknown number of susceptibles and

infectives of the same type for obvious reasons).

3.4.6 Types: Definition And Notation

If we consider partially observed multi-type epidemics, it is unwieldy to consider H types, where some types have unobserved components and others not. Instead, we decompose a type into a set of covariates. Consider the example where each individual is classed as either an adult, child or infant, and is either observed or unobserved.

Let the type of an individual be defined by W mutually exclusive covariates, e.g. age and component. Define the set $\mathbb{H}_w$ to be all possible values of the w^{th} covariate, for $1 \leq w \leq W$. Thus there are $H_w = |\mathbb{H}_w|$ values of each covariate, with each individual having a specific value. For an individual i, define the function $\mathcal{H}_w(i)$ be the value of the w covariate, which we shall denote by h_w. We can enumerate the set $\mathbb{H}_w$ such that, $1 \leq \mathcal{H}_w(i) \leq H_w$ for $1 \leq w \leq W$. Define $\mathcal{H}(i)$ to be the vector of covariates of individual i and $\mathbb{H}$ to be the set of all possible W-tuples of $\mathbb{H}_1 \times \cdots \times \mathbb{H}_W$. Then in total there are $H = |\mathbb{H}| = H_1 \times \cdots \times H_W$ possible types of individual, though there need not be an individual that exhibits every possible set of covariates.

Returning to our example, a partially observed three-type epidemic consisting of adults, children and infants that have both an observed and unobserved component. There are two covariates age and component, i.e. $W = 2$. For age there are three values of individual and each individual is in one of the two components, i.e. $H_1 = 3$ and $H_2 = 2$. Using meaningful labels, instead of numbers, $h_1 \in \mathbb{H}_1 = \{\text{adult}, \text{child}, \text{infant}\}$ and $h_2 \in \mathbb{H}_2 = \{\text{ob}, \text{un}\}$. Then, for each individual i, $\mathcal{H}(i)$ is a length two vector in the

following set,

$$\mathcal{H}(i) = (h_1, h_2) \in \mathbb{H} = \Big\{(\text{adult}, \text{ob}), (\text{child}, \text{ob}), (\text{infant}, \text{ob}), \\ (\text{adult}, \text{un}), (\text{child}, \text{un}), (\text{infant}, \text{un})\Big\} \text{ for all } 1 \leq i \leq N.$$

By considering the type of an individual as a set of disjoint covariates, we can more easily express the relationship between individuals. Clearly, considering the above as a single list of six types of individual is equivalent, though perhaps more confusing notation wise. In this thesis we only consider models with two covariates, one being a type of individual and the other being observed or unobserved.

The infection matrix Λ, is now an $H \times H$ sized matrix in general. Thus it is appropriate to use a model with additional constraints in order to reduce the number of parameters that we make inference for.

3.5 Multi-type Multi-level Epidemics

In Section 3.2 we began with a simple one-type one-level model for a homogeneous and homogeneously mixing population. By extending the generation representation in Sections 3.3 and 3.4 we have a general framework for multi-type models allowing for partially observed populations (a special type of individual).

For the multi-type model we can consider varying infectious period distributions, either as additional parameters or integrated out of the likelihood using expectations. We have not shown this explicitly for the multi-type case, a more complete derivation will be presented in Chapter 4.

Our final extension to the generation representation is to allow for multiple levels of mixing, as introduced in Section 1.2.5.2. Specifically, the inclusion of households where individuals are able to make local infections within their household and global infections with the population are differing rates.

The addition of multiple levels is similar to multiple types, in Section 3.5.1 we introduce the definition and notation of levels. We present a general framework for an arbitrary number of types and levels.

As was discussed in Section 3.4.1, a model with arbitrary numbers of types and levels will over fit the available data. Therefore we present suitable constrained forms for the infection matrix Λ in Section 3.5.2.

For this chapter we have mainly focused on a fixed infectious period, for simplicity and computational ease. Continuing in this vein, we present the multi-type multi-level model likelihood for fixed infectious periods in Section 3.5.3.

Before applying the generation method to a data set in Section 3.6, Section 3.5.4 summarises the MCMC update algorithms developed.

3.5.1 General Framework For Multi-type Multi-level Models

Returning to the case of a fully observed epidemic with a homogeneous population, i.e. one-type, we now address the need to account for non-homogeneous mixing that real populations exhibit.

As discussed in Section 1.2.5.2, initial work has focused on a two-level mixing model, where individuals are members of households, see for example Ball et al. (1997).

3.5.1.1 Multi-level Notation

Let V denote the number of levels of mixing, so for the household model we have $V = 2$. For the first level of mixing, we consider all individuals able to mix with all others at some rate, termed global contacts. Each individual is a member of a single household (for the current discussion), then the second level consists of all possible households. Within a household, individuals make contacts at a different, typically larger, rate called local.

Hence, we can introduce a label for each individual determining their relationship to other individuals, similarly as we did for the different covariates that comprise an individuals type in Section 3.4.6.

Let L_v be the number of groups within the v^{th} level and $\mathbb{L}_v$ be the set of such groups (commonly given numerical labels) for $1 \leq v \leq V$. Similarly, for an individual i in the population, define the function $\mathcal{L}_v(i)$ to be the group individual i belongs to in level v, the value of which we shall often denote by l_v. Finally, let $\mathcal{L}(i)$ be the vector of groups individual i belongs to, such that

$$\mathcal{L}(i) = (\mathcal{L}_1(i), \ldots, \mathcal{L}_V(i)) = (l_1, \ldots, l_V) \in \mathbb{L} = \mathbb{L}_1 \times \cdots \times \mathbb{L}_V.$$

Consider the example setting of Figure 3.8, consisting of individuals that have two levels of mixing, within the entire population and within their own group. In this case, $V = 2$ and $(L_1, L_2) = (1, 4)$ and each individual is a member of a distinct level set. In this case, there are only four such combinations of levels, i.e. determining to which group an individual belongs.

As for the multi-type model, we encode the varying rates between levels in terms of

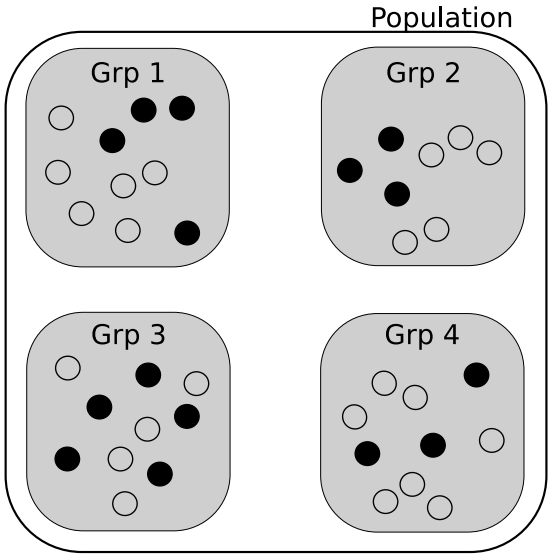


Figure 3.8: Diagram of an example one-type two-level setting with four groups (commonly called households), i.e. $V = 2$ and $(L_1, L_2) = (1, 4)$. Filled circles indicate individuals who were infective during the course of the epidemic.

the infection matrix Λ. In the most general model, we consider the matrix to have L^2 unique rates, where $L = |\mathbb{L}|$. We shall consider the form of Λ in Section 3.5.2, for now consider each entry λ_{ij} as a rate from an individual in group set i to an individual in group set j, for $i, j \in \mathbb{L}$.

For a fixed infectious period, the likelihood of the path can be decomposed into independent step probabilities as for the multi-type case. Again, we omit varying the infectious period until Chapter 4.

Thus, for the path z we have a column for individuals for each unique level set. Importantly, the model does not allow for migration of individuals between groups during the epidemic, it is assumed such events do not occur.

For the infection rates thus far we have always normalised by the total population size N. For multi-level epidemics it is common to normalise by different constants for different levels of mixing. For example, in the two-level household model the local rate

is usually not normalised.

Since the entry λ_{ij} in the infection matrix will be for a prescribed form, as discussed in Section 3.5.2, we shall include the normalising constants within λ. Hence, in the following expressions consider λ_{ij} to be a function of other parameters.

Then using the example case of Figure 3.8, the probability an individual in group 1 avoids infection from all individuals in generation t is,

$$\exp\left(-\lambda_{11}\left(\zeta^{(1)}+\cdots+\zeta^{(x_{1,t})}\right)\right)\ \exp\left(-\lambda_{21}\left(\zeta^{(1)}+\cdots+\zeta^{(x_{2,t})}\right)\right)$$
$$\exp\left(-\lambda_{31}\left(\zeta^{(1)}+\cdots+\zeta^{(x_{3,t})}\right)\right)\ \exp\left(-\lambda_{41}\left(\zeta^{(1)}+\cdots+\zeta^{(x_{4,t})}\right)\right),$$

where ζ is the infectious period of an individual (we index the individuals within each generation, hence the parenthesis around the superscript), and $x_{i,t}$ is the number of individuals of groups i in generation t. Since we are considering a fixed infectious period, i.e. $\zeta = c$ for all individuals, then the above product reduces to

$$\exp\left(-\sum_{i=1}^{4}\lambda_{i1}\ c\ x_{i,t}\right),$$

and the probability of being infected is one minus the avoidance probability.

In full, the step probability for a two-level model with four distinct level sets and a fixed infectious period is

$$\begin{aligned}\mathrm{P}_\theta^I\left(z_{t+1}=\begin{bmatrix}x_{1,t+1} & x_{2,t+1} & x_{3,t+1} & x_{4,t+1}\\ y_{1,t+1} & y_{2,t+1} & y_{3,t+1} & y_{4,t+1}\end{bmatrix}\middle| z_t=\begin{bmatrix}x_{1,t} & x_{2,t} & x_{3,t} & x_{4,t}\\ y_{1,t} & y_{2,t} & y_{3,t} & y_{4,t}\end{bmatrix}\right)\\ =\prod_{j=1}^{4}\binom{N_j-y_{j,t}}{x_{j,t+1}}\left(1-\exp\left(-A_j\right)\right)^{x_{j,t+1}}\left(\exp\left(-A_j\right)\right)^{N_j-y_{j,t+1}},\end{aligned}\tag{3.23}$$

where

$$A_j = \sum_{i=1}^{4} \lambda_{ij}\, x_{i,t}\, c, \qquad j = 1, 2, 3, 4.$$

The data in the matrix is either

$$\psi = \begin{pmatrix} N_1 & N_2 & N_3 & N_4 \\ D_1 & D_2 & D_3 & D_4 \end{pmatrix} \quad \text{or} \quad \theta = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 \\ n_1 & n_2 & n_3 & n_4 \\ d_1 & d_2 & d_3 & d_4 \end{pmatrix},$$

depending on whether the number of initial infectives is assumed known.

3.5.1.2 Combining Multi-type and Multi-level Models

The likelihood for the multi-type and multi-level models given by Equations (3.18) and (3.23) respectively are almost identical in form.

Definition 3.3

An individual has a type defined in terms of it's covariates and a level set that define its relation to all other individuals. We call the combined covariates and levels of an individual its class, which we shall denote by ω. That is for every individual i, for $1 \leq i \leq N$,

$$\omega = (\mathcal{H}(i), \mathcal{L}(i)) = \mathcal{S}(i) \in \mathbb{S},$$

where $\mathbb{S} = \mathbb{H} \times \mathbb{L}$ and $S = |\mathbb{S}| = \prod_{w=1}^{W} H_w \prod_{v=1}^{V} L_v$. The function $\mathcal{S}$ applied to an individual i gives the class of that individual.

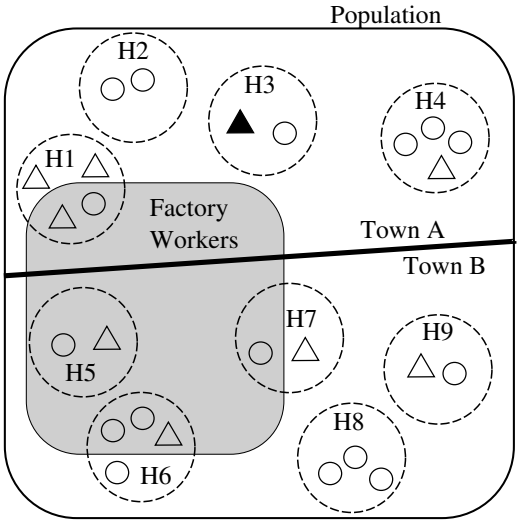


Figure 3.9: Diagram of an example multi-type multi-level setting with two types of individuals: circles and triangles; and four levels of mixing: global, within town, within households and within workplace.

Then the step probability for a multi-type multi-level fixed infectious period is

$$\begin{aligned} \mathrm{P}_\theta^I \left(z_{t+1} = \begin{bmatrix} x_{\omega_1,t+1} & \cdots & x_{\omega_S,t+1} \\ y_{\omega_1,t+1} & \cdots & y_{\omega_S,t+1} \end{bmatrix} \middle| z_t = \begin{bmatrix} x_{\omega_1,t} & \cdots & x_{\omega_S,t} \\ y_{\omega_1,t} & \cdots & y_{\omega_S,t} \end{bmatrix} \right) \\ = \prod_{\omega_j \in \mathbb{S}} \binom{N_{\omega_j} - y_{\omega_j,t}}{x_{\omega_j,t+1}} \left(\exp\left(- \sum_{\omega_i \in \mathbb{S}} \lambda_{\omega_i \omega_j} \, x_{\omega_i,t} \, c \right) \right)^{N_{\omega_j} - y_{\omega_j,t+1}} \\ \times \left(1 - \exp\left(- \sum_{\omega_i \in \mathbb{S}} \lambda_{\omega_i \omega_j} \, x_{\omega_i,t} \, c \right) \right)^{x_{\omega_j,t+1}} . \end{aligned} \tag{3.24}$$

Where ω_i and ω_j denote two classes of individual and $\lambda_{\omega_i \omega_j}$ is the entry in the infection matrix Λ corresponding to the rate between these two classes.

Example Setting We present a complicated example setting, with a highly structured population in Figure 3.9. The total population size is small, $N = 25$, therefore we would not attempt to fit such a complex model in practice, we only use this as a demonstration.

The population has one characteristic of gender for each individual, male (triangle, $\triangle$) and female (circle, $\circ$), i.e. $W = 1$, $H_1 = H = 2$ and $\mathbb{H} = \{\triangle, \circ\}$. There are four levels ($V = 4$) of mixing: within population ($L_1 = 1$), within the two towns A and B ($L_2 = 2$), within each household ($L_3 = 9$) and within an individuals place of work ($L_4 = 2$). Note that the within workplace level has two groups, those that work in the factory and those that do not; it is important to include a void group in each level so that every individual belongs to a group. Hence there are S classes of individual, where

$$S = \prod_{w=1}^{W} H_w \prod_{v=1}^{V} L_v = (2) \times (1)(2)(9)(2) = 72.$$

The general infection matrix Λ would require 72^2 separate parameters, which is clearly inappropriate. Thus we consider constrained forms of Λ shortly.

The data matrix ψ has S columns, one for each class of individual. We can summarise the ψ based on any characteristic or level by counting the number of individuals desired. For example, the number of males is

$$N_{\triangle} = |\{i : \mathcal{H}(i) = \triangle, 1 \leq i \leq N\}| = \sum_{i=1}^{N} \mathbb{I}_{\{\mathcal{H}(i)=\triangle\}}.$$

Similarly for any other identifier, or group of identifiers, for example the number of males in town B,

$$N_{\triangle,B} = |\{i : \mathcal{H}(i) = \triangle \text{ and } \mathcal{L}_2(i) = B, 1 \leq i \leq N\}| = \sum_{i=1}^{N} \mathbb{I}_{\{\mathcal{H}(i)=\triangle\}} \mathbb{I}_{\{\mathcal{L}(i)=B\}}.$$

Recall, since there is no temporal information in the generation representation the infection rates must be interpreted with care. For example, the rate between individuals

in the factory could implicitly model the factory not being open all day.

3.5.1.3 Types Or Levels

Mathematically, types and levels are equivalent and there is no reason to differentiate between them. In fact, in terms of the likelihood they contribute the same terms. Thus we could rephrase types as levels or vice versa.

The mechanism they differ by is in the form of the infection matrix entries. Recall, we consider the entry $\lambda_{\omega_i\omega_j}$ to be a function of some other set of parameters we have not yet defined. Returning to the three-type one-level partially observed model, it would be equivalent to consider this as a model with six types of individual. Using a more structured notation is useful for clarity of expressions and when implementing algorithms. For example, using the observed or unobserved covariate, we know only to apply d-updates or n-updates to unobserved classes. The interface is clearer and easily extended.

Also, though there may be a large number of classes, some of them may be empty, i.e. there are no individuals matching the specific combination of covariates and levels. The additional notation has been introduced to account for the most general case, for real life data sets a more specific notation may be more appropriate.

3.5.2 Models under consideration

We shall now restrict our attention to models with a single characteristic that consists of multiple-types and two levels of mixing: within population and within group (which we shall call households).

As mentioned, the full general infection matrix contains too many parameters for proper inference, thus we reduce the model by introducing a new set of parameters, with Λ being a function of these sub-parameters. Let $\Lambda_\psi(\cdot)$ be used to emphasis that the infection rate matrix is a function of these sub-parameters and the data ψ.

In Section 3.4.2 we presented the product model (3.20), susceptibility model (3.21) and infectivity model (3.22) for the multi-type one-level case. We now present the Product Model (PM) again, in terms of classes of individuals.

We present two specific models for the two-type two-level case, namely the Global-Local-Susceptibility (GLS) and Global-Susceptibility (GS). We shall use these in Section 3.6 to make comparison with the results of Demiris and O'Neill (2005a). Thus, there is one characteristic $W = 1$, with two types $\mathbb{H} = \{1, 2\}$ and there are two levels $V = 2$ a global population mixing and local mixing within the households, i.e. $\mathbb{L} = \{(1, 1), \ldots, (1, L_2)\}$ where there are L_2 households. For clarity we shall re-label the level sets as $\mathbb{L} = \{1, \ldots, L\}$ where there are L households.

Thus, the class of an individual can be expressed as $\omega = (il)$ for a type $1 \leq i \leq 2$ and a household $1 \leq l \leq L$. Then we write the rate from class ω to class υ for $\omega, \upsilon \in \mathbb{S}$, as

$$\lambda_{\omega\upsilon} = \lambda_{(il)(jm)}$$

Where $\lambda_{(il)(jm)}$ is the rate from an individual of type i in household l to an individual of type j in household m.

Recall, we have included the normalising constants in the function $\lambda_{\omega_i\omega_j}$. Thus in the following models for Λ we shall also specify the normalising constant, usually a function of the current data ψ, specifically the total number, N_ω, of a class ω. Note, we say current since altering ψ is possible via n-updates.

Global-Local-Susceptibility Model The Global-Local-Susceptibility model is an adaption of the susceptibility model to the case of global and local mixing. The rates depend only on the type of the target individual, i.e. the susceptible, thus the model accounts for varying susceptibility to the disease among different types of individual. There are two rates, global and local, to account for the two-levels of mixing and both depend only on the type of the target individual.

Therefore we reduce to the following sub-parameters, a vector of local rates $\Lambda^L = (\lambda_1^L, \lambda_2^L)$, a vector of global rates $\Lambda^G = (\lambda_1^G, \lambda_2^G)$ and ψ, i.e. four infection rate parameters instead of the $4L^2$ or $2L$ under the general and product models respectively. Then the infection rate matrix is a function

$$\Lambda_\psi(\lambda_1^G, \lambda_2^G, \lambda_1^L, \lambda_2^L).$$

We define the additive Global-Local-Susceptibility model as

$$\lambda_{(il)(jm)} = \begin{cases} \dfrac{\lambda_j^G}{N} & \text{for } l \neq m \\ \lambda_j^L + \dfrac{\lambda_j^G}{N} & \text{for } l = m, \end{cases}$$

note that we do not normalise the local infection rates, only the global. This is a reasonable model for small households of relatively uniform size.

For complicated models there will be some additive levels of mixing and some not. For the example in Figure 3.9, it would be reasonable to assume that the within workplace rate was additive with the within household rate, although the within town might not be additive with the within workplace. Any such model is valid.

Global-Susceptibility Model The Global-Susceptibility model is similar to the Global-Local-Susceptibility, except the model allows different rates depending on origin and target type within a household, i.e. local infections, but global infections still only depend on the target type. That is

$$\Lambda_\psi(\lambda_1^G, \lambda_2^G, \lambda_{11}^L, \lambda_{12}^L, \lambda_{21}^L, \lambda_{22}^L),$$

and the entries are

$$\lambda_{(il)(jm)} = \begin{cases} \frac{\lambda_j^G}{N} & \text{for } l \neq m \\ \lambda_{ij}^L + \frac{\lambda_j^G}{N} & \text{for } l = m. \end{cases}$$

3.5.3 Likelihood

The likelihood of a path z for a fixed infectious period is the product of step probabilities derived in Equation (3.24). Together with a form for the infection rate matrix Λ and θ.

In Chapter 4 we shall consider the likelihood of an arbitrary infectious period. However, the fixed period gives similar point estimates of the parameters, with varying uncertainty.

In Section 3.7 we shall discuss practical issues concerning computing the likelihood. For an MCMC algorithm, the likelihood must be computed for every Metropolis-Hastings update, i.e. all updates described for the generation representation.

3.5.4 Summary Of Update Algorithms And Seeds For Multi-type Multi-level Model

In Sections 3.2, 3.3 and 3.5 we have presented update algorithms for an MCMC scheme. Many of them require minor adaption to the the general setting or to a specific model.

3.5.4.1 Λ-update

The Λ-update is specific to the form of the infection matrix. However, given the set of sub-parameters used to determine the entries of Λ, the update for the sub-parameters can be done using a symmetric Random Walk Metropolis with a multivariate normal proposal distribution, as in Section 3.4.3.

Seed And Tunable Hyperparameters There are no general guidelines for seed values of the sub-parameters nor the covariance matrix Σ for the multivariate normal proposal. A small sample chain must be run to obtain estimates, though this will only be necessary if the number of sub-parameters is large. For the case studies that follow arbitrary seed values were sucessfully used.

