Inference on historical Ebola outbreaks using hierarchical models: a particle filtering approach

Dylan Morris

June 18, 2021

Thesis submitted for the degree of
Master of Philosophy
in
Applied Mathematics
at The University of Adelaide
Faculty of Engineering, Computer and Mathematical Sciences
School of Mathematical Sciences

THE UNIVERSITY
of ADELAIDE

Contents

Signed Statement **v**

Acknowledgements **vii**

Abstract **ix**

1 Introduction **1**
1.1 Thesis Outline 5

2 Technical Background **7**
2.1 Continuous-time Markov Chains 7
2.2 Epidemic modelling 9
2.2.1 The SIR model 10
2.3 Bayesian inference 13
2.3.1 The likelihood 17
2.4 Particle filters 18
2.4.1 Bootstrap particle filter 18
2.4.2 Alive particle filter 20
2.4.3 Importance sampling 21
2.4.4 Importance sampling in particle filters 22
2.5 The SPSA algorithm 26

3 MAP estimation using noisy likelihood estimates **31**
3.1 The MAP estimation problem 31
3.2 Tuning particle filters 32
3.3 MAP estimation as a minimisation problem 33
3.4 SPSA for MAP estimation 34
3.5 Application to the SIR model 43
3.5.1 Choosing the key parameters 44
3.5.2 Choosing the number of iterations 48
3.5.3 Choosing the number of particles 49

3.5.4 Sensitivity to the initial point . . . 49
3.6 Application to the SEIAR model . . . 56
3.6.1 Model details . . . 56
3.6.2 Data . . . 58
3.6.3 Search setup . . . 58
3.6.4 Results . . . 59
3.7 Summary . . . 63

4 Importance sampling for multiple observations of a single outbreak 65
4.1 Case study: 1995 DRC Ebola outbreak . . . 65
4.2 Model . . . 68
4.2.1 SIR . . . 71
4.2.2 SEIR with partial detection of onset and removal events . . . 73
4.3 Inference on a simulated outbreak . . . 77
4.4 Inference on the 1995 outbreak . . . 85
4.5 Summary . . . 91

5 Hierarchical modelling of Ebola 93
5.1 The data . . . 93
5.2 Epidemic model development . . . 97
5.3 Independent inferences . . . 100
5.4 Hierarchical modelling process . . . 109
5.5 Results . . . 116
5.6 Summary . . . 126

6 Discussion and further research 127
6.1 SPSA and MAP estimation . . . 127
6.2 Particle filtering and Ebola . . . 132

A Algorithms 143

B Posterior simulations for independent inferences 147

Bibliography 151

Signed Statement

I certify that this work contains no material which has been accepted for the award of any other degree or diploma in my name, in any university or other tertiary institution and, to the best of my knowledge and belief, contains no material previously published or written by another person, except where due reference has been made in the text. In addition, I certify that no part of this work will, in the future, be used in a submission in my name, for any other degree or diploma in any university or other tertiary institution without the prior approval of the University of Adelaide and where applicable, any partner institution responsible for the joint-award of this degree.

I give permission for the digital version of my thesis to be made available on the web, via the University's digital research repository, the Library Search and also through web search engines, unless permission has been granted by the University to restrict access for a period of time.

I acknowledge the support I have received for my research through the provision of an Australian Government Research Training Program Scholarship.

Signed: ..... Date: .....

Acknowledgements

First and foremost, I would like to give my thanks to my supervisors Andrew Black and Joshua Ross for all they have done over these past two years. They have both been excellent in providing me support through professional and personal matters and I appreciate all our insightful discussions over the course of my MPhil. I would also like to thank the Phoenix support team and acknowledge that the work in this thesis was supported with supercomputing resources provided by the Phoenix HPC service at the University of Adelaide.

Thank you to my friends and family for their support over the course of the last two years. In particular I would like to thank my mother, Lea, my partner, Christie, and my father, Steve, for providing me with unending amounts of support and encouragement. Lastly, I would like to thank my step father, Julian, who provided me with such an interesting perspective of the world around me.

Abstract

Particle filters are commonly used to estimate the likelihood for epidemic models when it is analytically intractable. These methods marginalise over missing data and do not suffer from the scaling issues present in traditional *data-augmented Markov chain Monte Carlo* methods. In this thesis we present a particle filtering methodology which extends upon recent advances in the field to simulate realisations of a process which are consistent with observations of two events from the same outbreak, such as symptom onsets and recoveries. This particle filtering approach is used in a *particle marginal Metropolis-Hastings* (pmMH) algorithm to fit a hierarchical model to four outbreaks of Ebola simultaneously for the first time. We estimated R_0 above 1 for all four outbreaks (as expected by the threshold theorem), with three of the outbreaks having values above 3. Our results also indicated that transmission began to reduce before the implementation of major intervention measures, which may be due to changes in community awareness. An additional area of work in this thesis relates to the efficiency of pmMH methods. Mixing of the overall Markov chain is crucial to the efficiency of a pmMH method, and is controlled by the variance in the log-likelihood estimates from the particle filter at the *maximum a posteriori* (MAP). This variance is in turn controlled by the number of particles used. We develop a more sophisticated methodology for estimating the MAP when only noisy estimates of the log-likelihood are available. This enables *a priori* tuning of the inference methods and the possibility of automating the tuning process.

Chapter 1

Introduction

Wide spread diseases, whether they be epidemics or pandemics, have been a recurring feature in human history. The Black Death in the fourteenth century is one such disease that is still discussed and studied to this day [39]. In more recent times we have seen major outbreaks of COVID-19, influenza, HIV/AIDS and Ebola [38, 47, 61, 71]. The impact of each of these infectious diseases has had widespread, unprecedented and unexpected impacts. This has particularly been the case in the current ongoing pandemic. From the Black Death to COVID-19 we have tried various techniques to quell the impact and control the spread. Over this time we have gotten better at handling outbreaks through improved understanding of the dynamics of these diseases. While we have improved our ability to deal with infectious diseases, we still have a long way to go.

The difficulty with infectious diseases is while there are many commonalities between them, the dynamics of a given outbreak are unique. There is not one model fits all and there is increasing importance for development of efficient methods to conduct inference on emerging diseases. Two outbreaks of the same disease in different communities could lead to vastly different outcomes [39]. Understanding the factors which contribute to different outbreaks can provide insights for potential public health measures. By studying past diseases we can attempt to avoid past mistakes and provide better strategies for managing future outbreaks. Historical outbreaks also act as a testing ground for new computational methods and these methods can be used to analyse new and emerging diseases.

Ebola virus disease[1] (EVD), is a viral hemorrhagic fever of humans and other primates caused by ebolaviruses [13, 41]. The most severe of these ebolaviruses is the Zaire strain, which was first identified in 1976 and has seen case fatality rates of 60–80% [14, 59, 61]. Ebola is responsible for a range of symptoms which vary in severity with common

[1] Also commonly referred to as Ebola hemorrhagic fever (EHF) or simply Ebola

symptoms including high fever, vomiting, diarrhoea, muscle pain, headaches, and bleeding [13, 59, 61]. The most devastating outbreak of Ebola to date was a global scale outbreak in 2014–2016 which saw more than 28,000 cases [20]. A subsequent outbreak of Ebola which began in 2018 and was deemed over in June 2020 was the largest outbreak in the Democratic Republic of the Congo (DRC) to date and saw over 3,000 confirmed cases with over 2,000 deaths [20]. In the shadow of COVID-19, the 11th (major) outbreak of Ebola (in the DRC) occurred in June 2020. This outbreak was deemed to be over in November that same year and there were 130 confirmed cases [20]. As of the time of submission there is an ongoing outbreak of Zaire strain of Ebola in the DRC and an emerging outbreak of the same strain in Guinea, the first outside of the DRC in five years. These outbreaks demonstrate the need to understand how Ebola is able to persist in communities, and provide potential intervention strategies which can be used in conjunction with the recently developed vaccines [15].

Mathematical modelling enables us to use deterministic and/or stochastic frameworks to fit *epidemic models* to outbreak data. Epidemic modelling is one of the oldest areas of mathematical biology, which has enabled us to develop understanding of the mechanisms that drive outbreaks. This improved understanding enables more informed public health decisions to be made [2, 39]. A common epidemic model is a *compartmental model* which divides the population up into compartments which reflect the status of individuals over the course of the outbreak [1, 2, 39]. Events—such as infections or recoveries—correspond to individuals transitioning between these compartments [1]. These models can be made more complex by incorporating additional compartments and transitions. This added model complexity enables more complex dynamics to be replicated which can more appropriately reflect the true dynamics of a particular outbreak [1, 39]. In this thesis we will be accounting for stochasticity present in outbreaks by working with stochastic compartmental models. Stochasticity is important in the modelling of outbreaks in small populations as individual events can have a strong impact on the outcome of an outbreak [1]. A stochastic compartmental model can be described in terms of a *continuous-time Markov chain* (CTMC) [39]. CTMC models capture the discrete nature of individuals in the population as well as the inherent randomness in outbreaks.

Well documented outbreaks of Ebola, like the 1976 outbreak in Yambuku and the 1995 outbreak in Kikwit, can be used to gain some insights into the dynamics of Ebola [14, 41, 59, 61]. Models have previously been fitted to these outbreaks which attempt to quantify the effectiveness of intervention measures and/or changes in transmission as a result of community awareness [14, 73, 45, 51, 61]. The understanding of the effectiveness of previous interventions provides us with knowledge of suitable secondary intervention strategies alongside current measures—like vaccines—to help reduce transmission chains and lead to lower total case counts in future outbreaks.

We use a collection of Ebola outbreak data sourced from the *Direction de Lutte contre*

la Maladie (DLM) collated for the first time, in Rosello *et al.* [61]. The DLM is the public body in charge of containing EVD outbreaks in the DRC [61]. These data contain a detailed line list of six of the major outbreaks (five instances of the Zaire strain, one of the Bundibugyo) and one minor outbreak [61]. The line list is a dataset featuring dates (where available) of different key events for each individual during the outbreak. These can include hospital admission dates, symptom onset dates and removal dates, either through recovery and hospital discharge, or death. This provides an opportunity to study independent outbreaks of Ebola in different provinces of the DRC in the same model, pooling the information stored across the different outbreaks. This enables us to gain better understanding of the intervention measures in each outbreak. In this thesis we look to use epidemic models to quantify the change in transmission as a result of interventions and changes in community interactions. A difficulty in analysis of such models is that current inference methods scale poorly due to the sheer amount of data. We focus on four of the outbreaks (Yambuku, Kikwit, Mweka (2008) and Boende) in this thesis as the data for these outbreaks are of high quality. In order to analyse the four outbreaks of Ebola in a hierarchical framework we require a methodology for conducting efficient inference [61].

Insights into the dynamics of an outbreak can be gained through inferring the parameters which govern an epidemic model. Quantities of interest are dependent on the choice of model and can be used to represent various dynamics of the outbreak. These same quantities enable modellers to inform governing bodies as to appropriate courses of action which can involve interventions of various forms [14, 39]. One common and informative quantity is the basic reproductive number R_0, defined as the average number of infections from a single infectious individual in an otherwise entirely susceptible population [1, 39]. This number can be used to determine whether an outbreak is likely to invade the population. If $R_0 \leq 1$ then the disease will fadeout and if $R_0 > 1$ then there is a chance that the infection will invade the population and lead to an outbreak [39].

A Bayesian framework is used as it enables us to incorporate known information about the spread of infectious diseases through the prior distributions [23, 27]. This can often prove to be useful for parameters which may be unidentifiable from the data. The main difficulty in performing inference using typical epidemic models arises in the calculation of the likelihood [3, 52]. Typically this means that the posterior distribution is analytically intractable and so we must resort to *Markov chain Monte Carlo* (MCMC) methods. MCMC provides us with a framework of sampling from analytically intractable distributions, and one of the most common methods which we use throughout this thesis is the *Metropolis-Hastings* algorithm [28, 27, 35, 53].

Previous studies of Ebola have employed *Data-augmented MCMC* (DA-MCMC) methodology which seek to infer the missing event times alongside the parameters as part of the overall Markov chain, allowing for direct calculation of the likelihood [45, 51, 56]. DA-

MCMC is an exact method which can be considered as integrating over the missing data and the parameters [70]. The main issues with DA-MCMC methods are that assessing whether the chains have converged can prove difficult and the algorithm is known to scale poorly [45, 52]. The scaling is a particularly relevant issue, as it makes the method poor for conducting inference on multiple datasets as a result of the full augmented space being high-dimensional, which can lead to prohibitively slow mixing [52].

An alternative approach to DA-MCMC is to take advantage of the ease simulation of a CTMC epidemic model through the *stochastic simulation algorithm* (SSA) [21]. *Sequential Monte Carlo* (SMC) methods leverage simulation to estimate the state of an epidemic and as a by-product of this process, obtain an unbiased estimate of the likelihood [21, 54, 64]. In contrast to DA-MCMC integrating over the missing data as part of the Markov chain, SMC can be considered as marginalising over the missing data in estimating the likelihood [3, 52]. *Pseudo-marginal Metropolis-Hastings* (PMMH) methods exploit the estimate of the likelihood obtained from these methods to target the exact posterior distribution of the parameters [3].

Particle filters are a commonly used SMC method which rely on the sequential filtering of particles to marginalise over latent states [3, 22, 52]. When used in a PMMH method we refer to this as *particle-marginal Metropolis Hastings* (pmMH) [3]. Simpler particle filters–like the bootstrap particle filter and alive particle filters–suffer from issues that stem from trying to produce realisations consistent with the data when there is little noise in the observations [3, 8, 33, 36, 52]. The bootstrap filter is particularly naive in that it just simulates particles without taking into account the observation. This often means the algorithm struggles with producing realisations consistent with rare events, and the number of particles required to produce low variance estimates of the likelihood as a result of this is usually quite large [8, 36].

Recently, importance sampling has been used in particle filters to alleviate the issues present in simpler particle filters. Importance sampling enables us to simulate realisations which are consistent with the observed data, without introducing too much computational expense [8, 52]. Particle filters which use importance sampling have been shown to enable more efficient inference routines by maximising the effective sample size per unit of computation time [8]. In this thesis we look to extend the work of [8], to design simulation algorithms capable of producing realisations which are consistent with observations of more than one event in the same outbreak. This amount of data is common for diseases with high fatality rates like Ebola where the number of deceased individuals (due to Ebola) is tracked, as well as the number of symptomatic individuals [41, 61]. This level of surveillance is often necessary due to individuals being infectious following death and prior to burial [12, 14]. These importance sampling based particle filters provide a methodology for carrying out efficient inference and enable inference on multiple outbreaks simultaneously, something which DA-MCMC approaches do not facilitate.

A key finding from fitting the hierarchical model was that changes in transmission occurred prior to the major intervention measures across all outbreaks. It appears likely that the growing concern among members of community led to reduced attendance at traditional burials and a reduced number of hospital visits [14]. Our analysis also confirmed some reported estimates from previous studies relating to the average latent and infectious periods, estimated at 6.1 and 8.8 days respectively [14, 73]. We also found that the Yambuku outbreak had a noticeably higher R_0 in comparison to the other outbreaks. This appears to be consistent with known knowledge of hospital practices in 1976, whereby syringes were shared between patients in the Yambuku mission hospital [14, 61].

The efficiency of pmMH is dependent on the mixing of the chains, which is in turn dependent on the number of particles used in the particle filter [8, 57, 65]. The optimal number of particles required can be chosen based on the variance in the likelihood estimates (obtained by a particle filter) at or near the *maximum a posteriori* (MAP) [22, 65]. Ensuring the variance lies within reasonably tight bounds is critical to obtaining optimal performance [22, 65]. The process of obtaining the MAP for *a priori* tuning of the number of particles used in a particle filter is not well defined. Most conventional methods rely on algorithms which adopt a Metropolis-Hastings methodology (simulated annealing and kernel density estimation following pilot runs of pmMH) and hence lead to all the well known tuning considerations when working with such algorithms [29]. Furthermore these algorithms tend to be slow and can require some *a priori* knowledge of the parameter space as well as understanding of some additional factors (such as the temperature and cooling schedule for simulated annealing) [69]. In this thesis we develop an alternative methodology for MAP estimation. This methodology relies solely on using noisy estimates of the unnormalised log-posterior—such as those obtained through a particle filter—to search in an approximate steepest descent direction and estimate the MAP [69]. The MAP estimate can then be used in the *a priori* tuning of the particle filter, which enables us to decouple this aspect of the tuning process from the tuning of the proposal distribution.

1.1 Thesis Outline

Chapter 2 outlines the technical background for the work in this thesis. In Chapter 3 the methodology of the MAP estimation routine is explored and then rigorously tested on an example of the SIR model. The chapter concludes with the algorithm being applied to estimation of the MAP of a more complex model with a high dimensional parameter space.

Chapter 4 focuses on extending the importance sampling techniques in [8] to produce

realisations which are consistent with multiple observations of the same outbreak. This methodology is then used to conduct inference on a simulated outbreak of Ebola, as well as the 1995 outbreak of Ebola in the Democratic Republic of the Congo. The results are then compared against previously published estimates in the literature.

Chapter 5 takes the methodology from Chapter 4 and extends upon it, by conducting inference on a hierarchical model of four independent outbreaks of the Zaire strain of ebolavirus. This section uses a hierarchical framework to pool the information in the more informative datasets to help fit the model to the less informative datasets. We compare the results of the hierarchical fit against fitting the model to each outbreak independently.

Finally, Chapter 6 is a discussion and outlines areas of future research. The discussion and future research sections are divided into considerations for the search algorithm and Ebola studies, respectively. The first part of the chapter discusses the primary contributions of the work in this thesis, and seeks to highlight some limitations of the simultaneous-perturbation-stochastic approximation (SPSA) method and the importance sampling based particle filters. The future work section focuses primarily on providing potential solutions to some of these issues.

Chapter 2

Technical Background

This chapter will provide an overview of the technical background and literature relevant to this thesis. The topics will include continuous-time Markov chains, epidemic models, Bayesian inference, sequential Monte Carlo methods, particle filtering, and stochastic optimisation.

2.1 Continuous-time Markov Chains

Throughout this thesis we will formulate all stochastic models as *continuous-time Markov chains* (CTMCs). A CTMC is a stochastic process $\{X(t)\}_{t\geq 0}$ that evolves in continuous time and satisfies the Markov property. A continuous-time stochastic process $\{X(t)\}_{t\geq 0}$ on a countable state space $\mathcal{S}$ is a CTMC if and only if,

$$\Pr(X(t+s)=j|X(u)=k,X(s)=i,u<s)=\Pr(X(t+s)=j|X(s)=i) \quad \forall s,t,u\geq 0, i,j,k\in\mathcal{S}.$$

A CTMC is time homogeneous if and only if,

$$p_{ij}(t):=\Pr(X(t+s)=j|X(s)=i)=\Pr(X(t)=j|X(0)=i), \quad \forall s,t\geq 0, i,j\in\mathcal{S}.$$

The $p_{ij}(t)$ are the transition probabilities of the CTMC $\{X(t)\}_{t\geq 0}$ and denote the probability of transitioning from state i to state j in elapsed time $t\geq 0$. These probabilities can be written in matrix form,

$$P(t)=\left(p_{ij}(t):i,j\in\mathcal{S}\right),$$

which is referred to as the transition function. In this thesis we will assume all our CTMC models are time-homogeneous and defined on a finite state space.

With the transition function defined, we can now define the infinitesimal generator or "Q-matrix" of a CTMC which denotes the instantaneous transition rates of the process. The generator matrix, $Q = (q_{ij} : i, j \in \mathcal{S})$, of a CTMC is defined as,

$$Q = \lim_{h \to 0^+} \frac{P(h) - I}{h},$$

where I is the identity matrix.

The off-diagonal elements, $q_{ij}, i \neq j$, are the instantaneous rate of transitioning from state i to state j. The diagonal elements, q_{ii}, follow from the properties of the transition function and the definition of Q,

$$q_i := -q_{ii} = \sum_{\substack{j \in \mathcal{S} \\ j \neq i}} q_{ij},$$

where q_i is the rate at which the process leaves state i.

The holding time of $\{X(t)\}_{t \geq 0}$ being in state i, T_i, is exponentially distributed with mean [62] ,

$$\mathbb{E}[T_i] = \begin{cases} \frac{1}{q_i}, \text{ if } q_i > 0, \\ \infty, \text{ if } q_i = 0. \end{cases}$$

Following the random holding time, the process leaves the current state i and transitions to another state $j \neq i$ and the probability of this transition can be calculated as follows [62],

$$\Pr(X(t) \text{ enters state } j \neq i \mid X(t) \text{ leaves state } i) = \frac{q_{ij}}{q_i}.$$

The elements of the generator matrix of a CTMC enable Monte Carlo simulation of the process through the *stochastic simulation algorithm*[1] (SSA) [30]. In order to simulate the process, we require an initial state of the system $X(0) = x_0$, a stopping condition, and a way to calculate the propensities (or rates) from the current state. The SSA is given in Algorithm 1.

[1] Also referred to as the Gillespie algorithm.

Algorithm 1 Stochastic Simulation Algorithm (SSA)

Inputs Stopping condition, initial state x_0

Set $k = 0, t_0 = 0$

while Stopping condition not met **do**

$k \leftarrow k + 1$

From current state $x_{t_{k-1}}$ calculate rates corresponding to each event (where J is the number of events),

$$a_j, j = 1, 2, \ldots, J$$

Calculate total rate,

$$a_0 = \sum_{j=1}^{J} a_j$$

Sample $u_1, u_2 \sim U(0, 1)$

Sample holding time,

$$t' = -\frac{1}{a_0} \log(u_1)$$

Update time, $t_k = t_{k-1} + t'$

Determine next event by finding smallest integer μ such that,

$$\sum_{j=1}^{\mu-1} a_j < u_2 a_0 \leq \sum_{j=1}^{\mu} a_j$$

Update state, $x_{t_k} = \mu$

end while

return Realisation of the process $\{x_t\}_{t=0}^{T}$ and event times $\boldsymbol{t} = (0, t_1, \ldots, T)$

2.2 Epidemic modelling

Compartmental models are one way to model epidemics [1, 39]. A compartmental model divides the population up into compartments with the assumption that individuals in the same compartment share the same characteristics [1]. These characteristics typically refer to the state of individuals and can refer to phases such as susceptibility, infectiousness, potential immunity and more [1]. All individuals in the population are assumed to exist in one of these compartments at any given time and transition instantaneously between them. In this thesis, we work exclusively with *stochastic compartmental models* which offer a more 'realistic' model of the dynamics. Stochastic models are typically more challenging to analyse, however capturing the stochasticity in the system is important in smaller populations where individual events can have a large influence on the outcome of

the epidemic [1]. Stochastic compartmental models can be described in terms of CTMC models and capture the discrete nature of individuals in the population as well as the randomness inherent in events that occur during an outbreak. We now introduce one of the simplest epidemic models, the *SIR* model.

2.2.1 The SIR model

The *Susceptible, Infectious, Recovered*, or, *SIR* model is one of the simplest epidemic models [1, 39, 40]. Many, more complex models are built upon the framework of the SIR model and as such this model serves as a good introduction to epidemic modelling. There are three compartments of which individuals within the population can be in:

1. Susceptible (S) - able to be infected;
2. Infectious (I) - infected and infectious; and,
3. Recovered (R) - recovered and permanently immune to the disease.

An example of a compartmental diagram is provided in Figure 2.1. The circles represent the compartments an individual may be in and the arrows indicate transitions between compartments. The arrow from $S \to I$ represents an infection event and the arrow from $I \to R$ represents a recovery event.

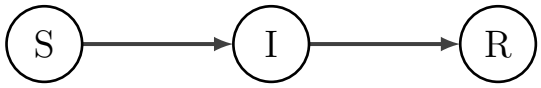

Figure 2.1: The SIR compartmental model. Details the transitions of individuals through the compartments.

Let $S(t)$ and $I(t)$ denote the number of susceptible and infectious individuals at time $t \geq 0$, respectively. In the instance where the population size, N, is constant, the number of recovered individuals, $R\,(t)$, can be easily recovered, as $R(t) = N - S(t) - I(t)$. As such the SIR model can be represented as a bivariate CTMC $\{(S(t), I(t))\}_{t\geq 0}$ with state-space (dropping the dependency on t),

$$\mathcal{S} = \Big\{ (S, I)\,|S, I \in \mathbb{N}, 0 \leq S, I, S + I \leq N \Big\}. \tag{2.1}$$

An alternative formulation of this process, which we use in this thesis, is the *Degree-of-Advancement* (DA) representation [8, 37, 52]. This representation uses state variables which count the number of transitions (infections and recoveries), as opposed to the number of individuals within each compartment. The DA representation and population

representations are related through the following system of equations. Let $\boldsymbol{X}(t)$ denote the state of the system in population representation, with some known initial condition $\boldsymbol{x}(0)$. Let $\boldsymbol{Z}(t)$ be a vector with elements corresponding to the number of events that have occurred up to time $t \geq 0$. The two representations are related through a stoichiometry matrix $\mathbb{M}$ where,

$$\boldsymbol{X}(t) = \boldsymbol{x}(0) + \mathbb{M}\boldsymbol{Z}(t). \tag{2.2}$$

The stoichiometry matrix $\mathbb{M}$ encodes the information of how an event influences the count of individuals within compartments [37]. Provided an initial population vector, Eq. (2.2) enables us to determine the population process from the DA process. In general it is not possible to obtain the DA process from the population process unless $\mathbb{M}$ is invertible [37].

In terms of the SIR model, the relationship between the two formulations is,

$$\begin{pmatrix} S \\ I \end{pmatrix} = \begin{pmatrix} S_0 \\ 0 \end{pmatrix} + \begin{pmatrix} -1 & 0 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} Z_1 \\ Z_2 \end{pmatrix}.$$

The stoichiometry matrix describes the change in S and I for an infection and recovery event. The change in population representation can be determined by reading down the columns of the stoichiometry matrix. Hence, if an infection event occurs (reading down the first column) then we remove 1 person from the susceptible compartment and add them to the infectious compartment and if a recovery event occurs (reading down the second column) then an individual is removed from the infectious compartment. For this model it is easy to see that the DA representation is more informative than the population representation. Using the DA representation the SIR model can be described by a bivariate CTMC model $\{(Z_1(t), Z_2(t))\}_{t\geq 0}$, with state space,

$$\mathcal{S} = \Big\{ (Z_1, Z_2) \,|\, Z_1, Z_2 \in \mathbb{N}, 0 \leq Z_2 \leq Z_1 \leq N \Big\}. \tag{2.3}$$

Note that there is a direct ordering of $Z_2 \leq Z_1$ which is as the number of recovery events cannot exceed the number of infection.

The transitions and corresponding rates from a given state (Z_1, Z_2), are given in Table 2.1. The denominator for the infection rate in Table 2.1 is $N - 1$ as this is the number of other individuals that a single individual can make contact with, excluding themselves.

The infection rate can be expressed as the product, *Scpv* [6], where S is the number of susceptible individuals, c is the rate of contact between individuals, $p = I/(N-1)$ is the probability that the contact is with an infectious individual, and v is the probability that the contact results in the transmission of the disease.