3.5.4.2 Z-update

The path z is now a high dimensional matrix, consisting of a column for each unique class of individual. We have developed several Z-updates, each with a specific purpose. The K-jump, with tunable hyperparameter $K_{\max}$, is the primary Z-update. In general we shall perform multiple independent K-jumps sequentially within a single iteration. For all of the MCMC runs performed we compute two K-jumps per iteration, this

seems to give reasonable mixing characteristics.

There are two possible a-updates, the first is a modification to the K-jump allowing jumps to and from the zeroth generation. Then the total number of initial infectives may vary. The second maintains a fixed number of initial infectives, commonly $a = 1$, and updates which class (Algorithm 3.7 is in terms of types, but it is simple to adapt to classes) the initial infectives are.

The d-updates and n-updates are for unobserved classes with an unknown number of infectives or susceptibles respectively. Both updates are easily extended to the case of classes of individuals where the number of classes is fixed.

Seed And Tunable Hyperparameters The seed path is non-trivial to construct for the multi-type multi-level case. We desire to seed near to the posterior modal path to minimise the burn in period. From Section 3.2 and the investigation on Chapter 2 we determined that the approximate length is $2\sqrt{d}$, where d is the total final size across all classes.

Thus to construct the seed path, begin with the first class, ω, and let each generation be of size one, i.e. $x_{\omega,1} = \cdots = x_{\omega,d_\omega} = 1$ and $s_\omega = d_\omega$. Then, for the second class, υ, begin in the generation after the first class ended, i.e. $r_\upsilon = s_\omega + 1$, again letting each generation be of size one. Once the length has reached $2\sqrt{d}$, reset to the first generation. For classes with an initial infective, begin in the first generation to ensure a reasonable likelihood.

For example, consider a model with 15 classes, each with a final size of two, i.e. $D_\omega = 2$ for all ω, and a single initial infective in the fourth class. Then $d = 29$, giving an initial

seed length of $2\sqrt{29} \approx 11$. Then we construct the seed path as below,

$z =$

t	ω_1	ω_2	ω_3	ω_4	ω_5	ω_6	ω_7	ω_8	ω_9	ω_{10}	ω_{11}	ω_{12}	ω_{13}	ω_{14}	ω_{15}
0				1											
1	1			1				1						1	
2	1							1						1	
3		1							1						1
4		1							1						1
5			1							1					
6			1							1					
7					1						1				
8					1						1				
9						1						1			
10						1						1			
11							1						1		
12							1						1		

.

Instead of a single individual we could stack them to reduce the length of each class, but still reset to the first generation upon crossing the $2\sqrt{d}^{\text{th}}$ generation.

The only tunable parameter for the Z-updates is $K_{\max}$. Our investigation indicates the length of the path to vary between $\sqrt{d}$ and $3\sqrt{d}$. Thus, setting $K_{\max} = 3\sqrt{d}$ would allow jumps of the appropriate length. These results are based on the one-type one-level case and experience with partially observed epidemics.

3.5.4.3 Unknown Number Of Classes

We now present two new updates for the multi-type multi-level model, though the first is only presented for completeness. We do not implement the case of an unknown number of classes. The second is a new type of Z-update to aid mixing of high dimensional paths.

So far we have always considered the number of columns in the path z to be fixed, i.e. the number of classes is fixed.

This is primarily of interest for partially observed multi-level epidemics. For the one-level case altering the number of unobserved infectives or susceptibles required updating a single column in the path z. Whereas if we introduce an unknown number of households, then we do not know how many classes there are (since each household defines its own class).

To avoid this issue, we may assume the unobserved population has households exactly in proportion to the observed component. Then the number of classes is known. However, this is only valid if we assume the total population size is known, i.e. we are considering an unknown number of infectives.

For the case of an unknown number of susceptibles, we may add or remove susceptibles from known classes, but we may also wish to add or remove households as a whole. This requires adding or removing a column from the path z.

To add a new column, we must propose a new data vector for the class to be added, $\theta_\omega(a, n, d)$, then add the d individuals to the current path in some manner. A sensible proposal for θ_ω would be from the distribution of observed classes.

Removing a class is simpler, since we can set $\theta = (0, 0, 0)$ and the column then has no contribution to the likelihood. It is more efficient to zero a column than remove it entirely, since when adding a new column we can reuse the removed class. This last point is for computational efficiency.

We do not implement the adding and removing of entire households, though the method outlined above would be relatively straightforward to program, there are issues on mixing and convergence that are unresolved.

3.5.4.4 Slip-update

If there is a large number of non-empty classes, with a relatively small final size, d_ω and the epidemic is locally driven, i.e. $\lambda^L >> \lambda^G$; then the path z will consist of many small clusters of local outbreaks.

Since each local outbreak will generally progress in consecutive generations, for a given class ω we would expect r_ω and s_ω to be close, relative to the size of d_ω. We defined r and s in Section 3.3.6.1, where $r_\omega = \min\{t : x_{\omega,t} > 0\}$ and $s_\omega = \max\{t : x_{\omega,t} > 0\}$.

For a locally driven epidemic, where each household outbreak will be initiated by a single outside infection, we may consider the household outbreak in isolation. This is commonly assumed to derive limiting results and for other theoretical work.

Thus we expect the length of each class to be in proportion to twice the square root of the final size, that is

$$s_\omega - r_\omega \approx 2\sqrt{d_\omega},$$

following the reasoning as for the seed path.

If the epidemic is locally driven, then proposing a long K-jump on an individual to a generation away from the others of its class will lead to a low acceptance probability. In particular, the class would then have two global infections into it at separate points.

Thus it will take a number of shorter K-jumps to shift all individuals of a class within the path. Hence, we introduce the slip-update to enable quick movement of an entire class.

First, we must determine if all individuals within a class can be moved. Define the function $J_s(z_\omega)$ to be one if the class ω can all be moved, zero otherwise. Let $\mathcal{J}_s = \sum_{\omega \in \mathbb{S}} J_s(z_\omega)$, that is the total number of slip-able classes. The slip indicator function is defined as,

$$J_s(z_\omega) = \begin{cases} 1 & \text{if } x_t - x_{\omega,t} > 0 \text{ for } 0 \leq t \leq \tau_{-\omega} \\ & \quad \text{and if } x_\tau - x_{\omega,\tau} = 0 \text{ for } \tau_{-\omega} < t \leq \tau \\ 0 & \text{otherwise,} \end{cases}$$

where $\tau_{-\omega}$ is the largest end generation excluding class ω. Namely,

$$\tau = \max\{s_\upsilon : \upsilon \in \mathbb{S}\}$$
$$\tau_{-\omega} = \max\{s_\upsilon : \upsilon \in \mathbb{S} \backslash \omega\},$$

The special attention needed for the final generation is such that a slip can reduce τ in the candidate path.

We then propose to move the class to a new starting generation at random in the path excluding the current generation so as not to propose a path identical to the current one, i.e. $r'_\omega \sim \mathrm{U}[1, \tau_{-\omega} + 1]$ not letting $r'_\omega = r_\omega$. To sample a new starting generation, choose uniformly from $[1, \tau]$ and if r_ω is chosen, set $r'_\omega = \tau_{-\omega} + 1$.

Hence, the proposal distribution is

$$q(z'|z) = \frac{1}{\mathcal{J}_s}\,\frac{1}{\tau_{-\omega}}.$$

We summarise the slip-update in Algorithm 3.8.

Algorithm 3.8: Slip-update within Z for multi-type multi-level model with fixed a

Calculate $J_s(z)$ and $\mathcal{J}_s$;
Sample $g \sim \text{Uni}[1, \mathcal{J}_s]$;
Determine the class to be slipped, ω corresponding to g;
Calculate r_ω, s_ω and $\tau_{-\omega}$;
Sample $h \sim \text{U}[1, \tau_{-\omega}]$;
if $h = r_\omega$ **then**
    $r'_\omega = \tau_{-\omega} + 1$
else
    $r'_\omega = h$
Construct z' by moving class ω to new starting generation;
Calculate acceptance probability α;
Draw $A \sim \text{U}(0, 1)$;
if $\alpha < A$ **then**
    reject z'
else
    accept z'

Example Of Slip-update Consider the path,

$$z = \left[\begin{array}{cccc} \omega_1 & \omega_2 & \omega_3 & x_t \\ \hline 1 & 0 & 0 & 1 \\ \hline 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 2 \\ 0 & 1 & 1 & 2 \\ 1 & 0 & 0 & 1 \end{array}\right] \quad \text{removing second column} \quad z = \left[\begin{array}{ccc|c} \omega_1 & \omega_2 & \omega_3 & x_t \\ \hline 1 & 0 & 0 & 1 \\ \hline 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \end{array}\right].$$

Only the second column is free to slip, since if we remove the second column the path remains valid.

Then there are four valid candidate paths,

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

All the candidate paths have the ω_2 as a slip-able class, thus the slip-update is reversible.

We use this simple example to justify the slip-update, clearly a slip-update is equivalent to a sequence of K-jumps applied to the same class. However, each K-jump is performed and accepted independently in sequence, hence the intermediate paths may have a very low likelihood. Whereas, a slip performs multiple K-jumps in parallel, which will lead to higher acceptance probabilities.

The slip-update is a method to increase mixing within the path z, as for the d-update, it is best to have a number of iterations between each slip-update.

The slip-update performs well for one-type multi-level models, but only when there are a large number of small classes in a locally driven epidemic. However if there are multiple types, the columns are slipped separately, thus we will separate local individuals and the acceptance probability will decrease. Instead we can slip sets of columns matching a given level set, say a household.

3.6 Case Study

In this section we apply the general framework developed in Section 3.5 to two specific data sets. The first is a one-type two-level case and the second is a two-type two-level case.

Both data sets contain a moderate number of individuals, combined with the additional structure within the data means we test the ability of the general framework to handle such structured data. For the moment, we still restrict attention to a fixed infectious period which is identical for all individuals in the population.

The two data sets were presented in Longini et al. (1988), who consider an approach using independent households to investigate the community level infection. We compare our results to those of Demiris and O'Neill (2005a), though they use a gamma infectious period for the one-type case with shape parameter 2 and rate 1/2.05, we shall make a comparison with a fixed infectious period of 4.1 days. For the two-type case Demiris and O'Neill (2005a) use a fixed period of 4.1 days, thus direct comparison is possible.

3.6.1 Data

The first data set, $\psi^{(1)}$, is presented in Table 3.9, consisting of individuals of one type that are grouped into households. The table lists the number of households that contain N_ω individuals of which D_ω were infective during the course of the epidemic.

Using the notation of Section 3.5.1 we have, $(W, V) = (1, 2)$ with $H_1 = 1$, $L_1 = 1$ and $L_2 = 287$. That is there are 287 groups, which in this case represent households. The data summarise an outbreak of influenza A(H3N2) in Tecumseh, Michigan in the period 1980–1981, see Haber et al. (1988) for details, the data are summarised in Table 3.9.

		Number in House, N_ω							
		1	2	3	4	5	6	7	
Number Infected, D_ω	0	44	62	47	38	9	3	2	
	1	10	13	8	11	5	3		
	2		9	2	7	3			
	3			3	5	1			
	4				1	0			
	5					1			
	6								
	7								
	Total	54	84	60	62	19	6	2	287

Table 3.9: Data for one-type two-level case, $\psi^{(1)}$, an outbreak of influenza A(H3N2) in Tecumseh, Michigan in 1980–1981. Counts for the number of households matching a given configuration, $\psi_\omega = (N_\omega, D_\omega)$, are given. Reproduced from Demiris (2004).

We can convert into the form required for our path, i.e. $\theta = (a, n, d)$, since we have the final numbers infected in the population on a per household basis and shall assume a single initial infective. We have reduced Table 3.9 to counts of identical household configurations for clarity, instead of listing each household separately.

The second data set, $\psi^{(2)}$, contains information about two types of individual, depending on the antibody titre level, which is termed low (type 1) or high (type 2). We can readily apply the method to the data, since we have the number of individuals and how many of them became infected on a per household basis. The complete data are presented in Table 3.10, reproduced from Longini et al. (1988) which contains further details on the collection and typing of individuals.

There is a single characteristic with two types, i.e. $H_1 = 2$, corresponding to low (1) and high (2) titre levels. Each individual belongs to one of the 567 groups (households). In 13 of the household configurations (†) we do not know the final outcome, also for a further 9 (‡) configurations are unreported. Hence we consider only the 545 households

for which we have data, i.e. $L_2 = 545$.

The two-type data set is actually from two separate periods, 1965–1971 and 1976–1981, though this is not important in illustrating the generation method. More importantly, the study is a random sample of all households, accounting for 10% of the total population. Hence, for a full analysis we should include an unobserved component for the remaining 90%. However, we must decide how to impute the unobserved households, either randomly or as nine exact copies of the observed component. For our results we assume $\eta = 1$, similar to the analysis by Demiris and O'Neill (2005a), this simplifies the analysis at the cost of not fully modelling the data and underestimating the uncertainty in our parameter estimates.

3.6.2 Results

For $\psi^{(1)}$ we determine the infection matrix using two sub-parameters, i.e. $\Lambda_{\psi^{(1)}}(\lambda^L, \lambda^G)$, using the additive form defined in Section 3.5.2.

For the two-type data, we apply both the Global-Local-Susceptibility and Global-Susceptibility additive models as defined in Section 3.5.2. Hence there are four and six sub-parameters respectively,

$$\Lambda_{\psi^{(2)}}(\lambda_1^L, \lambda_2^L, \lambda_1^G, \lambda_2^G) \qquad \text{and} \qquad \Lambda_{\psi^{(2)}}(\lambda_1^L, \lambda_2^L, \lambda_3^L, \lambda_4^L, \lambda_1^G, \lambda_2^G).$$

Since the two types are based on low or high antibody titre levels, it seems reasonable to use the susceptibility model.

We implemented the algorithm as outlined in Section 3.5.4, using computational methods we shall discuss in Section 3.7. Specifically, for $\psi^{(1)}$ and $\psi^{(2)}$ we perform an update

N_1	N_2	D_1	D_2	Observed
1	0	0	0	45
		1	0	18
0	1	0	0	65
		0	1	5
2	0	0	0	52
		1	0	11
		2	0	8
1	1	0	0	52
		0	1	2
		1	0	8
		1	1	4
0	2	0	0	45
		0	1	6
		0	2	1
3	0	0	0	17
		1	0	4
		2	0	3
		3	0	5
2	1	0	0	28
		0	1	1
		1	0	6
		1	1	0
		2	0	2
		2	1	2
1	2	0	0	16
		0	1	6
		0	2	0
		1	0	2
		1	1	1
		1	2	0
0	3	0	0	11
		0	1	4
		0	2	0
		0	3	0
4	0	0	0	16
		1	0	4
		2	0	6
		3	0	0
		4	0	2
3	1	0	0	13
		0	1	0
		1	0	6
		1	1	1
		2	0	1
		2	1	0
		3	0	5
		3	1	0
2	2	0	0	11
		0	1	0
		0	2	1
		1	0	1
		1	1	3
		1	2	1
		2	0	3
		2	1	0
		2	2	0
1	3	0	0	10
		0	1	5
		0	2	0
		0	3	0
		1	0	2
		1	1	1
		1	2	2
		1	3	0
0	4	0	0	10
		0	1	2
		0	2	0
		0	3	0
		0	4	0
5	0	0	0	3
		other		3 †
4	1	0	0	2
		other		4 †
2	3	0	0	4
		other		6 †
3	2	other		4 ‡
1	4	other		2 ‡
0	5	other		3 ‡

Table 3.10: Data for two-type two-level case, $\psi^{(2)}$, combined outbreaks of influenza A(H3N2) in Tecumseh, Michigan in 1965–1971 and 1976–1981. Counts for the number of households matching a given configuration, $\psi = (N_1, N_2, D_1, D_2)$, are given. Reproduced from Longini et al. (1988)

	Edge Method		Generation Method	
	λ^L	λ^G	λ^L	λ^G
Mean	0.050	0.193	0.048	0.189
Median	0.049	0.192	0.047	0.188
Standard deviation	0.010	0.025	0.008	0.021
95% highest posterior density region	(0.032,0.072)	(0.15,0.25)	(0.032,0.065)	(0.15,0.23)

Table 3.11: Comparison of results for one-type two-level data set, $\psi^{(1)}$, between Demiris and O'Neill (2005a) and generation method. The edge method assumes a gamma infectious period and the generation a fixed infectious period, both with mean $\mathrm{E}[I] = \iota = 4.1$ days.

of the Λ sub-parameters every iteration and two K-jump updates every iteration.

We compare our estimates for the sub-parameters in each model with those of Demiris and O'Neill (2005a), Tables 3.11 and Table 3.12 compare the one and two type cases respectively. Note, for the one-type case the comparison is not direct as a different infectious period distribution is used.

Given the posterior means and standard deviations in Table 3.12, it is interesting to consider the relationship between the four sub-parameters. We update the parameters all at once using a multivariate normal with a defined covariance matrix Σ, for the current case set to a diagonal matrix, implying independent components. If two parameters are highly dependent then the MCMC chain may mix poorly.

Figure 3.10 shows the pairwise plots for each parameter, using every 100^{th} iteration after the burn in period. The corresponding correlations are in the diagonally opposite position.

As expected, the local and global rates are negatively correlated within the same type. Since if we increase the local susceptibility we must balance the corresponding global

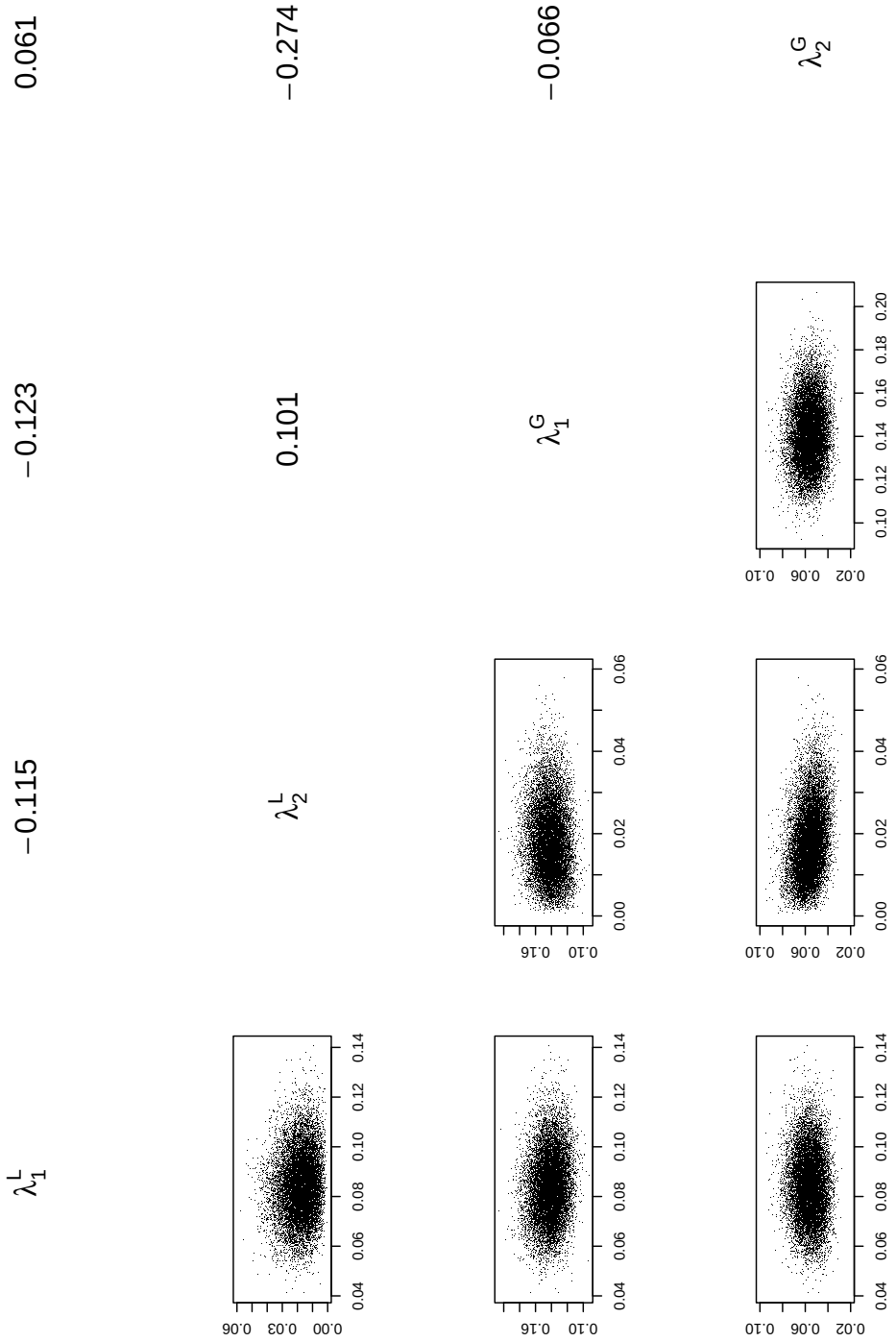


Figure 3.10: Pairwise plots of infection rate parameters using every 100^{th} iteration. The correlations are given in diagonally opposite position

		Global-Local-Susceptibility Model
Edge	Λ^L	$\begin{pmatrix} 0.0817 & (0.013) & 0.0124 & (0.0073) \\ 0.0817 & (0.013) & 0.0124 & (0.0073) \end{pmatrix}$
	Λ^G	$\begin{pmatrix} 0.143 & (0.014) & 0.0576 & (0.0090) \\ 0.143 & (0.014) & 0.0576 & (0.0090) \end{pmatrix}$
Generation	Λ^L	$\begin{pmatrix} 0.0842 & (0.013) & 0.0182 & (0.0084) \\ 0.0842 & (0.013) & 0.0182 & (0.0084) \end{pmatrix}$
	Λ^G	$\begin{pmatrix} 0.142 & (0.014) & 0.0564 & (0.0090) \\ 0.142 & (0.014) & 0.0564 & (0.0090) \end{pmatrix}$

Table 3.12: Comparison of results for two-type two-level data set, $\psi^{(2)}$, using the Global-Local-Susceptibility model and a fixed infectious period of 4.1 days for all individuals. The edge results are reproduced from Demiris and O'Neill (2005a)

susceptibility, the balance is due to the conditioned final outcome of the path, to match the observed data.