The concept of transmission is related to the mixing of individuals in the population. Mixing relates to how individuals in the population interact with one another. There are

Event	ΔState	Rate
Infection	$Z_1 \to Z_1 + 1$	$\frac{\beta(N - Z_1)(Z_1 - Z_2)}{N - 1}$
Recovery	$Z_2 \to Z_2 + 1$	$\gamma(Z_1 - Z_2)$

Table 2.1: Events and rates of the SIR model.

two common forms of mixing in epidemic models, homogenous and in-homogenous [1]. Homogeneous mixing assumes that all individuals in the population are equally likely to contact anyone else in the population. This is typically an unrealistic assumption but one which is commonly assumed in the modelling of epidemics [1, 2]. Non-homogeneous mixing assumes that subgroups of the population exist, and there exists contact rates within, and between, these subgroups. An example of in-homogeneous mixing would be household models where two levels of contact exist, within household and between household [9]. Individuals within a household interact more closely with members of the same household than with those from other households and so we can clearly establish two distinct levels of mixing. Throughout this thesis we will assume a homogeneous mixing of individuals and *frequency-dependent* transmission. Frequency-dependent transmission assumes that the rate of contact c is constant [6]. We can then let $\beta = cv$ be the frequency-dependent transmission term and as a result, the infection rate is,

$$a_1 = \frac{\beta SI}{N - 1}. \tag{2.4}$$

The other parameter that controls the dynamics of a SIR model is γ, the recovery rate of an infectious individual. Since we are using CTMCs this means that the infectious period of an individual is exponentially distributed with mean $1/\gamma$.

The basic reproduction number R_0, is defined as the average number of infections from a single infectious individual in an otherwise entirely susceptible population [1, 39]. For the SIR model, this quantity is,

$$R_0 = \frac{\beta}{\gamma}. \tag{2.5}$$

The importance of R_0 relates to the *threshold phenomenon* [39]. This is easily stated as the following; if $R_0 \leq 1$ then the disease will die out and not invade the population. If $R_0 > 1$ then there is a chance that the disease will invade the population and hence there is a major outbreak. This result means that R_0 is one of the most critical quantities in the prediction and control of epidemics during the early stages of an outbreak. The basic reproduction number R_0 is a quantity which is heavily influenced by community behaviour and two outbreaks of the same disease may not have the same R_0 [39].

The SIR model also assumes a closed population of size N and this means that there are no migrations, births or deaths. The SIR model is a very simplistic epidemic model with some relatively strong assumptions. The limiting assumptions of the SIR model are those relating to mixing of individuals and the simplistic outbreak dynamics. These assumptions dramatically influence the types of problems the model can be applied to as outbreaks are usually much more complicated than this and as such we need to appeal to more complex models to capture more interesting dynamics. This additional complexity is easily accounted for by adding additional compartments and event types to a compartmental model [1].

2.3 Bayesian inference

One of the primary interests in modelling epidemics is to infer the parameters of the models [28]. The parameters relay the dynamics of the outbreak in question and inferring them enables us to gain insights into the outbreak. These insights are vital in preparation for future outbreaks and providing intervention, or management strategies, to current and future threats. The understanding of a given outbreak arises from using the inferred parameters to simulate realisations of the epidemics. These simulations can be used for forecasting, or to assess possible intervention strategies and provide predictions and/or comparisons between the effectiveness of different strategies.

To conduct inference on the parameters, we appeal to a Bayesian framework, which aims to obtain the posterior distribution of the parameters given some observation data. A Bayesian framework is used here as this enables us to capture prior information which aids in ensuring biologically reasonable inferences [23]. Furthermore, the results from a Bayesian analysis can be updated as further data becomes available, which is very common for outbreak data. By Bayes' theorem, the posterior distribution of some vector of parameters $\boldsymbol{\theta}$ given some data $\mathcal{D}$ is,

$$p\left(\boldsymbol{\theta}|\mathcal{D}\right) = \frac{p\left(\mathcal{D}|\boldsymbol{\theta}\right)p\left(\boldsymbol{\theta}\right)}{p\left(\mathcal{D}\right)}. \tag{2.6}$$

Applying the law of total probability, yields,

$$p\left(\boldsymbol{\theta}|\mathcal{D}\right) = \frac{p\left(\mathcal{D}|\boldsymbol{\theta}\right)p\left(\boldsymbol{\theta}\right)}{\int_{\Theta} p\left(\mathcal{D}|\boldsymbol{\theta}\right)p\left(\boldsymbol{\theta}\right)d\boldsymbol{\theta}}, \tag{2.7}$$

where we integrate over Θ in the denominator which is the set of feasible parameters.

In Eq. (2.7), $p\left(\boldsymbol{\theta}\right)$ is the prior distribution of the parameters and is determined from past information [27, 67]. The prior distribution captures our understanding of the disease

before additional data and is typically informed from previous sources [27]. These sources can include previous mathematical studies, epidemiological studies and field studies. A well informed prior enables us to ensure biologically reasonable inferences by assigning distributions to parameters which reflect real dynamical behaviours [23].

The other key quantity is $p(\mathcal{D}|\boldsymbol{\theta})$ which is referred to as the likelihood. In parameter inference we are interested in the likelihood of the parameters given some fixed data, and as such we make the small notational change [67] of letting,

$$L(\boldsymbol{\theta}) \coloneqq p(\mathcal{D}|\boldsymbol{\theta}). \tag{2.8}$$

The likelihood, and more specifically, its calculation, are important topics in their own right with regards to inference. The calculation of the likelihood is necessary to calculate the posterior, and it is not always a simple task to perform this calculation when working with complex epidemic models. This topic is of such importance that it will be discussed further in Section 2.3.1.

The final quantity is the normalisation constant,

$$p(\mathcal{D}) = \int_{\Theta} L(\boldsymbol{\theta})\, p(\boldsymbol{\theta})\, d\boldsymbol{\theta}. \tag{2.9}$$

For models of reasonable complexity, this term is intractable [29]. Typically this means we have to sample from the posterior using Monte Carlo methods, of which *Markov chain Monte Carlo* (MCMC) is one of the most popular [26, 42]. MCMC incorporates an extensive class of algorithms that enable sampling from a desired distribution by constructing a Markov chain that has the desired distribution as its stationary distribution [26, 42]. The *Metropolis-Hastings* algorithm is one MCMC method that is commonly used, and one that we will use in this thesis [35, 53].

The Metropolis-Hastings algorithm enables us to draw samples from some target distribution, which within Bayesian inference is the posterior $p(\boldsymbol{\theta}|\mathcal{D})$. This is done by creating a discrete-time Markov chain with a stationary distribution equal to the target distribution [27, 28, 29]. A general Metropolis-Hastings scheme is provided in Algorithm 2.

There are several considerations of the user when implementing the Metropolis-Hastings algorithm. These include the choice of proposal distribution, the burn-in time, and initial state. The *mixing* of the resultant chain is the indicator of the performance of the algorithm [27, 29]. Mixing is how well the overall Markov chain explores the support of the posterior. The first and most important consideration which influences the mixing of the resultant chain is the choice of proposal distribution, $q(\cdot|\boldsymbol{\theta}_{k-1})$. The ideal proposal distribution is the full posterior distribution but this is not available. Instead, a common

Algorithm 2 Metropolis-Hastings

Inputs Initial parameters $\boldsymbol{\theta}_0$, number of samples n, proposal distribution $q(\boldsymbol{\theta}'|\boldsymbol{\theta})$, number of burn-in samples B

Set $k = 0$
while $k \leq n$ **do**
    $k \leftarrow k + 1$
    Sample candidate point,

$$\boldsymbol{\theta}' \sim q(\boldsymbol{\theta}|\boldsymbol{\theta}_{k-1})$$

    Calculate the acceptance probability,

$$\alpha(\boldsymbol{\theta}', \boldsymbol{\theta}_{k-1}) = \min\left\{1, \frac{p(\boldsymbol{\theta}'|\mathcal{D})q(\boldsymbol{\theta}_{k-1}|\boldsymbol{\theta}')}{p(\boldsymbol{\theta}_{k-1}|\mathcal{D})q(\boldsymbol{\theta}'|\boldsymbol{\theta}_{k-1})}\right\}$$

    Sample $u \sim U(0, 1)$
    if $\alpha(\boldsymbol{\theta}', \boldsymbol{\theta}_{k-1}) \geq u$ **then**
        $\boldsymbol{\theta}_k = \boldsymbol{\theta}'$
    else
        $\boldsymbol{\theta}_k = \boldsymbol{\theta}_{k-1}$
    end if
end while
return Posterior sample,$\{\boldsymbol{\theta}_k\}_{k=B+1}^{n}$

choice of proposal distribution is a multivariate normal proposal distribution centred at the current iterate $\boldsymbol{\theta}_{k-1}$ with some covariance matrix, Σ. Tuning of the proposal involves selecting the elements of Σ and tuning them accordingly to promote reasonable mixing. Mixing can be assessed by observing the *acceptance ratio* and by looking at *trace-plots* of multiple independent chains [27]. The acceptance rate is the number of samples accepted out of the total number of samples drawn. The optimal acceptance rate for a general implementation for Metropolis-Hastings is stated to be 0.234 [5]. While 0.234 signifies optimal mixing, an acceptance rate in the range of 0.1–0.5 is still acceptable but this often requires longer chains.

When Metropolis-Hastings is run from an arbitrary starting point, it can take a number of samples for the chain to converge to the target distribution. These samples are called the burn-in of the chain and are not samples drawn from the posterior distribution and as such are discarded [29]. The starting point can influence the number of burn-in samples and hence the overall performance of the inference process. An appropriate burn-in can be chosen by starting multiple independent chains from different starting points sampled from the prior distribution and observing estimators (such as an average) to see when the multiple chains return comparable results [27, 29].

A key way to quantify the performance of an MCMC method is to assess the *effective sample size* (ESS) of the posterior sample. The lack of independent-and-identically distributed (IID) samples causes issues by underestimating the variance of point estimates like the mean. This can be accounted for by using the ESS which gives the number of IID samples in a posterior sample obtained through a MCMC algorithm. The univariate ESS is defined as,

$$n_{\text{ESS}} = \frac{n}{1 + 2\sum_{i=k}^{\infty} \rho_k(\theta)}, \tag{2.10}$$

where n is the total number of samples and $\rho_k(\theta) = \text{cor}(\theta_k, \theta)$. In practice we truncate the summation in the denominator to the first value such that $\rho_k < 0$.

Pseudo-marginal MCMC replaces the exact calculation of the likelihood with an unbiased estimate [3]. *Particle Marginal Metropolis-Hastings* (pmMH) is a pseudo-marginal MCMC method which uses an unbiased estimate of the likelihood obtained from a particle filter (particle filters are explained in Section 2.4). The considerations for tuning pmMH are the same as those for general Metropolis-Hastings with only one major difference. Since the exact likelihood calculation is replaced with an estimate obtained through a particle filter, the optimal acceptance rate is stated to be smaller than that for Metropolis-Hastings [57, 65]. Sherlock *et al.* [65] states that the optimal acceptance rate in a pseudo-marginal method is approximately 0.07.

The tuning of pmMH methods is extremely important to optimal implementations. This tuning is carried out by adjusting the number of particles used in a particle filter such that the variance at the MAP lies within tight bounds [57, 65]. There is an obvious trade-off here as the number of particles used in the methods translates to increased computational expense and hence larger runtimes [8]. As such when using a particle filter for inference we want to balance the number of particles against the runtime of the method and this is typically determined through the ESS per unit of computation time.

Once a sample from the posterior distribution has been obtained we want to assess the model fit. A common approach for doing this is to use predictive distributions [27, 72]. The posterior predictive distribution for some new data, $\mathcal{D}_0$, given some observed data, $\mathcal{D}$, is defined as,

$$p\left(\mathcal{D}_0|\mathcal{D}\right) = \int_{\Theta} p\left(\mathcal{D}_0|\mathcal{D}, \boldsymbol{\theta}\right) p\left(\boldsymbol{\theta}|\mathcal{D}\right) d\boldsymbol{\theta}. \tag{2.11}$$

This integral is clearly intractable as the posterior $p(\boldsymbol{\theta}|\mathcal{D})$ is intractable. However, we can estimate the posterior predictive distribution using the sample from the posterior. First, we sample parameters from the posterior distribution $p(\boldsymbol{\theta}|\mathcal{D})$ through an MCMC method and then using this, we sample from $p(\mathcal{D}_0|\mathcal{D}, \boldsymbol{\theta})$. The sampling from $p(\mathcal{D}_0|\mathcal{D}, \boldsymbol{\theta})$ can be done by noting that the new data depends only on previous data through the parameters, $p(\mathcal{D}_0|\mathcal{D}, \boldsymbol{\theta}) = p\left(\mathcal{D}_0|\boldsymbol{\theta}\right)$. As such, this term is essentially the likelihood, where we condition on the posterior sampled parameters. In order to sample from $p(\mathcal{D}_0|\boldsymbol{\theta})$,

we simulate the model through the SSA. This process can be repeated some number of times and yields a sample from the posterior predictive distribution. These samples will be posterior simulated outbreak data and can be used for comparison with the observed data. Typical comparisons can involve summaries of observed incidences, final size and duration for the observed data against the posterior simulated data.

Some useful tools for summarising a posterior sample are *credible intervals (CI)* and *highest posterior density (HPD) intervals* [67]. A $100(1-\alpha)\%$ CI (for a scalar parameter θ) is defined as the following,

$$\Pr(\theta \in C|\mathcal{D}) = 1 - \alpha, \tag{2.12}$$

where $C \subseteq \Theta$. This interval expresses the notion that the probability of θ being in C given some data, $\mathcal{D}$, is $1-\alpha$. The $100(1-\alpha)$ HPD interval is defined as the credible region C with the highest posterior values,

$$C = \{\theta \in \Theta | p(\theta|\mathcal{D}) \geq \pi_\alpha\}, \tag{2.13}$$

where π_α is the largest constant such that $\Pr(\theta \in C) \geq 1-\alpha$ [48].

2.3.1 The likelihood

Here we discuss how to calculate the likelihood for a partially observed CTMC. This problem of estimating the internal states of a dynamical system when only partial observations are made is referred to as the *filtering problem* [3, 31, 63]. Let $\boldsymbol{x}_{0:T} = \{\boldsymbol{x}_0, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_T\}$, and let $\boldsymbol{y}_{1:T} = \{\boldsymbol{y}_1, \boldsymbol{y}_2 \ldots, \boldsymbol{y}_T\}$, where at each time t the state of the system is represented by a vector $\boldsymbol{x}_t$ and the observation of this state (which can be a scalar or vector quantity) is $\boldsymbol{y}_t$. It is not typically the case that $\boldsymbol{x}_t$ and $\boldsymbol{y}_t$ contain the same information, most often, $\dim \boldsymbol{y}_t < \dim \boldsymbol{x}_t$. Often this means that the observations are only of one or two events, which means there are latent (or unobserved) events. In this thesis we use the convention that the state of the system $\boldsymbol{x}_t$ begins at time $t = 0$ but that observations are made from time $t = 1$ onwards. The likelihood is given as,

$$\begin{aligned} L(\boldsymbol{\theta}) &= p(\boldsymbol{y}_{1:T}|\boldsymbol{\theta}) \\ &= p(\boldsymbol{y}_1|\boldsymbol{\theta}) \prod_{t=2}^{T} p\left(\boldsymbol{y}_t|\boldsymbol{y}_{1:t-1}, \boldsymbol{\theta}\right). \end{aligned} \tag{2.14}$$

The process of estimating the likelihood relies on the following recursion, where we assume the problem has been solved up to time $t-1$,

$$p\left(\boldsymbol{x}_{0:t}|\boldsymbol{y}_{1:t-1}\right) = p\left(\boldsymbol{x}_{0:t-1}|\boldsymbol{y}_{1:t-1}\right) p\left(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}\right) \tag{2.15}$$

$$p\left(\boldsymbol{x}_{0:t}|\boldsymbol{y}_{1:t}\right) = p\left(\boldsymbol{x}_{0:t}|\boldsymbol{y}_{1:t-1}\right) \frac{p\left(\boldsymbol{y}_t|\boldsymbol{x}_t\right)}{p\left(\boldsymbol{y}_t|\boldsymbol{y}_{1:t-1}\right)}. \tag{2.16}$$

Using the recursion, the contribution to the likelihood over $[t-1, t)$ is,

$$\begin{aligned} p\left(\boldsymbol{y}_t|\boldsymbol{y}_{1:t-1}\right) &= \int p\left(\boldsymbol{y}_t|\boldsymbol{x}_{0:t}\right) p\left(\boldsymbol{x}_{0:t}|\boldsymbol{y}_{1:t-1}\right) d\boldsymbol{x}_{0:t} \\ &= \int p\left(\boldsymbol{y}_t|\boldsymbol{x}_t\right) p\left(\boldsymbol{x}_{0:t}|\boldsymbol{y}_{1:t-1}\right) d\boldsymbol{x}_{0:t}, \end{aligned} \tag{2.17}$$

where we have used the conditional independence of the current observation $\boldsymbol{y}_t$ given the underlying states of the system $\boldsymbol{x}_{0:t}$ to obtain the result in the second integral. The term $p\left(\boldsymbol{y}_t|\boldsymbol{x}_t\right)$ is simply the observation density and is typically chosen to be some simple and known probability distribution. The second term $p\left(\boldsymbol{x}_{0:t}|\boldsymbol{y}_{1:t-1}\right)$ can be calculated using the recursive definition in Eq. (2.15).

When the size of the state space is small, we can leverage exact methods for solving this series of equations. These typically involve the use of matrix exponentials or using an implicit Euler approach [37, 66]. Each of these methods grows increasingly inefficient as the size of the state space increases so instead we can use Monte Carlo methods [3].

2.4 Particle filters

Sequential Monte Carlo (SMC) methods can be applied to any computational task that may be formulated as a filtering problem [17]. When used for state-space inference, SMC is commonly referred to as particle filtering [3, 17]. A *particle filter* is a Monte Carlo approach to solving the filtering problem.

Particle filters enable us to estimate the state of a dynamical system through the use of simulation. In doing so, an unbiased estimate of the likelihood can be obtained which can be used in a *particle MCMC* method referred to as *particle marginal Metropolis-Hastings (pmMH)* [3]. This makes them invaluable for use within an inference setting when the likelihood is intractable.

2.4.1 Bootstrap particle filter

The Bootstrap particle filter is a simple and intuitive SMC algorithm [33]. In the following we drop the conditioning on the parameters $\boldsymbol{\theta}$ but note that the conditioning is implied. Assume we begin with a set of N_p particles at time t,

$$\left\{\boldsymbol{x}_{0:t}^{(i)}\right\}_{i=1}^{N_p} \sim p\left(\boldsymbol{x}_{0:t}|\boldsymbol{y}_{1:t}\right).$$

We can use *prediction* and *update* steps to generate a set of particles which are consistent with the next observation at time $t+1$. The prediction step involves simulating each particle over $[t, t+1)$ using the current particle states $\boldsymbol{x}_{0:t}^{(i)}, i = 1, \ldots, N_p$ as the initial conditions. This yields a set of particles distributed as,

$$\left\{\tilde{\boldsymbol{x}}_{0:t+1}^{(i)}\right\}_{i=1}^{N_p} \sim p\left(\boldsymbol{x}_{0:t+1}|\boldsymbol{y}_{1:t}\right).$$

Particles are then assigned weights $w_t^{(i)} = p\left(\boldsymbol{y}_{t+1}|\tilde{\boldsymbol{x}}_{0:t+1}^{(i)}\right)$, and a Monte Carlo estimate of the likelihood (given by Eq. 2.17) over $[t, t+1)$ is,

$$\hat{p}\left(\boldsymbol{y}_{t+1}|\boldsymbol{y}_{1:t}\right) = \frac{1}{N_p}\sum_{i=1}^{N_p} w_t^{(i)}. \tag{2.18}$$

The update step begins by resampling the particles using the normalised weights. This returns a set of particles distributed as,

$$\left\{\boldsymbol{x}_{0:t+1}^{(i)}\right\}_{i=1}^{N_p} \sim p\left(\boldsymbol{x}_{0:t+1}|\boldsymbol{y}_{1:t+1}\right).$$

This process can be repeated to obtain estimates of the likelihood contributions $\hat{p}\left(\boldsymbol{y}_t|\boldsymbol{y}_{1:t-1}\right)$ for $t = 1, \ldots, T$. Noting that $p\left(\boldsymbol{y}_1|\boldsymbol{y}_{1:0}\right) = p\left(\boldsymbol{y}_1\right)$, this can be estimated using the same sequential process. These estimated likelihood contributions can be used to obtain an unbiased estimate of the total likelihood by using Eq. (2.14). Pseudo-code for the bootstrap particle filter is provided in Appendix A.

In the update step, resampling is used to increase the likelihood of particles with large weights being propagated into the next time step over those with small weights. This means particles which are more consistent with the data are likely to be replicated and those which have lower weights are removed. While resampling is definitely a beneficial inclusion in most implementations of particle filters, it should be noted that the way in which the particles are resampled can influence the variance in the likelihood estimates [46]. In this thesis we rely on systematic resampling (Algorithm 3) as this form of resampling eliminates the chance of only a few particles to be resampled on any particular iteration. This approach also has higher flexibility than methods like multinomial resampling as we are more likely to resample low-weight particles [46].

Depending on the observation density the main issue with the bootstrap particle filter is that naively forward simulating states is inefficient. This typically means we require a large number of particles to appropriately estimate the state of the system and obtain low variance estimates of the likelihood [3, 25, 36, 57]. This greater number of particles leads to increased computational expense and hence lower ESS per unit of computation time. This typically arises due to only a handful of particles being consistent with the

Algorithm 3 Systematic resampling

Inputs Vector of weights $\boldsymbol{w} = (w_1, w_2 \ldots, w_{N_p})$ and vector of particle indices $\boldsymbol{\nu} = (1, 2, \ldots, N_p)$

Set $j = 0$, and initialise the cumulative weight $w_C = w_1$, initialise vector of resampled indices $\boldsymbol{\mu} = (1, 2, \ldots, N_p)$
Generate random starting point, $u = r_1/N_p$ where $r_1 \sim U(0, 1)$
for $i = 0, 1, \ldots, N_p - 1$ **do**
    while $w_C < u$ **do**
        $j \leftarrow j + 1$
        $w_C \leftarrow w_C + w_j$
    end while
    Update index $\mu_i = \nu_j$
    Update current point $u \leftarrow u + 1/N_p$
end for
return Resampled indices $\boldsymbol{\mu}$

observations and hence the estimates of the likelihood grow increasingly worse. This is a particularly prominent issue with simulation of rare events where the filter will struggle to return a non-zero likelihood. This poses large issues in parameter inference as the variance in the likelihood estimates leads to poor mixing and hence inefficient pmMH method.

2.4.2 Alive particle filter

Jasra *et al.* [36] propose an alternative to the bootstrap particle filter called the *alive particle filter*. The alive filter follows a similar process to the bootstrap filter but now a number of particles are simulated until they are consistent with the observation and hence the algorithm is more robust but also potentially more expensive [25]. The major difference between the bootstrap and the alive particle filter is that at each time step, we simulate the process forward in time until we obtain $N_p + 1$ consistent particles. Then an estimate of the likelihood is given as,

$$p\left(\boldsymbol{y}_t | \boldsymbol{y}_{1:t-1}\right) = \frac{N_p}{n_t - 1},$$

where N_p is the number of particles and n_t is the number of simulations required to obtain the $N_p + 1$ consistent particles. See Algorithm 8 for a direct implementation of the alive particle filter. This means a non-zero weight is assigned to all particles. Degeneracy only arises in the case where it is very unlikely for particles to match the observed data as the number of simulations required for consistency with the observation increases [8, 25, 52]. In practical application this can cause the algorithm to become stuck and so one adds

an additional simulation parameter, K to prevent this. K is chosen such that if on any particular iteration, should the current number of simulations n be such that $n > K$ then the filter returns 0 as the estimated likelihood [25, 36]. This check prevents the filter from getting stuck in the instances where the observed event is very unlikely but also means the algorithm is no longer unbiased. Pseudo-code for the Alive particle filter is provided in Appendix A.

2.4.3 Importance sampling

A prominent issue which arises with using more naive particle filters, like the bootstrap and alive filters, is that they can produce high variance estimates [3, 52]. In both filters, as the size of the state space grows or there is a rare event which needs to be simulated, then it can take a large number of particles to appropriately estimate the state of the system. It can simply be difficult to obtain low variance estimates of the likelihood using these methods, but a secondary issue is that this larger number of particles is computationally expensive and leads to inefficient inference methods [8, 52]. Instead of these naive approaches, we can use the variance reduction technique, *importance sampling*. Importance sampling can be used to design simulation algorithms which increase the probability of a simulation being consistent with the data.

Consider calculation of the following expectation,

$$\mathbb{E}_p\left[f(X)\right] = \int f(x)p(x)dx,$$

where the expectation is taken with respect to the density $p(x)$. A Monte Carlo estimate $\hat{\mu}$ of this expectation can be obtained by drawing samples $x_i \sim p(x)$, and calculating the following [42],

$$\hat{\mu} = \frac{1}{N_p}\sum_{i=1}^{N_p} f(x_i), \tag{2.19}$$

where N_p is the number of samples drawn. This approach relies on us being able to sample from the probability distribution $p(x)$, however, this is not always possible [42]. Instead of sampling directly from the target density $p(x)$ we can sample from an alternative distribution, called an *importance distribution*, $q(x)$ [42]. Samples are now drawn according to this importance density, $x_i \sim q(x)$, and correcting for this, an estimate of the expectation is,

$$\hat{\mu} = \frac{1}{N_p}\sum_{i=1}^{N_p} f(x_i)\frac{p(x_i)}{q(x_i)}, \tag{2.20}$$

where N_p is again the number of samples drawn. This importance density should be chosen such that the support of $q(\cdot)$ dominates that of the target density $p(\cdot)$ and its easy to sample from [42].

Importance sampling will be used in this thesis to design particle filters which improve the ESS per unit of computation time in comparison to more naive filters. In Chapter 3 we use importance sampling in particle filters to obtain likelihood estimates in the estimation of the *maximum a posteriori*. Chapters 4 and 5 extend recent work on importance sampling based particle filtering techniques [8]. These chapters extend importance sampling in a particle filter to produce realisations which exactly match two time series of observations, one for onset events and one for removals. This methodology will then be used to perform parameter inference on models of historical outbreaks of Ebola.

2.4.4 Importance sampling in particle filters

Here we outline the importance sampling based approach to particle filtering. This relies on the work in Black [8] to develop efficient particle filtering methods. In Section 2.4 we introduced the basic idea of the bootstrap particle filter and how it can be used to estimate the likelihood. To demonstrate the difference between the bootstrap method and the importance sampling approach, suppose we have a single observation y of an underlying state z_1. Assuming a known initial condition z_0 then the problem of calculating the likelihood $p(y)$ equates to calculating the following,

$$p(y) = \int p(y|z_1)\, p(z_1|z_0)\, dz_1.$$

The bootstrap particle filter relies on producing a Monte Carlo estimate of this integral by sampling states from the transition density. Sampling $z_1^{(i)} \sim p(z_1|z_0)$, for $i = 1, \ldots, N_p$ where N_p is the number of realisations produced. an estimate of the likelihood is given by,

$$\hat{p}(y) = \frac{1}{N_p} \sum_{i=1}^{N_p} p(y|z_1^{(i)}).$$

As mentioned in Section 2.4 this can produce high variance estimates. Importance sampling can be used here to produce realisations which are more consistent with the observations at little additional computational expense [52]. To do this we sample states z_1 from an importance distribution $q(z_1|z_0, y)$ which accounts for the observation, instead of the true transition density $p(z_1|z_0)$, and correcting for this, an estimate of the likelihood

is [42],

$$\hat{p}(y) = \frac{1}{N_p} \sum_{i=1}^{N_p} p(y|z_1^{(i)}) \frac{p(z_1^{(i)}|z_0)}{q(z_1^{(i)}|z_0, y)}, \tag{2.21}$$

where N_p is the number of realisations produced.

The purpose of the importance density is that this can be chosen in such a way to increase the probability of the simulations being consistent with the observed data. This leads to a large improvement in the runtime of the particle filter as all simulations are consistent with the observations. This results in overall improvements to the performance of the inference methods and hence a larger ESS per second [8].

A key insight by McKinley *et al.* [52] is that we are able to design simulation based algorithms such that the observation likelihood $p(y|z_1) = 1$ for all realisations. This essentially means that all particles end up in a state which is consistent with the next observation. Hence, the resulting likelihood estimate is,

$$\hat{p}(y) = \frac{1}{N_p} \sum_{i=1}^{N_p} \frac{p(z_1^{(i)}|z_0)}{q(z_1^{(i)}|z_0, y)}. \tag{2.22}$$

We refer to the term inside the summation,

$$w^{(i)} = \frac{p(z_1^{(i)}|z_0)}{q(z_1^{(i)}|z_0, y)},$$

as the weight. This weight can be calculated iteratively as the simulation progresses. The work of Black [8] builds on the earlier work of McKinley *et al.* [52] and provides an efficient and numerically stable simulation method for estimating the likelihood for more complex epidemic models.

Algorithm 4 details the importance sampling based approach to particle filtering over a single observation period $[0, 1)$. This algorithm can be applied to each day of a time series and results in all particles being exactly consistent with the observed state of the system. The use of the modified process which forces consistency greatly improves the efficiency of the simulations.

In Step 23 of Algorithm 4 the continuation of the simulation allows for all unforced events to be simulated. The unforced events are those which are not the observed state as these are accounted for inside the while loop. All other modified events are adjusted in such a way to keep the simulation from terminating (where appropriate). In this time the process taken is the same as Steps 14 to 17. The final contribution in Step 24 of Algorithm 4 arises as the probability of no further events in the interval $(t, 1]$ where t is the time of the current event.

Algorithm 4 Importance Sampling Particle Filter

Inputs Data y, initial state $Z(0)$, propensities as a function of the state $a_i(\cdot)$, for $i = 1, \ldots, M$

1: Generate the times of the y observed events from a uniform distribution and add them to the stack ψ in reverse order.

2: Set $e_n, t_n, t = 0$.

3: Set initial weight contribution from generation of order statistics,

$$w = \log(y!)$$

4: **while** Stack ψ not empty **do**

5: Calculate the rates of the original process given the current state of the system,

$$a_i = a_i(Z(t)), \text{ for } i = 1, \ldots, M,$$

where M is the total number of events.

6: Calculate the total rate of the original process,

$$a_0 = \sum_{i=1}^{M} a_i$$

7: **if** The system is inconsistent with the next forced event e_n **then**

8: Generate the time of this event (assumed to be of type l) on the truncated interval $[t, t_n]$,

$$s \sim \text{TruncExp}(a_l, 0, t_n - t)$$

9: Update the weight,

$$w \leftarrow w - \log(a_l) + a_l s + \log\left(1 - e^{-a_l(t_n - t)}\right)$$

10: **end if**

11: Calculate the rates of the modified process which depend on the current state and the next forced event,

$$b_i = b_i(Z(t), e_n), \text{ for } i = 1, \ldots, M$$

12: Calculate the total rate of the modified process,

$$b_0 = \sum_{i=1}^{M} b_i$$

13: Propose a time to the next event (under the modified process) $t' \sim \text{Exp}(b_0)$
14: **if** $t' < t_n - t$ **then**
15: Choose an even index $j \in \{1, \ldots, M\}$ with probability $\Pr(j = i) = b_i/b_0$.
16: Update state and time,

$$Z_j \leftarrow Z_j + 1$$
$$t \leftarrow t + t'$$

17: Update weight,

$$w \leftarrow w + \log\left(\frac{a_j}{b_j}\right) - (a_0 - b_0)t'$$

18: **else if** $t' \geq t_n - t$ **then**
19: Implement the next forced event at time t_n,

$$Z_{e_n} \leftarrow Z_{e_n} + 1$$
$$t \leftarrow t_n$$

20: Update the weight,

$$w \leftarrow w + \log(a_{e_n}) - (a_0 - b_0)(t_n - t)$$

21: Pop event e_n and event time t_n off the stack
22: **end if**
23: **end while**
24: Continue simulation until $t + t' > 1$.
25: Once $t + t' > 1$ the final contribution is,

$$w \leftarrow w - (a_0 - b_0)(1 - t)$$

26: **return** Weight w

The final inclusion for an efficient and numerically stable particle filter is working with logarithms. As mentioned in Section 2.3, we work with logarithms as the probabilities we are working with are very small. Working with log-weights inside of a particle filter still requires exponentiation when the simulation terminates. This is prone to numerical instabilities and a common approach used to avoid exponentiation is the *Log-Sum-Exponential* (LSE) "trick" [10]. This can easily be applied alongside particle filters to reduce numerical instability and is essentially a way of evaluating the summation in Eq. 2.22. The LSE, φ, can be calculated as,

$$\varphi = c + \log \sum_{i=1}^{N_p} \exp\left(w^{(i)} - c\right), \tag{2.23}$$

where $w^{(i)}$ now denotes the log-weights and $c = \max\{w^{(1)}, w^{(2)}, \ldots, w^{(N_p)}\}$ ensures that the largest normalised weight is 1 [10]. Using the LSE, the log-likelihood is then calculated as,

$$\log \hat{p}(y) = -\log N_p + \varphi, \tag{2.24}$$

which is simply the log of Eq. (2.22). The weights normalised in this way can then be used in the resampling strategies avoiding issues of direct renormalisation.

2.5 The SPSA algorithm

This section introduces the *simultaneous perturbation stochastic approximation* (SPSA) algorithm. This provides the foundations for Chapter 3 were we develop a method for estimating the *maximum a posteriori* (MAP) in Bayesian models when the likelihood is only able to be estimated. This section will detail some basic theory related to steepest descent style methods of optimisation, and how these can be adapted for use in a stochastic optimisation framework.

Optimisation in this thesis is concerned with determining the minimiser of a function. In many optimisation problems we assume that the following are known, or able to be estimated;

- A loss function $f(\boldsymbol{x})$; and,
- a set of feasible points $\Omega \subseteq \mathbb{R}^m$.

The general minimisation problem can be stated as,

$$\boldsymbol{x}_{\min} = \arg\min_{\boldsymbol{x} \in \Omega} f\left(\boldsymbol{x}\right). \tag{2.25}$$

There are various approaches for minimising a function when it has a closed form [16, 69]. Many of these rely on calculation of the gradient of $f(\boldsymbol{x})$ [16]. In instances where the exact derivatives are not easily attainable, approximations are usually made. A particular method we build on in this thesis, the SPSA algorithm, relies on using a first order finite difference approximation to the gradient [69]. There are two first order finite difference approximations used in stochastic optimisation, a two-sided approximation;

$$\nabla f(\boldsymbol{x}) \approx \frac{f(\boldsymbol{x}+\boldsymbol{h}) - f(\boldsymbol{x}-\boldsymbol{h})}{2}\boldsymbol{h}^{-1}, \tag{2.26}$$

and, a one-sided approximation;

$$\nabla f(\boldsymbol{x}) \approx (f(\boldsymbol{x}+\boldsymbol{h}) - f(\boldsymbol{x}))\,\boldsymbol{h}^{-1}, \tag{2.27}$$

where $\boldsymbol{h}$ is some step vector and $\boldsymbol{h}^{-1}$ is a vector of inverses,

$$\boldsymbol{h}^{-1} = \left(\frac{1}{h_1}, \ldots, \frac{1}{h_m}\right).$$

When we estimate the loss function there is some noise in the result, and the loss function is more appropriately expressed as,

$$f(\boldsymbol{x}) = \tilde{f}(\boldsymbol{x}) + \epsilon(\boldsymbol{x}), \tag{2.28}$$

where $\epsilon(\boldsymbol{x})$ is some noise term which is often state dependent, $\tilde{f}(\boldsymbol{x})$ is the actual loss measurement and $\boldsymbol{x}$ is some point in the set $\Omega \in \mathbb{R}^m$ which may be constrained. The function in Eq. (2.28) is referred to as a loss function. In the context here—as our loss function is stochastic in nature—this form of optimisation is referred to as *stochastic optimisation* [16, 69].

Spall [69] provides the technical link between the two sided approximation and the true gradient for particular choices of the step vector. This derivation can be easily adapted to the one-sided finite difference and we provide the key result here. At a given iteration k, let $\boldsymbol{h}_k = c_k\boldsymbol{\Delta}_k$ where $\{c_k\}$ is a specially defined sequence and $\boldsymbol{\Delta}_k$ has independent and identically distributed elements $\Delta_{k,i}$ which satisfy an inverse moments property. Let $g_{k,j}(\cdot)$ denote element j of the estimate of the gradient as per Eq. (2.27). Under the assumptions on c_k and $\boldsymbol{\Delta}_k$,

$$\mathbb{E}\left[g_j(\hat{\boldsymbol{\theta}}_k)|\hat{\boldsymbol{\theta}}_k\right] \approx f_j'(\boldsymbol{\theta}_k) + \sum_{l\neq m} f_l'(\boldsymbol{\theta}_k)\,\mathbb{E}\left[\frac{\Delta_{k,l}}{\Delta_{k,m}}\right]$$

where f' denotes the true gradient and $\Delta_{k,m}$ denotes element m of $\boldsymbol{\Delta}_k$. The '$\approx$' arises as there are some neglected higher order terms. Furthermore, the error terms on the loss function may introduce an additional source of error, however this often proves negligible

as the function estimate is often distributed about the true function value and hence the error has mean 0. The error terms introduced by the Taylor expansion are of order $O(c_k^2)$ and so can be neglected for suitably chosen sequences c_k. Under the assumption on the elements of $\boldsymbol{\Delta}_k$ satisfying an inverse moment property and being a mean 0 distribution, $\mathbb{E}\left[\Delta_{k,m}/\Delta_{k,j}\right] = 0$. As a result the summation term is equal to zero and hence,

$$\mathbb{E}\left[g_{k,j}(\hat{\boldsymbol{\theta}}_k)|\hat{\boldsymbol{\theta}}_k\right] \approx f_j'(\hat{\boldsymbol{\theta}}_k).$$

We provide details regarding the choices of c_k and $\boldsymbol{\Delta}_k$ in Chapter 3. The details in [69] provide the detailed walkthrough for the derivation of this result when assuming a central finite difference. The same derivation can easily be adapted to produce the result for the single sided finite difference used here.

In a stochastic optimisation framework there are a range of commonly used techniques such as *simulated annealing* and *genetic algorithms* [16]. The issue with these types of algorithms is that they are computationally expensive, and (often) slow approaches which can prove difficult to tune [69]. These methods are most suited in instances where the problem is solely to solve the optimisation problem to a certain level of precision. The key purpose of obtaining the MAP in this thesis is to tune particle filters and as a result of this we prefer lower runtimes over highly precise estimates. An alternative to these methodologies is the *simultaneous perturbation stochastic approximation* (SPSA) algorithm which relies on the idea of a finite difference approximation to the gradient. The SPSA algorithm searches in an approximate steepest descent direction and uses gain sequences which increase the step sizes during the earlier iterations, while reducing it later on [69]. This accounts for noise in the estimation of the loss function.

The SPSA algorithm was developed in Spall [68] and while this provided most of the analysis of the properties of the algorithm, later work provided a detailed explanation and proof of the convergence properties of the SPSA algorithm [69]. We refer the interested reader to [69] for the detailed analysis and proof of the performance of the general algorithms. We begin by presenting the basic version of the SPSA algorithm (Algorithm 5) with many generalised steps to demonstrate the main ideas.

In Algorithm 5, $\boldsymbol{\Delta}$ is referred to as the perturbation vector and is used to determine the direction of the update to the current estimate of the parameters. The gain sequences, $c^{(k)}$ and $a^{(k)}$ are sequences of positive decreasing numbers approaching 0. These gain sequences allow the search to improve precision during later iterations. The parameters of the SPSA algorithm are crucial to its implementation and while there are no certain choices for each of these, Spall [69] provides many reasonable starting choices. We leave discussion on these sequences until Chapter 3 as we provide indications of some of these suggestions as well as some adaptations that we have made in order to improve the performance in a Bayesian framework where SMC methods are used to obtain estimates of the likelihood.

Algorithm 5 Simultaneous Perturbation Stochastic Approximation (SPSA) algorithm

Inputs Initial parameters $\boldsymbol{x}_0$, number of iterations M, loss function f

1: **for** $k = 1 : M$ **do**

2: **for** $i = 1 : m$, where m is the number of parameters **do**

3: Sample $\Delta_i^{(k)}$ from a mean 0 distribution satisfying an inverse moments property (Eq. 3.5).

4: **end for**

5: Set $\boldsymbol{\Delta}^{(k)} = \left(\Delta_1^{(k)}, \Delta_2^{(k)}, \ldots, \Delta_m^{(k)}\right)$

6: Set $\boldsymbol{h}^{(k)} = c^{(k)}\boldsymbol{\Delta}^{(k)}$

7: Estimate the derivative of the loss function,

$$\nabla f\left(\boldsymbol{x}^{(k)}\right) \approx \hat{\boldsymbol{g}}^{(k)}$$

8: Update the current estimate of the minimum using the standard stochastic approximation form,

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} - a^{(k)}\hat{\boldsymbol{g}}^{(k)},$$

9: **end for**

Return Search path $\left\{\boldsymbol{x}^{(k)}\right\}_{k=1}^{M}$ and estimate of the minimum $\boldsymbol{x}^{(M)}$

Chapter 3

MAP estimation using noisy likelihood estimates

This chapter details the development of a stochastic optimisation algorithm to determine an estimate of the *maximum a posteriori* (MAP) of a posterior distribution when only noisy estimates are available. This chapter explores the *simultaneous perturbation stochastic approximation* (SPSA) algorithm and the alterations needed in order to estimate the MAP when the likelihood is estimated using a particle filter. The modified SPSA algorithm is tested on several simulated datasets in order to assess its practical performance.

3.1 The MAP estimation problem

The MAP is the vector of parameters $\boldsymbol{\theta}_{\text{MAP}} \in \Theta$, where Θ is the support of the posterior, such that,

$$\boldsymbol{\theta}_{\text{MAP}} = \underset{\boldsymbol{\theta} \in \Theta}{\arg\max}\, p\left(\boldsymbol{\theta}|\mathcal{D}\right), \tag{3.1}$$

where $\mathcal{D}$ is the observed data.

From a Bayesian modelling perspective, the MAP is important for two main reasons. The first is that an estimate of the MAP provides information about the most likely parameters to have resulted in our data while accounting for our prior beliefs. This can be used to then make quick decisions in different contexts without having to sample the whole posterior. The second reason is to tune the particle filters used to obtain estimates of the log-likelihood and hence lead to improved efficiency of pmMH methods [8, 57, 65]. This latter point is key to improved use of pmMH methods to learn posterior distributions

of parameters. Within this thesis we place emphasis on determining the MAP estimate in order to improve the computational efficiency of the pmMH methods by determining the optimal number of particles to use in a given particle filter. This optimal number of particles is problem dependent and a suitable choice of particles is one which keeps the variance of the log-likelihood within tight bounds [3, 22, 65].

3.2 Tuning particle filters

The goal of tuning a particle filter is to optimise the ESS per unit of computation time. How well the pmMH routine explores the posterior space influences the ESS and this depends on the mixing of the overall chain. The mixing is directly controlled by the variance in the particle filters, which is in turn dependent on the number of particles used in the particle filter [8, 22, 57]. Variance in the log-likelihood estimates can be reduced by increasing the number of particles, but this leads to higher computational expense as the algorithm scales linearly with the number of particles used [3, 8]. There is an obvious tradeoff between the variance in the likelihood estimates and the computational expense. Making more informed choices as to the number of particles to use in a particle filter can improve the efficiency of the overall inference method.

In order to tune particle filters, we want to keep the variance in the log-likelihood estimates at the MAP within tight bounds [22, 57, 65]. We can assess previous studies of the target variances for efficient pmMH procedures to gain some insight into appropriate bounds on the variance. Pitt *et al.* [57] assumes the variance (in the log-likelihood estimates) is Gaussian and further assumes that the Metropolis-Hastings algorithm used is an independence sampler which proposes from the posterior distribution and states that the optimal variance in this case is $0.92^2 = 0.84$ and that a reasonable range is $(0.25, 2.25)$. This is an unrealistic situation for most problems as mentioned in Sherlock *et al.* [65] and Doucet *et al.* [22]. Sherlock *et al.* [65] relaxes the assumption that the noise is Gaussian but restricts the proposal to a normal random walk proposal and assumes that the posterior density factorises into independent and identically distributed components. In this framework they show that the optimal variance is around 3.283 with an acceptance rate of approximately 7%. One of the more recent approaches is that of Doucet *et al.* [22]. They relax the assumptions in these previous two studies to general proposal and target densities and relax the Gaussian noise assumption. They determine the optimal variance to be around 1.4.

The results from the three studies can be aggregated into a reasonable range of between $[0.5, 3.3]$. We choose to target variances in the log-likelihood estimates in this range which should provide more optimal implementations of pmMH procedures. The problem

of identifying the number of particles to use is a one-dimensional optimisation problem, which simply requires the user to choose the number of particles to obtain variances approximately in this range.

3.3 MAP estimation as a minimisation problem

A naive approach to determine the MAP would be to simply consider the posterior distribution as the target function. The problem with this is that with most useful epidemic models, the posterior is often analytically intractable. A better target function can be determined by first taking the log of the posterior distribution as this is a one-to-one function. This has the added benefit of improving the numerical stability of the function estimates. Hence, a decent initial target function is,

$$y_1(\boldsymbol{\theta}) = \ell(\boldsymbol{\theta}) + \log(p(\boldsymbol{\theta})) - \log(p(\mathcal{D})),$$

where $\ell(\boldsymbol{\theta}) = \log(L(\boldsymbol{\theta}))$. This can be simplified further by noting that the intractable term $\log(p(\mathcal{D}))$ is constant with respect to the parameters, and that,

$$\arg\max_{\boldsymbol{\theta}\in\Theta} \Big\{ \ell(\boldsymbol{\theta}) + \log(p(\boldsymbol{\theta})) - \log(p(\mathcal{D})) \Big\} = \arg\max_{\boldsymbol{\theta}\in\Theta} \Big\{ \ell(\boldsymbol{\theta}) + \log(p(\boldsymbol{\theta})) \Big\}.$$

Noting that SPSA operates in a minimisation framework—and that maximising the posterior is equivalent to minimising the negative—we can instead work with the negative of this function,

$$y_2(\boldsymbol{\theta}) = -\ell(\boldsymbol{\theta}) - \log(p(\boldsymbol{\theta})). \tag{3.2}$$

Optimising Eq. (3.2) is a trivial task when both the likelihood and prior are analytically tractable. In models where the likelihood is only able to be estimated, we instead have a noisy estimate of the objective function and the problem becomes much more complicated. The loss function in this case is,

$$f(\boldsymbol{\theta}) = -\hat{\ell}(\boldsymbol{\theta}) - \log(p(\boldsymbol{\theta})), \tag{3.3}$$

where $\hat{\ell}(\cdot)$ is an estimate of the log-likelihood. If we assume this noise is additive and approximately has mean 0, then the estimate of the log-likelihood can be expressed as,

$$\hat{\ell}(\boldsymbol{\theta}) = \ell(\boldsymbol{\theta}) + \epsilon(\boldsymbol{\theta}),$$

where $\ell(\cdot)$ is the true log-likelihood and $\epsilon(\cdot)$ is a noise term which we assume,

$$\epsilon(\boldsymbol{\theta}) \sim N\left(0, \sigma^2_{N_p,\boldsymbol{\theta}}\right). \tag{3.4}$$

In Eq. (3.4) we assume that the variance of the noise is dependent on the parameters. This is a reasonable point to make as the estimates of the log-likelihood obtained from a particle filter will often have higher variance in the tails of the posterior than around the MAP. This noise term is also dependent on the number of particles used and so this can also influence the overall performance of the algorithm.

Stochastic optimisation techniques enable us to solve problems of this form and obtain estimates of the minimum [16, 69]. There are some common methods used to minimise these kinds of problems, such as *simulated annealing* and *genetic algorithms* [16, 69]. We consider an alternative framework to these methods which requires only two function evaluations at each iteration, and can often prove to be more efficient and easier to tune [69]. Spall [68] provides an alternative framework for stochastic optimisation referred to as the *simultaneous perturbation stochastic approximation* (SPSA) algorithm. The SPSA algorithm is briefly outlined in Section 2.5 however we note that this version of the algorithm is presented in its simplest form and we will discuss extensions in the following section. As the primary source of computational expense is the simulation component involved with estimating the loss function, the SPSA algorithm offers improved efficiency over other stochastic optimisation techniques as it requires only one to two function evaluations on each iteration [69].

3.4 SPSA for MAP estimation

When conducting inference on epidemic models, we are working in parameter spaces where our only *a priori* knowledge is captured in the prior distributions. The parameter space typically allows only biologically valid quantities and this results in points outside the support leading to an undefined value of the loss function (Eq. (3.3)), and can lead to convergence issues if the SPSA algorithm is used naively. The particle filter may also struggle to obtain estimates of the log-likelihood within the support, as certain parameter configurations may simply be unlikely to produce the observed data. This means we also need to be careful to explore the parameter space appropriately, particularly during the early iterations. Solutions to these, and other issues, are outlined in this section.

If we simply applied Algorithm 5 to the loss function in Eq. (3.3) for a given model, problems are quickly encountered. The issues which arise are due to the constraints on the parameter space, the use of a logarithmic transformation and the variance in the estimates of the log-likelihood. In order to appropriately determine the MAP, we need to modify the general SPSA algorithm to promote more careful exploration of the parameter space. To begin with we will detail some choices regarding the direct implementation of the algorithm which will focus on the perturbation vector $\boldsymbol{\Delta}$, a blocking process and the

gradient approximation.

Perturbation distribution

The inverse moments property of the perturbation vector $\boldsymbol{\Delta}$, mentioned in Step 3 of Algorithm 5, is important to the overall performance of the algorithm. This condition states that for the elements Δ_i of the perturbation vector $\boldsymbol{\Delta}$, the following must hold for it to be a reasonable perturbation vector,

$$\mathbb{E}\left[\Delta_i\right] = 0, \tag{3.5}$$

$$\mathbb{E}\left[\left|\frac{1}{\Delta_i}\right|\right] < \infty. \tag{3.6}$$

Distributions that satisfy these conditions are those with no point mass arbitrarily close to, or at, 0 but that still have mean 0 [7, 69]. This yields some obvious choices of appropriate distributions as per the conditions in [69]. A common and optimal distribution for practical implementation is to assume that the Δ_i are Bernoulli random variables which can take values $+1$ or -1 with equal probability (called a Rademacher distribution). This is the distribution that we will assume for the work in this thesis, as the guidelines in [69] provide a reasonable starting point for the choice of other search parameters under this assumption. Examples of distributions which do not satisfy the conditions are a standard normal distribution as well as a uniform distribution symmetric about 0.

Blocking processes and gradient estimation

In Step 7 of Algorithm 5, we can estimate the gradient using either of the finite difference approximations (Eq. (2.26) and (2.27)). A one-sided approximation, which does not require simultaneous perturbations, proves to be much more efficient for our problems. This enables us to reuse the estimate from the previous iteration and still maintain the consistency of the simultaneous perturbation part of SPSA whereby the noise terms cancel in the Taylor-series expansion [69]. In Step 7 we estimate the gradient as,

$$\nabla f\left(\boldsymbol{\theta}^{(k)}\right) \approx \hat{\boldsymbol{g}}^{(k)} = \left(f\left(\boldsymbol{\theta}^{(k)} + \boldsymbol{h}\right) - f\left(\boldsymbol{\theta}^{(k)}\right)\right)\left(\boldsymbol{h}^{(k)}\right)^{-1}, \tag{3.7}$$

where $\boldsymbol{h}$ is the perturbation step which we will define shortly.

The single sided finite difference has also been shown to perform better when working in a constrained space, as we can project violating parameters to the nearest feasible point [7]. This can be carried out as we can just evaluate the estimate away from the boundary,

and so the search is less likely to have instabilities in regions of the parameter space near boundaries of the support. This version of the gradient approximation is advantageous in cases where we are near a boundary, or in a region of the posterior space where the simultaneous perturbation required for the central finite difference only results in one (or both) of the loss evaluations, at the perturbed points, being undefined. The function can be re-evaluated at the beginning of each iteration at the cost of doubling the runtime of the algorithm, which can potentially lead to better results. Choosing to reevaluate the function reduces the chance of the algorithm getting stuck as a result of underestimating the loss. This is a particularly important consideration if there is a reasonable degree of noise in loss function estimates. An alternative approach could be to average multiple evaluations of the loss function but this will typically come at large computational expense.

A way to increase the likelihood of convergence of the algorithm is to consider a blocking process, whereby we reject a proposed move if there would be a suspiciously large change in the parameters, or if there is no relative improvement to the value of the loss function [69]. Both of these blocking processes require some understanding of the problem at hand. The first approach can be demonstrated to perform better, but it is difficult to classify what can be considered a "reasonable" change in the parameters without appropriate knowledge of the target distribution [69]. For this reason, the latter approach is used for the problems in this thesis. The reasonable change in the objective function is defined on a problem-by-problem basis, but should typically be chosen to enable a move to a point with a lower value of the loss function, in order to search towards a potential better minimum. Consider two points $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$. If $\boldsymbol{\theta}_2$ is a better estimate of the MAP compared to $\boldsymbol{\theta}_1$ then,

$$f(\boldsymbol{\theta}_2) < f(\boldsymbol{\theta}_1)$$

or,

$$f(\boldsymbol{\theta}_2) - f(\boldsymbol{\theta}_1) < 0.$$

Using this as a blocking process would mean that we only accept moves to points with smaller loss. A way to enable moves to slightly worse points is to have,

$$f(\boldsymbol{\theta}_2) - f(\boldsymbol{\theta}_1) < \epsilon.$$

where $\epsilon > 0$. This results in us being able to move to a worse estimate in order to eventually find a better estimate in the future. This parameter can be a little difficult to choose but if one evaluates the loss function at some number of starting points (which can be sampled from the prior) the difference in loss value between the sampled points can be used to make an informed choice. As $\epsilon \to \infty$ this simply means that the search will not block any move, regardless of size or whether it dramatically worsens the estimate.

A useful addition to the SPSA algorithm is to store the current best estimates of the MAP and loss function. This ensures we allow the search to continue to explore the

region near the MAP and potentially improve the current estimate. This also reduces the influence in the process of searching away from the optimum, which can occur due to the stochastic nature of the perturbation vector as well as the noise in the measurements of the objective function. An issue with this approach is that if the current estimate is from a large overestimate of the loss function then the search can become stuck. We can mitigate the effect of this by re-evaluating the loss function at the minima each iteration. While this does introduce at least twice the computational expense at each iteration, testing showed strong improvements to the results of the search.