Given the correlation between the parameters, we investigate the mixing of the MCMC algorithm using the Auto Correlation Function (ACF). Figure 3.11 shows the ACF for the four sub-parameters of Λ, the two local and two global rates. We can partially explain the large auto-correlation of λ_2^L if we examine the two-type data, $\psi^{(2)}$. There are 18 households that consist of only type 2 individuals where a non-zero number of them become infected, out of 545 households in total. Thus the amount of information directly related to the local susceptibility of type 2 individuals is limited. We must also consider the high dependence between the Λ parameters, especially between local and global rates, which will reduce the information about λ_2^L in households containing both types of individual.

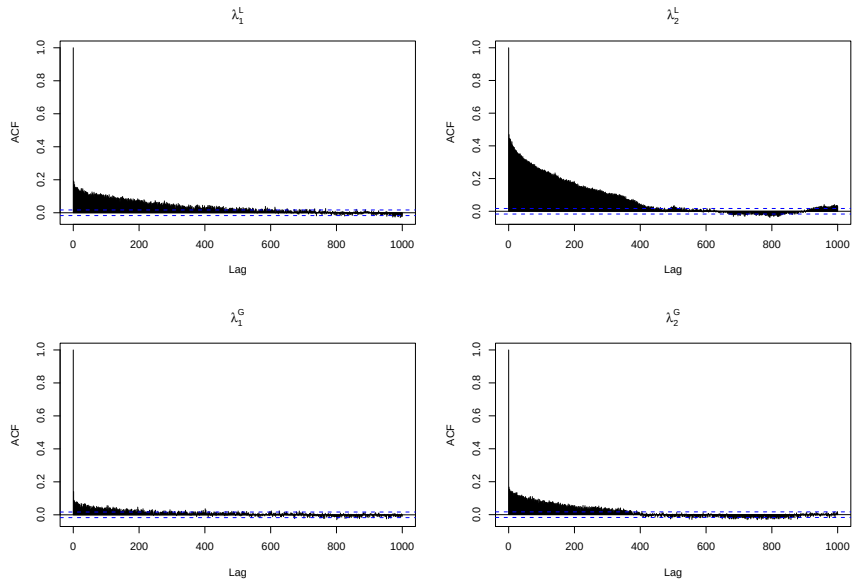


Figure 3.11: ACF Plots for Λ sub-parameters under the GLS Model from the generation method, using every 50^{th} iteration

3.7 A Note On Computation And Parallel Computing

Markov Chain Monte Carlo methods using Metropolis-Hastings update steps require the calculation of the acceptance probability, consisting of the posterior and proposal distribution, at each step. For the method to be viable we need to calculate the acceptance probability accurately and efficiently.

In Section 3.7.1 we present several optimisations to the acceptance probability calculations, each reduces the number of computer operations, increasing the speed of each iteration. These optimisations may have a secondary effect on accuracy, removing unnecessary error. However, eventually the limits of standard computer accuracy will be met and the calculations will become invalid. In Section 3.7.2 we discuss the issue of accuracy. Finally, we consider parallel computing techniques in Section 3.7.3. As the data sets become larger, the time to generate a sufficient number of iterations becomes prohibitive. Parallel computing can help to tackle such issues.

Other practical concerns exist, though we shall not consider them further. For example, the amount of memory available to hold a large data set, not only must we store θ but also the whole path z (as well as temporary candidate versions, θ' and z'). For the actual final size data, the memory required is very small, but the imputed path may be a very high dimensional object.

The amount of output generated by an MCMC algorithm can be unmanageable. We have used the moments of the path z as a summary, reducing the amount of data output for each generation to four values. If instead we recorded the actual path z at each iteration we would generate a lot more output. For example, assume each iteration produces 1 kilobyte of output (in terms of computer file sizes), then running 10^6 iterations will produce log files of approximately 1 gigabyte in size.

Optimisation is a balance between efficiency and complexity. Reducing an iteration from 0.2 seconds to 0.15 seconds will reduce the time for 10^6 iterations from 2 days and 8 hours to 1 day and 18 hours, a reduction of fourteen hours run-time. Whether such a reduction is worthwile depends on the effort required to optimise the 0.05 seconds per iteration.

3.7.1 Computational Efficiency

Naive evaluation of the likelihood will in general be an inefficient operation. For each update algorithm, the acceptance probability must be analysed to find cancellations and unnecessary terms that can be optimised.

For example, when performing an update of Λ the likelihood must be evaluated, as in Equation (3.24). However, the binomial coefficients depend on θ which is unchanged between the current and candidate Λ'. The coefficients will be identical in the denominator and numerator, hence we need not waste computer operations in calculating them.

Likewise, a K-jump alters a single class in z from the origin generation to the target generation, effecting an additional generation either side. Assume $t_O < t_{O+\delta k}$, i.e. a forward jump, then the candidate path z' is identical to the current path z up until generation $t_O - 1$ and from generation $t_{O+\delta k} + 1$ to τ. Since the likelihood of a path z is the product of independent step probabilities, the identical steps will cancel.

In fact for the K-jump we can go further, since only the binomial coefficients associated with the updated class will differ we can ignore the binomial coefficients of all other classes.

Using temporary variables to hold intermediate results can greatly increase efficiency. For example, the two exponential terms in Equation (3.24) are the same, we need only calculate this once and reuse the value.

For Λ-updates, if we store the contribution of the current state to the acceptance probability, i.e. the denominator, should the proposed candidate be rejected and the path not alter in the Z-updates between Λ-updates, we may reuse the denominator value. Since an acceptance rate of 0.234 is optimal (see Section 1.3.2.4), if the hyperparameters are tuned to achieve this acceptance rate then the probability of the Λ-update and two subsequent K-updates being rejected is $(1 - 0.234)^3 = 0.45$ or 45%. This saving on forty-five percent of 10^6 iteration updates could have a significant effect if the cost of computing the likelihood is high.

3.7.2 Computational Accuracy And GNU MPFR

Accuracy is dependent upon practical issues, theoretically any mathematical expression given can be numerically evaluated, however physical limits may cause inaccuracy.

Consider the binomial coefficient,

$$\binom{a}{b} = \frac{a!}{(a-b)!\, b!},$$

a common term in the likelihood of a path z. The formula is valid for any $a \in \mathbb{Z}_+$ and $0 \leq b \leq a$, giving an integer number of ways to choose b objects from among a objects.

If we consider a naive function to calculate the binomial coefficient, we define

$$f(x) = x! = (x)(x-1)(x-2)\cdots(2)(1),$$

then the binomial coefficient is calculated as

$$\binom{a}{b} = \frac{f(a)}{f(a-b)f(b)}.$$

However, a computer stores integers in fixed amounts of memory, that is an integer is represented as a binary number of fixed length. For a 32-bit computer the standard unsigned integer type is 32 bits, i.e. it can store an integer up to $2^{32} - 1$. Thus, if any of the factorials exceed this limit we have an error called buffer overflow, i.e. the computer is attempting to store a number larger than the maximum it can represent.

The naive approach fails for $a \approx 20$, since the factorial of 20 causes a buffer overflow in the numerator, despite the fact that $\binom{20}{b}$ is easily represented by a standard integer for all values of b.

The problem is the intermediate calculations, principally the numerator, causing a buffer overflow. A solution is to express the binomial coefficient in an alternate way,

$$\binom{a}{b} = \frac{a!}{(a-b)!\,b!} = \frac{(a)(a-1)\cdots(a-(b-1))}{(b)(b-1)\cdots(1)} = \prod_{i=1}^{b} \frac{a+1-i}{i}.$$

Thus each intermediate value is the ratio of two smaller numbers. Obviously, the ratio is not in general an integer until the entire product is taken. Using this ratio will stop a buffer overflow of the standard float type (the C programming language stores non-integers as floating point numbers).

A product propagates any inaccuracy in the terms, since we take a product of rationals the computer needs to store these internally as base two (binary) numbers. Again, a computer has a fixed size to store any number, meaning rounding error may occur.

Instead, we can take natural logarithms of the product to give a sum of floating point

values, taking the exponential of the sum to give the binomial coefficient,

$$\binom{a}{b} = \prod_{i=1}^{b} \frac{a+1-i}{i} = \exp\left(\sum_{i=1}^{b} \log(a+1-i) - \log(i)\right).$$

Note that the final answer may not be an integer due to rounding and representation errors. However, using this form to calculate the binomial coefficient we can attain accurate results for $a \approx 90$.

As an aside on efficiency, for the sum of logs expression there are b terms to calculate. The binomial coefficient obeys the identity,

$$\binom{a}{b} = \binom{a}{a-b}.$$

Hence, if $2b > a$ then calculating $\binom{a}{a-b}$ instead will be quicker.

For larger coefficients, there are basic types of variable that are stored using more bytes, the so-called double is equivalent to the size of two floats, allowing much larger numbers to be stored. Eventually, there is an upper bound using standard data types, though this will vary depending upon the programming language used and the architecture (e.g. x86 32-bit, x86 64-bit, PowerPC, etc.) of the machine used.

The GNU MP (Multi-Precision) library is an extension to the C programming language allowing integers (and rationals) of arbitrary size to be stored. It uses a custom variable type that the user can specify the number of bytes of storage. The GNU MPFR (Multi-Precision Float) is based on GNU MP, allowing floating point numbers of arbitrary precision. Hence, we may define a variable to use sufficient memory to calculate any binomial coefficient desired.

For acceptance probabilities, we have the opposite problem, attempting to store num-

bers that are too small. This is called buffer underflow, a problem briefly mention in Section 2.4.7.3 on the Forward-Backward Algorithm (FBA).

This was also a problem when calculating the probability of a certain connectedness of a digraph using a brute force path search in Section 2.4.7.2. That is, some paths resulting in the correct connectedness are highly unlikely, contributing a small but significant amount to the total probability when summing over all such unlikely paths.

Increasing the precision dramatically affects computation time, so it is always more efficient to rewrite an expression cancelling terms and removing products before using GNU MPFR.

For example, using a fixed infectious period we have the likelihood of a path z as the product of independent step probabilities. This is very numerically unstable, hence if we consider the logarithm of the likelihood the product becomes a sum. Additionally, each step is is fact a product of avoidance and infection terms which itself will become a sum of logs. For the fixed infectious period, standard data types are sufficient to perform MCMC. However, for alternate infectious periods that are integrated out in large populations we must resort to using higher precision arithmetic.

3.7.3 Parallel Computing Using GNU OpenMP

An MCMC algorithm must be run for a large enough number of iterations to first converge and then give a sufficient sample to estimate the posterior density.

Using efficient methods of calculation and removing unnecessary computations is within the ability of casual programmers, requiring the ability to manipulate the mathematical expressions.

Code optimisation is another skill entirely. In the C programming language for example, the architecture will determine if post or pre incrementing variables is quicker in loops. Such low level optimisation usually performed by the compiler and is beyond the scope of this thesis.

There are two further options for decreasing the run-time of our MCMC algorithm. Using a faster processor, i.e. performing more operations per second, will generally reduce the run-time. Though other factors play a part. The second option is to use parallel computing, that is perform disjoint operations in parallel to reduce the overall run-time.

Recently multi-core processors have become more readily available, these single processor chips contain multiple cores that can run their own processes and share access to the common computer memory. To use this functionality we use GNU OpenMP, a library that allows simple modification of existing code to a multi-core environment. OpenMP (Open Multi-Processor) uses a shared memory model for the program, where all processors share access to the same memory space and are thus all located on the same physical machine. For details of OpenMP in the C programming language see for example Chapman et al. (2007).

A separate technology called MPI (Message Passing Interface) is used for running a program across multiple machines that are connected via a data network. Thus each machine, called a node, is a separate unit, with no direct access to the memory of another node in the cluster (the collection of all nodes). To facilitate parallel computation, each node must exchange data over the network linking the cluster, called message passing.

See Quinn (2004) for a discussion of the two approaches and details of their implementation, possibly combining both technologies in the same application. When to use either

technology is dependent upon the specific program, data structures and calculations required. As a guide, MPI is more suited to lengthy calculations on small fragments of the entire data. Whereas OpenMP is more suited to many short calculations, requiring repeated reference to the entire data structure.

As an additional consideration, to use MPI the program should be designed for that purpose from the start. The mechanism of message passing requires careful planning of functions and access to specific data and variables. Conversely, it is far simpler to convert an existing program to use OpenMP. The shared memory aspect of OpenMP can cause so-called race conditions, where two processors attempt to read and write the same variable giving unpredictable results; care must be taken to avoid such errors.

For our MCMC algorithm, OpenMP is the more appropriate technology. The acceptance probability is a function of two large data structures, the current and candidate paths, thus it is reasonable to use the shared memory model with each processor calculating a disjoint part of the likelihood. For an MPI approach, we would need to pass the entire path to all nodes. As discussed, the path is a sequence of independent step probabilities each of which can be calculated in parallel and then combined. This is the primary used of parallel computing in our algorithm, we omit specific details of implementing the parallel code.

OpenMP is relatively easy to implement, requiring minor alterations in existing code. However, this does not always give the best improvement. Figure 3.12 shows the run-time ratios (relative to a single processor) for the two-type two-level and one-type two-level models. Each test used a sample run of 10^4 iterations and limited the number of processors as indicated on the x-axis. The theoretical ratio is also plotted, it is clear that the actual speedup is far below that in theory.

There are many reasons for the less than optimal speedup, many are technical in nature

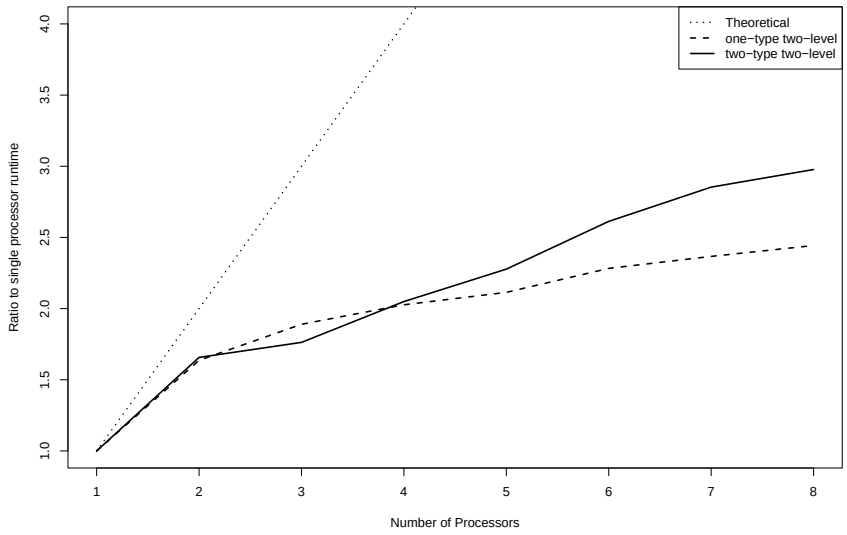


Figure 3.12: Run-Time Ratios for the one-type two-level and two-type two-level algorithms using OpenMP running on a twin quad core machine using various numbers of processors.

and are to do with computer architectures including: memory access, processor cache sizes and number of cores on each chip (in this case we use two quad-core processors). Two other issues can also be easily identified, the algorithm logic and job scheduling. OpenMP is designed to be added to existing programs, however it is sometimes better to rewrite the entire code to accommodate more possibilities for parallel sections. Scheduling is how parallel sections of code are divided among the available processors, there are several schemes built into OpenMP and selecting which to use for the best gain in efficiency is not trivial. The Computer Science literature contains more on this and related topics, see for example Ayguad et al. (2003) on OpenMP scheduling. These issues are beyond the scope of this thesis, thus we must accept sub-optimal improvements.

Chapter 4

Equine Influenza

4.1 Introduction And Motivation

In Chapter 3 we developed a general framework to analyse multi-type multi-level final size data given that all individuals have an independent and identically distributed (i.i.d.) infectious period, specifically a constant period, i.e. $I = c$, and a specified infection matrix, Λ, in terms of a set of sub-parameters that we wish to make inference for using Markov Chain Monte Carlo (MCMC). A variety of MCMC updates were considered, including the ability to model partially observed epidemics. In this chapter we extend the framework of Chapter 3 and use this to analyse final size data for an outbreak of Equine Influenza (H3N8) at Newmarket in 2003.

In Section 4.2 we shall adapt our generation method to accommodate an arbitrary infectious period for each class of individual, subject to conditions upon its moment generating function. The population may be partitioned into subsets, which we define as classes in Section 3.5, based on group membership, e.g. households or place of work, and a set of covariates that describe each individual, e.g. gender, age or observed.

The study of equine influenza within racehorse populations has received attention in the veterinary literature, though applied examples in the epidemics literature are lacking. We follow the analysis of Baguelin et al. (2009), who apply standard results from epidemic modelling and use Approximate Bayesian Computation (ABC) to estimate

parameters in a standard one-type two-level mixing model. We shall present the data and model in Sections 4.4 and 4.5 respectively, then compare the results of Baguelin et al. (2009) with our MCMC algorithm in Section 4.6.

The estimates obtained using the two techniques, ABC and MCMC, are noticeably different, more so than can be explained by the variance of the estimators. To investigate this further we perform a simulation study, which allows us to demonstrate the behaviour of the epidemic given the differing parameter estimates. Ultimately, the estimates differ because they are based on different likelihoods and assumptions.

4.2 General Infectious Period

Before considering the Newmarket outbreak data, we expand upon the general model for fixed infectious periods described in Section 3.5.1 and incorporate general infectious period distributions into the likelihood of Section 3.5.3. Further, we allow each class to have its own specified infectious period distribution.

Using the notation for the general model in Section 3.5, we now define the infectious period distribution of the class ω to be I_ω, for $\omega \in \mathbb{S}$. As before, the population is closed without migration between groups, that is the class of an individual i, denoted $\mathcal{S}(i) = \omega$, is fixed for the duration of the epidemic. Then an individual i is a member of the class ω and has an infectious period distributed according to I_ω which we shall denote I^i_ω, i.e. $I^i_\omega \stackrel{d}{=} I_\omega$, and a realisation of this random variable shall be denoted ζ^i. It is not necessary to include a class subscript on ζ^i since the individual's label i gives this implicitly, though it may be included for clarity. Then, the vector of infectious

periods is

$$I = \left(I_{\mathcal{S}(1)}, I_{\mathcal{S}(2)}, \ldots, I_{\mathcal{S}(N)}\right).$$

Recall, we can reduce the vector to only those individuals that are ultimately infected, i.e. a length D vector. For a path z, let x_t and $x_{\omega,t}$ denote the total number of individuals and the number of individuals of class ω in generation t respectively, i.e. $\sum_{\omega \in \mathbb{S}} x_{\omega,t} = x_t$. Let I_t be the vector of infectious periods of individuals in generation t. For clarity, we relabel individuals within I_t using the index j, for $1 \leq j \leq x_t$, such that j corresponds to the label of an individual i in the population. In this case it is necessary to indicate the class of individual j in generation t, specifically ω_j. Then we define $I^j_{\omega_j}$ and $\zeta^j_{\omega_j}$ as the infectious period and a realisation for individual j in generation t. Finally, we denote the N length vectors of independent infectious period distributions as I and a realisation as ζ, similarly we define I_t and ζ_t to be the X_t length vectors of the infectious period distributions of individuals in generation t.

This construction allows for a large variety of models using different infectious period distributions for different classes of individuals. We may include parameters from the infectious period distributions into our MCMC algorithm, though these parameters may be unidentifiable. This is the case for a fixed infectious period in the one-type one-level model, where the length $I = c$ and rate λ always appear as a product in the likelihood, hence we cannot identify both in the model and must specify the length of the infectious period prior to making inference. Given the issues of identifiability we shall only consider fully specified infectious period distributions. Note, the issue of identifiability is non-trivial for complicated models with highly correlated parameters, being dependent upon the data and model.

We wish to calculate the likelihood of the path z. Then, as before we have

$$\begin{aligned}
&\mathrm{P}[Z = z | I = \zeta] \\
=&\mathrm{P}[Z = (z_0, z_1, \ldots, z_\tau, z_{\tau+1}) | I = (\zeta^1, \ldots, \zeta^N)] \\
=&\prod_{t=0}^{\tau} \mathrm{P}[Z_{t+1} = z_{t+1} | Z_t = z_t, I_t = \zeta_t] \\
=&\prod_{t=0}^{\tau} \mathrm{P}[Z_{t+1} = z_{t+1} | Z_t = z_t, I_t = (\zeta^1_{\omega_1}, \ldots, \zeta^{x_t}_{\omega_{x_t}})] \\
=&\prod_{t=0}^{\tau} \mathrm{P}[Z_{t+1} = z_{t+1} | Z_t = z_t, I_t = (\zeta^1_{\omega_1}, \ldots, \zeta^{x_{\omega_1,t}}_{\omega_1}, \zeta^1_{\omega_2}, \ldots, \zeta^{x_{\omega_2,t}}_{\omega_2}, \cdots, \zeta^1_{\omega_S}, \ldots, \zeta^{x_{\omega_S,t}}_{\omega_S})].
\end{aligned} \tag{4.1}$$

Here we have listed individuals in generation t by their class. In general not all S classes will appear in each generation, this notation is convenient in the following expression of the general step probability.

As discussed in Section 3.2.4.3, there are two approaches to non-constant infectious periods. We may include the infectious periods as new parameters in the MCMC algorithm adding more imputed data, this would require the individuals to be labelled within each generation such that we can associated a specific infectious period with each individual. Then path updates moving individuals between generations must take the correct infectious period associated with that individual to the new generation. This adds extra book keeping to the algorithm and increases the size of the imputed state space, which must be explored by the MCMC algorithm to insure adequate mixing over the parameters of interest, i.e. the infectious rates. Alternatively, we may integrate the infectious periods out of the likelihood. This creates a more complicated and computationally costly likelihood, but has the benefit of removing the need to impute parameters that are of no interest and may cause problems with mixing.