Choice of gain sequences and parameters

An important consideration in the implementation of SPSA are the choices of the gain sequences $c^{(k)}$ and $a^{(k)}$. These sequences serve different purposes in the algorithm but express the same idea. As the number of iterations goes on, we expect to be approaching the minimum of the function, and as such we want to make smaller moves to reduce the influence of the measurement noise. As per the suggestions in [69], this can be asserted by assuming the following forms for the gain sequences for $k \in \mathbb{Z}^+ \cup \{0\}$,

$$c^{(k)} = \frac{s}{(k+1)^\gamma}, \tag{3.8}$$

$$a^{(k)} = \frac{A}{(B+1+k)^\alpha}, \tag{3.9}$$

where $s, A, B > 0$ and $\alpha \in (0,1)$ and $\gamma \in (0,1)$ are chosen such that,

$$\begin{aligned} \alpha - 2\gamma &> 0 \\ 3\gamma - \frac{\alpha}{2} &\geq 0. \end{aligned} \tag{3.10}$$

In practice, the user is open to specify these parameters on the basis of their problem, however Spall [69] does provide some reasonable starting points, assuming a Rademacher perturbation distribution. Under this assumption, it is optimal to set $\alpha = 0.602$ and $\gamma = 0.101$. This choice maintains a relatively large step size for each of the gain sequences. Under this assumption it is also reasonable to set s equal to the standard deviation in the measurements of the initial estimate of the gradient at the initial point $\boldsymbol{\theta}_0$ [69]. In this thesis we will use ς_0 to denote the standard deviation in the initial gradient estimate. To calculate this, we evaluate $\hat{\boldsymbol{g}}^{(0)}$ some number of times and take the standard deviation of the results.

Another consideration is that elements of the target point $\boldsymbol{\theta}$ may have different magnitudes. To deal with this, Spall [69] suggests a matrix scaling of the gain sequences (which is just a one-to-one mapping of the gain sequences). We define a vector $\boldsymbol{s} = (s_1, \ldots, s_m)$,

and a vector valued gain sequence (which we denote in bold) for the perturbation sequence as,

$$\boldsymbol{c}^{(k)} = \frac{1}{(k+1)^{\gamma}}\boldsymbol{s}. \tag{3.11}$$

This mapping allows us to perturb different elements of $\boldsymbol{\theta}$ by different amounts, consistent with their magnitudes.

The remaining parameters A and B can be defined together. Define B as a value equal to 10% or less of the expected total iterations [69]. This can be chosen by assessing how much movement is expected in the $\boldsymbol{\theta}$ estimates during the earlier iterations. The larger the initial value of B, the smaller the initial update steps. Suppose that the smallest expected magnitude movement in the elements of $\boldsymbol{\theta}$ during the early iterations is δ, then calculate A as,

$$A = \frac{B^{\alpha}}{\bar{g}_0}\delta, \tag{3.12}$$

where, $\bar{g}_0$ is defined as the average magnitude of the elements in the initial gradient estimates. $\bar{g}_0$ is calculated using independent perturbations $\boldsymbol{\Delta}_j$, for $j = 1, \ldots, J$ and we evaluate the estimate of the gradient $\hat{\boldsymbol{g}}^{(0)}$ for each of these to give $\hat{\boldsymbol{g}}_j^{(0)}$, for $j = 1, \ldots, J$ which is calculated using Eq. (3.7). In the case where these are the same magnitude we then simply average over the absolute value of one of the elements (noting that the elements will be the same with different signs).

In the case where the elements are of different magnitudes, we consider a vector $\bar{\boldsymbol{g}}_0$ and set element ℓ of this to,

$$\bar{\boldsymbol{g}}_{0,\ell} = \frac{1}{J}\sum_{j=1}^{J}\frac{\hat{\boldsymbol{g}}_{j,\ell}^{(0)}}{s_{\ell}}\boldsymbol{\Delta}_{j,\ell}.$$

This form takes into consideration the different sizes of parameters such as quantities constrained to intervals $(0, 1)$ as opposed to those constrained to $(0.1, 10)$.

We apply a similar extension to the update sequence to account for differing magnitudes of parameters. We use a vector $\boldsymbol{\delta} = (\delta_1, \ldots, \delta_m)$ where δ_i is approximately the expected movement of parameter $i = 1, \ldots, m$ during the early iterations. With this definition, a vector valued form of Eq. (3.9) is,

$$\boldsymbol{a}^{(k)} = \frac{B^{\alpha}}{(B+1+k)^{\alpha}}\bar{\boldsymbol{g}}_0^{-1} \circ \boldsymbol{\delta}, \tag{3.13}$$

where $\circ$ denotes element-wise multiplication. With these considerations the update step is,

$$\boldsymbol{\theta}^{(k+1)} = \boldsymbol{\theta}^{(k)} - \boldsymbol{a}^{(k)} \circ \hat{\boldsymbol{g}}^{(k)}. \tag{3.14}$$

Issues with constraints and convergence

The two primary issues when attempting to apply SPSA to estimate the MAP when we use a particle filter to estimate the log-likelihood are:

1. Constraints imposed by the support of the parameters; and,
2. the variance of the likelihood estimates obtained from the particle filter.

Both issues influence the performance and results of the search algorithm and hence need to be dealt with in order to obtain reliable MAP estimates.

For any model, the parameters $\boldsymbol{\theta} \in \Theta$ have support defined by intervals $[a_i, b_i]$ for $i = 1, \ldots, m$, where m is the number of parameters of the model. Clearly any point outside of the set,

$$\Theta = \Big\{ \boldsymbol{\theta} \Big| \theta_i \in [a_i, b_i], \text{ for } i = 1, \ldots, m \Big\},$$

returns a non-finite value of the loss function (Eq. (2.28)) as the prior probability would be 0. There are several ways to deal with constraint violations provided in [69], but each of these assumes that the loss function is able to be evaluated outside of the support, which is not the case here. We are working with the log-posterior density and so outside of the support, the log-density is non-finite which leads to divergence in the search.

An issue which further complicates matters is that the estimates of the log-likelihood are obtained through simulation. The reason this complicates matters is that there can be configurations of parameters which are very unlikely to be consistent with the observed data. This can result in non-finite estimates of the loss function and can be another cause of divergence. These constraints are not trivial to determine *a priori*. Furthermore, constraints are related to the number of particles used as well as the conditions in the simulations.

An obvious thought is that the issue of poor parameter configurations could be solved or reduced by considering more appropriate prior distributions. Unfortunately, this is not the case. Typically, a well chosen prior will not even guarantee we can estimate the log-likelihood over the full support. There are likely to be regions where the combinations of parameters are so unlikely to be consistent with the data, that the likelihood is essentially zero. The combination of appropriate prior distributions and efficient particle filtering approaches (such as those which use importance sampling) can aid in providing wider coverage over the parameter space, allowing us to estimate the log-likelihood for even unlikely parameter configurations. That being said, these considerations do not guarantee that the issues are alleviated and so we still need to be careful to ensure the search has converged to the MAP.

The issues relating to unlikely parameter configurations are quite influential in terms

of practical implementations. There are a few ways to modify the algorithm in order to work around these issues and increase performance. An adjustment to reduce issues of divergence in the search is to consider a rescaling of the additive factors from the perturbation step and the update steps in Algorithm 5 by some scalar factor $r > 0$ whenever the proposed $\theta \notin \Theta$. A rescaling can be seen to be suitable by considering when the gain sequences ($\boldsymbol{c}^{(k)}$ and $\boldsymbol{a}^{(k)}$) are used in the SPSA algorithm. The perturbation gain sequence, $\boldsymbol{c}^{(k)}$, appears only in the calculation of the estimate of the derivative,

$$\hat{\boldsymbol{g}}^{(k)} = \left(y(\boldsymbol{\theta}^{(k)} + \boldsymbol{C}^{(k)} \circ \boldsymbol{\Delta}^{(k)}) - y(\boldsymbol{\theta}^{(k)})\right) \left(\boldsymbol{C}^{(k)} \circ \boldsymbol{\Delta}^{(k)}\right)^{-1}, \tag{3.15}$$

and rescaling by $r \in (0, 1)$ yields,

$$\hat{\boldsymbol{g}}^{(k)} = \left(y(\boldsymbol{\theta}^{(k)} + r\boldsymbol{C}^{(k)} \circ \boldsymbol{\Delta}^{(k)}) - y(\boldsymbol{\theta}^{(k)})\right) \left(r\boldsymbol{C}^{(k)} \circ \boldsymbol{\Delta}^{(k)}\right)^{-1}, \tag{3.16}$$

which has the same form as Eq. (3.15). Hence, the rescaled estimate is still a suitable estimate of the gradient at $\boldsymbol{\theta}^{(k)}$ but one which uses a smaller step size. This approach can also be used if the log-likelihood at the perturbed point is undefined ($-\infty$). Rescaling alongside a suitable initial point can minimise this issue.

The other possible point of divergence is Step 8 of Algorithm 5. The update gain sequence, $\boldsymbol{a}^{(k)}$, is used only to update the current estimate of the MAP,

$$\boldsymbol{\theta}^{(k+1)} = \boldsymbol{\theta}^{(k)} - \boldsymbol{a}^{(k)} \circ \hat{\boldsymbol{g}}^{(k)},$$

and rescaling by $r \in (0, 1)$ yields,

$$\boldsymbol{\theta}^{(k+1)} = \boldsymbol{\theta}^{(k)} - r\boldsymbol{a}^{(k)} \circ \hat{\boldsymbol{g}}^{(k)},$$

which simply reduces the traversal distance in the approximate steepest descent direction but does not alter the direction of the search. While this appears inefficient, as with the other condition, it merely prevents large steps outside of the support and ensures consistency during the search.

Choosing parameters for SPSA

The parameters specified in Section 2.5, for practical implementation of the SPSA algorithm in estimating the MAP, are,

$$(\boldsymbol{\theta}_0, \epsilon, \boldsymbol{\delta}, \boldsymbol{s}, B, M, N_p), \tag{3.17}$$

where, $\boldsymbol{\theta}_0$ is the initial point, ϵ is the tolerance in the difference in the loss function at the proposed point compared to the current iteration, $\boldsymbol{\delta}$ is as specified for the update step

(Eq. (3.13)), $\boldsymbol{s}$ is as specified for the perturbation step size (Eq. (3.11)), B is chosen to be 10% or less of the total number of iterations, M is the number of iterations and N_p is the number of particles. These parameters are key to the performance of the SPSA algorithm. While each problem will require individual attention, we provide some indication to the thought process in choosing each parameter values here. We note that we explore some approaches for finding better choices of the parameters in the results section later in this chapter. The advice presented here is to complement that and should be used as a guide.

Of these parameters, the initial point $\boldsymbol{\theta}_0$ is chosen quite easily. We can sample point(s) from the prior distribution such that the loss function (Eq. 3.3) has some defined function value. While any starting point sampled from the prior should be appropriate, there are some considerations for a starting point to be deemed acceptable. A decent starting point should be chosen some distance from the boundaries as this allows the movements around the posterior space to be larger during earlier iterations. An initial point should also not have an excessively large variance in the loss estimate. This helps with producing lower variance estimates of the gradient during the earlier iterations. The variance being in the range of $(0, 10)$ should provide good performance, and this provides some criteria for choosing the number of particles.

The number of iterations M should be chosen in the range of $1000 - 10000$. This showed promising results for a variety of problems during testing. The rescaling factor, r is set to 0.5 for most problems. If the search appears to be mainly targeting the boundary, it is reasonable to increase r allowing the search to take larger steps and be rescaled a lesser amount. The tolerance, ϵ for most problems will be somewhere in the range of $(0, 50)$. This allows for a decrease of between 0 and 50 in the proposed value of the loss function which can improve the convergence rate. The tolerance chosen in this fashion means that a better minimum will always be accepted and only those which are marginally worse will be accepted. In testing it was found that higher values of ϵ tend to work better with lower noise in the function estimates.

SPSA for MAP estimation

Algorithm 6 provides pseudo-code for the SPSA algorithm applied to MAP estimation, with all the improvements discussed in the previous two sections. In Algorithm 6, the operator $\circ$ refers to element-wise multiplication.

Algorithm 6 SPSA for MAP estimation

Inputs Initial parameters θ_0, number of iterations M, rescaling factor r, gain sequences $\{\boldsymbol{c}^{(k)}\}_{k=1}^{\infty}$ and $\{\boldsymbol{a}^{(k)}\}_{k=1}^{\infty}$

for $k = 1 : M$ **do**
  for $i = 1 : m$, where m is the number of parameters **do**
    Sample $\Delta_i^{(k)}$ from a mean 0 distribution satisfying an inverse moments property (Eq. 3.5).
  end for
  Set $\boldsymbol{\Delta}^{(k)} = \left(\Delta_1^{(k)}, \Delta_2^{(k)}, \ldots, \Delta_m^{(k)}\right)$
  Set $\boldsymbol{h}^{(k)} = \boldsymbol{c}^{(k)} \circ \boldsymbol{\Delta}^{(k)}$
  while $y(\boldsymbol{\theta}^{(k)} + \boldsymbol{h}^{(k)})$ undefined **do**
    Set $\boldsymbol{h}^{(k)} = r\boldsymbol{h}^{(k)}$
  end while
  Calculate estimate of derivative,

$$\hat{\boldsymbol{g}}^{(k)} = \left(y(\boldsymbol{\theta}^{(k)} + \boldsymbol{h}^{(k)}) - y(\boldsymbol{\theta}^{(k)})\right)\left(\boldsymbol{h}^{(k)}\right)^{-1}$$

  Update the current estimate of the minimum using the standard stochastic approximation form,

$$\boldsymbol{\theta}^{\star} = \boldsymbol{\theta}^{(k)} - \boldsymbol{a}^{(k)} \circ \hat{\boldsymbol{g}}^{(k)},$$

  while $y(\boldsymbol{\theta}^{\star})$ undefined **do**
    Set $\boldsymbol{a}^{(k)} = r\boldsymbol{a}^{(k)}$
    Update estimate,

$$\boldsymbol{\theta}^{\star} = \boldsymbol{\theta}^{(k)} - r\boldsymbol{a}^{(k)} \circ \hat{\boldsymbol{g}}^{(k)}$$

  end while
  if $y(\boldsymbol{\theta}^{\star}) - y(\boldsymbol{\theta}^{(k)}) < \epsilon$ **then**
    Accept move, $\boldsymbol{\theta}^{(k+1)} = \boldsymbol{\theta}^{\star}$
  else
    Reject move, $\boldsymbol{\theta}^{(k+1)} = \boldsymbol{\theta}^{(k)}$
  end if
  if $y(\boldsymbol{\theta}^{(k)}) < y(\boldsymbol{\theta}_{\min})$ **then**
    $\boldsymbol{\theta}_{\min} \leftarrow \boldsymbol{\theta}^{(k)}$
  end if
end for

Return Search path $\left\{\boldsymbol{\theta}^{(k)}\right\}_{k=1}^{M}$ and estimate of the minimum $\boldsymbol{\theta}_{\min}$

3.5 Application to the SIR model

In the remainder of this chapter we demonstrate the application of the SPSA algorithm in estimating the MAP for two epidemic models of increasing complexity. The first example relates to estimating the MAP of an SIR model. We simulated a dataset of daily infection counts for an outbreak which lasted 10 days and as such we expect no issue for the performance of appropriate particle filters as a result of the length of the time series. The daily counts are relatively small and there do not appear to be any rare events and so one would expect that the use of the bootstrap, alive, or importance sampling filters would work sufficiently in calculating estimates of the likelihood. However, there could be an increase in the required computation time of the algorithm as a result of less efficient methods due to the number of particles needed. For this example as well as subsequent implementations in this chapter we apply a particle filter which uses importance sampling as this is the most efficient method. The second example looks to estimate the MAP for a SEIAR model. The SEIAR model is an extension of the SIR model which incorporates an exposed phase following infection. This model also adds a presymptomatic infectious phase and the possibility of undetected cases. This requires a much more complex particle filter and increased computational expense arises as a result. This example provides us a foundation for demonstrating the types of problems the search can be successfully applied to. The SEIAR example also explores the care required in choosing the parameters of the search when elements of $\boldsymbol{\theta}$ have different magnitudes (i.e. constrained to $(0, 0.5)$ instead of $(0, 10)$). This model allows for much more complex epidemic dynamics and hence demonstrates the algorithms performance to target the MAP in higher dimensional problems.

For a reference point, we sample from the relevant posterior distributions using a pmMH procedure as outlined in Section 2.3. We then apply the adaptive kernel density estimator (AKDE) of Botev *et al.* [11] in order to obtain an estimate of the MAP. This will allow us to compare the results of SPSA against the gold standard of kernel density estimation. To use AKDE, we require a sample from the posterior which is representative, meaning that the parameter space has been explored appropriately. In order to obtain a sample that is representative, we can construct very long chains (with optimal acceptance rates) and this should ensure that we have reasonably explored the parameter space. Targeting an effective sample size (ESS) in the range of 1000–10000 should ensure this.

For the SIR model the basic parameters are (β, γ), however interest is in the practically interpretable parameters $R_0 = \beta/\gamma$ and $1/\gamma$ and hence we parametrise the model in terms of these,

$$\boldsymbol{\theta} = \left(R_0, \frac{1}{\gamma}\right). \tag{3.18}$$

We initially look at what can be considered an ideal scenario of a small outbreak and a simple model. In the simulation we assume an initial condition of $\boldsymbol{Z}(0) = (1, 0)$ and take the population size to be $N = 100$. The parameters used to simulate the data are $R_0 = 1.5$ and $1/\gamma = 1$ where $1/\gamma$ is the infectious period in days. The simulated time series is plotted in Figure 3.1.

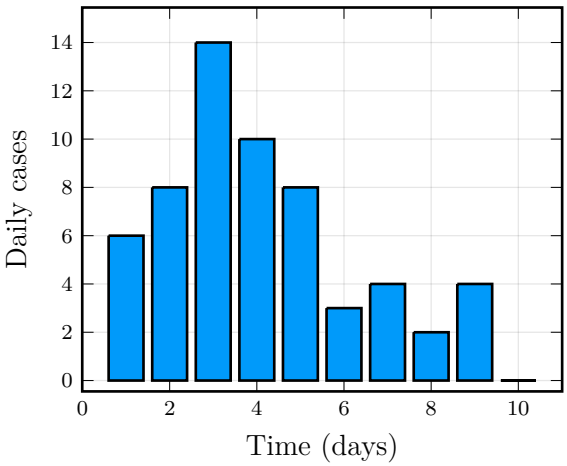

Figure 3.1: Daily incidence for the simulated epidemic, simulated using the SSA.

We assign priors to the parameters of,

$$R_0 \sim \text{Gamma}(10, 0.2)$$
$$\frac{1}{\gamma} \sim \text{Gamma}(1.4, 2.7).$$

The prior distribution of R_0 is chosen such that the mode is 1.8 and the mean is 2. This ensures that both measures of location are centred close to the value of R_0 used to simulate the data (2). The prior on $1/\gamma$ is chosen in a similar fashion but has mode of approximately 1, in line with the true value, but a much larger spread.

3.5.1 Choosing the key parameters

We begin by assessing the sensitivity of the search to varying $\boldsymbol{\delta}$ and $\boldsymbol{s}$. These two parameters are closely related and so it is more effective to assess combinations of them to provide some indication as to appropriate choices. In the remainder of this section we use initial points randomly sampled from prior and these are shown in Figure 3.2. While this is not the optimal implementation, the dimensionality of the problem enables this simplified approach of starting point selection to suffice. The initial set of points is chosen such

that there is high dispersion in the initial points and this enables us to quantify general search parameter choices which lead to convergence for most starting points.

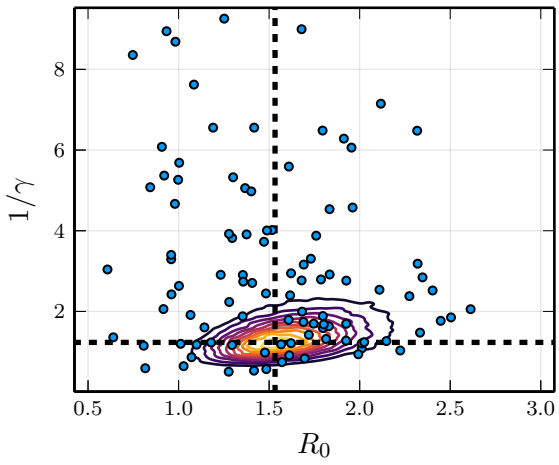

Figure 3.2: Scatter plot of the starting points, indicated in blue, for the 100 searches. The posterior distribution is indicated by the contour plot and the estimate of the MAP from the AKDE approach is indicated by the black dashed lines.

In order to assess the sensitivity of the search to the choices of $\boldsymbol{\delta}$ and $\boldsymbol{s}$ we fix the other values. We choose the number of particles to use as $N_p = 60$ which offered reasonable noise without introducing it being the dominating factor. We fix the number of iterations at $M = 1000$ which also enables us to quantify convergence over a smaller number of samples. The other parameters are set for the entirety of this section. The tolerance is set at $\epsilon = 5$ which we have observed to have minimal impact on the results and is mainly to limit aggressively large moves in more complex models. We assume $r = 0.5$ and take $B = 0.1M$ [69].

We consider the following values

$$\begin{aligned}\boldsymbol{\delta} &\in \{0.01, 0.1, 0.25, 1.00\},\\ \boldsymbol{s} &\in \{10\varsigma_0\boldsymbol{L}, 2\varsigma_0\boldsymbol{L}, \varsigma_0\boldsymbol{L}/2, \varsigma_0\boldsymbol{L}/10\},\end{aligned}$$

where $\boldsymbol{L}$ has elements which are the length of the interval parameter i is supported on. The variable ς_0 is the standard deviation in the initial gradient estimate. The full set of scatter plots are featured in Figure 3.3 and the better results from the bottom row of that plot are shown in Figure 3.4. Titles for each subplot show the combination, $(\boldsymbol{\delta}, \boldsymbol{s})$, of parameters used.

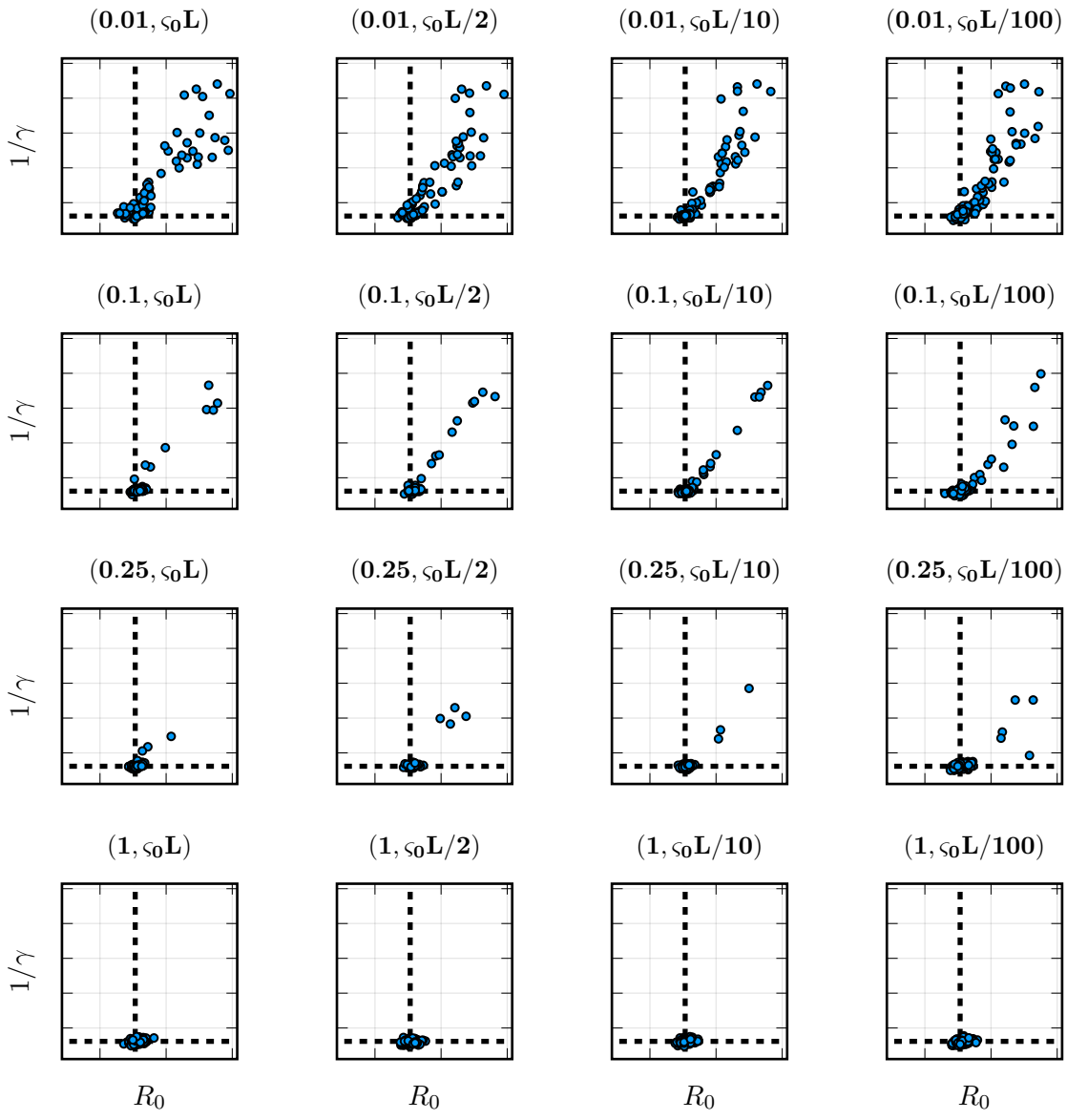

Figure 3.3: Scatter plot of MAP estimates, indicated by the blue points, for different combinations of $\boldsymbol{\delta}$ and $\boldsymbol{s}$. Dashed black lines indicate the estimate of the MAP from the AKDE approach. The combination of $(\boldsymbol{\delta}, \boldsymbol{s})$ is shown in the title of each subplot.

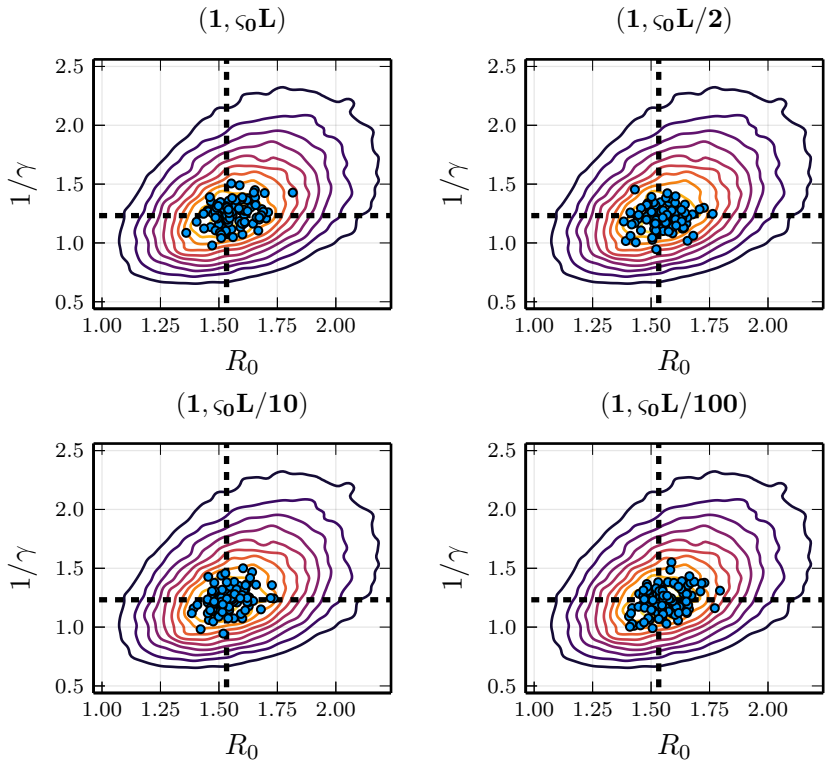

Figure 3.4: Scatter plot of MAP estimates, indicated by the blue points, for different combinations of $\boldsymbol{\delta}$ and $\boldsymbol{s}$. Dashed black lines indicate the estimate of the MAP from the AKDE approach. The contour plot shows the posterior distribution. The combination of $(\boldsymbol{\delta}, \boldsymbol{s})$ is shown in the title of each subplot.

We see that the bottom two rows of Figure 3.3 show improved convergence. Thus suggesting that larger $\boldsymbol{\delta}$ is crucial to improved convergence over a lower number of iterations. We can see in the bottom two rows of Figure 3.3 that with this choice, the choice of $\boldsymbol{s}$ is not as important. This is particularly relevant in the bottom row. In Figure 3.4 we have shown this bottom row in increased detail and we see no discernible differences between the different choices of $\boldsymbol{s}$.

The results here are for a smaller number of iterations and we expect that all estimates, regardless of the gain sequence parameters, will improve as the number of iterations increases. The number of iterations is something which directly influences the performance of the algorithm, as the runtime scales linearly with the number of function evaluations. We can control the performance of the search by adjusting the number of iterations and the number of particles used.

3.5.2 Choosing the number of iterations

There are two approaches to improve the accuracy of the search estimates, and that is to either run the search for a larger number of iterations, or to re-run the search using the newly obtained estimate as the starting point. In the case of a larger number of iterations, typically the user can get away with using larger values of $\boldsymbol{\delta}$ and $\boldsymbol{s}$. This is as the search will hone in on the MAP slower due to the form of the update equation. Repeated runs tend to solve the issue of determining a good starting point as we can simply run the search to obtain a better estimate. Repeated running also tends to favour a reduced number of total iterations and hence reduced computational budget as each time the search is run the precision tends to improve. This is an obvious approach in optimisation and hence we focus on altering the number of iterations here. We fix the search parameters of $\boldsymbol{\delta} = (0.25, 0.25)$ and $\boldsymbol{s} = \varsigma_0 \boldsymbol{L}/2$. We choose the gain sequence parameters to be slightly less optimal than what was found in the previous section to more appropriately assess the effect of the number of iterations on how well the searches converge.

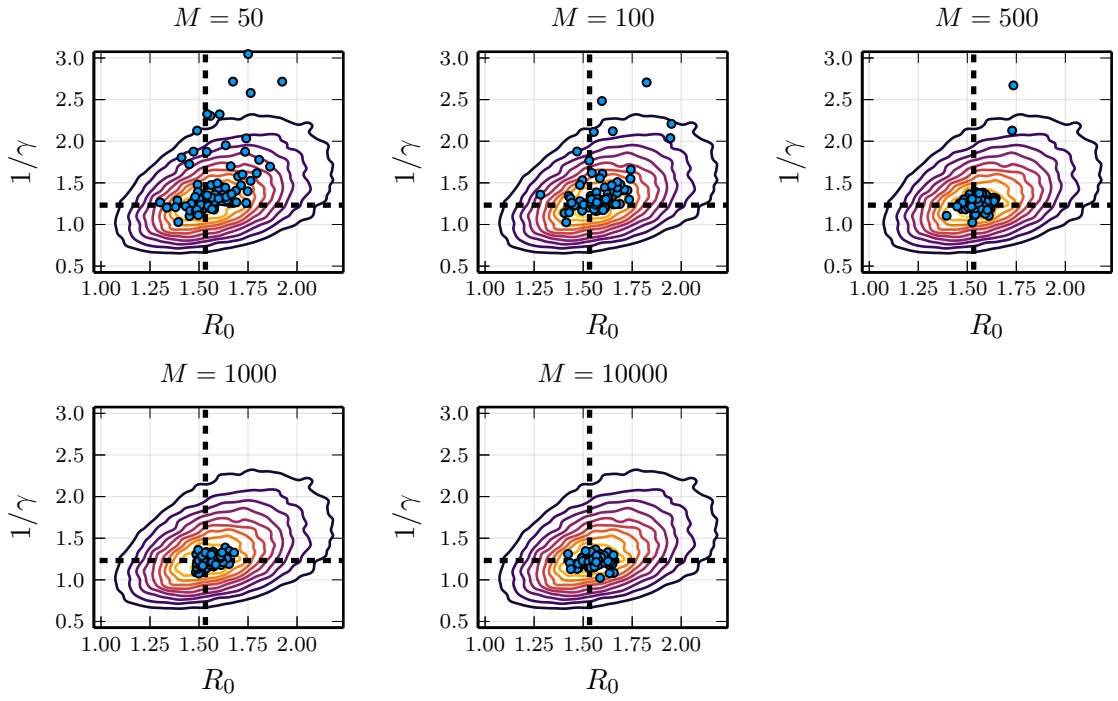

Figure 3.5: Scatter plot of MAP estimates, indicated by the blue points, for different number of iterations. The contour plot shows the posterior distribution. Dashed black lines indicate the estimate of the MAP from the AKDE approach. The number of iterations M is shown in the title of each subplot.

Figure 3.5 shows results we expect for a search algorithm. As the number of iterations increases, we see improved convergence to the global optimum. We see substantial improvement from increasing the number of iterations from $M = 100$ to those for $M = 500$ and $M = 1000$. The key observation is that increasing the number of iterations to 10000 results in all points converging quite reasonably. There is relatively low improvement from the $M = 500$ case to the $M = 1000$ and $M = 10000$ cases suggesting that a reasonable number of iterations would be in the range $M \in (500, 10000)$.

3.5.3 Choosing the number of particles

Another consideration is to see how the number of particles in the particle filter influences the MAP estimates. We again fix the search parameters of $\boldsymbol{\delta} = (0.25, 0.25)$ and $\boldsymbol{s} = \varsigma_0 \boldsymbol{L}/2$ for the same reason as per the iteration analysis. We also choose to fix $M = 1000$ as this assesses the finite sample performance. We consider the following number of particles,

$$N_{\text{part}} \in \{5, 10, 25, 50, 100\}$$

In Figure 3.6 we see that the number of particles does indeed influence the convergence of the search. As the number of particles increases we see more of the points converge reasonably close to the MAP. There is minimal difference between using $N_p = 50$ and $N_p = 100$ particles. This is likely a result of more than 50 particles resulting in low variance estimates and so the search will perform relatively consistently.

3.5.4 Sensitivity to the initial point

The final point we seek to investigate regarding the SIR example is the convergence properties and sensitivity due to the dispersion of the starting points. In order to understand the convergence of the search we use search parameter values of,

$$\Big(\boldsymbol{\theta}_0, N_p = 100, \boldsymbol{\delta} = (0.5, 0.5), \boldsymbol{s} = \varsigma \boldsymbol{L}/2, B = 1000, M = 10000, \epsilon = 0.4\Big).$$

The choice of $\boldsymbol{\delta}$ and $\boldsymbol{s}$ are motivated by the previous discussions. We choose to use 10000 iterations to allow the search a reasonable number of steps to converge regardless of where the initial point is in the parameter space. To assess the convergence we sample 1000 starting points from the prior a distance 0.5 from the nearest boundary and these are shown in Figure 3.7.

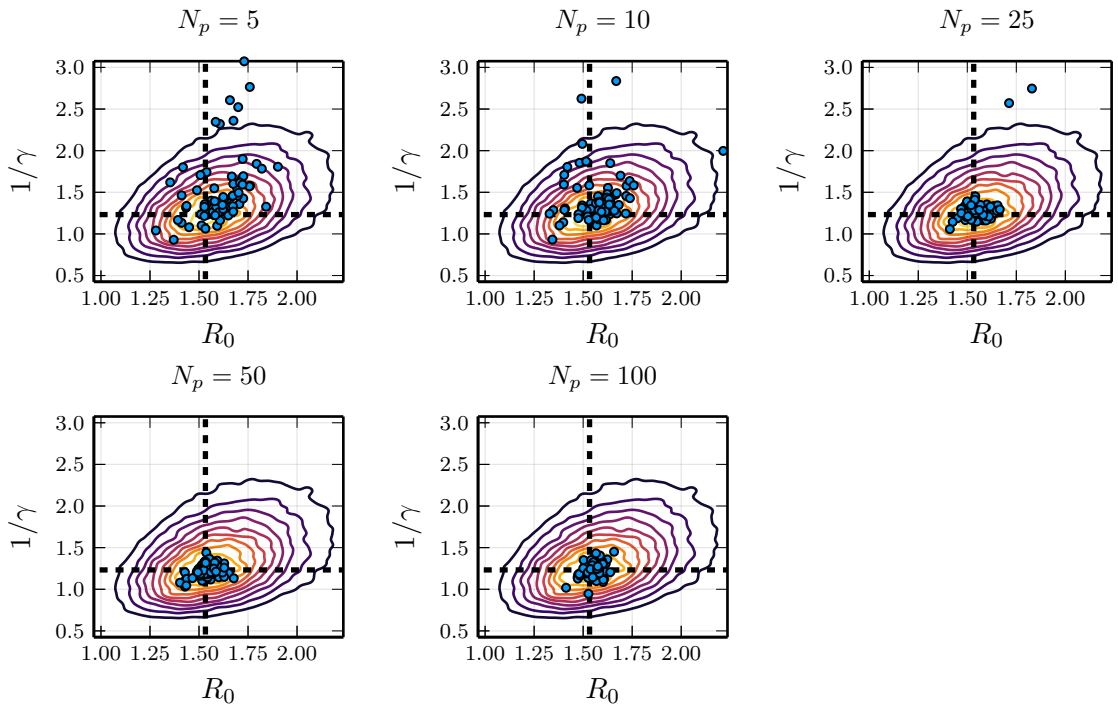

Figure 3.6: Scatter plot of MAP estimates, indicated by the blue points, for different numbers of particles. The contour plot shows the posterior distribution. Dashed black lines indicate the estimate of the MAP from the AKDE approach. The number of particles N_p is shown in the title of each subplot.

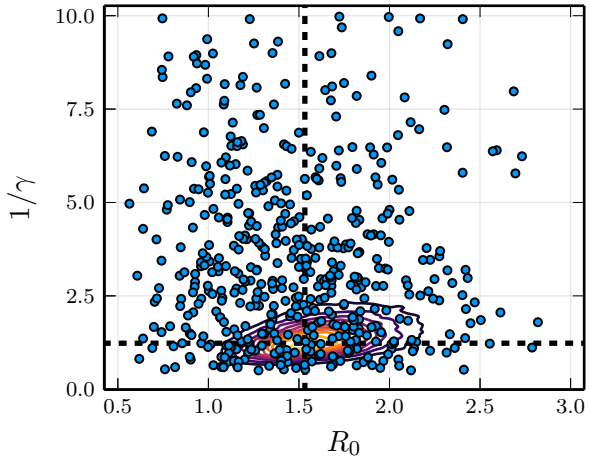

Figure 3.7: Scatter plot of the starting points, indicated in blue, for the 1000 searches. The posterior density is indicated by the contour plot and the estimate of the MAP from the AKDE approach is indicated by the black dashed lines.

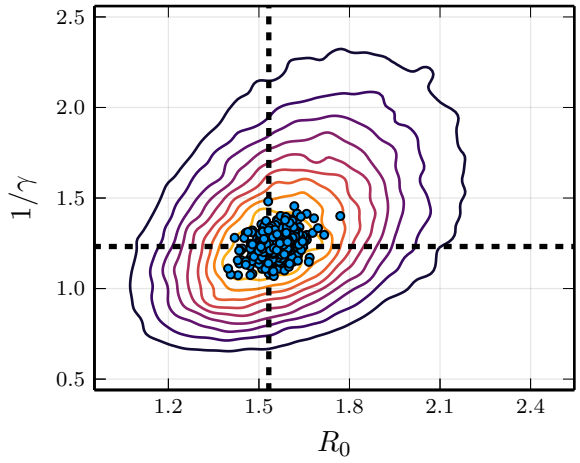

Figure 3.8: Scatter plot of MAP estimates, indicated by the blue points, for the 1000 searches. The contour plot shows the posterior distribution. Dashed black lines indicate the estimate of the MAP from the AKDE approach.

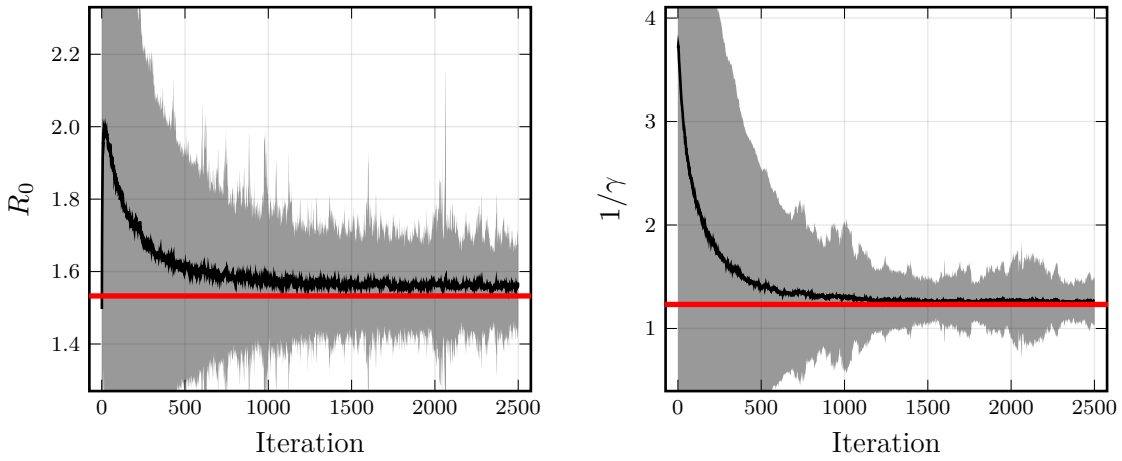

Figure 3.9: Plot of the sample path behaviour of 1000 independent searches. The grey band indicates 95% confidence interval of the search estimates and the black line denotes the average estimate across the searches. The red line indicates the estimate of the MAP from the AKDE approach.

Figure 3.8 shows a contour plot of the posterior distribution of $(R_0, 1/\gamma)$. The AKDE estimate of the MAP is indicated using the black dashed lines and the blue dots show the MAP estimates from the SPSA search. We see relatively strong convergence to the region of highest posterior density. At first, the spread across this region could appear to be slightly concerning, but this is simply the product of the dispersed set of starting points and a the stochastic nature of the search. Improved estimates can be obtained by rerunning the search with a smaller $\boldsymbol{\delta}$, however for tuning purposes all these points will suffice. The sample path behaviour of the search is shown in Figure 3.9. We see that the mean over the searches (indicated by the black line) converges to the MAP estimate obtained through the AKDE method (indicated by the red line). Convergence is seen to occur after around 1000 iterations and this agrees with the results of study of the choices of $\boldsymbol{\delta}$ and $\boldsymbol{s}$.

Method	R_0	$1/\gamma$
AKDE	1.533	1.232
SPSA	1.557 (0.048)	1.243 (0.061)

Table 3.1: MAP estimates for the SIR model. The estimates for the SPSA method are averaged across the 1000 independent searches. Each row of the table corresponds to the method used and each column corresponds to a parameter. Standard deviations are given in parentheses and are only provided for the search algorithm as AKDE returns the same result for the same bandwidth and choice of kernel.

Method	Loss (from search)	Loss (post search)
AKDE	–	26.807 (0.255)
SPSA	26.08 (0.286)	26.801 (0.220)

Table 3.2: Loss values for each method. Each row of the table corresponds to the method used and each column corresponds to the measured loss. Standard deviations for the evaluated loss are given in parentheses. The "Loss (from search)" column represents the best estimate of the loss found during the SPSA algorithm. The "Loss (post search)" is the estimated loss at the average value obtained through the different methods when evaluating the loss function 25000 times.

In Table 3.1 we see that the average result of the SPSA algorithm for each method and the AKDE algorithm are in agreement. In Table 3.2 we show the average loss estimated during the searches (from search) and following the search (post search). To obtain the loss estimates post search we run the particle filter at the average MAP estimate and

average the loss function estimates. The post search loss values show that the search algorithm returns a better average estimate of the MAP compared to the AKDE method. An interesting result is that the average loss value found during the search is much lower than the estimate found by the AKDE approach. This can be understood better by looking at the MAP estimates which yielded the better outcomes. The point with the best value of the loss function (during the search) was $\theta_1 = (1.64, 1.36)$ with an estimated loss function value of $f(\theta_1) = 25.78$.

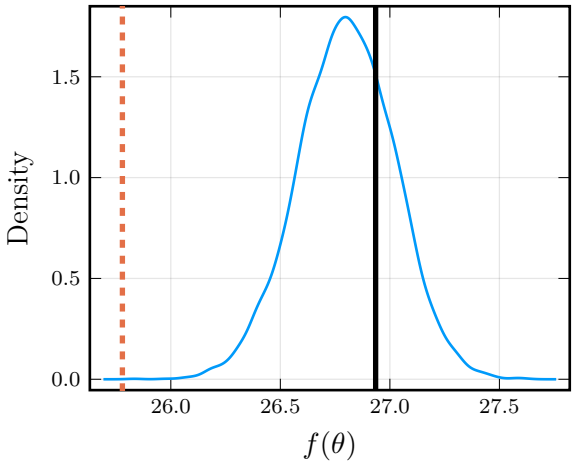

Figure 3.10: Kernel density estimate of the loss values at the best estimate of the MAP. Average loss value at the AKDE MAP is indicated by the solid black line. The estimated value of the loss function found during the search is indicated by the dashed red line.

Figure 3.10 shows the distribution of the loss function values evaluated at the best estimate of the MAP. The loss function was evaluated 50000 times at θ_1 and the kernel density estimate of the loss function value was obtained, this is shown by the blue curve. The dashed red line is the estimated value of the loss function obtained during the search algorithm. The black line indicates the value of the loss function for the point obtained through the AKDE method. It is clear that the loss function was underestimated for this particular point and hence the search stored this as the best estimate and the search got stuck. We can still see that this point would be slightly better than the estimate from the AKDE method, but it was grossly underestimated.

Restarting the search from this point would result in re-evaluation of the MAP estimate and start the search over again. Another approach is to the increase the number of particles which reduces the variance and hence the search is less likely to over- or under-estimate the loss function by a severe amount. Instead of increasing the number of particles we could parallelise the filters which would also reduce the variance in the likelihood estimates and further improve the searching capabilities near the optimum. This

would be the case because the variance in the estimates would reduce and so our search would perform better.

Another workaround is to reevaluate the loss function at the current minima. This drastically reduces the chance of the search getting stuck in this fashion as the loss function is unlikely to be drastically over- or under-estimated by a large amount over several tries. However this approach requires twice the number of loss evaluations hence leading to around twice the runtime. This is due to the overall runtime of the search effectively being the runtime of running the particle M (the number of iterations) times. This approach of reevaluation is something we have observed to be a more critical choice when a more precise estimate of the MAP is necessary. This approach is what we use for the SEIAR model in the next example.

Another way to improve the search is to apply parallelism to the search itself. This can be done by running multiple, independent searches on different cores from the same initial point. Through these searches we can store the best obtained estimate and then average across these searches and obtain an improved estimate. This approach means that we obtain multiple estimates of the MAP per search at the computational cost of running the algorithm sequentially.

This version of the algorithm was tested on the same example using 4 cores and we saw a halving in the variance of the MAP estimates obtained. While the implementation of parallelising the code is not trivial in most cases, it offers a drastic improvement. We used 80 particles in the filter and used the optimal values of the parameters for the searches.

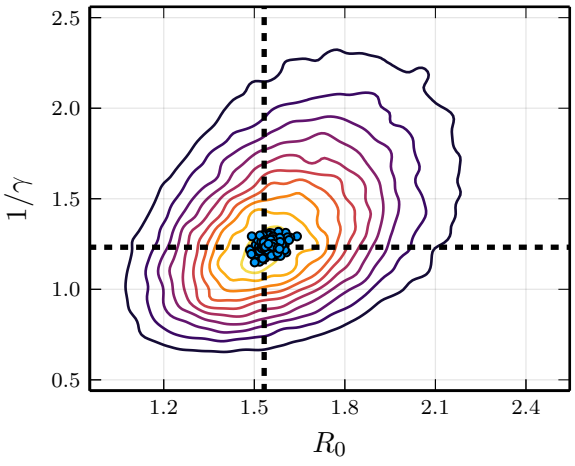

Figure 3.11: Contour plot of the posterior density. Estimates of the MAP from the SPSA algorithm are indicated in blue and the estimate of the MAP from the AKDE approach is indicated by the black dashed lines.

Method	R_0	$1/\gamma$
AKDE	1.533	1.232
SPSA (serial)	1.557 (0.048)	1.243 (0.061)
SPSA (parallel)	1.546 (0.024)	1.240 (0.031)

Table 3.3: MAP estimates for the SIR model. The estimates for the two versions of the SPSA method are averaged across the 1000 independent searches. See the caption of Table 3.1 for details.

Method	Loss (from search)	Loss (post search)
AKDE	–	26.807 (0.255)
SPSA (serial)	26.080 (0.287)	26.801 (0.220)
SPSA (parallel)	26.698 (0.088)	26.799 (0.221)

Table 3.4: Loss values for each method. Each row of the table corresponds to the method used and each column corresponds to the measured loss. See caption of Table 3.2 for details.

Figure 3.11 shows that running multiple searches from the same initial point in parallel tightened the spread about the MAP. This is reflected in the variances which have halved from the serial algorithm (Table 3.3). This suggests that considerable improvement arises from running multiple searches in parallel from the same starting point and averaging the result.

Noticeably there is still some minor difference between the MAP estimates from the AKDE approach and the SPSA approach. Table 3.4 shows that the average estimated value of the loss function at the MAP estimate is lowest for both the SPSA approaches (which are comparable to one another). Regardless of these minor differences, we can quite clearly observe that estimates converge and lie well within the highest posterior density region.

3.6 Application to the SEIAR model

3.6.1 Model details

The SEIAR model is a more complex epidemic model which is represented by Figure 3.12. This model allows individuals to pass through a phase of pre-symptomatic infectiousness I_p into a phase of symptomatic infectiousness I_s. There is also the possibility for asymptomatic individuals to pass directly to the recovered state without contributing to the force of infection.

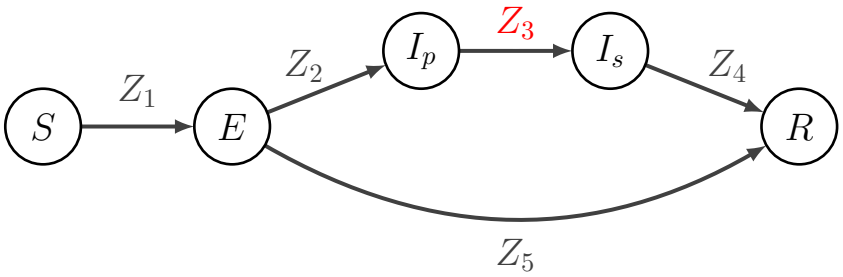

Figure 3.12: The SEIAR compartmental model. Details the transitions of individuals through the compartments. We highlight Z_3 as this corresponds to the observable event type.

The SEIAR model can be represented by a 5-dimensional CTMC, where the state of the system at time $t \geq 0$ is denoted $\boldsymbol{Z}(t) = (Z_1, Z_2, Z_3, Z_4, Z_5)$. The model has the following state space,

$$\begin{aligned} \mathcal{S} = \Big\{ \boldsymbol{Z}(t) | Z_j \in \mathbb{N}, Z_j \leq Z_i, \\ \text{for } i, j = 1, 2, \ldots, 4 \text{ and } j > i, \\ Z_5 \leq Z_1, Z_2 + Z_5 \leq Z_1, \\ 0 \leq Z_1 \leq N_F \Big\}, \end{aligned} \tag{3.19}$$

where N_F is defined as the total number of detection events over the course of the outbreak.

Table 3.5 shows the events and corresponding rates for the SEIAR model. In Table 3.5 we see that the model is specified by the parameters,

$$(\beta_p, \beta_s, \sigma, \gamma, q)\,,$$

where β_p and β_s are the transmission rates of pre-symptomatic and symptomatic individuals respectively, $1/\sigma$ and $1/\gamma$ are the average latent and infectious periods and q is the

Event	ΔState	Rate
Infection	$Z_1 \to Z_1 + 1$	$\frac{(S_0 - Z_1)\left[\beta_p(Z_2 - Z_3) + \beta_s(Z_3 - Z_4)\right]}{N-1}$
Become asymptomatic infectious	$Z_2 \to Z_2 + 1$	$q\sigma(Z_1 - Z_2 - Z_5)$
Become symptomatic infectious	$Z_3 \to Z_3 + 1$	$\gamma(Z_2 - Z_3)$
Symptomatic recovery	$Z_4 \to Z_4 + 1$	$\gamma(Z_3 - Z_4)$
Asymptomatic recovery	$Z_5 \to Z_5 + 1$	$(1-q)\sigma(Z_1 - Z_2 - Z_5)$

Table 3.5: Events and rates for the SEIAR model.

probability of an individual becoming asymptomatically infectious. A re-parametrisation of the parameters yields more practically interpretable quantities of interest. We begin by noting that for the SEIAR model, the basic reproduction number is defined as,

$$R_0 = \frac{q(\beta_p + \beta_s)}{\gamma}, \tag{3.20}$$

where frequency-dependent transmission has been assumed [8]. Define κ as the proportion of transmission due to pre-symptomatic individuals (in I_p), and note,

$$R_0 = \frac{q\beta_p}{\gamma} + \frac{q\beta_s}{\gamma}.$$

Hence, R_0 can be separated into the proportions attributed to pre-symptomatic and symptomatic individuals respectively, and thus,

$$R_0^p = \frac{q\beta_p}{\gamma} = \kappa R_0, \tag{3.21}$$

$$R_0^s = \frac{q\beta_s}{\gamma} = (1-\kappa)R_0. \tag{3.22}$$

Using this parametrisation, the parameters of the model are,

$$\boldsymbol{\theta} = \left(R_0, \frac{1}{\sigma}, \frac{1}{\gamma}, q, \kappa\right). \tag{3.23}$$

The priors assigned to each of the parameters are,

$$\begin{aligned} R_0 &\sim U(0.1, 8), \\ \frac{1}{\gamma}, \frac{1}{\sigma} &\sim \text{Gamma}\left(10, \frac{1}{10}\right), \\ q &\sim U(0.5, 1), \\ \kappa &\sim U(0, 1), \end{aligned}$$

and we assign lower-bounds to $1/\gamma$ and $1/\sigma$ of 0.5 and 0.1, respectively.