Using the notation of Equation (4.1), following the derivation and form of Equation (3.24), the probability of an individual of class ω_1 avoiding infection from generation t is

$$\begin{aligned}&\exp\Big(-\lambda_{\omega_1\omega_1}(\zeta_{\omega_1}^1+\cdots+\zeta_{\omega_1}^{x_{\omega_1,t}})-\lambda_{\omega_2\omega_1}(\zeta_{\omega_2}^1+\cdots+\zeta_{\omega_2}^{x_{\omega_2,t}})\\&\qquad-\cdots-\lambda_{\omega_S\omega_1}(\zeta_{\omega_S}^1+\cdots+\zeta_{\omega_S}^{x_{\omega_S,t}})\Big)\\=&\exp\Big(-\sum_{\omega_i\in\mathbb{S}}\lambda_{\omega_i\omega_1}(\zeta_{\omega_i}^1+\cdots+\zeta_{\omega_i}^{x_{\omega_i,t}})\Big),\end{aligned}$$

and probability of infection is one minus the probability of avoidance. Note we need the total infectious time of individuals of each class $\omega_i \in \mathbb{S}$ in generation t, i.e. the infectious pressure.

Thus the step probability given the infectious period vector is,

$$\begin{aligned}&\mathrm{P}[Z_{t+1}=z_{t+1}|Z_t=z_t,I_t=\zeta_t]=\\&\qquad\prod_{\omega_j\in\mathbb{S}}\binom{N_{\omega_j}-y_{\omega_j,t}}{x_{\omega_j,t+1}}\exp\Big(-\sum_{\omega_i\in\mathbb{S}}\lambda_{\omega_i\omega_j}(\zeta_{\omega_i}^1+\cdots+\zeta_{\omega_i}^{x_{\omega_i,t}})\Big)^{N_{\omega_j}-y_{\omega_j,t+1}}\\&\qquad\qquad\Big(1-\exp\Big(-\sum_{\omega_i\in\mathbb{S}}\lambda_{\omega_i\omega_j}(\zeta_{\omega_i}^1+\cdots+\zeta_{\omega_i}^{x_{\omega_i,t}})\Big)\Big)^{x_{\omega_j,t+1}}.\end{aligned}\tag{4.2}$$

Equation (4.2) is a product over each class of individual. A class contributes three terms to the product: a binomial coefficient for the number of ways to select the next generation; the probability of avoiding those not chosen and the probability of infecting the next generation.

If we include the infectious periods as additional parameters in the MCMC algorithm we may use Equation (4.2) to calculate the likelihood. Alternatively, if we wish to integrate out the infectious period parameters, we must take the expectation with

respect to I. Using the following relation,

$$(1-x)^n = \sum_{k=0}^{n} (-1)^{n-k} \binom{n}{k} x^{n-k},$$

we can rewrite Equation (4.2) as,

$$\begin{aligned}
&\mathrm{P}[Z_{t+1} = z_{t+1} | Z_t = z_t, I_t = \zeta_t] = \\
&\quad \prod_{\omega_i \in \mathbb{S}} \left\{ \binom{N_{\omega_i} - y_{\omega_i,t}}{x_{\omega_i,t+1}} \exp\Big(- \sum_{\omega_j \in \mathbb{S}} \lambda_{\omega_j \omega_i} (\zeta_{\omega_j}^1 + \cdots + \zeta_{\omega_j}^{x_{\omega_j,t}}) \Big)^{N_{\omega_i} - y_{\omega_i,t+1}} \right. \\
&\quad \left. \sum_{k=0}^{x_{\omega_i,t+1}} (-1)^{x_{\omega_i,t+1}-k} \binom{x_{\omega_i,t+1}}{k} \exp\Big(- \sum_{\omega_j \in \mathbb{S}} \lambda_{\omega_j \omega_i} (\zeta_{\omega_j}^1 + \cdots + \zeta_{\omega_j}^{x_{\omega_j,t}}) \Big)^{x_{\omega_i,t+1}-k} \right\}.
\end{aligned}$$

Then combining the exponential terms and using the expansion,

$$\prod_{n=1}^{N} \left(\sum_{k=0}^{\delta_n} A_{n,k} \right) = \sum_{k_1=0}^{\delta_1} \cdots \sum_{k_N=0}^{\delta_N} A_{1,k_1} \dots A_{N,k_N},$$

we obtain the expression,

$$\begin{aligned}
\mathrm{P}[Z_{t+1} = z_{t+1} | Z_t = z_t, I_t = \zeta_t] = &\left\{ \prod_{\omega_i \in \mathbb{S}} \binom{N_{\omega_i} - y_{\omega_i,t}}{x_{\omega_i,t+1}} \right\} \\
&\sum_{k_{\omega_1}=0}^{x_{\omega_1,t+1}} \cdots \sum_{k_{\omega_S}=0}^{x_{\omega_S,t+1}} \left[\prod_{\omega_i \in \mathbb{S}} (-1)^{x_{\omega_i,t+1}-k_{\omega_i}} \binom{x_{\omega_i,t+1}}{k_{\omega_i}} \right. \\
&\left. \exp\Big(- (N_{\omega_i} - k_{\omega_i} - y_{\omega_i,t}) \sum_{\omega_j \in \mathbb{S}} \lambda_{\omega_j \omega_i} (\zeta_{\omega_j}^1 + \cdots + \zeta_{\omega_j}^{x_{\omega_j,t}}) \Big) \right].
\end{aligned}$$

Rearranging we obtain (note by definition $y_{\omega,t+1} = y_{\omega_t} + x_{\omega,t+1}$ as used to combine the

exponential terms),

$$\mathrm{P}[Z_{t+1} = z_{t+1} | Z_t = z_t, I_t = \zeta_t] = \left\{ \prod_{\omega_i \in \mathbb{S}} \binom{N_{\omega_i} - y_{\omega_i,t}}{x_{\omega_i,t+1}} \right\} \times$$
$$\sum_{k_{\omega_1}=0}^{x_{\omega_1,t+1}} \cdots \sum_{k_{\omega_S}=0}^{x_{\omega_S,t+1}} \left[\left(\prod_{\omega_i \in \mathbb{S}} (-1)^{x_{\omega_i,t+1}-k_{\omega_i}} \binom{x_{\omega_i,t+1}}{k_{\omega_i}} \right) \right.$$
$$\left. \left(\prod_{\omega_i \in \mathbb{S}} \prod_{\omega_j \in \mathbb{S}} \exp\Big(-(N_{\omega_i} - k_{\omega_i} - y_{\omega_i,t}) \lambda_{\omega_j \omega_i} (\zeta_{\omega_j}^1 + \cdots + \zeta_{\omega_j}^{x_{\omega_j,t}}) \Big) \right) \right],$$

then altering the order of the products gives,

$$\mathrm{P}[Z_{t+1} = z_{t+1} | Z_t = z_t, I_t = \zeta_t] = \left\{ \prod_{\omega_i \in \mathbb{S}} \binom{N_{\omega_i} - y_{\omega_i,t}}{x_{\omega_i,t+1}} \right\} \times$$
$$\sum_{k_{\omega_1}=0}^{x_{\omega_1,t+1}} \cdots \sum_{k_{\omega_S}=0}^{x_{\omega_S,t+1}} \left[\left(\prod_{\omega_i \in \mathbb{S}} (-1)^{x_{\omega_i,t+1}-k_{\omega_i}} \binom{x_{\omega_i,t+1}}{k_{\omega_i}} \right) \right.$$
$$\left. \left(\prod_{\omega_j \in \mathbb{S}} \exp\Big(-(\zeta_{\omega_j}^1 + \cdots + \zeta_{\omega_j}^{x_{\omega_j,t}}) \sum_{\omega_i \in \mathbb{S}} (N_{\omega_i} - k_{\omega_i} - y_{\omega_i,t}) \lambda_{\omega_j \omega_i} \Big) \right) \right].$$

Define ϕ_ω to be the moment generating function of I_ω, so that

$$\phi_\omega[s] = \mathrm{E}_{I_\omega}[e^{-sI_\omega}], \qquad s \geq 0. \tag{4.3}$$

Then taking the expectation with respect to the infectious periods, and since all infec-

tious periods are independent (though not necessary identically distributed),

$$\begin{aligned}
&\mathrm{E}_I\left[\mathrm{P}\left[Z_{t+1}=z_{t+1}|Z_t=z_t,I_t\right]\right]\\
&=\mathrm{P}[Z_{t+1}=z_{t+1}|Z_t=z_t]=\left\{\prod_{\omega_i\in\mathbb{S}}\binom{N_{\omega_i}-y_{\omega_i,t}}{x_{\omega_i,t+1}}\right\}\times\\
&\sum_{k_{\omega_1}=0}^{x_{\omega_1,t+1}}\cdots\sum_{k_{\omega_S}=0}^{x_{\omega_S,t+1}}\left(\prod_{\omega_i\in\mathbb{S}}(-1)^{x_{\omega_i,t+1}-k_{\omega_i}}\right)\left(\prod_{\omega_i\in\mathbb{S}}\binom{x_{\omega_i,t+1}}{k_{\omega_i}}\right)\\
&\qquad\left(\prod_{\omega_j\in\mathbb{S}}\phi_{\omega_j}\left[\sum_{\omega_i\in\mathbb{S}}\lambda_{\omega_j\omega_i}(N_{\omega_i}-k_{\omega_i}-y_{\omega_i,t})\right]^{x_{\omega_j,t}}\right).
\end{aligned}$$

Thus we can specify an arbitrary infectious period distribution of each class $\omega\in\mathbb{S}$, provided the generating function in Equation 4.3 can be evaluated.

We now have the multi-type multi-level generation probability for any infectious period distribution and any infection matrix Λ. In the case of fixed infectious period, $I_\omega=c$, for all $\omega\in\mathbb{S}$ we have $\phi_\omega(s)=\exp(-cs)$ and expression (4.2) reduces to that given in Section 3.5.3.

4.3 Model And Optimisation

In this chapter we shall consider an infectious period determined by a discrete empirical distribution, thus $\phi(s)$ is a finite sum, which requires careful evaluation to keep reasonable run-times. For simplicity, as well as computational efficiency, we shall assume all classes of individual have the same infectious period distribution, i.e. $I_\omega\overset{d}{=}I$ for all $\omega\in\mathbb{S}$, returning to the case of i.i.d. infectious periods.

As discussed in Section 3.5.2, Equation (4.2) is for the general infection matrix Λ.

However, we shall impose a restricted model for the infection matrix Λ in terms of sub-parameters.

4.3.1 Form Of Infection Matrix

Recall the Global-Local-Susceptibility (GLS) Model as defined in Section 3.5.2, which consists of an H length vector of Global rates, λ^G and an H length vector of Local rates, λ^L; a local and global susceptibility for each type. An individual's type is the collection of values it takes for each covariate used to describe the population. Then the infection matrix, $\Lambda = (\lambda_{ij})$, giving the susceptibility of a type j individual to an infection from a type i individual, is a function of these sub-parameter,

$$\Lambda_\psi(\lambda^L_1, \ldots, \lambda^L_H, \lambda^G_1, \ldots, \lambda^G_H).$$

For the household data considered in Chapter 3 the size of each household was small in comparison to the population size. Also, the distribution of household sizes, and corresponding final sizes, was relatively narrow in the range from zero to seven. Hence the local rate was unnormalised, resulting in no variation between households.

For the Newmarket data, the number of horses in each yard varies considerably, the minimum and maximum yard sizes are 6 and 190, with a mean of 43.45 and a median of 31. Thus we include the size of each yard as a normalising factor in the local infection rate. Hence the GLS model is defined as,

$$\lambda_{ij} = \begin{cases} \dfrac{\lambda^G_{\mathcal{H}(j)}}{N} & \text{for } \mathcal{L}(i) \neq \mathcal{L}(j) \\ \dfrac{\lambda^L_{\mathcal{H}(j)}}{N_{\mathcal{H}(j)}} + \dfrac{\lambda^G_{\mathcal{H}(j)}}{N} & \text{for } \mathcal{L}(i) = \mathcal{L}(j), \end{cases}$$

giving the rate from an individual of type i to an individual of type j. Where $N_{\mathcal{H}(j)}$ is the number of individuals of the same type as individual j. Recall from Section 3.5, for an individual j in the population, $j \in \{1, \ldots, N\}$, then $\mathcal{H}(j)$ is the type of individual j in terms of the vector of covariates. Similarly, $\mathcal{L}(j)$ is the level of individual j as the vector of groups to which it belongs.

For the Newmarket outbreak we have complete data for a single type of infective with two levels of mixing, which reduces to only two sub-parameters, i.e. a local and global rate λ^L and λ^G respectively. Hence the class of an individual i, denoted ω is simply the yard to which that individual belongs.

4.3.2 Optimisation Of Likelihood

As discussed in Section 3.7.1, it is important to analyse the likelihood and find any cancellations or re-usable terms to prevent unnecessary calculations. For the one-type two-level GLS model with i.i.d. infectious periods, it is possible to store intermediate evaluations of the generating function $\phi(s)$. If the function can be expressed in a simple analytical form the saving may not be that great. However, if ϕ must be evaluated numerically then we may save many calculations.

Let ω and ν be two classes of individual, e.g. two yards, then consider the generation function in Equation (4.2),

$$\phi_\omega\Big[\sum_{\nu\in\mathbb{S}} \lambda_{\omega\nu}(N_\nu - k_\nu - y_{\nu,t})\Big].$$

If we partition the sum into local and global rates, we have

$$\begin{aligned}
&\phi_\omega\Big[\sum_{\nu\in\mathbb{S}}\lambda_{\omega\nu}(N_\nu - k_\nu - y_{\nu,t})\Big] \\
&= \phi_\omega\Big[\frac{\lambda^L}{N_\omega}(N_\omega - k_\omega - y_{\omega,t}) \;+\; \sum_{\nu\in\mathbb{S}}\frac{\lambda^G}{N}(N_\nu - k_\nu - y_{\nu,t})\Big] \\
&= \phi_\omega\Big[\frac{\lambda^L}{N_\omega}(N_\omega - k_\omega - y_{\omega,t}) \;+\; \frac{\lambda^G}{N}\sum_{\nu\in\mathbb{S}}(N_\nu - k_\nu - y_{\nu,t})\Big] \\
&= \phi_\omega\Big[\frac{\lambda^L}{N_\omega}(N_\omega - k_\omega - y_{\omega,t}) \;+\; \frac{\lambda^G}{N}(N - y_t - \sum_{\nu\in\mathbb{S}}k_\nu)\Big].
\end{aligned}$$

Where ϕ_ω is a function of k_ω and the sum of all k's. Equation (4.2) includes a cascading sum over sets of k's. Therefore, if we store the values of ϕ_ω for a given set of k's we can reuse the value for combinations with the same total, i.e. $\sum_{\nu\in\mathbb{S}} k_\nu$, and k_ω. For large generation sizes across multiple classes, the values will be reused a number of times saving the cost of repeatedly evaluating the generating function.

4.4 Data

We thank Marc Baguelin and Nikos Demiris for providing the data for the outbreak of Equine Influenza (H3N8) at Newmarket in 2003. The outbreak is described by Newton et al. (2006), though the full data set is not included. Though the outbreak obviously occurred over a period of time, it was not recorded in sufficient detail to permit temporal modelling; specifically, the times of infection and removal are missing. Only the time of detection of the first case is recorded for each yard and the overall length of the outbreak, which is insufficient to reliably use temporal inference techniques. However, there was a defined end point to the outbreak, which progressed over the period March to May 2003. Thus, we can apply our final size analysis based on the available counts of infected horses in each yard.

To apply our method we require the number of horses in each yard and the number that were infected during the course of the epidemic. This detail is not present in Baguelin et al. (2009), but was provided by the authors. However, the data provided for the outbreak is not exactly as described by Newton et al. (2006) or analysed by Baguelin et al. (2009). Specifically, these papers discuss the twenty-one yards that were infected during the outbreak, but Table 4.1 only includes twenty yards with cases. The data are compiled from several sources, including trainer surveys and other mis-matched sources, hence it is unsurprising that different versions exist. Also, we shall assume a single initial infective in a given yard, corresponding to the yard where the outbreak was first detected, it is uncertain whether Baguelin et al. (2009) are using the same seed yard, specifically yard 13 in Table 4.1. Our brief investigation of the placement of the initial infective in Chapter 3 determined there was little effect, though in that case the household sizes were small and relatively uniform, which is not the case for the yards. For the purpose of illustrating our method these differences are unimportant, though for the comparison of the parameter estimates these discrepancies must be considered.

Investigations into the factors associated with risk of infection have been reported by Barquero et al. (2007). In particular, there is a compulsory vaccination program for horses that attend race events. However, the effect of the vaccine is variable due to factors such as antigentic drift and characteristics of each horse, e.g. vaccine type and administration schedule. Such detail is unavailable in the data provided, though small numbers of horses were tested for immunity in each yard, the results of these tests should be used with caution since it is difficult to accurately determine immunity (see Park et al. (2004) for a discussion of the two common strains for Equine Influenza). Also, we do not have break downs of yards into male and female horses, a factor that Barquero et al. (2007) find significant in the risk of infection. In the latter part of their paper, Baguelin et al. (2009) consider vaccination effects and interventions, so-called

vaccination in the face of an outbreak. Our non-temporal method cannot accommodate such interventions, hence we do not consider this further.

The latent and infectious periods of equine influenza were investigated in a study by Park et al. (2004) on a sample of 24 horses. The distributions are estimated by an empirical distribution and are used by Baguelin et al. (2009), hence we also adopt these values. Recall that, for final size analysis, the Susceptible-Exposed-Infective-Removed (SEIR) model which includes a latent period (see Section 1.2.5.1) has the same final size distribution as for the Susceptible-Infective-Removed (SIR) model. Hence we apply our method using only the empirical infectious period as defined in Table 4.2. For comparison we use a fixed infectious period with mean equal to the empirical mean of $3\frac{1}{3}$ days. However, Baguelin et al. (2009) use an exponential infectious period and the empirical distribution. As discussed in Demiris and O'Neill (2005a), differing infectious period distributions give similar point estimates of the rate parameters, thus it is reasonable to compare a fixed period with an exponential.

The complete final outcome data for all 58 yards is given in Table 4.1. For each yard i we have the total number of horses and the detected final size, N_i and D_i respectively. Thus the total population is $\sum_i N_i = 2520$ and the total final size is $\sum_i D_i = 617$. For a subset of the yards, a sample of horses were tested using an anti-body level test, checking for immunity. These are reported as the total tested and the number of those tested that were found to be immune. This additional information is included for reference and to motivate further work. Due to the uncertainty over the testing procedure, the definition of immune and other unknown factors we do not consider this additional information.

A reduced data set, consisting of only ten yards, is presented in Table 4.3. The reduction is motivated by Baguelin et al. (2009) to enable to computation of parameter estimates,

a similar benefit in run-time is obtain for our MCMC algorithms and so we also analyse this smaller data set. These ten yards are identical to those in Baguelin et al. (2009), being provided by the authors. However, the ten yards are not an exact subset of Table 4.1, there is no corresponding entry for some of the rows in Table 4.3. The third yard is indicated to contain the single initial infective, this is after matching with the equivalent yard in the raw data provided, this matching may be incorrect. Lacking a one-to-one correspondence between the two data sets, we cannot include any of the additional immunity information.

As a final consideration, as indicated by Baguelin et al. (2009), the reported final sizes are actually from a sample of each yard. Thus, within a yard of size N_i, a number of horses, $\tilde{N}_i$, were tested for the disease of which D_i were found to be positive. The actual number tested within each yard is unknown, but the sampling was performed at random.

4.5 Model

As discussed in Newton et al. (2006) and Baguelin et al. (2009), the horses are moved along paths between yards and training areas, with horses passing close enough to permit infection. Thus it is reasonable to assume a one-type two-level mixing model, having a within yard and a global mixing rate. Though spatial data is available for the yards, there is no biological motivation for adding a more complex model than the two-level mixing.

For our MCMC method, define Λ as in Section 4.3.1 and let there be a fixed single initial infective in the indicated yard, i.e. 13 and 3 for the 58 and 10 yards respectively. We consider a fixed infectious period of $3\frac{1}{3}$ days and the empirical infectious period

Yard	N_i	D_i	Tested	Immune	Yard	N_i	D_i	Tested	Immune
01	13	0			30	11	0		
02	14	0			31	14	0		
03	89	0			32	19	17	17	15
04	27	0			33	97	0		
05	30	2			34	43	0		
06	67	0			35	24	0		
07	32	16	8	4	36	10	0		
08	83	39	80	67	37	45	0		
09	18	0			38	18	2	16	16
10	6	0			39	67	0	56	20
11	23	0			40	19	16	17	15
12	18	0			41	20	0		
13 †	103	78	78	45	42	39	0		
14	7	0			43	82	0		
15	18	0			44	23	0		
16	141	74	124	84	45	9	0		
17	17	0			46	65	0	26	26
18	7	0			47	190	134	159	137
19	35	0			48	32	29	34	30
20	110	33			49	62	0		
21	20	0			50	16	0		
22	13	0			51	49	0		
23	17	0			52	41	17	31	25
24	47	47			53	37	12		
25	70	13	62	51	54	59	0		
26	36	0	33	18	55	25	19	24	16
27	29	0			56	25	13		
28	103	25			57	70	9		
29	46	0	43	2	58	70	22		

Table 4.1: Data for the outbreak of equine influenza at Newmarket in 2003, $\psi^{(3)}$. Giving the size, N_i, and the reported final outcome, D_i, of each yard i; the first detected yard is indicated by †. Also, tests for immunity within selected yards where a number of horses were randomly tested.

Number of Days Infectious	0	1	2	3	4	5	6	7
Frequency	0	7	1	4	2	9	1	0

Table 4.2: Empirical infectious periods for a study of 24 horses that were heterologously vaccinated. Each horse was infected and observed to determine the latent period (not shown) and the infectious period, estimated as the number of days between the virus first being detected and the last detectable symptom, see Park et al. (2004) for further details.