3.6.2 Data

For the SEIAR model, we use a particle filter which uses importance sampling. The filter is coded following the outline and decisions in [8]. We use 100 particles in this filter. We use the simulated data found in the supplementary materials of [8] as a test of the optimisation approach. This is a good demonstration of the performance of the search algorithm as the model is 5-variate and requires the use of a more complex particle filter. This time series is simulated according to the model outlined in Section 3.6.1 and the parameters used to simulate the data were $R_0 = 2.2, \kappa = 0.7, 1/\sigma = 1, 1/\gamma = 1$ and $q = 0.9$. We use the simulated data with population size $N = 150$. This time series represents the number of individuals whom pass from pre-symptomatic infectiousness to symptomatic infectiousness and is shown in Figure 3.13 shows the daily number of cases. We also assume a final size observation has been made from later data, $N_F = 121$.

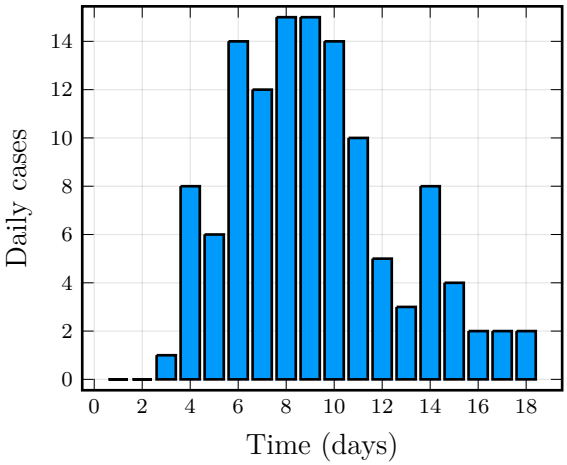

Figure 3.13: Observed incidence in the simulated SEIAR model.

3.6.3 Search setup

We begin by choosing the parameters of the search. Based off the preliminary analysis of the SIR model we choose $\boldsymbol{s} = \varsigma \boldsymbol{L}/4$. This ensures that when we are updating the gradient estimate we are only using points which are relatively close to one another. The parameters δ_4 and δ_5 are chosen to be smaller in comparison to the other values as the probabilistic parameters are constrained on a tighter space and so we want to explore this more carefully by taking smaller steps. Hence we take $\boldsymbol{\delta} = (0.25, 0.25, 0.25, 0.05, 0.05)$.

We choose to use $N_p = 100$ particles, which appeared to offer a reasonable amount of

noise without it dominating the search. The number of iterations is set at $M = 10000$ as this allows the search more than enough steps to converge. We choose to average the estimates obtained over 4 searches which are run in parallel from the same starting point. The tolerance ϵ was set equal to 2 which is similar to the case for the SIR model and allows some potentially worse points to be moved to. The rescaling factor was set at 0.5, but as mentioned in the SIR analysis, this value is not as important for the end result and just helps the search during the early iterations.

The set of starting points are shown in Figure 3.14. We obtained these points by sampling from the prior such that we could evaluate the posterior density 30 times over the course of 1000 trials. That is to say that the variance in the initial loss estimate was low (around 5). This can be automated in implementations of the search. We note that there is much less spread in the starting points for $1/\sigma$ and $1/\gamma$ and this is as these parameters are assigned informative priors. Without doing this, these parameters were unidentifiable and this explains why there is much higher dispersion in the initial R_0, q and κ values in Figure 3.14.

3.6.4 Results

Figure 3.15 demonstrates that all 50 of the searches obtained estimates of the MAP close to the estimate obtained from the AKDE method. There is close proximity to the MAP estimate across all parameters with some larger spreads prominent in the subplots with the probabilistic quantities (κ and q). This larger spread is slightly misleading as a result of the scale of these parameters and Table 3.6 demonstrates that there is not a large spread. There is some higher variability in the R_0 estimate and this is likely a result of the step size being slightly too large. It could also be a byproduct of the search comparing the current estimate of the minimum to the new estimate. This can cause the opposite issue to the SIR model where we over-estimate the loss function and hence the search can slightly diverge.

Method	R_0	$1/\sigma$	$1/\gamma$	q	κ
AKDE	2.83	1.07	1.18	0.88	0.75
SPSA	2.87 (0.15)	1.04 (0.06)	1.19 (0.06)	0.88 (0.06)	0.74 (0.06)

Table 3.6: MAP estimates for the SEIAR model. The estimates for the two versions of the SPSA method are averaged across the 50 independent searches. See the caption of Table 3.1 for details.

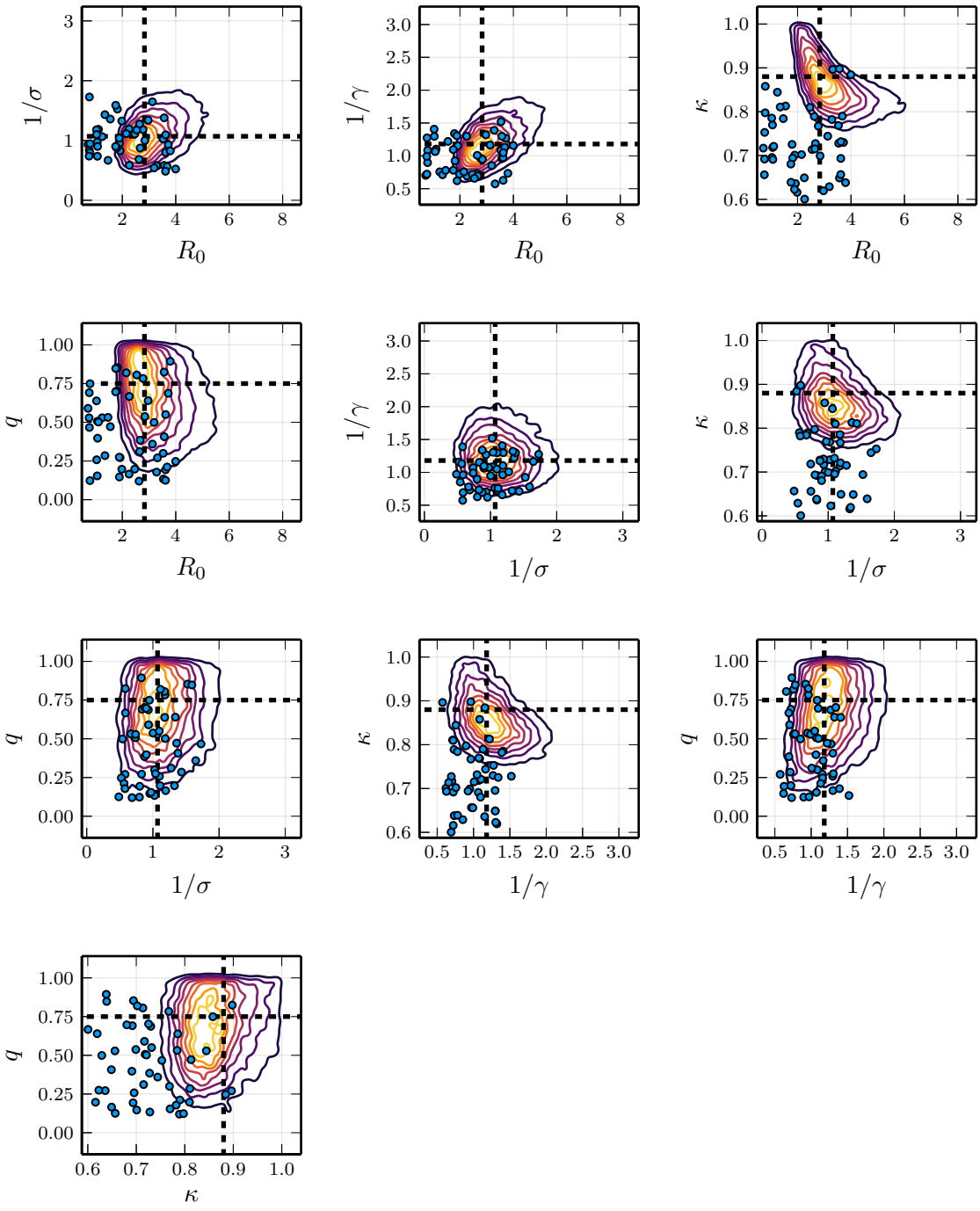

Figure 3.14: Contour plot of the bivariate marginal posterior densities. The black dashed lines indicate the estimate of the MAP from the AKDE approach. The blue dots represent 50 starting points.

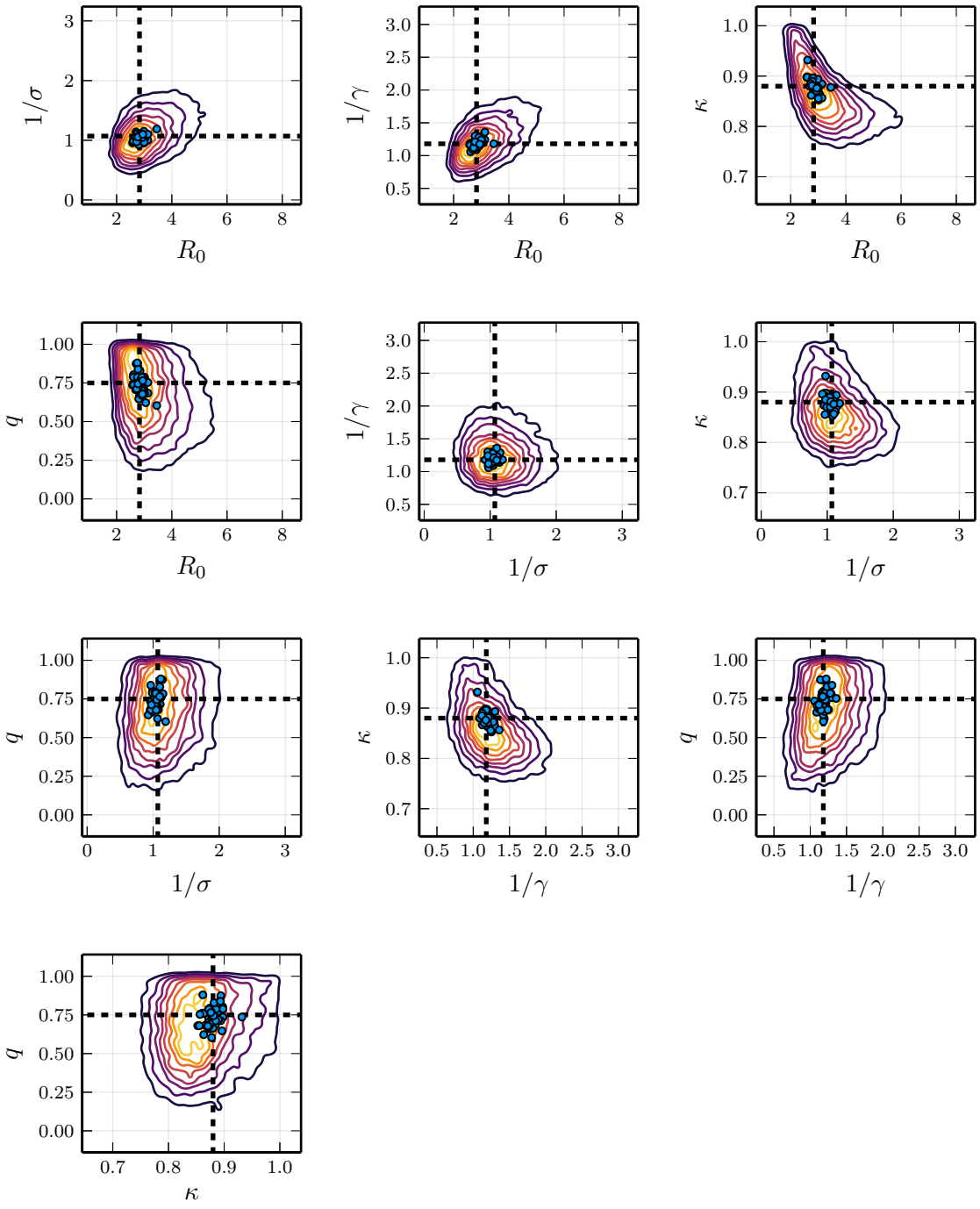

Figure 3.15: Contour plot of the bivariate marginal posterior densities. The black dashed lines indicate the estimate of the MAP from the AKDE approach. The blue dots represent the 50 estimates obtained through the SPSA algorithm.

Method	Loss (from search)	Loss (post search)
AKDE	–	40.91 (0.77)
SPSA	40.62 (0.42)	40.91 (0.77)

Table 3.7: Loss values for each method. Each row of the table corresponds to the method used and each column corresponds to the measured loss. See caption of Table 3.2 for details.

In Table 3.6 we see that there is agreement between the mean estimate of the SPSA algorithm and the estimate obtained through the AKDE approach. In Table 3.7 we run the particle filter independently 10000 times at the AKDE estimate of the MAP and mean SPSA estimate and average the results. We can see in Table 3.7 that the average estimate loss at the mean SPSA MAP estimate was directly comparable to the loss for the AKDE method. Interestingly we see the underestimation of the loss during the search again but as noted this does not influence the results a great deal. All the elements of the MAP estimates have a low standard deviation besides the R_0 term. This is likely a result of the step size chosen for the first term being slightly too large.

Running the particle filter sequentially, 1000 times at the mean MAP estimate we found the variance of the log-likelihood estimates to be 0.27 which is low. We found the variance in the log-likelihood estimates to be 1.32 when 60 particles are used which is reasonably close to the optimal value of 1.44.

In this Chapter we required the largest number of iterations for the SEIAR example. This example was run for $M = 10000$ iterations which equated to around 20000 likelihood evaluations (when accounting for recalculation of the current point at each iteration). If we budget the same amount of likelihood evaluations for the pmMH approach from the start, then the advantage of the SPSA method becomes clearer. With a budget of only 20000 likelihood evaluations, one needs to tune the proposal and choose an appropriate number of particles. This requires running the pmMH method, as well as running the tuned pmMH method to sample from the posterior. Without a reasonable initial guess as to the appropriate number of particles and the proposal, this budget will not provide precise estimates. These forms of pilot runs are something which are commonly used when using an MCMC method and while they are useful in providing an actual sample from the posterior, depending on the number of particles and the size and quality of the time series, this can require a non-trivial budget of likelihood evaluations. This is something that will be heavily problem dependent and there will be instances–such as the SIR example–where running the inference method will likely be quicker than running the search algorithm.

3.7 Summary

This Chapter outlines an approach for estimating the MAP in Bayesian models where only noisy estimates of the likelihood are available. The approach relies on some extensions to the SPSA algorithm of Spall [68]. This approach is optimal for these kinds of problems as the estimation of the likelihood is the computationally expensive part of the minimisation problem. Using a one-sided form of the SPSA algorithm mean only a single estimate is required on each iteration which allows for estimates of the MAP to be obtained in a reasonable amount of runtime.

In comparison to the AKDE approach the SPSA algorithm proves to be much more efficient as we do not need to sample the full posterior to obtain an estimate of the MAP. Furthermore, running AKDE often requires multiple runs of inference routines. This is a result of needing to tune the proposal distribution and number of particles at the same time to attain reasonable results. These issues are exacerbated for more complex epidemic models and/or longer time series as the cost of the simulations in the particle filter increases. In this instance the runtime (and cost) of running the pmMH method mixed with using the AKDE method is much higher than the SPSA approach.

The key contribution of such a methodology is to estimate the MAP for use in tuning of a particle filter. The performance of particle filters is crucial to the implementation of good mixing in the chains obtained through pmMH methods. This performance can be quantified by keeping the variance in the log-likelihood estimates within tight bounds. The search algorithm provides us with a framework of decoupling the tuning processes of pmMH, and the particle filters.

Chapter 4

Importance sampling for multiple observations of a single outbreak

In this chapter we develop a particle filtering approach for modelling epidemics when observations are made of more than a single event type. The particle filter uses importance sampling to simulate realisations of an epidemic model which are consistent with more than one observed event type. This particle filter is used in a pmMH procedure to conduct inference on the parameters of an outbreak of Ebola in Kikwit, 1995. The results are compared against previous studies of the Kikwit outbreak which use *data-augmented MCMC* (DA-MCMC) and we see strong agreement with our method.

4.1 Case study: 1995 DRC Ebola outbreak

The outbreak of Ebola in 1995 in the *Democratic Republic of the Congo* (DRC) was one of the largest outbreaks in the country [13, 41]. The outbreak began on January 6, 1995 and ended 191 days later on July 16 [41]. Intervention measures were introduced on May 9, 123 days into the outbreak, and these measures primarily concerned education of the public, as well as increased use of personal protective equipment (PPE) amongst health care workers [18, 41, 45]. It is documented that on January 6, a charcoal mine worker was identified with a case of Ebola, but that this was only confirmed to be the causative agent following an analysis of specimens from the early stages of the outbreak on the May 9 [41]. An outbreak was confirmed on March 2 which means there is a period of 55 days without reports. During this time it is reasonable to assume that there were not widespread cases, as otherwise authorities would have been alerted to the outbreak earlier.

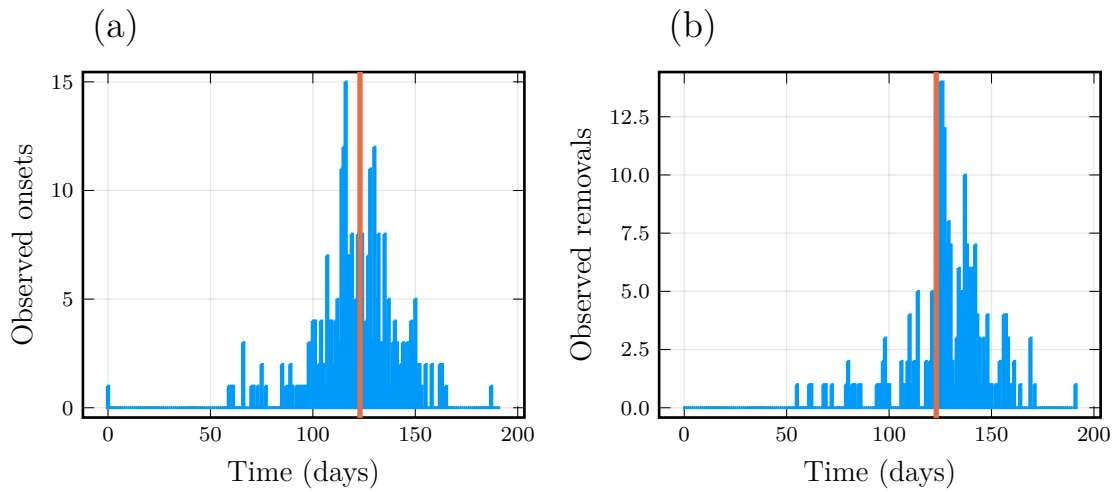

Figure 4.1: Daily onset and removal incidence for the 1995 DRC Ebola outbreak. Interventions were introduced on May 9th (123 days after the first case) and this date is indicated by the red line. The index case on 6th of January is taken as the initial condition.

The dataset is comprised of two time series: one for incidence of symptom onset which coincides with infectiousness (which is a reasonable assumption for Ebola), and another for removal incidence. The two time series are at daily resolution and feature the counts of the number of individuals who either developed symptoms, or died, respectively. A total of 316 individuals were identified to have been infected over the course of the outbreak including the index case. A number of the onset and removal times were missing and as such only the times of 291 onsets and 236 deaths were known up to daily precision. Daily incidence are shown in Figure 4.1 and we can see the lack of observations over the first 54 days as well as the time reporting began and the time interventions were introduced. From the difference between the total number of reported onsets and removals to the reported final size, this suggests that there are at least 25 missing dates of symptom onset and a further 80 missing dates of removal.

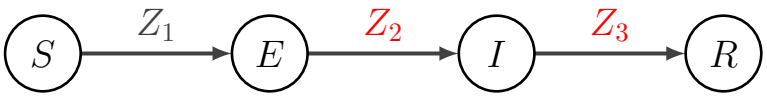

Figure 4.2: The SEIR compartmental model. Details the transitions of individuals through the compartments. The variables Z_2 and Z_3 correspond to the observed onset and removal events.

Throughout this chapter we consider the same SEIR model (see Figure 4.2) used by

Lekone and Finkenstädt [45] but provide some modifications to account for the missing onset and removal times. Our results will be compared against those obtained in previous studies [18, 45, 51, 55]. Each of these papers takes a different approach but produces (relatively) comparable results. Lekone and Finkenstädt [45] was the original motivation for development of these methods and consideration of this dataset, and their approach was to approximate the process in discrete-time using a chain-binomial model. Many of the assumptions surrounding both the model and their approach are reasonably explained but there is little model checking to demonstrate whether the approach they took appropriately captures the heterogeneity in the process. The more recent study of McKinley *et al.* [51] provides a comparison of DA-MCMC and Approximate-Bayesian-Computation (ABC) approaches on the same dataset. The approach in [51] is an extension of the approach taken in [45], but they do not approximate the process with a binomial or assume that the process evolves in discrete time. Removal of these assumptions leads to a model which appears to more reasonably capture the full dynamics of the process. The DA-MCMC approach in particular provides a good basis to compare our methods against as it targets the exact posterior. Using this model means that we assume all individuals are removed and transition to the same compartment and so may not necessarily die upon transitioning to the removal compartment. This means the average infectious period will be estimated over individuals who die and recover.

The issue with the analysis in [55] is that the MCMC approach is not well documented. Priors are not mentioned and it appears that they fit their model only to the available data and do not account for any missing information. A further issue with the study is that some of the parameter estimates are grossly inconsistent with the previously presented estimates, specifically the incubation period, which they estimate to be 1.65 and 1.69 days for the least squares approach and MCMC approaches, respectively. In the literature the incubation period is on average estimated to be around 6 days, and this tends to give weight to the idea that the priors used may have been poorly informed. For this reason we consider only comparisons against the other three analyses ([18, 45, 51]). The major inconsistency with comparing results to these previous studies are that Chowell *et al.* [18] and Lekone and Finkenstädt [45] appear to consider only the second phase (day 55 onwards) in their model fitting. In contrast, McKinley *et al.* [51] fit their model to the entire duration of the outbreak taking the 6th of January as the initial day. We also take the 6th of January to be the date the outbreak began. We assume the same population size used in [18, 45, 51] of $N = 5,364,500$.

The first and perhaps most appropriate approach to handle the missing data is that of data-augmented MCMC (DA-MCMC), which relies on inferring the missing times and the entire missing latent curve, as well as the parameters [33, 58]. When all the event times are known, the calculation of the likelihood is trivial. This approach is outlined in Gibson [28] and essentially involves proposing moves to the missing event times and

analytically calculating the likelihood. While this method is appropriate, it does suffer from some issues. It can be challenging to determine that the DA-MCMC approach has indeed converged to the appropriate stationary distribution [52, 58]. Furthermore, it scales poorly in higher dimensions [58]. We highlight this issue of scaling as we aim to conduct inference on multiple outbreaks of the Zaire ebolavirus using a hierarchical model in Chapter 5. Due to the sheer amount of data, the dimension of the augmented parameter space is in the order of around $1,000$ dimensions. The reason for this high-dimensional space is that the latent curve for each outbreak is completely missing which drastically increases the dimensionality of the problem. The DA-MCMC approach would prove to be computationally inefficient for such a problem and it would be very challenging to determine that the chains had converged. Here we propose a pmMH approach with a particle filter that uses importance sampling to match the counts in the two time series exactly and account for some missing data.

4.2 Model

The approach we consider is an alteration to the SEIR model (Figure 4.2) to allow for partial detection. This can be done by introducing two additional parameters, p_1 and p_2 which correspond to the observation probabilities of symptom onset and removal, for a binomial observation process each day. Individuals can pass from the exposed compartment to the infectious compartment (or infectious to recovered) without being detected. Whether this is an acceptable approach for modelling the missing data depends on the dynamics of the reporting process as well as the day-to-day variability in this reporting process. While we cannot ascertain this directly, we can assess posterior predictive distributions using our sample from the posterior distribution. This can provide us with some indication of the validity of our assumptions.

There have been a variety of serological studies following historical outbreaks and these showed that there were undetected and potentially asymptomatic cases [12]. As such it is possible that there were more than 316 cases, however, we take the reported final size as the total number of infections over the course of the outbreak. Specifically, this means that we assume there were $N_F = 316$ total detected cases and of this there were $N_1 = 291$ detected onset events and $N_2 = 236$ detected removals. This is captured by enabling an additional transition from E to I and I to R respectively, and is represented in the compartmental diagram in Figure 4.3. While this model can be reformulated as a simple SEIR model with a binomial observation process, there is still advantage in using an importance sampling methodology as this leads to reduced variance estimates at minimal computational expense.

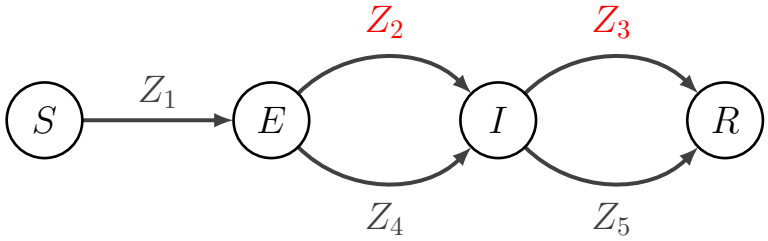

Figure 4.3: The SEIR compartmental model with partial detection. Details the transitions of individuals through the compartments. The variables Z_2 and Z_3 correspond to the observed onset and removal events.

The process can be modelled by a 5-variable CTMC $\{\boldsymbol{Z}(t)\}_{t\geq 0}$. The state space is given as,

$$\begin{aligned}\mathcal{S} = \Big\{\boldsymbol{Z}(t) : Z_i \geq Z_j, \text{ for } i \leq j, i \neq j,\\ Z_1 \leq N_F, Z_2 \leq N_1, Z_3 \leq N_2,\\ Z_4 \leq N_F - N_1, Z_5 \leq N_F - N_2\Big\}.\end{aligned} \tag{4.1}$$

The event types, change of state and transition rates are summarised in Table 4.1.

Event	ΔState	Rate
Infection	$Z_1 \to Z_1 + 1$	$\frac{\tilde{\beta}(t)(S_0 - Z_1)(Z_2 + Z_4 - Z_3 - Z_5)}{N-1}$
Become infectious (observed)	$Z_2 \to Z_2 + 1$	$p_1(t)\sigma(Z_1 - Z_2 - Z_4)$
Removal (observed)	$Z_3 \to Z_3 + 1$	$p_2(t)\gamma(Z_2 + Z_4 - Z_3 - Z_5)$
Become infectious (unobserved)	$Z_4 \to Z_4 + 1$	$(1 - p_1(t))\sigma(Z_1 - Z_2 - Z_4)$
Removal (unobserved)	$Z_5 \to Z_5 + 1$	$(1 - p_2(t))\gamma(Z_2 + Z_4 - Z_3 - Z_5)$

Table 4.1: Events and rates for the SEIR model with partial detection.

In this thesis we will be assuming an initially completely susceptible population, $S_0 = N$.Consistent with the literature [45, 51], we use a form of the effective transmission parameter—which we denote $\tilde{\beta}$—which decays exponentially to 0 following the introduction of intervention measures at time t_{int},

$$\tilde{\beta}(t) = \begin{cases}\beta, & \text{for } t \leq t_{\text{int}},\\ \beta e^{-q(t - t_{\text{int}})}, & \text{for } t > t_{\text{int}},\end{cases} \tag{4.2}$$

where β is the effective transmission parameter in the absence of interventions, and the additional parameter, q, can be thought of as the effectiveness of the intervention measures.

This version of a time-dependent transmission term appears unreasonable as,

$$\lim_{t\to\infty} \tilde{\beta}(t) = 0,$$

however, it has seen success in previous studies. While other forms of the time dependent transmission term have been proposed to varying degrees of success [14, 18, 44, 45]. There is some tension in capturing the effectiveness of the interventions and ensuring sufficient transmission over the observation period which can lead to parameter identifiability issues. For the analysis in this section we will assume the form of Eq. (4.2), which enables direct comparison with previous studies [45, 51].

As there was a period of 55 days which were unmonitored, we account for this by assuming a basic time-dependent functional form on the observation probabilities [41, 45, 51]. We assume small probabilities of $p_{1,0}$ and $p_{2,0}$ during the first 54 days. Following day 54 we then assume that the observation probability is much higher as this was when active monitoring for Ebola began. The functional forms for both observation probabilities are,

$$p_k(t) = \begin{cases} p_{k,0}, & \text{for } t \leq 54, \\ p_{k,1}, & \text{for } t > 54, \end{cases} \tag{4.3}$$

where $k = 1$ denotes detection probability for onsets and $k = 2$ denotes detection probability for removals.

The basic parameters of the model are,

$$(\beta, q, \sigma, \gamma, p_{1,0}, p_{2,0}, p_{1,1}, p_{2,1}),$$

however, interest is typically in the transformed parameters as they have more physical interpretations,

$$\boldsymbol{\theta} = \left(R_0 = \frac{\beta}{\gamma}, q, \frac{1}{\sigma}, \frac{1}{\gamma}, p_{1,0}, p_{2,0}, p_{1,1}, p_{2,1} \right).$$

We note here that we can also look at the effective reproduction number which is defined as,

$$R_{\text{eff}}(t) = \frac{\tilde{\beta}(t)}{\gamma} \tag{4.4}$$

as per Lekone and Finkenstädt [45] which allows us to establish when R_{eff} drops below the threshold of 1. This provides us with an indication of the effectiveness of the intervention measures. Throughout this section we will present both untransformed and transformed parameter values for direct comparison with previous studies.

4.2.1 SIR

In this section we detail the extension of the importance sampling to when we have multiple observations of the same process. We begin by motivating the idea of a particle filter which uses importance sampling to exactly match two events in a simple SIR model, before moving to the SEIR with partial observations. The SIR model is specified here in the same manner as in Section 2.2.1. While it is idealistic in that we observe all events, it suffices for demonstrating the key idea of how to simulate the process. First we will outline the process over a time interval $[0, 1)$ and then we will outline how this method can be extended to match time series of events. This relatively trivial example demonstrates the key idea of choosing which observed event occurs at each of the sampled times by selecting them in proportion to their rates.

Assume without loss of generality that y_1 infection, and y_2 recovery events occur over the interval $[0, 1)$ and so the observable state of the system at time 1 is $\boldsymbol{y} = (y_1, y_2)$. Assume we have an initial condition of $\boldsymbol{Z}(0) = (Z_1(0), Z_2(0))$. We begin by letting $y_{\text{tot}} = y_1 + y_2$ and calculating the upper bound on the number of events, $\boldsymbol{l} = \boldsymbol{Z}(0) + \boldsymbol{y}$ which allows us to constrain the number of each event type over $[0, 1)$.

Then the times of the y_{tot} observed events are generated as per Algorithm 4 by sampling from a uniform distribution and sorting the times [8]. We then set the initial contribution to the log-weight from the generation of these times,

$$w = -\log(y_{\text{tot}}!).$$

We initialise the current time $t = 0$ and let t_n, where n denotes next, be the time of the next observed event. The rates of the original process are then calculated, given the current state of the system (note that we omit the time dependency on the state variables),

$$\begin{aligned} a_1 &= \beta \frac{(N - Z_1)(Z_1 - Z_2)}{N - 1} \\ a_2 &= \gamma(Z_1 - Z_2) \end{aligned}$$

and let $a_0 = a_1 + a_2$ be the total rate.

Next we calculate the rates of the modified process, c_1, c_2. For the SIR model we just require the modified infection rate to be 0 if $Z_1 = \ell_1$. We require the modified recovery

rate to be set to 0 if $Z_2 = \ell_2$ or we need to prevent fadeout of the epidemic. That is,

$$c_1 = \begin{cases} 0, & \text{if } Z_1 = l_1, \\ a_1, & \text{otherwise,} \end{cases}$$
$$c_2 = \begin{cases} 0, & \text{if } Z_1 = l_1 \text{ or } Z_1 - Z_2 = 1, \\ a_2, & \text{otherwise.} \end{cases}$$

Then we set the total rate of the modified process as $c_0 = c_1 + c_2$.

The next step of the algorithm is to determine which event occurs next and we can do this by choosing an event index $j \in \{1, 2\}$ with probability,

$$\Pr(J = j) = \frac{c_j}{c_0}.$$

Then, the contribution to the weight under the original process is just the log-probability of event j happening at t_n,

$$w_{\text{orig}} = \log(a_j) - a_0(t_n - t),$$

and the contribution to the weight under the modified process is just the log of the probability of selecting event j since the times for this are accounted for when the event times are generated,

$$w_{\text{is}} = \log(c_j) - \log(c_0).$$

Then the weight is updated as,

$$w \leftarrow w + w_{\text{orig}} - w_{\text{is}}.$$

We then update the state and time,

$$Z_j \leftarrow Z_j + 1$$
$$t \leftarrow t_n.$$

This process is repeated until all the observed event times are iterated through and finally we account for the probability of no further events in the interval $[t, 1)$,

$$w \leftarrow w - a_0(1 - t).$$

We present only the SIR example to demonstrate the general idea of the algorithm. For more complex models we may need to force a chain of events in order for the state of the system to be consistent with the next observed event [8]. This forcing process is done in the same was as outlined in [8] whereby we sample times of the forced events from a truncated-exponential distribution (see Algorithm 4).

4.2.2 SEIR with partial detection of onset and removal events

In the design of a multi-observation importance sampling filter, we need to determine the rates of a modified process such that the importance process does not allow premature fadeout, or ruin the consistency of the simulations. The simulations are deemed to be consistent if at the end of each day the relevant state variables are equal to the observed counts over that day. This enables our simulations to proceed, exactly matching the incidence curves and this enables us to more efficiently estimate the log-likelihood [8].

In the algorithm we essentially have three concurrent processes to keep track of. There is the original process, the modified events which correspond to our observed events, as well as those which do not. The rates of the original process are denoted by a_i, for $i = 1, \ldots, 5$, the rates of the modified and unobserved events are denoted b_i for $i = 1, 4, 5$ and the rates of the modified and observed events are denoted c_i for $i = 2, 3$. The idea of the algorithm is to generate the times of the modified and observed events over the observation period. We then check the state of the system to see if it is consistent with the next forced event and if not we need to force an additional event. If the state of the system is consistent then we calculate the modified rates of the unobserved events and then sample a time until the next event using these. This is then compared with the time of the next forced event, and we increment the state variable and time accordingly.

In this version of the algorithm we work with a stack, ψ, to store the event times and types. A stack data-structure operates under a last-in-first-out regime. That is to say, the item which is added to the stack most recently will be the one on top (see [8] for details). Let t_n and e_n—where n means next—point to the top of the stack, which corresponds to the time of the next forced event and event type, respectively [8]. The use of a stack allows additional events to be simulated before any events which may need to be forced in order to maintain consistency with the observations (as in [8]). Assume we observe y_1 onset events and y_2 removal events over the interval $[0, 1]$ and hence $\boldsymbol{y}$ is the observed state of the system. Let the state of the system at time 0 be $\boldsymbol{Z}(0) = (Z_1(0), Z_2(0), Z_3(0), Z_4(0), Z_5(0))$. Let $\boldsymbol{l} = (Z_2(0), Z_3(0)) + \boldsymbol{y}$. Let $y_{\text{tot}} = y_1 + y_2$ be the total number of events.

We initialise the times of the observed events (as in Algorithm 4) and event indices and add them in reverse order to the stack,

$$\psi = \Big\{(t_{y_{\text{tot}}}, e_{y_{\text{tot}}}), (t_{y_{\text{tot}}-1}, e_{y_{\text{tot}}-1}), \ldots, (t_1, e_1)\Big\},$$

and let $e_j = 0$ for $j = 1, 2, \ldots, y_{\text{tot}}$ denote the next event being an observed event. Let t_n and e_n point to t_1 and $e_1 = 0$ (the top of the stack) respectively and set $t = 0$. Then account for the generation of the order statistics by initialising the weight,

$$w = -\log(y_{\text{tot}}!).$$

First we calculate the rates under the original process (note that we omit the time dependency on the state variables),

$$\begin{aligned}
a_1 &= \frac{\tilde{\beta}(t)}{N-1}(N - Z_1)(Z_2 + Z_4 - Z_3 - Z_5) \\
a_2 &= p_1(t)\sigma(Z_1 - Z_2 - Z_4) \\
a_3 &= p_2(t)\gamma(Z_2 + Z_4 - Z_3 - Z_5) \\
a_4 &= (1 - p_1(t))\sigma(Z_1 - Z_2 - Z_4) \\
a_5 &= (1 - p_2(t))\gamma(Z_2 + Z_4 - Z_3 - Z_5)
\end{aligned}$$

and let $a_0 = \sum_{i=1}^{5} a_i$ be the total rate under the original process.

We then calculate the rates of the first modified process (the observed events). In this step we set the rate of latent progression to 0 if we have simulated all onsets, $Z_2 = l_1$. The rate of removal is set to 0 if we have simulated all removals, $Z_3 = l_2$ or we need to prevent fadeout. To avoid fadeout we check whether the number of $E + I = 1$ and whether at least one of $Z_1 < N_F, Z_2 < N_1$, or $Z_3 < N_2 - 1$ is true. Checking if $Z_1 < N_F$ or $Z_2 < N_1$ ensures we do not allow fadeout while there are still future infections or onsets to detect. The condition $Z_3 < N_2 - 1$ ensures we only allow fadeout in this step if the next forced event is a detected removal and the total number of allowed infections and onsets have occurred. The rates under the first modified process are,

$$c_2 = \begin{cases} 0, & \text{if } Z_2 = l_1, \\ a_2, & \text{otherwise,} \end{cases}$$

$$c_3 = \begin{cases} 0, & \text{if } Z_3 = l_2 \text{ or } Z_1 - Z_3 - Z_5 = 1 \text{ and} \\ & \quad Z_1 < N_F \text{ or } Z_2 < N_1 \text{ or } Z_3 < N_2 - 1 \\ a_3, & \text{otherwise,} \end{cases}$$

and let $c_0 = c_2 + c_3$ be the total rate of the first modified process.

If $c_0 = 0$ and the next event is a forced event, $e_n = 0$, then we need to force an additional event for consistency. The type of the next event can be determined by looking at the current state of the system. The ways in which we can determine which event to force is to determine the conditions leading to $c_0 = 0$.

1. If $Z_2 = l_1$, we have simulated all onsets over the current time interval but still require removals. There are two instances to consider, if $E = 0$ or not.
   (a) If $E = 0$, then the only way for $c_0 = 0$ is to have $I = 1$ and we need to prevent fadeout. We can only do this provided $Z_1 < N_F$ meaning we have not exhausted the total number of possible infection events. In this case we force an infection event, $e_n = 1$.

(b) Else, $E > 0$ and $I = 0$. If we have not simulated the maximum number of non-detections $Z_4 < N_F - N_1$, we force an undetected onset, $e_n = 4$.

2. Else, $Z_2 < l_1$ and $E = 0$. If the simulation is in this state then we need to force an infection event which can only occur if $Z_1 < N_F$.

During simulations we set the weight of any realisation which attempts to simulate more than the total number of allowed infections or non-detections to 0. This can be directly accounted for in the code and doing it at this step prevents unnecessary calculation at later stages of the simulation. For the model presented here, we do not need to check whether the next event is a type 1 or 4 as the forced events are added in terms of what is required for the state to be consistent. This will not always be the case and is something we will discuss further in Chapter 6.

Assuming the next forced event is of type k then we sample a time of this event,

$$t' \sim \text{TruncExp}(a_k, 0, t_n - t),$$

and the log-weight is updated,

$$\begin{aligned} w_{\text{is}} &= \log(a_k) - a_k t' - \log(1 - \exp(-a_k(t_n - t))) \\ w &\leftarrow w - w_{\text{is}}. \end{aligned}$$

Then we push $t + t'$ and index k onto the stack and set $e_n = k, t_n = t + t'$.

Following this, we set the rates of the second modified process. In the calculation of these rates we need to account for the next forced event. We also need to account for no fadeout which can be done by checking that $E + I = Z_1 - Z_3 - Z_5 = 1$ and whether $Z_1 < N_F, Z_2 < N_1$ or $Z_3 < N_2$. The rates of the modified and unforced events are,

$$b_1 = \begin{cases} 0, \text{ if } Z_1 = N_F, \text{ or } e_n = 1, \\ a_1, \text{ otherwise,} \end{cases}$$

$$b_4 = \begin{cases} 0, \text{ if } Z_4 = N_F - N_1, \text{ or } e_n = 4, \\ a_4, \text{ otherwise,} \end{cases}$$

$$b_5 = \begin{cases} 0, \text{ if } Z_5 = N_F - N_2, \text{ or } Z_1 - Z_3 - Z_5 = 1 \text{ and} \\ \qquad Z_1 < N_F \text{ or } Z_2 < N_1 \text{ or } Z_3 < N_2, \\ a_5, \text{ otherwise.} \end{cases}$$

The total rate of the second modified process is then set as $b_0 = b_1 + b_4 + b_5$.

Next we sample a time until the next unforced event from the second modified process,

$$t' \sim \text{Exp}(b_0)$$

to be compared to the time of the next forced event.

If the proposed time is less than the time until the next forced event $t' < t_n - t$ then we choose event index $j \in \{1, 4, 5\}$ with probability,

$$\Pr(J = j) = \frac{b_j}{b_0}.$$

We update the weight,

$$\begin{aligned} w_{\text{is}} &= \log(b_j) - b_0 t' \\ w_{\text{orig}} &= \log(a_j) - a_0 t' \\ w &\leftarrow w + w_{\text{orig}} - w_{\text{is}} \end{aligned}$$

and update the time and state,

$$\begin{aligned} Z_j &= Z_j + 1 \\ t &\leftarrow t + t'. \end{aligned}$$

If instead $t' \geq t_n - t$, we begin by accounting for the probability that the proposed time was larger than the time until the next forced event,

$$\begin{aligned} w_{\text{is}} &= -b_0(t_n - t) \\ w &\leftarrow w - w_{\text{is}}. \end{aligned}$$

Then there are two cases which can occur, either the next forced event is one of $\{1, 4\}$ or it is one of the first modified process, i.e. the observed events $\{2, 3\}$. This can be ascertained by looking at the event type on the top of the stack, if $e_n \neq 0$ then the next event is of type 1 or 4. We then account for the contribution to the log-weight under the original process, noting that the contribution to the log-weight under the importance process is already accounted for during the generation of the event times. Hence,

$$\begin{aligned} w_{\text{orig}} &= \log(a_{e_n}) - a_0(t_n - t) \\ w &\leftarrow w + w_{\text{orig}}, \end{aligned}$$

and the state and time are updated as,

$$\begin{aligned} Z_{e_n} &\leftarrow Z_{e_n} + 1 \\ t &\leftarrow t_n. \end{aligned}$$

Then we remove the event time and type from the stack.

If $e_n = 0$ then one of the observed events in the first modified process is next. We choose event index $j \in \{2, 3\}$ with probability,

$$\Pr(J = j) = \frac{c_j}{c_0}.$$

The contribution to the weight is,

$$\begin{aligned} w_{\text{is}} &= \log(c_j) - \log(c_0), \\ w_{\text{orig}} &= \log(a_j) - a_0 t', \\ w &\leftarrow w + w_{\text{orig}} - w_{\text{is}}. \end{aligned}$$

Then we set,

$$\begin{aligned} Z_j &\leftarrow Z_j + 1 \\ t &\leftarrow t_n, \end{aligned}$$

and we remove the event time and type from the stack.

We continue this process until all observed events have been iterated through. Then we run a simpler version of the simulation by noting that all the observed events have been accounted for in step 1, and so we only use the original process and second modified process.

One thing to note on the modelling assumptions is whether it is reasonable to assume the known total number of infections is precise. For the DRC outbreak it is known that there were 316 total cases and so we can constrain the total possible number of non-detections by this. A minor caveat with this approach is that while we have certainty of knowing there were N_F infections over the course of the outbreak, it is entirely feasible for the simulation to allow less than $N_F - N_1$ or $N_F - N_2$ undetected onset or removal events. The expectation is that assuming this final size observation will provide a greater deal of information to the simulations which should result in a more informative posterior. Furthermore, an issue we will touch on in the discussion is that it is entirely plausible that there may be more than N_F infections. Assuming this final size observation, limits the number of possible events which can be forced. In the step where we force an event for consistency, we also need to be careful in ensuring we do not allow more unobserved onsets or infections than we know occurred. This can be accounted for by setting the weight of this realisation to 0 if either of these two instances occur.

4.3 Inference on a simulated outbreak

In this section we conduct inference on a simulated dataset. For comparative purposes, in this and the following section we use the basic form of the parameters as in [45]. We

simulate a realisation of the SEIR with partial detection taking the basic parameters as,

$$(\beta, q, \sigma, \gamma, p_{1,0}, p_{2,0}, p_{1,1}, p_{2,1}) = (0.2, 0.2, 0.2, 0.14, 0, 0, 0.90, 0.80),$$

where all parameter values are chosen to be the same as those used in [45] and the probabilities of observation are chosen empirically based on the proportion of cases where the date of onset or removal was known.

An initial condition of $\boldsymbol{Z}(0) = (1, 1, 0, 0, 0)$ is assumed. We choose simulations which have similar characteristics to the observed outbreak, namely that there were few missing cases during the early phase of the outbreak, which would have meant the outbreak could have gone unnoticed in such a large population. We then seek it to quickly take off and follow a similar shape to the 1995 outbreak. Furthermore, we wanted an outbreak of comparable duration and final size. The full simulated outbreak is shown in Figure 4.4. This shows all onsets and removals and we have coloured the events in the plot based on whether they were detected (blue) or not (orange). The simulated outbreak saw 292 onsets and 274 removals detected across 159 days. There were 374 total infections over the course of the outbreak. We simulated this outbreak by removing the first 54 days which essentially allows the process to evolve as an SEIR where none of these cases are detected. For day 55 onwards, we set $p_1 = 291/316$ and $p_2 = 236/316$. We introduce interventions on day 123 and use the time dependent form of the transmission term. Figure 4.5 shows the result of this censoring process.

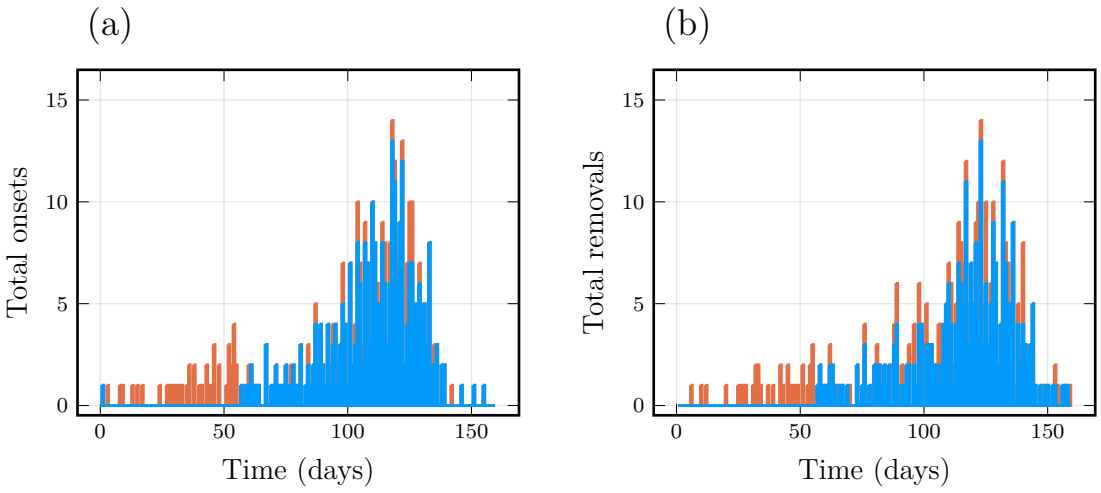

Figure 4.4: Total daily onset and removal incidences for the simulated Ebola outbreak. Blue indicates detected events and orange indicates undetected cases.

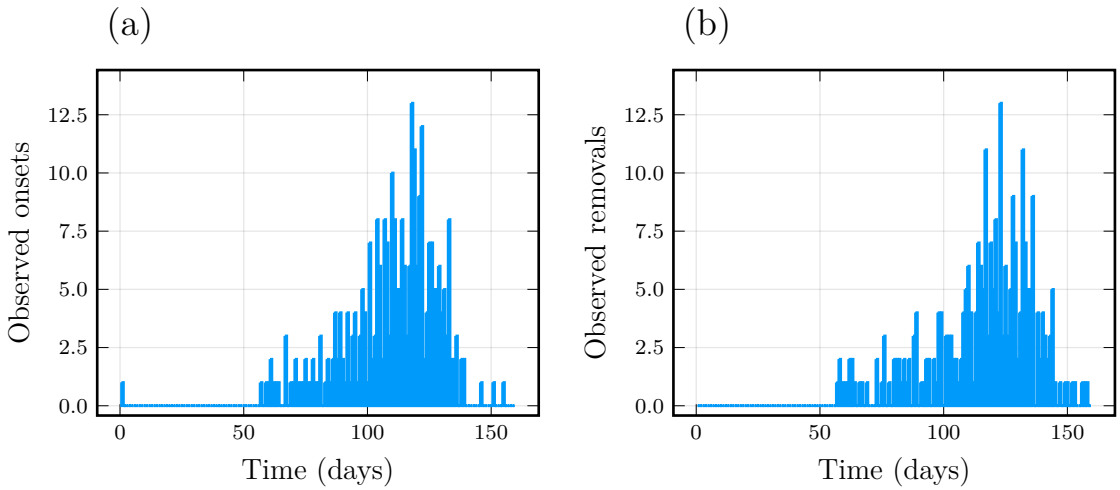

Figure 4.5: Observed daily onset and removal incidences for the simulated Ebola outbreak.

We assume the same prior densities as those provided in [45]. Prior densities of Beta$(3, 7)$ are assumed on the observation probabilities $p_{1,0}, p_{2,0}$ which have mode 0.25 which captures the idea that over the first 55 days it is unlikely to observe events. When the reporting phase begins we assume prior distributions of Beta$(7, 3)$ which has mode 0.75, which captures the idea that majority of cases would be observed when we are actively monitoring for Ebola cases. The prior distributions for each of the parameters are,

$$\begin{aligned}
\beta &\sim \text{Gamma}(2, 1/10),\\
q &\sim \text{Gamma}(2, 1/10),\\
\sigma &\sim \text{Gamma}(2, 1/10),\\
\gamma &\sim \text{Gamma}(2, 1/14),\\
p_{1,0} &\sim \text{Beta}(3, 7),\\
p_{2,0} &\sim \text{Beta}(3, 7),\\
p_{1,1} &\sim \text{Beta}(7, 3),\\
p_{2,1} &\sim \text{Beta}(7, 3).
\end{aligned}$$

The SPSA algorithm was run in parallel on 4 cores—meaning we average over 4 search estimates—using 200 particles in the particle filter and the MAP was estimated to be

$$(0.17, 0.22, 0.22, 0.12, 0.03, 0, 0.89, 0.81),$$

and the variance in the log-likelihood at this point (when assuming 200 particles) is 3.8. This suggests that more particles should be used. An alternative to increasing the number

of particles is to run multiple independent particle filters in parallel and average the likelihood estimates, which reduces the variance in the likelihood estimate at no increased computational runtime [24]. Running the particle filter in parallel on 8 cores using the 150 particles results in a variance in the log-likelihood estimate of 1.21 which lies in the tolerance discussed in Section 3.2.

We ran a pilot run of the pmMH algorithm to obtain a suitable covariance matrix. Following this the pmMH algorithm was run until the resultant sample had a minimum ESS of 900 with all parameters having ESS in the range of 900–4000. The marginal posterior distributions, priors and true values are shown in Figure 4.6. The bivariate posterior densities are shown in Figure 4.7 and the MAP estimate is indicated on these. The posterior distributions are summarised by their mean and standard deviations in Table 4.2.

In Figures 4.6 we see that the likelihood dominates the posterior density. This can be seen by the shift from the priors (red) to the marginal posterior densities (blue). Figure 4.6 shows that all marginal posteriors are smooth, unimodal distributions. There appear to be no discernable irregularities such as funnel degeneracies or sharp edges in the posteriors outside of the parameters $p_{1,0}$ and $p_{2,0}$ which lie on the boundary of the support. We see in Figure 4.7 that the values of the true parameters (indicated by the grey lines) lie close to the medians of the marginal posterior densities. In Table 4.2 we see that there is a good agreement between the means obtained for each parameter compared to those used to simulate the dataset. This suggests the method is able to appropriately recapture the true parameter values. We note here that the prior on q is necessary for this parameter to be identifiable. This was assigned following the discussion in [45] and essentially captures the idea that changes to transmission will be slow following interventions.