Yard	N_i	D_i
1	60	18
2	83	78
3	103	80 †
4	141	78
5	36	6
6	25	19
7	19	16
8	190	139
9	32	30
10	41	21
	$\sum_{i=1}^{10} N_i = 730$	$\sum_{i=1}^{10} D_i = 485$

Table 4.3: Reducing the full data set to 10 yards, $\psi^{(4)}$, to ease computation of the parameter estimates. The single initial infective is assumed to be in the third yard, †, i.e. $D_3 = 80$, $a_3 = 1$ and $d_3 = 79$. Note, the yards do not match exactly with those in Table 4.1, thus we do not have immunity data.

distribution of Table 4.2. Baguelin et al. (2009) analyse the same model, though using different techniques and an exponential infectious period with mean $3\frac{1}{3}$ days instead of a fixed period.

Baguelin et al. (2009) also consider the issue of initially immune individuals, that is a proportion of horses begin in the removed state of the SIR epidemic, which our non-temporal final size inference can accommodate. The proportions of truly susceptible horses in each yard are estimated using a risk parameter derived from a sample of 400 horses in 10 yards. These 10 yards are not those of Table 4.3 (the total number of horses do not match), so we cannot reproduce this part of the model. In the following section, it is assumed the entire population is susceptible, i.e. a homogeneous one-type model, for the MCMC method and also for the results of Baguelin et al. (2009) unless otherwise stated.

4.6 Results

Given the data and model, we make inference on the parameters of interest, i.e. λ^L and λ^G. In Section 4.6.1 we present the results of Baguelin et al. (2009) and discuss the methods used to obtain these estimates. The description of these methods is from interpreting the published paper (and a preprint). At times the exact method is unclear and we have made assumptions on the authors method and analysis. To maintain consistency within the thesis, the methods are translated into our notation.

Section 4.6.2 gives the posterior estimates from our MCMC generation method. As always, it must be checked that the MCMC chain has indeed converged and a technique to check is outlined in Section 4.6.3. The immunity of horses is briefly discussed in Section 4.6.4, with potential future work considered.

Data	Infectious Period	λ^L	λ^G
Full	Exponential	1.03	0.015
	Empirical	0.7	0.015
10 Yards	Exponential	0.78	0.017
	Empirical	0.69	0.016

Table 4.4: Summary of results presented by Baguelin et al. (2009), obtain using a method similar to Approximate Bayesian Computation for the outbreak of equine influenza at Newmarket in 2003. Both infectious period distributions have a mean of $3\frac{1}{3}$ days.

Finally, since the estimates differ between the methodologies, this difference is investigated by performing a simulation study in Section 4.6.5 and explanations of the simulations are given.

4.6.1 Published Results And Methods

The results of Baguelin et al. (2009) are summarised in Table 4.4, namely the point estimates of the local and global infection rate for both data sets using an exponential and empirical infectious period. We now discuss the methods used to obtain these estimates. Unfortunately, the variance of these estimates are not explicitly stated. Baguelin et al. (2009) consider two methods to estimate the parameters, the first is applied to the 58 yard data set, the second to the reduced 10 yard data. In this section we reproduce these methods, as stated in Baguelin et al. (2009), though several issues are noted, the comparison between methods and estimates in deferred until Section 4.6.5.

4.6.1.1 First Method

Using the full data set, Baguelin et al. (2009) implement an ABC rejection algorithm, as defined in Section 1.3.4. There is a growing literature for ABC, in particular in population genetics where the method was originally applied, see for example Beaumont et al. (2002), Sousa et al. (2009) and Toni et al. (2009).

Recall, the data described by Baguelin et al. (2009) is not the same as presented in Table 4.1, an additional yard has infections detected in it, i.e. twenty-one yards instead of the twenty in Table 4.1. However, more importantly the total number of yards in their analysis is unknown, i.e. additional yards that were not infected during the outbreak. Clearly, yards that avoid infection are a source of information on the global rate, λ^G. Newton et al. (2006) describe the outbreak as occurring in twenty-one yards involving over 1300 horses, from Table 4.1, the total number of horses in the twenty yards that are infected is 1185. Further, the map of the yards at Newmarket in Newton et al. (2006) contains 68 yards. From now on, we shall assume the estimates in Table 4.4 are based on the data in Table 4.1 (though this is clearly not the case).

ABC is a technique to obtain samples from a posterior distribution using an approximation to the likelihood. This is achieved by generating realisations of a process using a set of parameters, then defining a metric to determine if the realisation is close to the observed data. If it is close, then the parameter values used to generate it are a sample from the approximate posterior.

Formally, let λ^L and λ^G be the parameters of interest. Given $N = (N_1, \ldots, N_{58})$ and $a = (0, \ldots, 1, \ldots, 0)$ (a single initial infective in the 13^{th} yard), Baguelin et al. (2009) generate a realisation of the continuous time SEIR epidemic. Details of the simulation method are not given, in particular for the empirical infectious period distribution the

process is no longer Markovian. Let $f_B(D|\lambda^L, \lambda^G, N, a)$ denote the likelihood of a final size vector $D = (D_1, \ldots, D_{58})$ given the rates and yard sizes, then generate the i^{th} realisation as $D^{[i]} \sim f_B(D|\lambda^L, \lambda^G, N, a)$.

For the distance metric, Baguelin et al. (2009) consider the total final size and the number of yards infected, specifically the final size must be between 24% and 25% of the population, i.e. $612 \leq \sum_k D_k \leq 638$ and twenty yards must be infected (though not necessarily the twenty yards in the observed data). The final size restriction is applied on its own, then together with the required number of yards. Formally, we can express this as the following, for the i^{th} realisation, let Δ_{B_1} and Δ_{B_2} denote the distances between a realisation and the observed final size vector, $D = (D_1, \ldots, D_{58})$, then

$$\Delta_{B_1}(D, D^{[i]}) = \left(1 - \mathbb{I}\left(612 \leq \sum_{k=1}^{58} D_k^{[i]} \leq 638\right)\right), \tag{4.4}$$

and

$$\Delta_{B_2}(D, D^{[i]}) = \left|1 - \mathbb{I}\left(612 \leq \sum_{k=1}^{58} D_k^{[i]} \leq 638\right)\right| + \left|20 - \sum_{k=1}^{58} \mathbb{I}\left(D_k^{[i]} > 0\right)\right|, \tag{4.5}$$

where $\mathbb{I}(E)$ is the indicator function for event E. Note, we must define the distances as in Equations (4.4) and (4.5) such that, for two arbitrary final size vectors of length K, i.e. $D^{[i]} = (D_1^{[i]}, \ldots, D_K^{[i]})$ and $D^{[j]} = (D_1^{[j]}, \ldots, D_K^{[j]})$, the Δ_{B_2}-distance is

$$\Delta_{B_2}(D^{[i]}, D^{[j]}) = \left|\mathbb{I}\left(612 \leq \sum_{k=1}^{K} D_k^{[i]} \leq 638\right) - \mathbb{I}\left(612 \leq \sum_{k=1}^{K} D_k^{[j]} \leq 638\right)\right| + \left|\sum_{k=1}^{K} \mathbb{I}\left(D_k^{[i]} > 0\right) - \mathbb{I}\left(D_k^{[j]} > 0\right)\right|,$$

then $\Delta_B(D^{[i]}, D^{[j]})$ is a well defined metric on the space of valid final size vectors,

satisfying,

$$\begin{aligned}
&\Delta_B(D^{[i]}, D^{[i]}) = 0,\\
&\Delta_B(D^{[i]}, D^{[j]}) > 0,\\
&\Delta_B(D^{[i]}, D^{[j]}) = \Delta_B(D^{[j]}, D^{[i]}),\\
&\Delta_B(D^{[i]}, D^{[l]}) \leq \Delta_B(D^{[i]}, D^{[j]}) + \Delta_B(D^{[j]}, D^{[l]}),
\end{aligned}$$

for all final size vectors, i, j and l.

For a Bayesian approach, there must be a prior distribution on the parameters; Baguelin et al. (2009) perform a brute force grid search, for 6400 pairs (λ^L, λ^G), they perform 5000 ABC iterations per pair, this is effectively a uniform prior, though its range is not stated. Using the distances defined in Equations (4.4) and (4.5), with an iteration being accepted only if the distance is zero. Note, that despite having a distance of zero, these samples are not exact since we are using summary statistics.

The results using the empirical infectious period are plotted in Baguelin et al. (2009) (see Figure 1); note the axes are on the log scale. Using the accepted sample pairs, point estimates for the rates are calculated and reproduced in Table 4.4. This is a Monte Carlo Maximum Likelihood approach, see Diggle and Gratton (1984).

4.6.1.2 Second Method

For the second method, Baguelin et al. (2009) include an additional parameter into the model, namely the susceptibility of each yard, α_i. Assuming this is given, as well as N_i and D_i, they assume each yard has a single external infection, approximating the epidemic using the independent households model (see for example Addy et al. (1991)). Due to computation issues, presumably the run-time, the reduced data set of 10 yards

(see Table 4.3) is introduced.

The susceptibility is estimated for each yard as follows, using the estimate of λ^L from the first method and assuming the final proportion of infected animals is $\gamma = 0.7287$ for each yard, then Baguelin et al. (2009) propose,

$$\alpha_i \approx \frac{\gamma_i}{1 - \exp(-\gamma_i \frac{\lambda^L}{g_i})},$$

where $\gamma_i = \gamma$ for all yards and g_i is the removal rate for yard i, i.e. for an exponential infectious period the removal rate is g_i. This forumla is derived by algebraic manipulation of the final size limiting result, see Theorem 1.1.

Given the α_i's, then for yard i, $(1 - \alpha_i)N_i$ horses are completely immune and removed from the epidemic, such that the initial state of the SEIR model for each yard i is: $S_i(0) = \alpha_i N_i$, $E_i(0) = 0$, $I_i(0) = 0$ and $R_i(0) = (1 - \alpha_i)N_i$.

Given the modified number of initial susceptibles, i.e. N_i and α_i, for a given local rate, λ^L, it is possible to calculate the probability of a given final size, D_i, for each yard assuming a single initial infective, i.e. $\mathrm{P}[D_i|N_i, \alpha_i, \lambda^L]$. Thus, if each yard is independent, then

$$\mathrm{P}[D|N, \alpha = (\alpha_1, \ldots, \alpha_K), \lambda^L] = \prod_{i=1}^{K} \mathrm{P}[D_i|N_i, \alpha_i, \lambda^L].$$

It is well known that the final size probabilities are difficult to calculate explicitly. A set of triangular equations were derived by Ball (1986) (see also Andersson and Britton (2000)). However, these are numerically unstable for large populations. Arbitrary precision computing, as described in Section 3.7.2, has been used by Demiris (2004), though Baguelin et al. (2009) report that this was too costly to compute. Instead, they approximate the final size probabilities using simulations; further details are not given.

The local rate is then estimated by performing simulations for a range of values of λ^L, the details are not given in the published paper. However, a preprint provided by the authors gives the number of simulations as 2×10^6 and 2×10^5 for the exponential and empirical distributions respectively, using the 10 yard data set of Table 4.3. These simulations are used to approximate the probability of the observed outbreak within each yard, $\tilde{\mathrm{P}}[D|N, \alpha, \lambda^{[i]}]$ for values of λ^L, indexed by i, i.e. $\lambda^{[i]}$, and the approximate probability of the overall outbreak is the product of the independent yards. Then, the estimate for the local rate is given by the weighted mean,

$$\lambda^L_{\text{est}} = \frac{\int_0^\infty \lambda \mathrm{P}[D|N, \alpha, \lambda] \mathrm{d}\lambda}{\int_0^\infty \mathrm{P}[D|N, \alpha, \lambda] \mathrm{d}\lambda} \tag{4.6}$$

$$\approx \frac{\sum_i \lambda^{[i]} \tilde{\mathrm{P}}[D|N, \alpha, \lambda^{[i]}]}{\sum_i \tilde{\mathrm{P}}[D|N, \alpha, \lambda^{[i]}]}. \tag{4.7}$$

Equation (4.6) is as stated by Baguelin et al. (2009), using our notation. Equation (4.7) is our intepretation of their estimator, in terms of their simulations.

Given the estimated local rate, λ^L_{est}, Baguelin et al. (2009) then estimate the global rate using simulations of the full model. We assume the estimate is through rejection sampling, using the distance Δ_{B_1}, i.e. such that the final size is between 24% and 25% of the total population. Details in a preprint from the authors, state an unspecified range of values for λ^G were each simulated 4000 times, using the accepted samples a point estimate for λ^G was calculated, using a similar estimator to Equation (4.7).

4.6.1.3 Further Results

As mentioned, the remaining sections of Baguelin et al. (2009) consider vaccination, based on the local and global rates estimated using the second method described in Section 4.6.1.2. We do not consider vaccination, though further investigation would be

of interest, see Section 4.6.4.

We draw attention to a single section, on quantifying the impact of the size of the seeding yard (Baguelin et al., 2009, see Section 2.4.1), that we will discuss in Section 4.6.5, namely that the yard with the single initial infective has a large effect on the final size distribution of the epidemic.

4.6.2 Results From MCMC Method

The multi-type multi-level algorithm derived in Chapter 3, using the arbitrary infectious period integrated likelihood derived in Section 4.2, was applied to the two data sets in Tables 4.1 and 4.3, using the GLS model defined in Section 4.3.1, i.e. a one-type two-level model, for both a fixed infectious period and an empirical form defined in Table 4.2.

Assume a single initial infective, which we place in the first yard to record an infected horse, yard 13 or 3 respectively (this is yard one in the description given by Newton et al. (2006)). We consider this to be fixed and do not perform any a-updates on the path.

Our results are presented in Table 4.5, giving the point estimates and standard deviations. We check convergence of the chains in the following section.

4.6.3 Checking Convergence Of MCMC Chains

In Chapter 3 we use the length, τ, of the imputed path z, to gauge if the chain has converged. Recall z may be a high dimensional object, consisting of the size of each class for every generation, thus the summary by a single number does not always

Data	Infectious Period	λ^L	λ^G
Full	Fixed	0.466 (0.0367)	0.0446 (0.0331)
	Empirical	0.585 (0.0509)	0.0501 (0.0424)
10 Yards	Fixed	0.425 (0.0486)	0.0514 (0.0422)
	Empirical	0.473 (0.0549)	0.0841 (0.0411)

Table 4.5: Summary of results using generation method for the outbreak of equine influenza at Newmarket in 2003, posterior means and standard deviations in parentheses. Both infectious period distributions have a mean of $3\frac{1}{3}$ days.

capture enough detail. For the household data in Section 3.6 each group was small in comparison to the total population size. Hence the mixing of individuals within a given class of z was sufficiently summarised by the overall length of the path.

For the yard data, individual yards represent a larger proportion of the population and corresponding final size, for example in the ten yard data set, $\psi^{(4)}$, the eighth yard is $(N_8, D_8) = (190, 139)$, which accounts for 26% of the population and 29% of the total final size. Thus, if the epidemic is locally driven, the column of z corresponding to the eighth class will resemble the one-type epidemics of Section 3.2. For many consecutive K-jumps the path length may not alter, yet the chain is mixing within the space of all paths.

To overcome this, we consider five new summary statistics of the path z at each iteration, in addition to the length τ. Consider the generation totals of z, denoted x_t such that

$$x_t = \sum_{\omega \in \mathbb{S}} x_{\omega,t}.$$

For the i^{th} iteration of the MCMC algorithm, the path z may be summarised by the

vector of generation totals,

$$z^{[i]} = (x_0, x_1, \ldots, x_\tau).$$

Consider the vector of generation totals to be a set of values, then we can compute the moments of this set. Specifically the first four standard centred moments: mean, $\mu = \mathrm{E}[z]$; variance, $\sigma^2 = \mathrm{E}[(z-\mu)^2]$; skew, $\gamma_1 = \frac{\mu_3}{\sigma^3}$ and kurtosis $\gamma_2 = \frac{\mu_4}{\sigma^4} - 4$. Where we define $\mu_n = \mathrm{E}[(z-\mu)^n]$. Note that the mean of z and the path length τ are related,

$$\mu = \mathrm{E}[z] = \frac{1}{\tau+1}\sum_{t=0}^{\tau} x_t = \frac{D}{\tau+1},$$

since D is a constant the two statistics are the scaled inverse of each other. Finally, we consider the maximum generation size, i.e. $x_{\max} = \max\{x_t : 0 \leq t \leq \tau\}$, as another summary of the path.

Figure 4.1 shows the six summary statistics for the start of a specific MCMC run, for the 10 yard data set using a fixed infectious period of $3\frac{1}{3}$ days. The first 2×10^5 iterations are shown, these are clearly part of the burn in period, i.e. the chain has yet to converge. In contrast, Figure 4.2 is the same MCMC run, showing all the iterations after convergence. The relation of the mean, μ, and the length, τ, of the path, z, is clearly seen.

Whereas the length of the path, has a reasonable interpretation, the higher moments are purely for checking the chain has converged. For example, the variance of z has no direct meaning to the epidemic. The posterior mean of the length is 19.6 generations for a fixed infectious period of $3\frac{1}{3}$ days. Thus, crudely the epidemic has a duration of 33 days, which is comparable to the recorded length of the outbreak, from March to May. Note this is a crude approximation to the actual temporal length of the outbreak, since a generation does not represent a known period of time nor are individuals in a

generation infected at the same time. Similarly, Figures 4.3 and 4.4 show the trace plots, for the same MCMC run as above, of the infection rate parameters, λ^L and λ^G, during the burn in period and after convergence respectively.

Similar plots were generated for the 10 yard data using the empirical infectious period and the 58 yard data, using both infectious periods. These are not included, though all showed a similar convergence after a burn in period of approximately the same length. Recall, the number of iterations for the burn in period is dependent upon the initial seed of the path z, see Section 3.2.3.3, several short MCMC chains were run to determine an appropriate seed path.

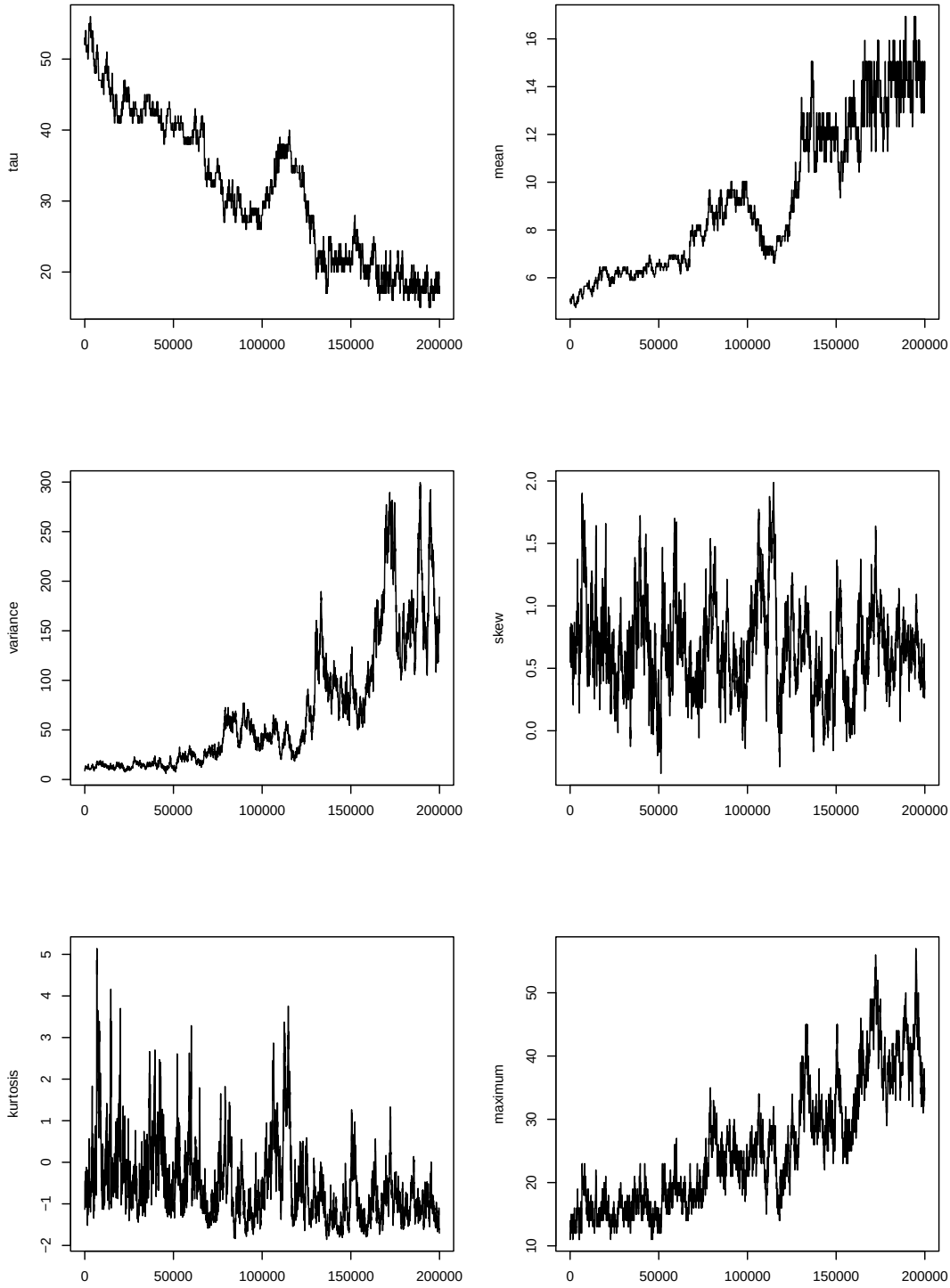


Figure 4.1: Trace plots for the moments of the imputed path z for the 10 yard data set, $\psi^{(4)}$, using a fixed infectious period of 3.3 days. The plots show the start of the burn in period, i.e. before convergence.

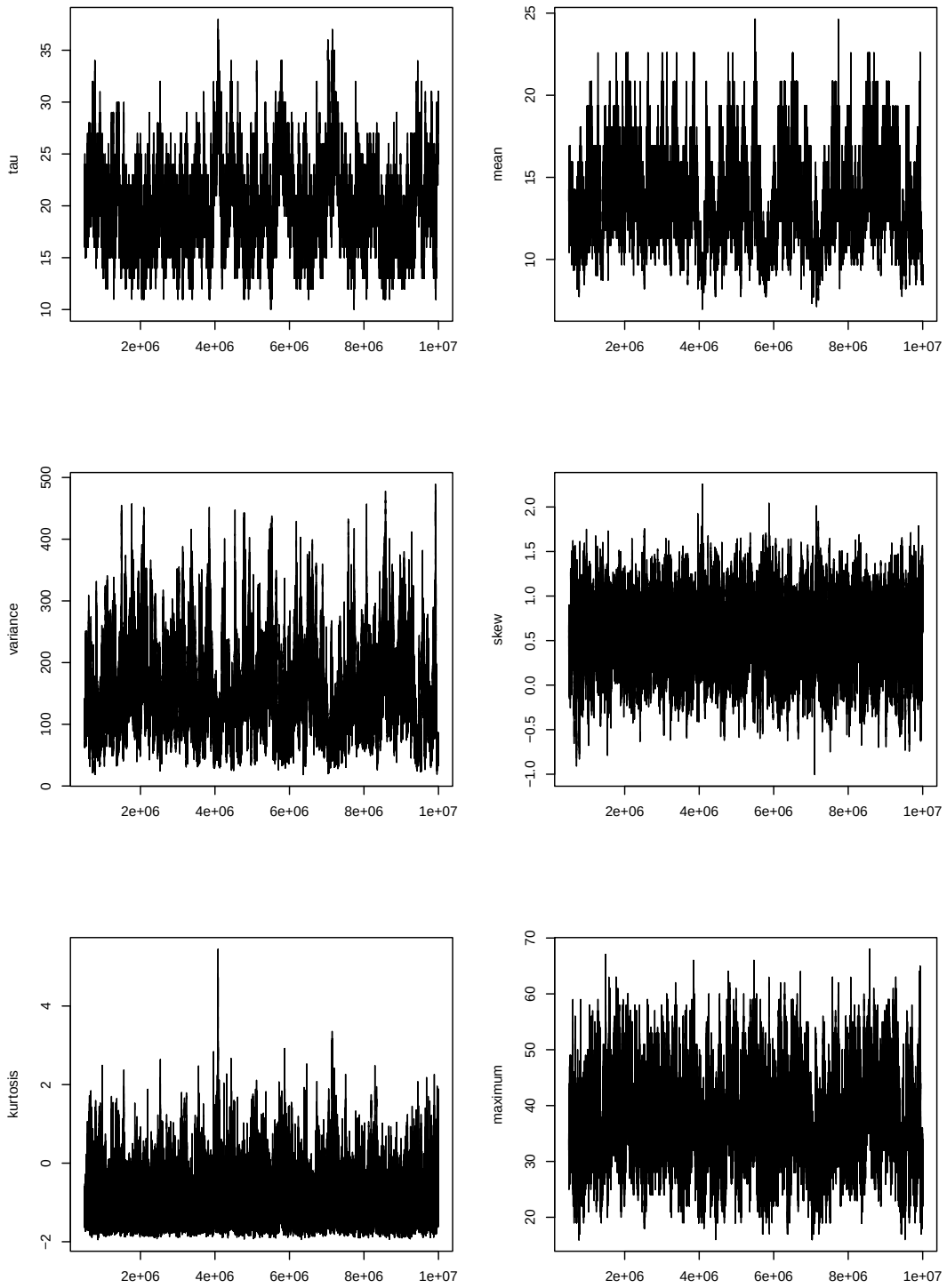


Figure 4.2: Trace plots for the moments of the imputed path z for the 10 yard data set, $\psi^{(4)}$, using a fixed infectious period of 3.3 days. The plots show the chain after the burn in period.

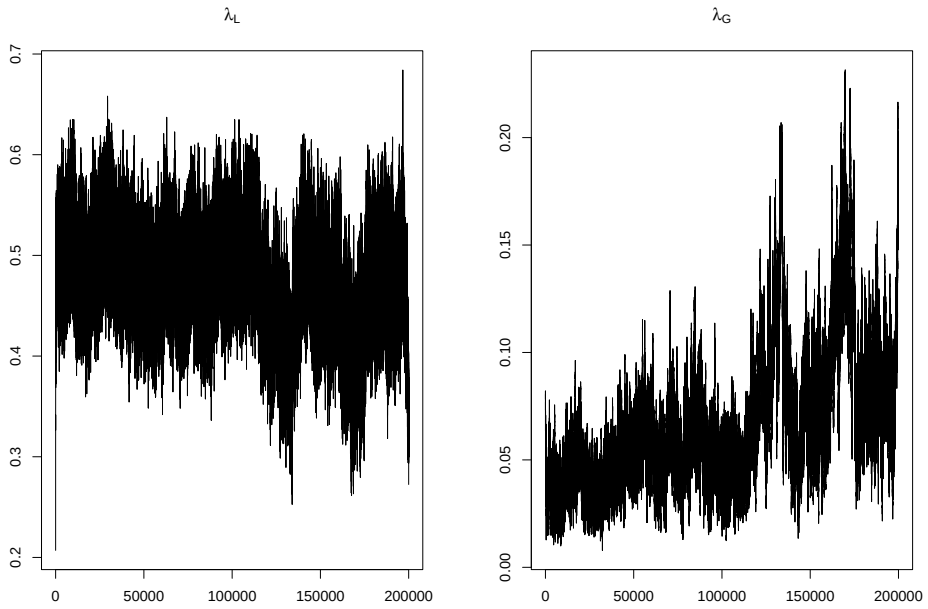


Figure 4.3: Trace plots for the local and global infection rates for the 10 yard data set, $\psi^{(4)}$, using a fixed infectious period of 3.3 days. The plots show the start of the burn in period, i.e. before convergence.

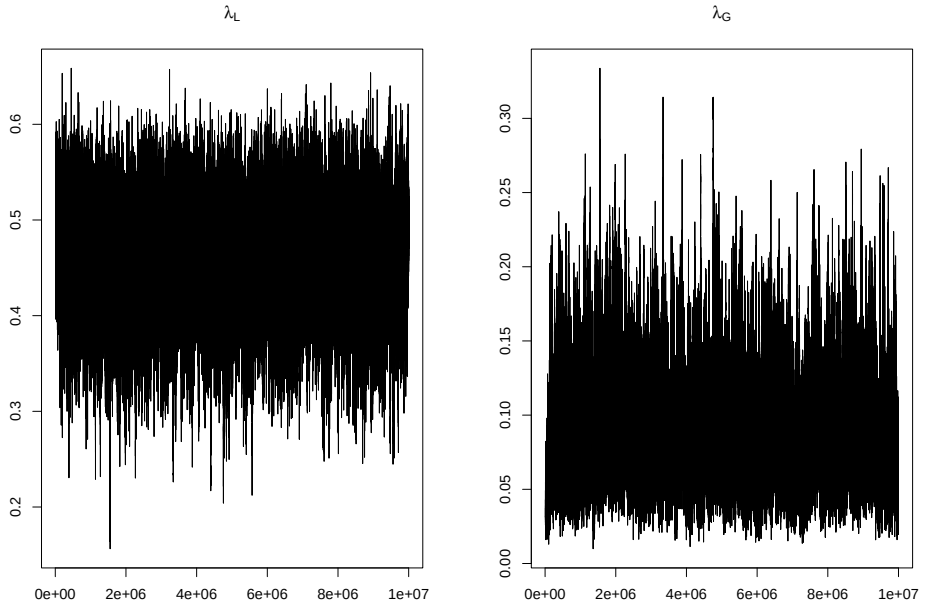


Figure 4.4: Trace plots for the local and global infection rates for the 10 yard data set, $\psi^{(4)}$, using a fixed infectious period of 3.3 days. The plots show the chain after the burn in period.

4.6.4 Partially Observed Extension

The time to obtain the 10^7 iterations for an MCMC run using the full data set, assuming a fully observed epidemic, was several days for the constant infectious period and up to two weeks for the empirical distribution. This is after using GNU OpenMP on eight processors to achieve a significant decrease in computation time. The empirical distribution required longer runs due to the increased computation time when using GNU MPFR to ensure accurate acceptance probabilities. These long run times motivated us to use the reduced data set of only ten yards, from among the full set of fifty eight. Baguelin et al. (2009) also reduce to the smaller number of yards for computational reasons. By reducing to the 10 yards, we are effectively only partially observing the epidemic. We do not account for the unoberved yards, since the aim is to reduce the complexity of the problem in order to reduce the algorithm run-times.

The issue of immunity is emphasised by the study of Park et al. (2004) and the analysis of Barquero et al. (2007). In particular, by observing the immunity information of Table 4.1, there is cause to believe that different yards have varying susceptibility due to the horses immunity. The variability in the administration of a vaccine, due to different trainers using varying types and dosages, implies that we are not accounting for an important factor in modelling the outbreak.

Baguelin et al. (2009) attempt to make inference for the yard level immunity, as discussed in Section 4.6.1. For our generation method, having an unknown number of horses initially immune (where immunity gives complete protection from the disease) corresponds to having an unknown number of susceptibles, i.e. for each yard i, $N_i = N_{i,\text{ob}} + N_{i,\text{un}}$, as discussed in Section 3.3.8. In that case, the immunity data in Table 4.1 could be used to form informed prior distributions for each $N_{i,\text{un}}$. However, given the increased number of iterations required for adequate mixing when consider-

ing a partially observed epidemic, it was infeasible to consider an uncertain number of susceptibles. Hence the immunity testing data was not incorporated as a random effect in our model. It would be interesting to attempt such analysis, though the run-times would be particularly long.

Further, as briefly mentioned in Section 4.4, the reported final sizes are actually from random samples of each yard. This means there is actually an unknown number of infectives in each yard, as well as an unknown number of susceptibles. However, the number of susceptibles is bounded above and the number of infectives is bounded below, meaning the problem is at least well defined; though such a model is unlikely to yield reliable estimates given data as in Table 4.1.

In fact, there is also uncertainty on the effect of the vaccine, specifically a vaccinated horse is not necessarily completely protected from the disease. Hence, incorporating an unknown number of susceptibles is not an accurate model, instead the vaccine may effect the susceptibility of a horse to infection, altering the model for the infection matrix, Λ.

4.6.5 Comparison Of Results

Inspection of Tables 4.4 and 4.5 immediately reveals very different estimates for the infection rate parameters. Baguelin et al. (2009) do not state the standard deviation of their estimates, hence we cannot compare the variability. Though for the generation method, as expected the empirical distribution has a greater variance than the fixed period for both data sets in Table 4.5.

Our estimates give a lower local infection rate than the methods used by Baguelin et al. (2009). However, to compensate the estimated global rate is larger, as we expect,

since the global and local rate are in general negatively correlated for two-level mixing models.

In Section 4.6.5.1 we compare the parameter estimates in terms of a threshold result, and determine they are in some sense equivalent. Then in Section 4.6.5.2 we simulate realisations of the model for a fixed infectious period given the estimates. Finally, in Section 4.6.5.3, we make some concluding remarks.

4.6.5.1 Calculating R_*

As discussed in Section 1.2.3, the value of the threshold parameter, $R_0 = \iota\lambda$, in the one-type one-level model is an indicator of the behaviour of the epidemic process. Specifically, if $R_0 > 1$, then there is a non-zero probability that there is a major outbreak, as the population size tends to infinity. Also, for a given scaling of the infectious period and the inverse scaling of the infection rate, the threshold parameter is invariant. For the one-type two-level model a similar threshold parameter is derived by Ball et al. (1997), termed R_*

Theorem 4.1 (Ball et al. (1997))

For a two-level mixing model with unequal households, with local rate λ^L and global rate λ^G, and all individuals having an i.i.d. infectious period, I. Let the total number of households be m, of which m_n are of size n, i.e. $m = \sum_{n=1}^{\infty} m_n$, and the total population is $\sum_{n=1}^{\infty} nm_n$. Then

$$R_* = \lambda^G \, \mathrm{E}[I] \, \frac{1}{\sum_{n=1}^{\infty} nh_n} \sum_{n=1}^{\infty} (1 + \mu_{1,n-1}) nh_n,$$

where $h_n = \lim_{m\to\infty} \frac{m_n}{m}$ and $\mu_{1,n-1}$ is the expected final size, not including the single initial infective, of an epidemic with rate λ^L and $n-1$ susceptibles.

To calculate R_* requires the expected final size of an epidemic within each yard, which may be obtained by solving a recursive set of equations; expressed in terms of the Gontcharoff polynomials. However, as noted in Baguelin et al. (2009), the solutions are numerically unstable for the given yard sizes, specifically the yards containing more than 80 horses.

Instead of exact solutions, we may estimate the expected final size in a yard using simulations. Note that, Baguelin et al. (2009) use such simulation of a continuous time epidemic process for inference, namely using rejection sampling on a likelihood approximated by the distance metric, whereas our simulations are to investigate the reported parameter estimates.

Hence, for a fixed infectious period, $c = 3\frac{1}{3}$ days, we can use Equation (3.24) to simulate epidemic paths for a given infection matrix, which in this case is a function of the sub-parameters, i.e. $\Lambda(\lambda^L, \lambda^G)$. Note that, in Equation (3.24) the exponential term is a function of the class, ω and the generation t, define

$$A_{\omega,t} = \sum_{\nu \in \mathbb{S}} \lambda_{\nu\omega}\, x_{\nu,t}\, c.$$

To simulate generation $t+1$, given generation t, we can determine each class independently. Thus, for each class $\omega \in \mathbb{S}$,

$$\mathrm{P}\big[Z_{\omega,t+1} = z_{\omega,t+1} \big| Z_{\omega,t} = z_{\omega,t}\big] = \binom{N_\omega - y_{\omega,t}}{x_{\omega,t+1}} \left(e^{-A_{\omega,t}}\right)^{(N_\omega - y_{\omega,t+1})} \left(1 - e^{-A_{\omega,t}}\right)^{x_{\omega,t+1}}.$$

Then, we may calculate the probability of $X_{\omega,t+1} = x$ for $x = 0, 1, 2, \ldots, (N_\omega - y_{\omega,t})$ and sample from this distribution. Recall, the size of the next generation for class ω is only dependent upon the total infectious pressure from all individuals in generation t, i.e. $A_{\omega,t}$.

The simulations were very quick, requiring less than an hour per yard on average. Computing R_* was only done for the 10 yard estimates, not only due to the time to compute the simulations (since it would require a few days at most for the 58 yards), but because of the unknown full data set used by Baguelin et al. (2009). During approximately an hour of run-time, 2×10^6 epidemics were simulated for a single yard, given a single initial infective and the appropriate rate parameters. This was then repeated for each of the 10 yards; in fact it was possible to run eight yards in parallel using our eight core machine, further reducing the time to compute R_*.

The expected final sizes are shown in Table 4.6, as well as R_* computed as in Theorem 4.1. Specifically, all the yards are of different sizes, thus

$$h_n = \begin{cases} \frac{1}{10} & n \in \{19, 25, 32, 36, 41, 60, 83, 103, 141, 190\} \\ 0 & \text{otherwise,} \end{cases}$$

which gives an of $R_* = 5.18$ from the estimates of Baguelin et al. (2009) and $R_* = 5.12$ from our MCMC estimates.

The estimated threshold parameters are very similar. Note that, for the 10 yard data set all yards are infected, thus there is very little information in the data directly relating to the global rate λ^G. That is, we cannot easily distinguish between a very high global rate with near zero local rate (effectively a globally driven epidemic) and a very small global rate with large local rate. There are issues of identifiability, however the threshold parameter is invariant, being a function of both λ^L and λ^G.

Thus, given the data's lack of information directly relating to the global rate, the posterior density surface will exhibit a ridge, showing the negative correlation between the two parameters. This can be seen in Figure 4.5, which shows a scatter plot for the MCMC run using a fixed infectious period on the 10 yard data set. Note the strong

Yard	Expected Final Size, $\mu_{1,n-1}$	
	$(0.78, 0.017)$	$(0.425, 0.0514)$
1	48.51	16.73
2	67.39	22.37
3	83.76	27.31
4	114.88	36.90
5	28.79	10.88
6	19.77	8.16
7	14.86	6.64
8	155.01	49.57
9	25.51	9.90
10	32.91	12.11
R_*	5.18	5.12

Table 4.6: Estimating R_* using 2×10^6 simulations of epidemics in each yard, giving the expected final sizes shown, and using Theorem 4.1.

negative correlation of -0.709 between the parameters.

As described in Section 4.6.1.2, Baguelin et al. (2009) use a three step process to obtain their estimates for the 10 yard data set. First, using estimates from the full data, they calculate the proportion of susceptible individuals in each yard, α_i, based on the estimates from the rejection ABC algorithm. These ABC estimates are discussed further in Section 4.6.5.2, there are doubts to the validity of the assumptions underlying the approximation. Second, using an independent households approach, the local rate is estimated using the modified 10 yards. Third, in a separate step, given the estimated local rate, λ^L_{est}, the global rate is then estimated, λ^G_{est}.

Thus, there is a dependence of the second method estimates on the first method, which is itself a potentially poor approximation. Also, the second method estimates the global rate conditional upon the local rate. That is, the second method gives the following

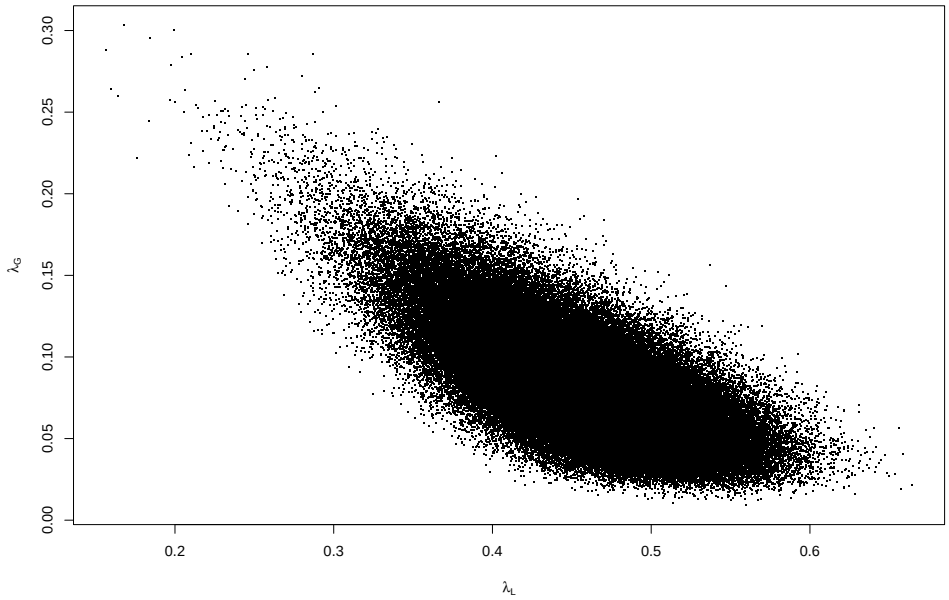


Figure 4.5: Scatter plot for local and global rate from MCMC run using a fixed infectious period of $3\frac{1}{3}$ days on the 10 yard data set, with a correlation of -0.709.

posteriors

$$\pi(\lambda^L|\text{independent yards}, \alpha) \quad \text{and} \quad \pi(\lambda^G|\lambda^L, \text{independent yards}, \alpha),$$

which is not a joint posterior as given by our MCMC method, namely

$$\pi(\lambda^L, \lambda^G, z|\theta).$$

Hence we explain the very low global infection rate as a consequence of conditioning upon the high local rate, which itself is conditional upon the α_i's, which were derived from an approximation giving a high local infection rate. Given the invariance of R_*, it is unsurprising to discover the two methods give similar estimated threshold parameters. Since the conditional estimation of $\lambda^G|\lambda^L$ would tend to scale the estimate appropriately.

Finally, the range of likely pairs, (λ^L, λ^G), can be seen in the increased standard deviations of the estimates between the full data and the 10 yards, Table 4.5, from the high negative correlation between the parameters due to the lack of yards avoiding infection.

4.6.5.2 Simulations Of Final Size

Using the generation representation of an epidemic it is efficient to simulate realisations of an epidemic given parameter values. Using the procedure, as outlined in Section 4.6.5.1, though including multiple yards, we can generate paths very quickly for the fixed infectious period case.

To continue the comparison between the estimates of Baguelin et al. (2009) and our generation method, we generated 10^7 simulations using our generation representation. For the parameter values, we use the fixed parameters $(1.03, 0.015)$ from Baguelin et al. (2009) and draws from the posterior distribution from our MCMC. The full data set, $\psi^{(3)}$ in Table 4.1 took approximately eight hours to simulate. Also, we generated 10^7 simulations of the 10 yard data set, $\psi^{(4)}$ in Table 4.3, taking approximately three hours. Again, we use the fixed parameters $(0.78, 0.017)$ from Baguelin et al. (2009) and draws from the posterior distribution from our MCMC.

Note, these simulations are of epidemic paths, i.e. the generation structure. The simulations performed by Baguelin et al. (2009) for their ABC algorithm were continuous time SEIR models. For the purpose of final size analysis, simulating paths is quicker and equivalent in terms of the final size distribution.

However, we are not making a fair comparison, firstly the underlying models used for inference are different. More importantly, by only using a point estimate we are under estimating the variability of the Baguelin et al. (2009) estimates.

Using our notation, the total final size of an epidemic, $\sum_k D_k$, is plotted for the full and 10 yard comparison in Figures 4.6 and 4.8 respectively. In both cases, the plot has been broken at a final size of 50, with both parts then scaled to highlight the features of interest. For our MCMC estimates, the final size distributions are as expected, a mass at zero representing all minor epidemics and an approximately normal peak on the right-hand side. The Baguelin et al. (2009) plots are more interesting (the upper plots in both figures). Beyond the cut point of 50, there is a small peak, following by several other peaks which reduce in size. The large local infection rate means that the epidemic is likely to take off within the yard containing the initial infective. However, the low global rate means the epidemic may not spread to other yards, giving the first peak. The smaller peaks are when the epidemic spreads to one of the smaller yards, then fails to spread to a third or fourth yard. In all cases, the observed final size (indicated by a vertical line) has a low probability of occurring. Finally, when an epidemic does take off, the final size is larger given the estimates by Baguelin et al. (2009).

For the latter sections of their paper, Baguelin et al. (2009) use the infection rates estimated using their second method, i.e. the 10 yard results in Table 4.4. It is noted that the final size distribution is dependent upon the location of the initial infective. Given the very low global infection rate, this is unsurprising, the behaviour being exhibited in Figures 4.6 and 4.8. Our MCMC algorithm did not perform any a-updates, i.e. the initial infective was assumed fixed, thus further investigation would be interesting.

One of the criteria to accept or reject a simulated epidemic is the number of yards that are infected during the outbreak. For the full data, twenty of the fifty-eight yards were infected. For the ten yard data, all the yards were infected, which affects the estimation of the parameters. Thus, for our 10^7 simulations, we have recorded the number of yards that are infected. Figures 4.7 and 4.9 consist of two columns, one for each pair of parameters for the 58 and 10 yard data respectively. The first row plots the

final size, labelled D in an abuse of notation, against the number of yards infected in that simulation. The second row gives the distribution of the number of yards infected.

For Figure 4.7, the full data, we see that the MCMC estimates produce a final size distribution that is 'closer' to the observed, i.e. $D = 617$, but the number of infected yards is very different to the twenty observed. Conversely, the ABC estimates give a 'closer' infected yard distribution, but a wider spread of final sizes. It is also clear that the MCMC estimates produce a greater number of minor outbreaks that do not take off.

The two rows are repeated in Figure 4.9 for the 10 yard data and corresponding estimates. The most striking plot is the number of infected yards for the MCMC estimates, though given the relative global rates between the two estimates this is not surprising.

Approximate Bayesian Computation requires a distance metric to determine if a realisation is 'close' to the observed value. We have defined Δ_{B_1} and Δ_{B_2} as in Baguelin et al. (2009). However, for comparison we define a third metric as follows. For the observed final size vector and the i^{th} realisation, $D = (D_1, \ldots, D_K)$ and $D^{[i]} = (D_1^{[i]}, \ldots, D_K^{[i]})$ respectively, define the distance Δ_M as,

$$\Delta_M(D, D^{[i]}) = \sqrt{\sum_{k=1}^{K} (D_k - D_k^{[i]})^2},$$

the Euclidean distance, also known as the L2-norm.

For each simulation, we calculate the Δ_M-distance from the observed final size vector and these are plotted on the third row of Figures 4.7 and 4.9. In both cases, the MCMC estimates generate a larger proportion of 'close' realisations. The peaks at certain distances are easily explain, consider an epidemic that immediately goes extinct,

i.e. $D^{(0)} = a = (0, \ldots 0, 1, 0 \ldots, 0)$, then for the data sets,

$$\Delta_M(D_{\psi^{(3)}}, D^{(0)}) = 195.7 \quad \text{and} \quad \Delta_M(D_{\psi^{(4)}}, D^{(0)}) = 200.4.$$

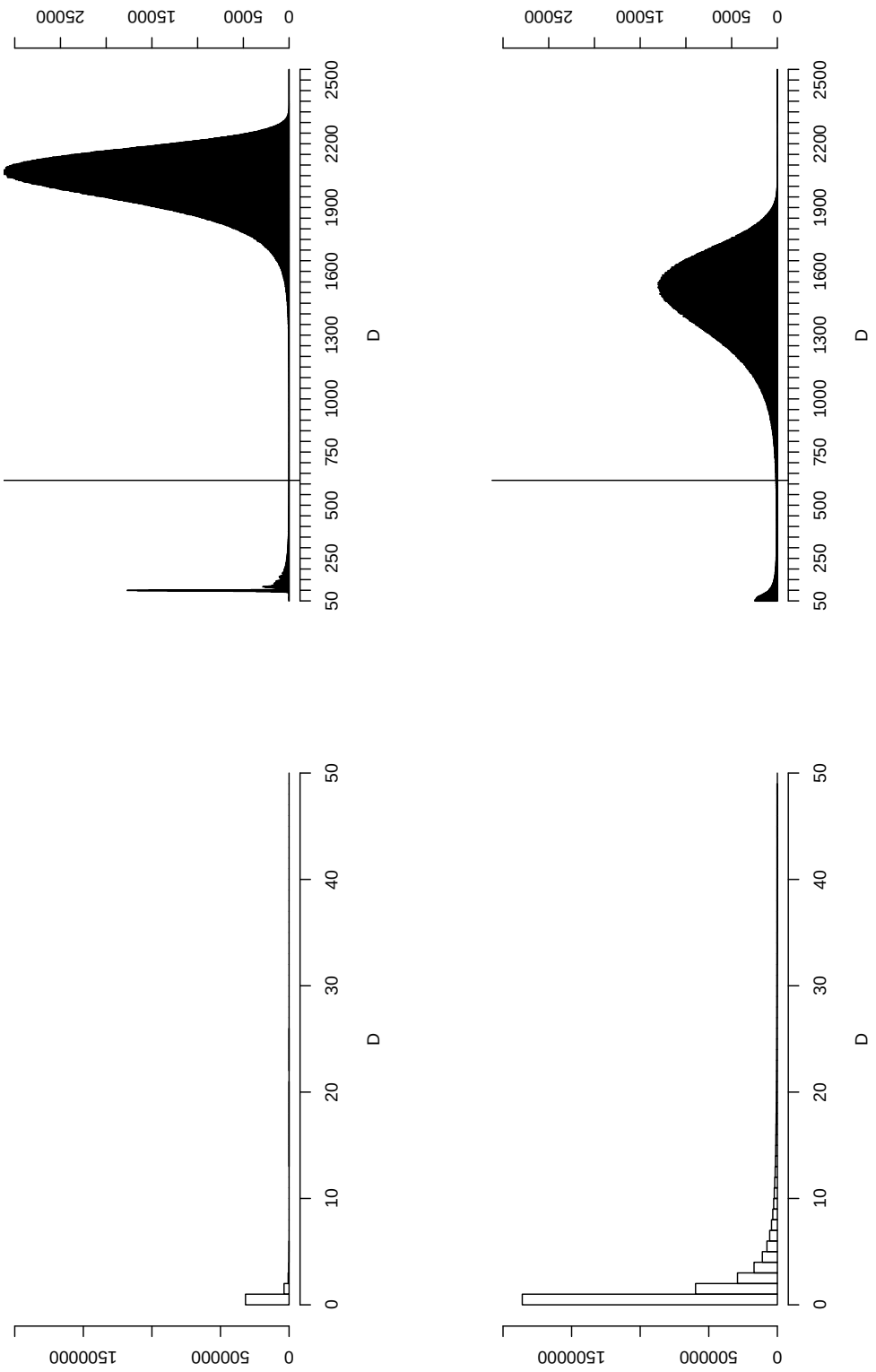


Figure 4.6: Total final size for 10^7 simulations of an epidemic on the 58 yards data set, $\psi^{(3)}$, using a constant infectious period of $3\frac{1}{3}$ days. The upper plot is for the fixed parameter values $(\lambda^L, \lambda^G) = (1.03, 0.015)$ from Baguelin et al. (2009) and the lower uses posterior samples from our MCMC algorithm. The vertical line corresponds to the observed total final size, i.e. $D = 617$.

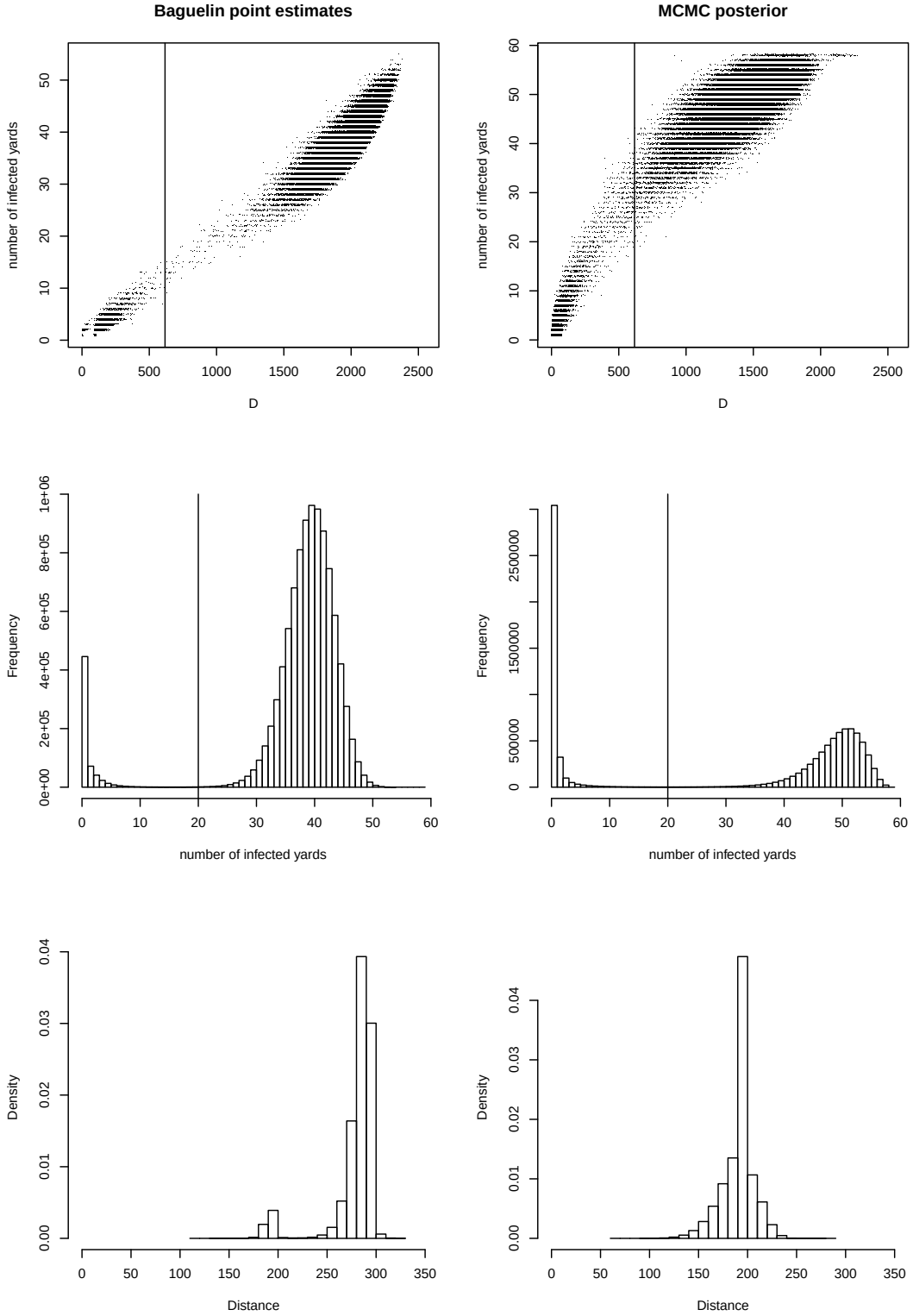


Figure 4.7: Summary plots for 10^7 simulations of an epidemic on the 58 yards data set, $\psi^{(3)}$, using a constant infectious period of $3\frac{1}{3}$ days. The left plots are for the fixed parameter values $(\lambda^L, \lambda^G) = (1.03, 0.015)$ from Baguelin et al. (2009) and the right uses posterior samples from our MCMC algorithm.

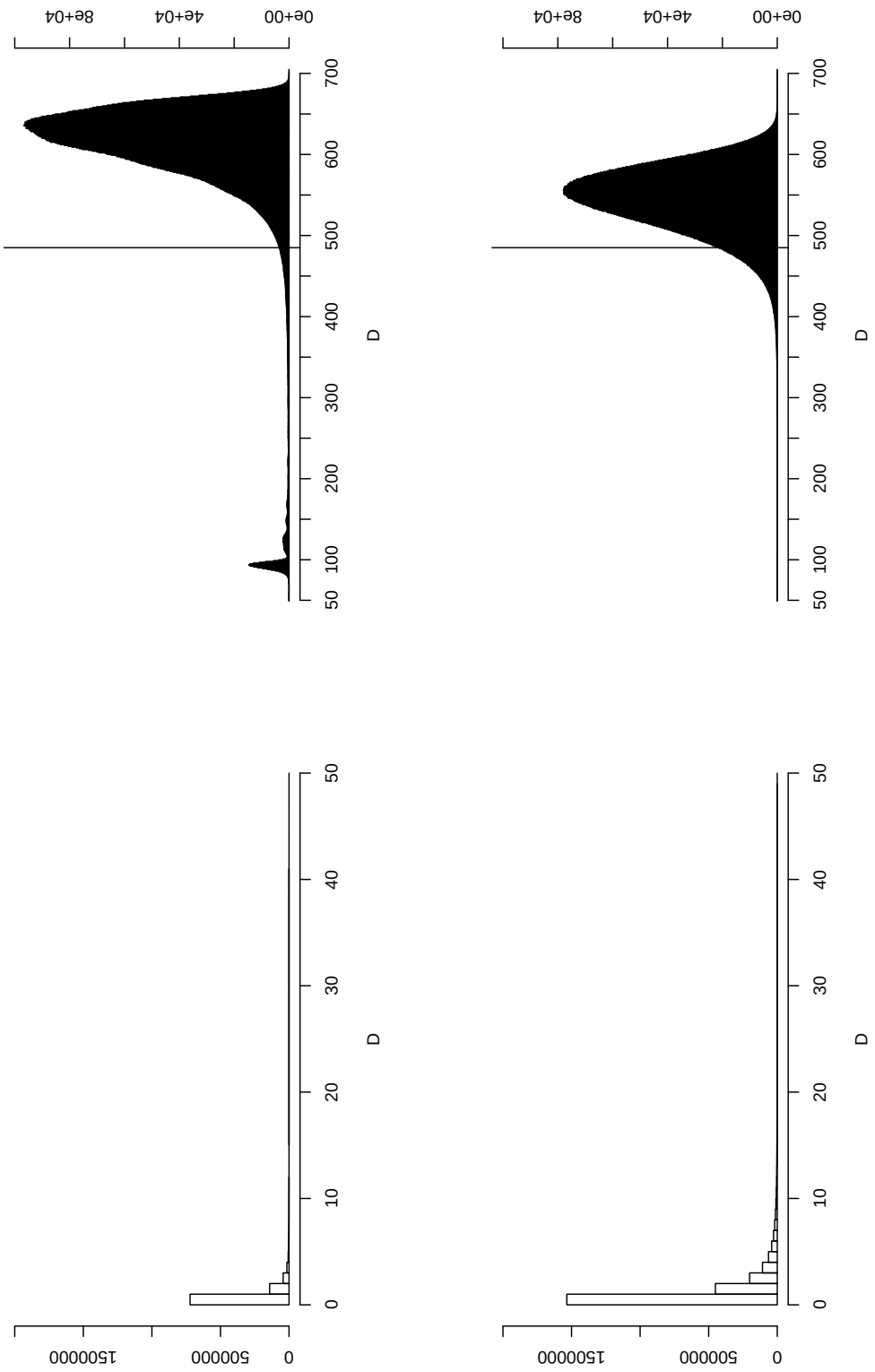


Figure 4.8: Total final size for 10^7 simulations of an epidemic on the 10 yards data set, $\psi^{(4)}$, using a constant infectious period of $3\frac{1}{3}$ days. The upper plot is for the fixed parameter values $(\lambda^L, \lambda^G) = (0.78, 0.017)$ from Baguelin et al. (2009) and the lower uses posterior samples from our MCMC algorithm (as shown in Figure 4.5). The vertical line corresponds to the observed total final size, i.e. $D = 485$.

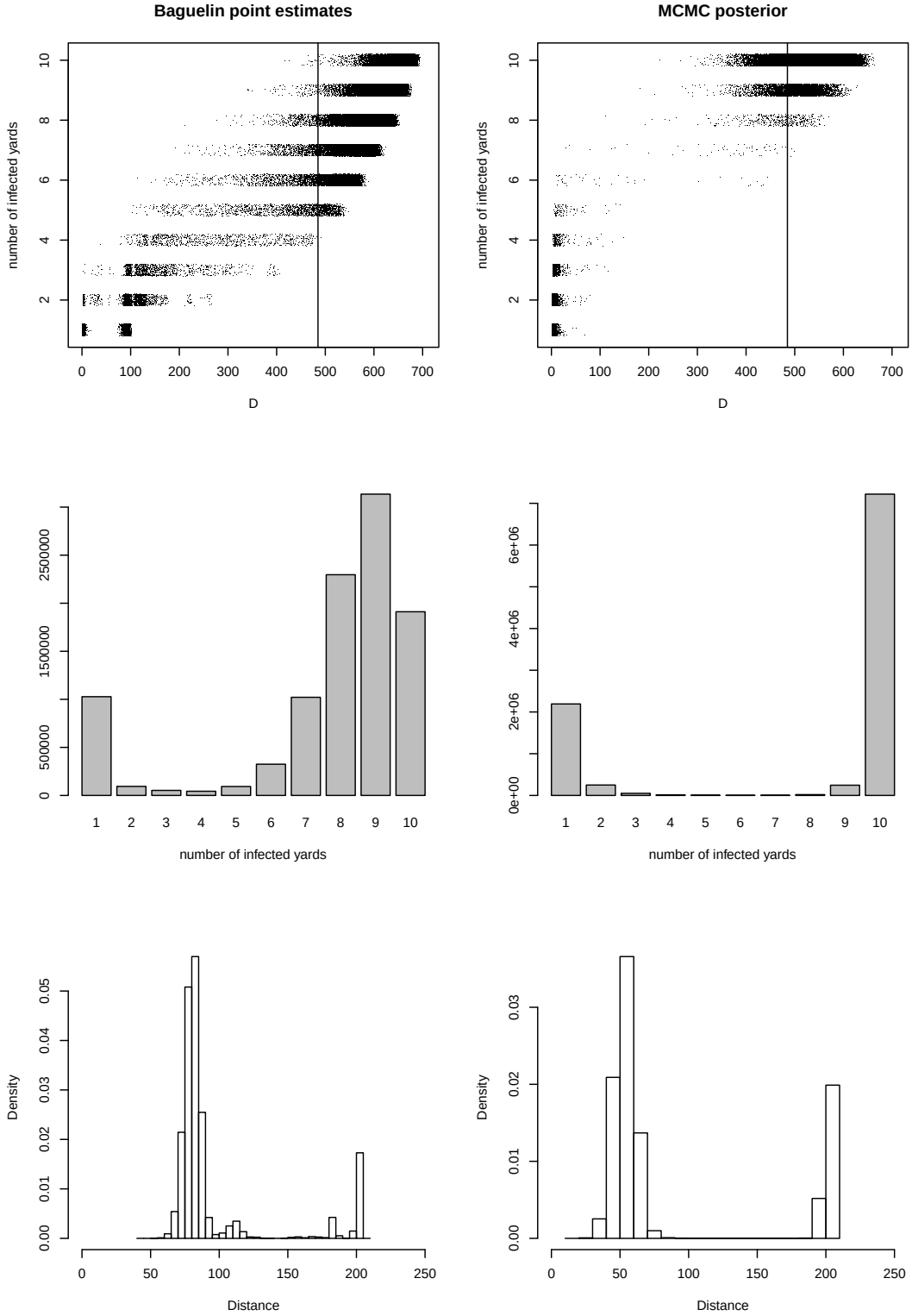


Figure 4.9: Summary plots for 10^7 simulations of an epidemic on the 10 yards data set, $\psi^{(4)}$, using a constant infectious period of $3\frac{1}{3}$ days. The left plots are for the fixed parameter values $(\lambda^L, \lambda^G) = (0.78, 0.017)$ from Baguelin et al. (2009) and the right are posterior samples from our MCMC algorithm (as shown in Figure 4.5).

4.6.5.3 Remarks

For the full data set, ignoring any effect from potential differences with Table 4.1, the estimates from Baguelin et al. (2009) using simple rejection Approximate Bayesian Computation (ABC) are different to our generation Markov Chain Monte Carlo (MCMC) method since they are drawing samples from different likelihoods.

Denote the imputed epidemic as z, i.e. the continuous time process or the generation path, then the marginal posterior density of the infection rates is estimated in Baguelin et al. (2009) by drawing samples from,

$$\pi\left(\lambda^L,\ \lambda^G,\ z \mid \Delta_{B_2}\left(D, D^{[i]}\right) = 0\ , \mathcal{M}\right),$$

whereas the MCMC chain draws samples from,

$$\pi\left(\lambda^L,\ \lambda^G,\ z \mid \theta,\ \mathcal{M}\right).$$

Here $\mathcal{M}$ denotes the one-type two-level model, with specificed infectious period, I, and infection rate matrix, $\Lambda(\lambda^L, \lambda^G)$.

This can be most clearly seen in the second row of Figure 4.7, where the ABC estimates generate realisations that infect a number of yards similar to the observed data, i.e. twenty infected yards, whereas the MCMC estimates do not. The choice of metric to use in an ABC algorithm is very important, in particular the Δ_{B_2}-distance does not capture the same structure of the observed data and hence produces estimates that are inconsistent with the MCMC method.

However, analysing the MCMC estimates shows that they too poorly reflect the observed data, in terms of the Δ_M-distance. It is clear from Table 4.1 and the literature

that the effect of vaccination must be accounted for in the model. As such, the MCMC estimates are probably more accurate, given the model $\mathcal{M}$, though this does not reflect the data.

The second analysis of Baguelin et al. (2009), as interpreted from their paper, is based on the ABC estimates and an unreasonable assumption. The aim in applying a two-level mixing model is to account for the interaction of global and local infections, thus to assume all ten yards are independent and seeded by a single initial infective is questionable in light of our estimate for the global infection rate. Further, estimating the local infection rate and then the global rate, conditional upon the local rate, in such a highly correlated setting is prone to erroneous results.

Chapter 5

Discussion And Further Work

In this thesis we have presented and applied a general approach to making Bayesian statistical inference, using Markov Chain Monte Carlo (MCMC), for final size data under the standard Susceptible-Infective-Removed (SIR) model; and any model whose final size is equal in distribution to the SIR model. Several extensions to the simple SIR model have been explicitly investigated, namely partially observed populations, multiple types of individuals and multiple levels of mixing within the population.

In Chapter 1 we introduced standard results and background theory for the stochastic SIR epidemic model of interest and the Bayesian paradigm for statistical inference. Extensions to the simple epidemic model are presented, which are subsequently incorporated into our approach. A general outline of MCMC is given, as well as a brief review of several adaptions.

Our approach is to use a representation of the epidemic process, its generation structure, to augments the parameter space. Imputing these additional unknown parameters, the size of each generation, allows us to form a likelihood that can be efficiently computed given the additional imputed data. This enables us to design efficient MCMC algorithms to make inference on the parameter models of interest. The form of the imputed data is investigated in Chapter 2, namely the generation representation of an epidemic process. Using generations, we impute the minimal information sufficient to represent the epidemic process in terms of final size analysis.

Using the generation representation, throughout Chapter 3 we develop a series of MCMC algorithms to perform analysis of final size data. We begin with the simplest case, consisting of a homogeneous population that is homogeneously mixing. Our approach is compared to standard estimates in the literature. The simple model is extended to include partially observed epidemics and we produce results comparable to Demiris and O'Neill (2005b). The representation is suited to such inference problems due to its characteristics, in that the MCMC algorithms run efficiently and quickly.

The series of algorithms conclude with a multi-type multi-level framework for fixed infectious periods. The algorithm is successfully applied to make parameter estimates for an outbreak of Influenza A (H3N2), presented by Longini et al. (1988), using a two-type two-level model with fixed infectious periods. The approach makes no assumptions on whether the epidemic process is above threshold nor does it use any approximations in the likelihood; the posterior density is approximated from the MCMC samples after convergence.

The MCMC algorithm uses the exact likelihood of a given epidemic path, thus for large final sizes the computational cost becomes greater. Conversely, approximate methods such as the Gaussian approximation to the final size distribution, see Demiris (2004), do not increase as greatly in time to compute. To overcome this, so that a large enough sample of the posterior may be taken in a reasonable time, we employ parallel computing and careful optimisation of the likelihood for each update step to improve efficiency.

The algorithms developed perform well, in terms of mixing and convergence of the MCMC chain, and the estimates are in agreement when compared to previous techniques in the literature. The run-time, using optimised likelihoods and parallel computing, is of a reasonable scale such that a sufficient number of samples from the posterior

can be generated in a reasonable time scale; from a few minutes for the simplest one-type one-level model up to a few hours for the Haber et al. (1988) and Longini et al. (1988) data sets (Tables 3.9 and 3.10), using fixed infectious periods.

Finally, we derive the likelihood for an arbitrary infectious period for each class of individual. Expressions are given in Chapter 4 for including the infectious periods as additional variables, to augment the parameter space further, or to integrate the infectious periods out from the likelihood. The latter approach is taken in applying the method to an outbreak of Equine Influenza (H3N8) at Newmarket (Table 4.1 and 4.3). The parameter estimates are compared to those of Baguelin et al. (2009), who analyse the same data using a type of Approximate Bayesian Computation (ABC), an alternative inference technique using an approximation to the likelihood briefly outlined in Section 1.3.4, for parameter inference.

For the Equine data set, the parameter estimates using our MCMC algorithm are substantially different to the estimates by Baguelin et al. (2009), our approach determining the outbreak was more globally driven. The parameter estimates for the local and global rates are negatively correlated, as is expected, thus given the higher global rate we estimate a correspondingly lower local infection rate. The disparity in estimates is explained in terms of the likelihoods being approximated. The method used by Baguelin et al. (2009) is a form of ABC and it would be interesting to investigate this method and the disparity in estimates further.

The non-temporal nature of final size data means it is difficult to make inference for certain extensions. In particular, our imputed generation representation does not support temporal information. For example, we cannot include explicit seasonally varying infectivity or susceptibility, population migration or threshold triggered interventions, e.g. vaccination or isolation of individuals at time of detection. However, the generation

method is well suited to models for partially observed final size epidemics, due to the minimal imputed information for the likelihood. The MCMC algorithms perform very well for these models, in terms of run-time, convergence and exploring the parameter space. Further investigation and applications would be interesting. For example, applying the partially observed generation method to the one and two type household data of Section 3.6, where the data are actually only a sample of the population. Also, as discussed in Section 4.6.4, a more suitable model is needed for the outbreak of Equine Influenza at Newmarket, many ideas are suggested and these could be investigated further.

In summary, the MCMC algorithms developed using the generation representation are a viable approach to inference for final size data. There are no restrictive assumptions to the inference, except the type of non-temporal models that can be fitted and some practical concerns for implementation of elaborate models in terms of their run-time.

Bibliography

Addy, C. L., Longini, I. M., and Haber, M. (1991). A generalized stochastic model for the analysis of infectious disease final size data. *Biometrics*, 47:961–974.

Anderson, R. M. and May, R. M. (1991). *Infectious Diseases of Humans; Dynamics and Control*. Oxford University Press.

Andersson, H. (1997). Epidemics in a population with social structures. *Math. Biosci.*, 140:79–84.

Andersson, H. (1999). Epidemic models and social networks. *Math. Sci.*, 24(2):128–147.

Andersson, H. and Britton, T. (2000). *Stochastic epidemic models and their statistical analysis*, volume 151 of *Lecture Notes in Statistics*. Springer-Verlag, New York.

Ayguad, E., Blainey, B., Duran, A., Labarta, J., Martinez, F., Martorell1, X., and Silvera, R. (2003). Is the schedule clause really necessary in OpenMP? In *OpenMP Shared Memory Parallel Programming: International Workshop on OpenMP Applications and Tools, WOMPAT 2003, Toronto, Canada, June 26-27, 2003. Proceedings*, volume 2716/2003, pages 147–159. Springer Berlin / Heidelberg.

Baguelin, M., Newton, J., Demiris, N., Daly, J., Mumford, J., and Wood, J. (2009). Control of equine influenza: scenario testing using a realistic metapopulation model of spread. *J R Soc Interface*. rsif.2009.0030v1-rsif.2009.0030.

Bailey, N. T. J. (1975). *The mathematical theory of infectious diseases and its applications*. Hafner Press [Macmillan Publishing Co., Inc.] New York, second edition.

Ball, F. G. (1983). The threshold behaviour of epidemic models. *J. Appl. Probab.*, 20(2):227–241.

Ball, F. G. (1986). A unified approach to the distribution of total size and total area under the trajectory of infectives in epidemic models. *Adv. in Appl. Probab.*, 18(2):289–310.

Ball, F. G. (1995). Coupling methods in epidemic theory. In *Epidemic models: Their structure and relation to data*. Publications of the Newton Institute, Cambridge University Press, Campbridge.

Ball, F. G., Britton, T., and O'Neill, P. D. (2002). Empty confidence sets for epidemics, branching processes and Brownian motion. *Biometrika*, 89(1):211–224.

Ball, F. G. and Clancy, D. (1993). The final size and severity of a generalised stochastic multitype epidemic model. *Adv. Appl. Prob.*, 25:721–736.

Ball, F. G. and Donnelly, P. (1995). Strong approximations for epidemic models. *Stochastic Process. Appl.*, 55(1):1–21.

Ball, F. G., Mollison, D., and Scalia-Tomba, G. (1997). Epidemics with two levels of mixing. *Ann. Appl. Probab.*, 7(1):46–89.

Ball, F. G. and Neal, P. (2002). A general model for stochastic SIR epidemics with two levels of mixing. *Math. Biosci.*, 180:73–102. John A. Jacquez memorial volume.

Ball, F. G. and Neal, P. J. (2008). Network epidemic models with two levels of mixing. *Math. Biosci.*, 212:69–87.

Barquero, N., Daly, J. M., and Newton, J. R. (2007). Risk factors for influenza infection in vaccinated racehorses: Lessons from an outbreak in Newmarket, UK in 2003. *Vaccine*, 25:7520–7529.

Baum, L. E. (1972). An inequality and associated maximization technique in statistical estimation for probabilistic functions of Markov processes. In *Inequalities, III (Proc. Third Sympos., Univ. California, Los Angeles, Calif., 1969; dedicated to the memory of Theordore S. Motzkin)*, pages 1–8. Academic Press, New York.

Baum, L. E., Petrie, T., Soules, G., and Weiss, N. (1970). A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. *Ann. Math. Statist.*, 41:164–171.

Beaumont, M. A., Zhang, W., and Balding, D. J. (2002). Approximate Bayesian Computation in Population Genetics. *Genetics*, 162(4):2025–2035.

Becker, N. G. (1989). *Analysis of Infectious Disease Data*. London: Chapman and Hall.

Berger, J. O. (1993). *Statistical Decision Theory and Bayesian Analysis (Springer Series in Statistics)*. Springer.

Bernardo, J. and Smith, A. (1994). *Bayesian Theory*. Wiley Series in Probability and Statistics. John Wiley and Sons Ltd.

Bernoulli, D. (1766). Essai d'une nouvelle analyse de la mortalité causée par la petite vérole et des advantages de l'inoculation pour la prévenir. *Histoire et Mémoires de l'Académie des Sciences*, 2:1–79.

Blum, M. (2009). Approximate bayesian computation: a non-parametric perspective. *ArXiv e-prints*. http://arxiv.org/abs/0904.0635.

Blum, M. G. B. and Tran, V. C. (2008). Approximate bayesian computation for epidemiological models: Application to the cuban hiv-aids epidemic with contact-tracing and unobserved infectious population. *ArXiv e-prints*. http://arxiv.org/abs/0810.0896.

Bollobás, B. (1985). *Random Graphs.* Academic Press Inc. (London) Ltd.

Bollobás, B. (1998). *Modern Graph Theory.* Springer-Verlag, New York.

Bondarenko, E. M. and Topchiĭ, V. A. (2001). Estimates for the expectation of the maximum of a critical Galton-Watson process on a finite interval. *Sibirsk. Mat. Zh.*, 42(2):249–257, i.

Britton, T. (1998). Estimation in multitype epidemics. *Journal of the Royal Statistical Society. Series B (Statistics in Society)*, 60(4):663–679.

Britton, T. and O'Neill, P. D. (2002). Bayesian inference for stochastic epidemics in populations with random social structure. *Scandinavian Journal of Statistics*, 29(3):375–390.

Chapman, B., Jost, G., and van der Pas, R. (2007). *Using OpenMP.* MIT Press.

Clancy, D. and O'Neill, P. D. (2007). Exact bayesian inference and model selection for stochastic models of epidemics among a community of households. *Scandinavian Journal of Statistics*, 34(2):259–274.

Demiris, N. (2004). *Bayesian Inference for Stochastic Epidemic Models using Markov chain Monte Carlo Methods.* PhD thesis, School of Mathematical Sciences, University of Nottingham.

Demiris, N. and O'Neill, P. (2005a). Bayesian inference for stochastic multitype epidemics in structured populations via random graphs. *J. R. Statist. Soc. B*, 67(5):731–745.

Demiris, N. and O'Neill, P. D. (2005b). Bayesian inference for stochastic epidemic models with two levels of mixing. *Scandinavian Journal of Statistics*, 32(2):265–280.

Devijver, P. A. (1985). Baum's forward-backward algorithm revisited. *Pattern Recognition Letters.*

Diekmann, O. and Heesterbeek, J. A. P. (2000). *Mathematical Epidemiology of Infectious Diseases: Model Building, Analysis and Interpretation*. Wiley Series in Mathematical & Computational Biology. WileyBlackwell, new edition edition.

Diggle, P. J. and Gratton, R. J. (1984). Monte carlo methods of inference for implicit statistical models. *Journal of the Royal Statistical Society. Series B (Methodological)*, 46(2):193–227.

Drmota, M. and Gittenberger, B. (1997). On the profile of random trees. *Random Structures Algorithms*, 10(4):421–451.

Dwass, M. (1969). The total progeny in a branching process and a related random walk. *J. Appl. Prob.*, 6:682–686.

Galassi, M., Davies, J., Theiler, J., Gough, B., Jungman, G., Booth, M., and Rossi, F. (2003). *Gnu Scientific Library: Reference Manual*. Network Theory Ltd.

Gelfand, A. E. and Smith, A. F. M. (1990). Sampling-based approaches to calculating marginal densities. *Journal of the American Statistical Association*, 85(410):398–409.

Gelman, A., Carlin, J. B., Stern, H. S., and Rubin, D. B. (2003). *Bayesian Data Analysis*. Texts in statistical science. Chapman & Hall, second edition.

Geman, S. and Geman, D. (1984). Stochastic relaxation, gibbs distributions and the bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(6):721–741.

Gibson, G. J. and Renshaw, E. (1998). Estimating parameters in stochastic compartmental models using Markov chain methods. *Math Med Biol*, 15(1):19–40.

Gilks, W. R., Richardson, S., and Spiegelhalter, D. J. (1996). *Markov chain Monte Carlo in practice*. Interdisciplinary Statistics. Chapman & Hall, London.

Gittenberger, B. (1998). Convergence of branching processes to the local time of a Bessel process. In *Proceedings of the Eighth International Conference "Random Structures and Algorithms" (Poznan, 1997)*, volume 13, pages 423–438.

Good, I. (1949). The number of individuals in a cascade process. *Proc. Camb. Phil. Soc.*, 45:360–363.

Haario, H., Saksman, E., and Tamminen, J. (2001). An adaptive metropolis algorithm. *Bernoulli*, 7(2):223–242.

Haber, M., Longini, I. M., and Cotsonis, G. A. (1988). Models for the statistical analysis of infectious disease data. *Biometrics*, 44(1):163–173.

Harary, F., Norman, R. Z., and Cartwright, D. (1965). *Structural Models: An Introductory to the Theory of Directed Graphs.* John Wiley & Sons.

Hastings, W. K. (1970). Monte carlo sampling methods using markov chains and their applications. *Biometrika*, 57(1):97–109.

Hayakawa, Y., O'Neill, P., Upton, D., and Yip, P. (2003). Bayesian inference for an epidemic model with uncertain numbers of susceptibles of several types. *Aust. N.Z. J. Stat.*, 45:491–502.

Jagers, P. (1975). *Branching Processes with Biological Applications.* John Wiley & Sons.

Jewell, C. P., Kypraios, T., Neal, P., and Roberts, G. O. (2009). Bayesian analysis for emerging infectious diseases. *Bayesian Analysis*, 4:465–495.

Kass, R. E. and Raftery, A. E. (1995). Bayes factors. *Journal of the American Statistical Association*, 90(430):773–795.

Kenah, E. and Robins, J. M. (2007). Second look at the spread of epidemics on networks. *Phys. Rev. E*, 76:036113.

Kendall, D. (1956). Deterministic and stochastic epidemics in closed populations. In *Proc. 3rd Berkeley Symp. Math. Statist. Prob*, volume 4, pages 149–165.

Kerbashev, T. B. (1999). On the maximum of a branching process conditioned on the total progeny. *Serdica Math. J.*, 25(2):141–176.

Kermack, W. O. and McKendrick, W. G. (1927). A contribution to the mathematical theory of epidemics. In *Proceedings of the Royal Society of London, Series A 115*, pages 700–721.

Kypraios, T. (2007). *Efficient Bayesian Inference for Partially Observed Stochastic Epidemics and A New Class of Semi-Parametric Time Series Models*. PhD thesis, Department of Mathematics and Statistics, Lancaster University.

Lindvall, T. (1976). On the maximum of a branching process. *Scand. J. Statist.*, 3(4):209–214.

Longini, I. M., Koopman, J. S., Haber, M., and Cotsonis, G. A. (1988). Statistical inference for infectious diseases: risk-specific household and community transmission parameters. *Am. J. Epidem.*, 128:845–859.

Ludwig, D. (1974). *Stochastic Population Theories*, volume 3 of *Lecture Notes in Biomathematics*. Springer-Verlag, Berlin.

Ludwig, D. (1975). Final size distributions for epidemics. *Math. Biosci.*, 23(1):33–46.

MacDonald, I. L. and Zucchini, W. (1997). *Hidden Markov and other models for discrete-valued time series*, volume 70 of *Monographs on Statistics and Applied Probability*. Chapman & Hall, London.

Marjoram, P., Molitor, J., Plagnol, V., and Tavare, S. (2003). Markov chain monte carlo without likelihoods. *Proc Natl Acad Sci U S A*, 100(26):15324–15328.

McKendrick, A. G. (1926). Applications of mathematics to medical problems. *Proceedings of the Edinburgh Mathematical Society*, 40:98–130.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., and Teller, E. (1953). Equation of state calculations by fast computing machines. *Journal of Chemical Physics*, 21(6):1087–1092. This article introduces the Metropolis algorithm, which the journal *Computing in Science and Engineering* cited in the top 10 algorithms having the "greatest influence on the development and practice of science and engineering in the 20th Century.".

Mode, C. J. (1971). *Multitype Branching Processes: Theory and Applications*. American Elsevier, New York.

Morris, C. N. (1983). Parametric empirical bayes inference: Theory and applications. *Journal of the American Statistical Association*, 78(381):47–55.

Neal, P. and Roberts, G. (2005). A case study in non-centering for data augmentation: Stochastic epidemics. *Statistics and Computing*, 15(4):315–327.

Neal, P. J. and Roberts, G. O. (2004). Statistical inference and model selection for the 1861 Hagelloch measles epidemic. *Biostat*, 5(2):249–261.

Newman, M. E. J. (2002). Spread of epidemic disease on networks. *Phys. Rev. E*, 66(1):016128.

Newton, J. R., Daly, J. M., Spencer, L., and Mumford, J. A. (2006). Description of the outbreak of equine influenza (H3N8) in the United Kingdom in 2003, during which recently vaccinated horses in Newmarket developed respiratory disease. *Veterinary Record*, 158:185–192.

O'Neill, P. (2009). Bayesian inference for stochastic multitype epidemics in structured populations using sample data. Under revision for Biostatistics.

O'Neill, P. D. (2002). A tutorial introduction to Bayesian inference for stochastic epidemic models using Markov chain Monte Carlo methods. *Maths. Bio.*, 180:103–114.

O'Neill, P. D. (2003). Perfect simulation for Reed-Frost epidemic models. *Statistics and Computing*, 13(1):37–44.

O'Neill, P. D., Balding, D. J., Becker, N. G., Eerola, M., and Mollison, D. (2000). Analyses of infectious disease data from household outbreaks by Markov chain Monte Carlo methods. *J. Roy. Statist. Soc. Ser. C*, 49(4):517–542.

O'Neill, P. D. and Becker, N. G. (2001). Inference for an epidemic when susceptibility varies. *Biostat*, 2(1):99–108.

O'Neill, P. D. and Marks, P. J. (2005). Bayesian model choice and infection route modelling in an outbreak of Norovirus. *Statistics in Medicine*, 24(24):2011–2024.

O'Neill, P. D. and Roberts, G. O. (1999). Bayesian inference for partially observed stochastic epidemics. *Journal of the Royal Statistical Society. Series A (Statistics in Society)*, 162(1):121–129.

Ore, O. (1967). *Theory Of Graphs*, volume 38 of *Colloquium Publications*. American Mathematical Society.

Panaretos, V. M. (2007). Partially observed branching processes for stochastic epidemics. *Journal of Mathematical Biology*, 54(5):645–668.

Papaspiliopoulos, O., Roberts, G. O., and Skøld, M. (2003). Non-centred parameterisations for hierarchical models and data augmentation. In *Bayesian Statistics 7*, pages 307–326. Oxford University Press, Oxford.

Park, A. W., Wood, J. L. N., Daly, J. M., Newton, J. R., Glass, K., Henley, W., Mumford, J. A., and Grenfell, B. T. (2004). The effects of strain heterology on the

epidemiology of equine influenza in a vaccinated population. *Proc. R. Soc. Lond. B*, 271(1548):1547–55.

Parzen, E. (1964). *Stochastic Processes*. Holden-Day Series in Probability and Statistics. Holden-Day, San Francisco.

Pellis, L., Ferguson, N. M., and Fraser, C. (2008). The relationship between real-time and discrete-generation models of epidemic spread. *Math Biosci*, 216(1):63–70.

Plagnol, V. and Tavare, S. (2004). Approximate Bayesian Computation and MCMC. In Niederreiter, H., editor, *Monte Carlo and Quasi-Monte Carlo Methods 2002*, pages 99–114.

Quinn, M. J. (2004). *Parallel Programming in C with MPI and OpenMP*. McGraw-Hill.

Rabiner, L. R. (1989). A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286.

Robert, C. P. and Casella, G. (1999). *Monte Carlo Statistical Methods*. Springer Texts in Statistics. Springer-Verlag, New York.

Roberts, G. O., Gelman, A., and Gilks, W. R. (1997). Weak convergence and optimal scaling of random walk metropolis algorithms. *Annals of Applied Probability*, 7:110–120.

Roberts, G. O. and Rosenthal, J. S. (2007). Coupling and ergodicity of adaptive markov chain monte carlo algorithms. *J. Appl. Probab.*, 44(2):458–475.

Scalia-Tomba, G. (1990). On the asymptotic final size distribution of epidemics in hetrogeneous populations. In *Lecture notes in Biomathematics: Stochasitc processes in epidemic theory*, pages 189–196. Springer, New York.

Sherlock, C., Fearnhead, P., and Roberts, G. (2009). The random walk Metropolis: linking theory and practice through a case study. submitted for publication.

Sousa, V. M., Fritz, M., Beaumont, M. A., and Chikhi, L. (2009). Approximate Bayesian Computation (ABC) Without Summary Statistics: The Case of Admixture. *Genetics*, pages genetics.108.098129+.

Streftaris, G. and Gibson, G. J. (2004). Bayesian inference for stochastic epidemics in closed populations. *Statistical Modeling*, 4(1):63–75.

Toni, T., Welch, D., Strelkowa, N., Ipsen, A., and Stumpf, M. P. (2009). Approximate bayesian computation scheme for parameter inference and model selection in dynamical systems. *Journal of The Royal Society Interface*, 6:187–202.

Uspensky, J. V. (1937). *Introduction to Mathematical Probability*. McGraw-Hill, New York.

Weiner, H. (1984). Moments of the maximum in a critical branching process. *J. Appl. Probab.*, 21(4):920–923.

Williams, T. (1971). An algebraic proof of the threshold theorem for the general stochastic epidemic. *Adv. Appl. Prob.*, 3:223–223. Nonlinearity in biology and medicine (Los Alamos, NM, 1987).