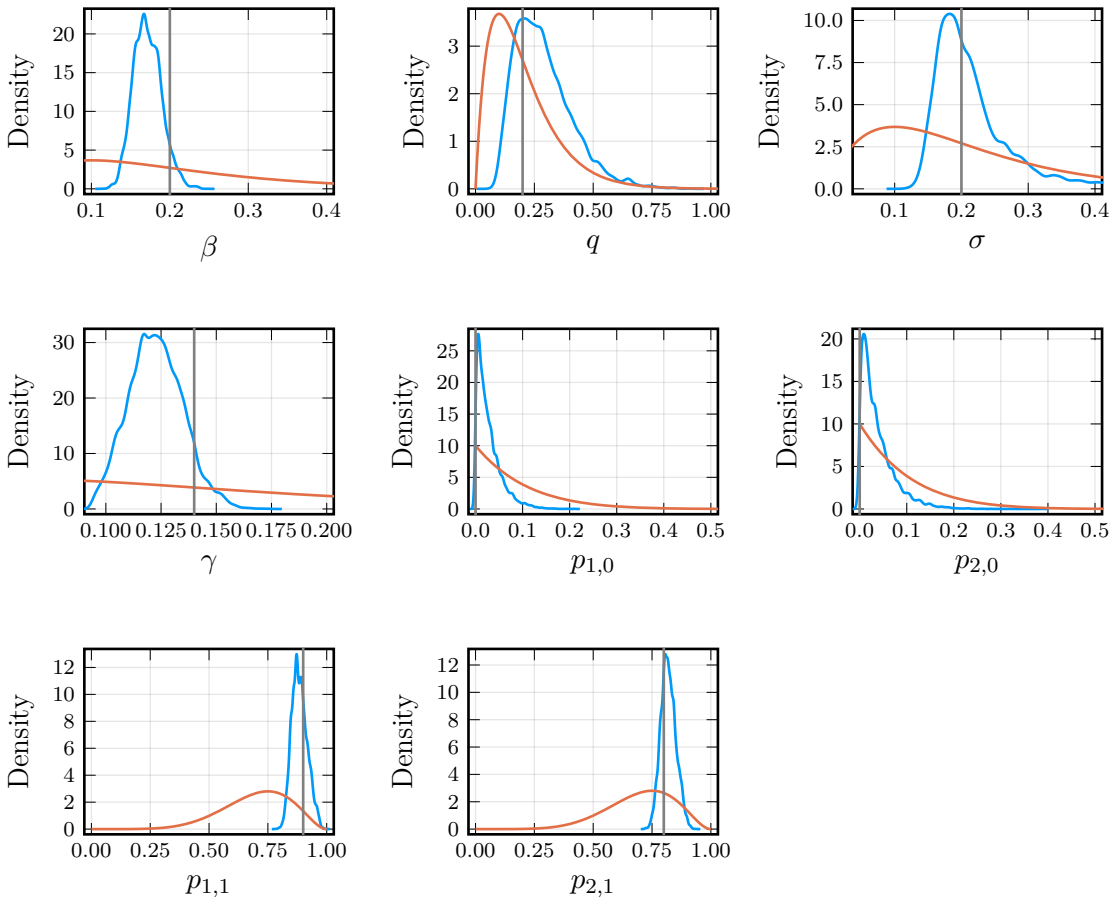

Figure 4.6: Marginal posterior distributions (blue) and priors (red) for the parameters of the model fitted to the simulated outbreak. True parameter values are indicated by the grey lines.

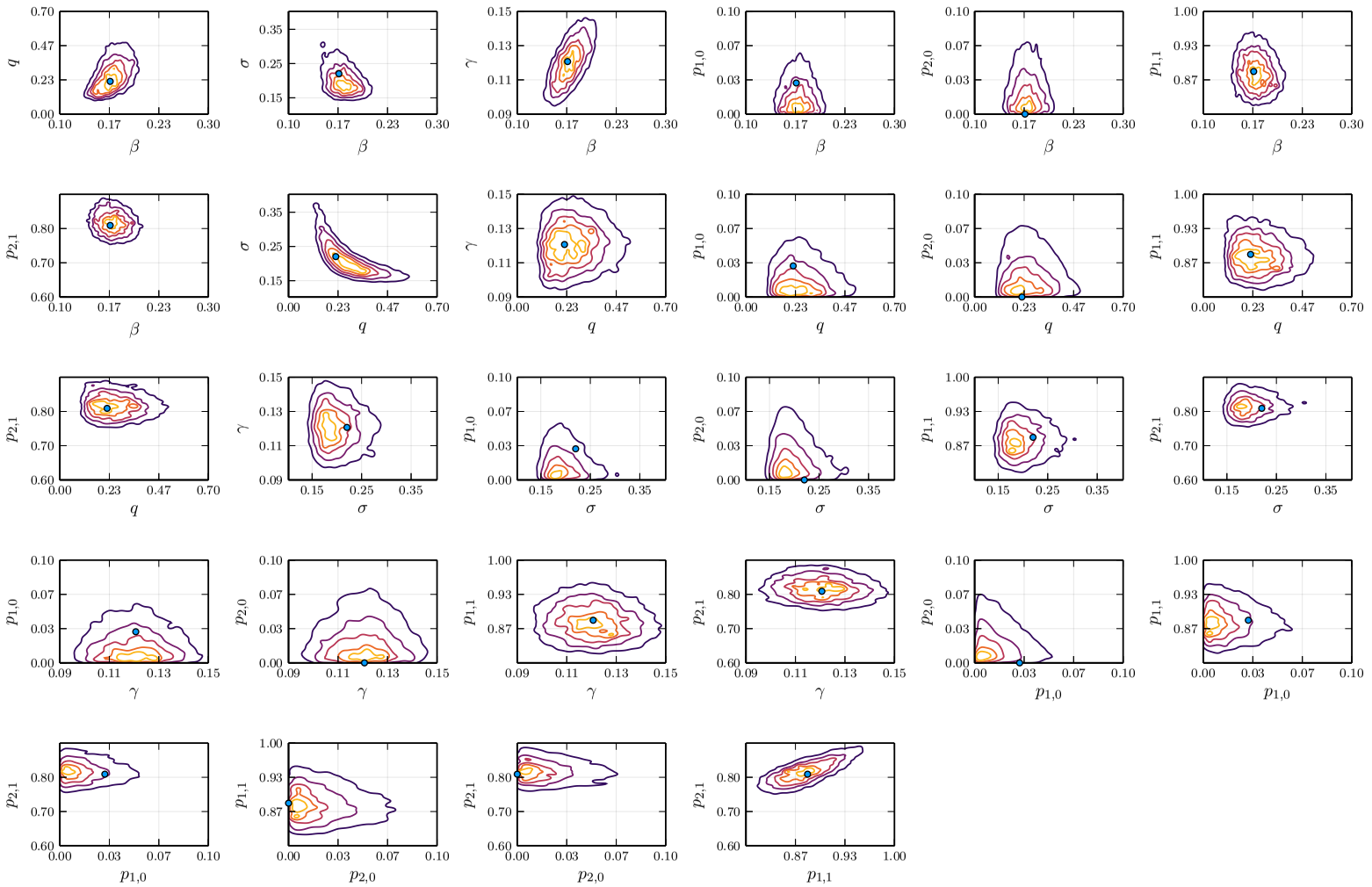

Figure 4.7: Bivariate marginal posterior distributions for the parameters of the model fitted to the simulated outbreak. The blue point indicates an estimate of the MAP obtained through the SPSA algorithm.

Parameter	True value	Estimate
β	0.2	0.17 (0.02)
R_0	1.43	1.41 (0.12)
q	0.2	0.29 (0.13)
$1/\sigma$	5.00	4.86 (1.17)
$1/\gamma$	7.14	8.22 (0.86)
$p_{1,0}$	0	0.03 (0.03)
$p_{2,0}$	0	0.04 (0.03)
$p_{1,1}$	0.90	0.88 (0.04)
$p_{2,1}$	0.80	0.82 (0.04)

Table 4.2: Posterior means and standard deviations (in parentheses). Values used to simulate the dataset are displayed in the "True value" column of the table.

The MAP esimate is shown in Figure 4.7 and we can see that this point lies in the region of high density. There are some minor discrepancies which are present mainly in the plots with the observation probabilities. Due to how small these values are there is likely to be little difference in the posterior density over the range of feasible values.

We can assess the goodness of fit of the models by checking posterior predictive plots. We choose to look at the incidence and outbreak duration inline with the analysis in [51]. In the following, we condition on the outbreak lasting at least 100 days and matching the observed detected onsets over this period.

In Figure 4.8 we see that the mean incidence of the simulations is consistent with the observed incidences. We see that the observed incidence lies well within the 95% credible interval and the model appears to capture the uncertainty in the outbreak. In Figure 4.9 the full outbreak incorporating the undetected events is shown. From Figure 4.9 we see that the mean of the simulations appear to reasonably capture the undetected events during the missing phase of the outbreak. The narrowing of the credible interval at the 55 day mark is simply as this is when active surveillance begins and so there is more certainty over the behaviour of the outbreak as the data is available. We can see that even with this barren reporting phase, the model is able to accurately estimate the underlying dynamics. We also see by checking the posterior predictive distribution of outbreak durations (Figure 4.10) that this was an outbreak which took off. Furthermore, it is a very likely duration for an outbreaks which does take off. These diagnostics demonstrate that the model appears to fit reasonably to the simulated data.

Other realisations were also simulated which had less similarity to the observed outbreak and the methods performed equivalently well on those. This realisation is presented here to demonstrate the fit on a large outbreak with a comparable degree of missing dates to the 1995 outbreak. It also had a more obvious outbreak nature whereby we obtain a peak in incidence which tapers off following interventions. These are all characteristics of the observed data and hence provided a good test case for the particle filtering method.

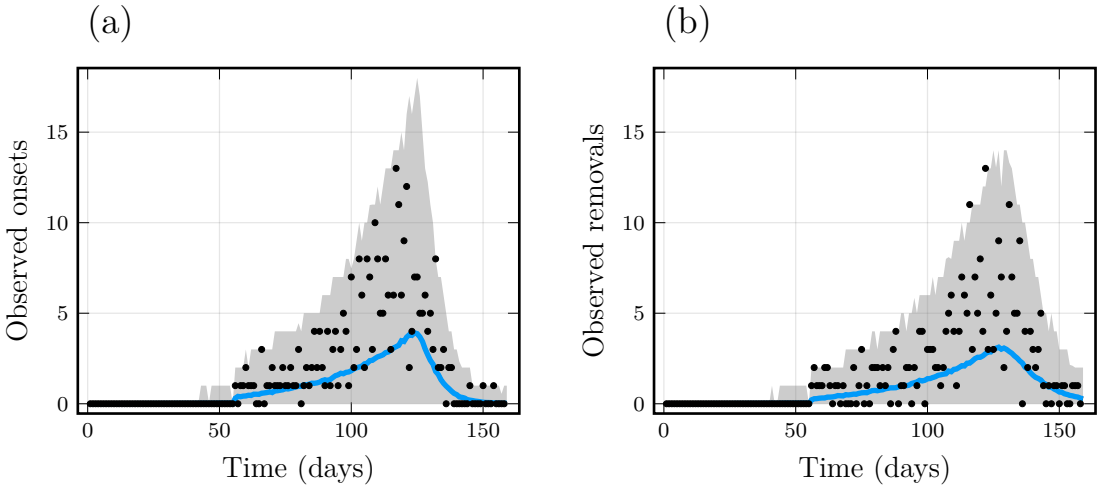

Figure 4.8: Simulated Ebola data versus the posterior simulations. Results are shown for 10000 outbreaks simulated according to the SEIR model with partial detection, using the posterior sample. The blue line indicates the daily average incidence across the simulations. The grey area corresponds to the 95% credible interval of the simulations. The black dots indicate the observed daily incidences.

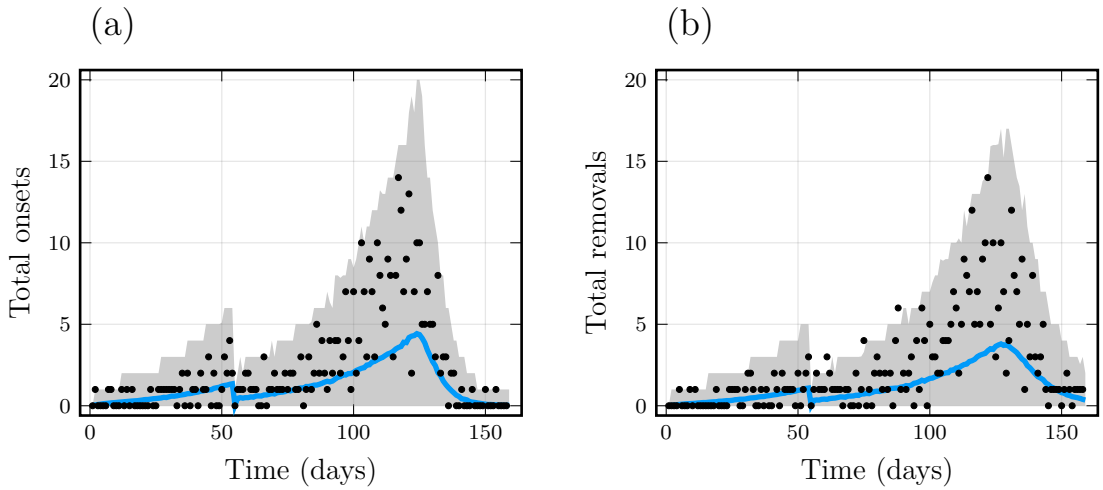

Figure 4.9: Simulated data versus the posterior simulations. Both the detected and undetected cases are plotted. See caption of Figure 4.8 for details.

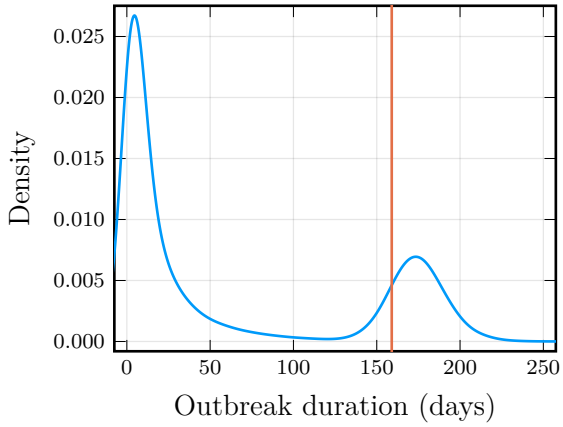

Figure 4.10: Posterior predictive distribution for the outbreak duration. The red line indicates the observed duration.

4.4 Inference on the 1995 outbreak

In this section we fit the model specified in Section 4.2 to the Ebola data from the 1995 outbreak in the DRC. We use the same priors as in the simulated example and assume an initial condition of $\boldsymbol{Z}(0) = (1, 1, 0, 0, 0)$ in the particle filters. From testing of the likelihood function we saw that there was a reasonably large variance (> 5) in the likelihood estimates for 200–400 particles. This is likely a result of the sheer amount of

missing data over the first phase of the outbreak and that there are only a small number of total missing dates. Due to the forward simulation approach we require enough particles to approximate the evolution of the outbreak appropriately and to do so over the first 55 days, we need to ensure enough particles are used so that we essentially capture all possible events. Increasing the number of particles to around 500 and parallelising the filter over 8 cores reduced the variance to within acceptable bounds allowing the search to move with increased precision in the tails. Applying the SPSA algorithm using the parallelised particle filter we obtained an estimate of the MAP of,

$$(0.24, 0.22, 0.08, 0.13, 0, 0.04, 0.96, 0.79).$$

Evaluating the parallelised particle filter at this point the variance in the log-likelihood estimates was 2.4 which lies within the tolerance specified in Section 3.2. Increasing the number of particles to 600 reduced this variance to around 1.9. Using the results of the search we use a particle filter with 500 particles run on 8 cores in a pmMH routine as while this offered a slightly higher variance, it still lies within the tolerance in Section 3.2. The inference method was run for 24 hours and obtained a sample which had a minimum ESS of 1400, with majority of parameters having ESS in the range of 1500–4000. The marginal posterior distributions and bivariate posterior distributions are shown in Figure 4.11 and Figure 4.12, respectively. MAP estimates from the SPSA search are indicated in Figure 4.12.

Parameter	2-IS	DA-MCMC	Chain-Binomial	Deterministic
β	0.23 (0.02)	0.25 (0.03)	0.24 (0.03)	0.33 (0.06)
R_0	2.00 (0.22)	2.00 (0.21)	1.40 (0.13)	1.83 (0.06)
q	0.35 (0.16)	0.37 (0.17)	0.16 (0.01)	–
$1/\sigma$	12.49 (1.38)	12.00 (1.20)	9.40 (0.62)	5.30 (0.23)
$1/\gamma$	8.80 (0.77)	8.1 (0.92)	5.70 (0.55)	5.60 (0.19)
$p_{1,0}$	0.07 (0.06)	–	–	–
$p_{2,0}$	0.07 (0.06)	–	–	–
$p_{1,1}$	0.95 (0.01)	–	–	–
$p_{2,1}$	0.78 (0.02)	–	–	–

Table 4.3: Posterior means and standard deviations (in parentheses) for the different inference methods used on the 1995 DRC outbreak data. The dashes indicate values that were not inferred for the given method.

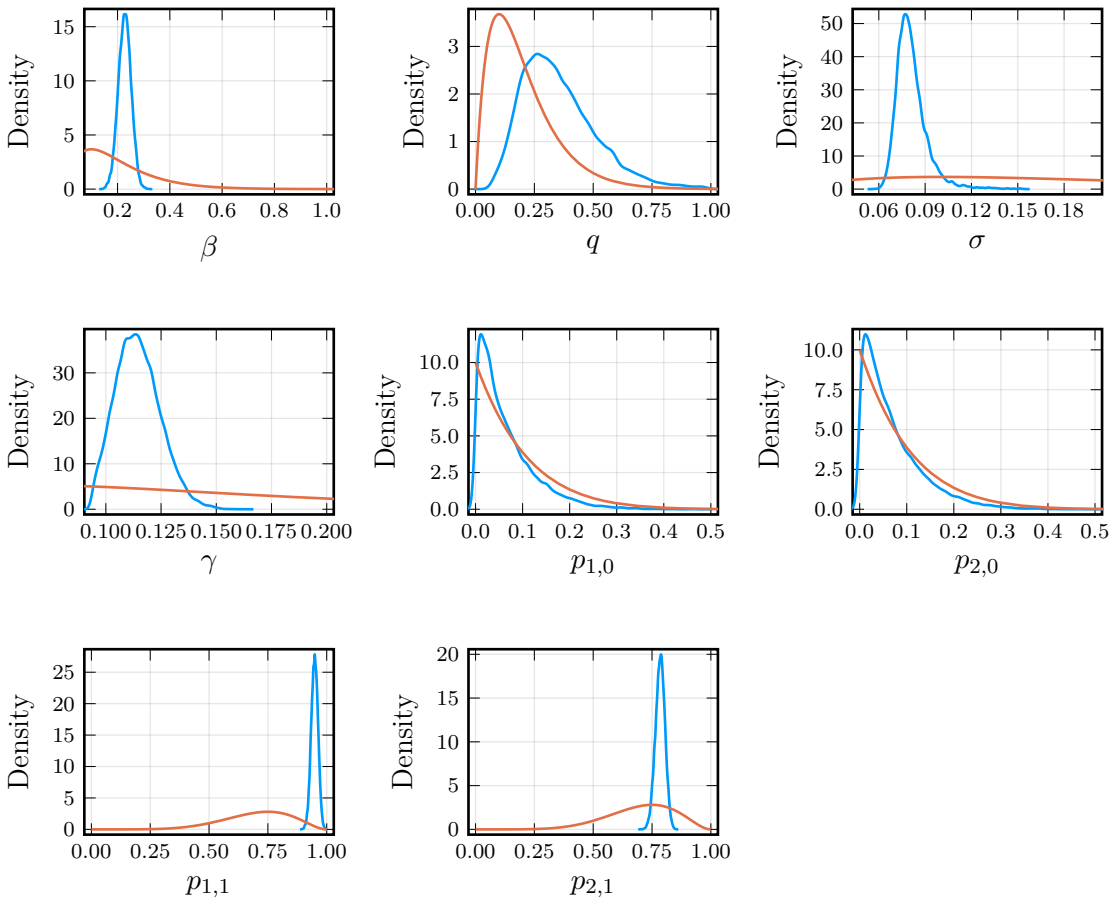

Figure 4.11: Marginal posterior distributions (blue) and priors (red) for the parameters of the model fitted to the 1995 outbreak.

A comparison with the results of the three different approaches from the literature are provided in Table 4.3. The 2-IS method is the current method; the DA-MCMC method refers to the stochastic continuous-time data augmented MCMC approach of McKinley *et al.* [51]; the chain-Binomial is the approach of Lekone and Finkenstädt [45] and they approximate the dynamics of the process with a stochastic, discrete-time binomial model; and the Deterministic model is the approach taken by Chowell *et al.* [18] which was fitted to a continuous-time model using least squares.

The 2-IS method proposed here produces posterior means and standard deviations that are most similar to the results of the DA-MCMC approach of [51]. In all cases, the parameter estimates appear quite different to the estimates from the chain-binomial and deterministic models. As mentioned in [51] this could be the result of the sheer number of missing data and the different approaches to dealing with this missingness in each method.

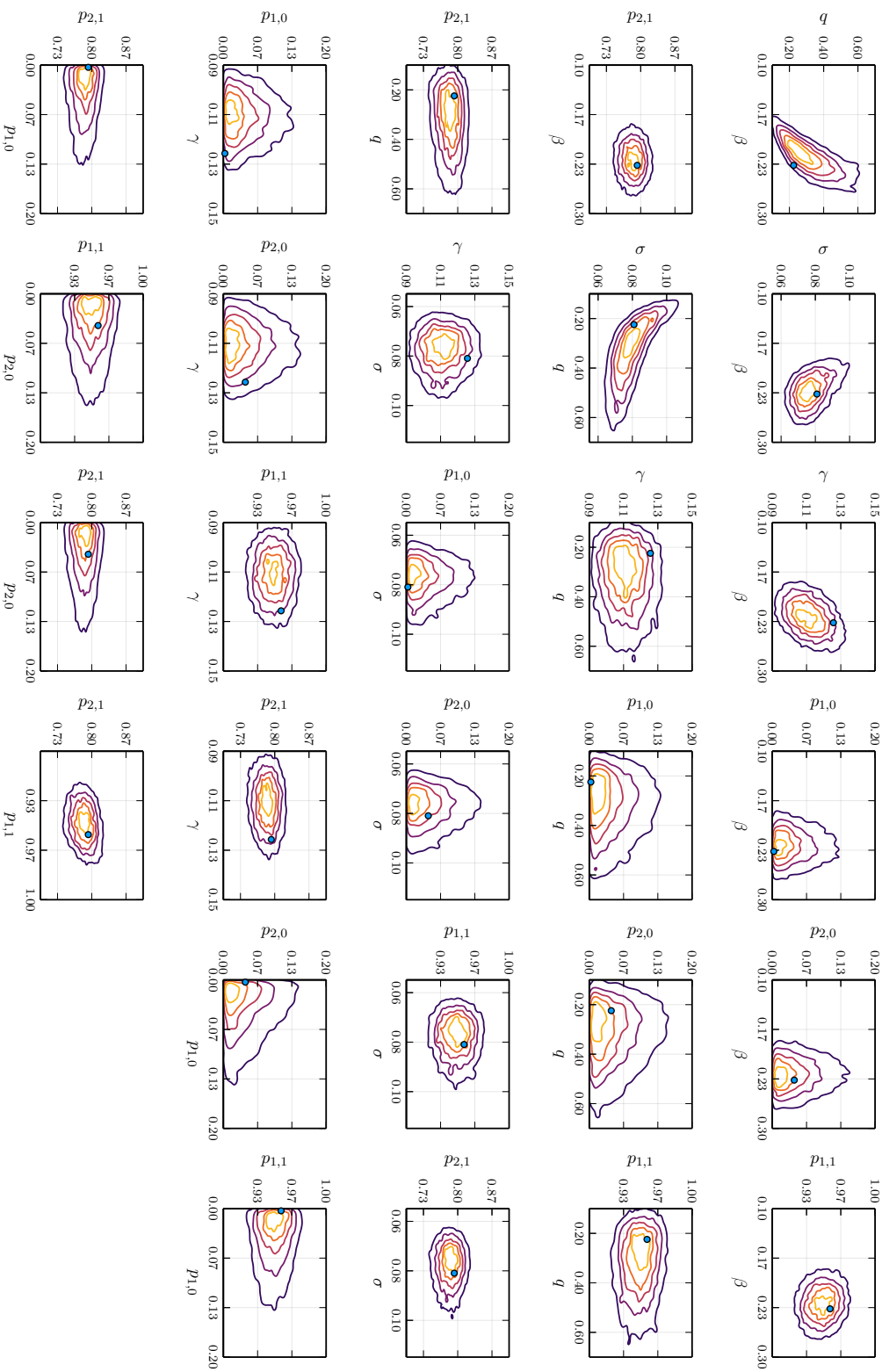

Figure 4.12: Bivariate marginal posterior distributions for the parameters of the model fitted to the 1995 outbreak. MAP estimate from the SPSA search are indicated by the blue point.

The methods used could easily result in posteriors which settle around a slightly different configuration of the parameters. It is also likely that since they fit to the data following the 55 day missing observation period that the parameter estimates are indicative of the missing data being only from the secondary phase of the outbreak.

The amount of missing data in each of the time series drives up the uncertainty for several of the parameters. Interestingly we see that the standard deviations for the incubation period is larger than that of the DA-MCMC estimate, which is approximately twice as large as the Chain-Binomial estimate. This suggests that the model here does indeed capture the uncertainty, but that it appears to primarily be in the incubation period as opposed to the infectious period, as is the case with the DA-MCMC approach. The inferred observation probabilities are relatively consistent with the empirically estimated proportions from the available data. This appears reasonable, but it is likely there is a greater time dependency to the observation probabilities.

Figures 4.11 and 4.12 show no spikes of density in the marginal posteriors and the bivariate marginal posterior distributions all appear smooth and unimodal. There do not appear to be any funnel degeneracies and there are no sharp edges in the marginal bivariate posterior densities outside of the observation probabilities $p_{1,0}$ and $p_{2,0}$, which are a result of these values being up against the boundary. There is an interesting relationship with the intervention effect q and latent period σ. This makes sense as the longer individuals take on average to show symptoms, the stronger we expect the intervention effect to be, as more individuals will be experiencing onsets in the later stages of the outbreak. The reverse trend is shown in the bivariate posterior for β and q. We see here that as β increases, then q increases also. In order to check the fit of the model we can assess the goodness of fit diagnostics. Again, we choose the outbreak duration and posterior predictive distributions for the incidences [51]. Here, as we know the outbreak took off, we condition the simulations on lasting at least 100 days and matching the detected incidence during this time.

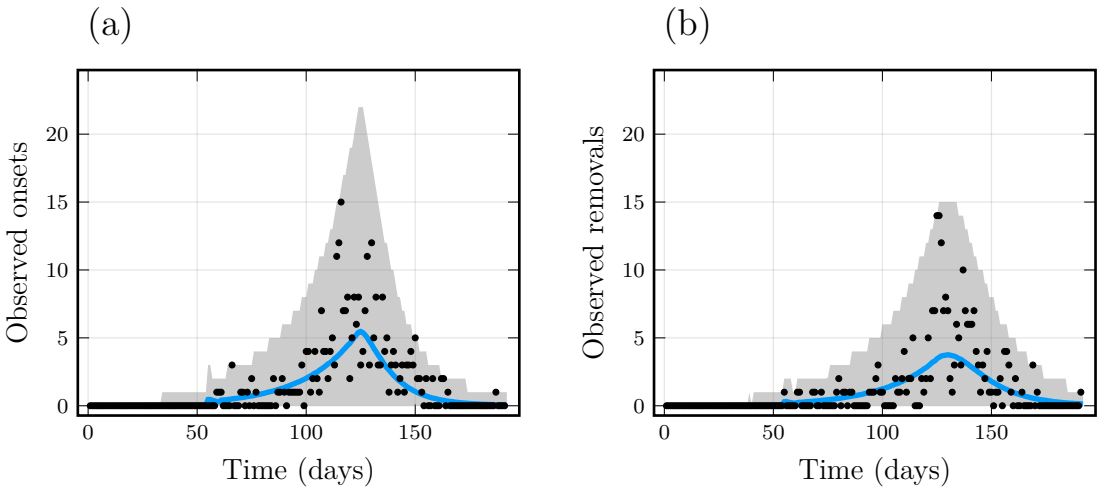

Figure 4.13: Kikwit data versus the posterior simulations. The black dots indicate the observed daily incidence. The grey area corresponds to the 95% credible interval of the simulations. The blue line indicates the mean incidence over the simulations.

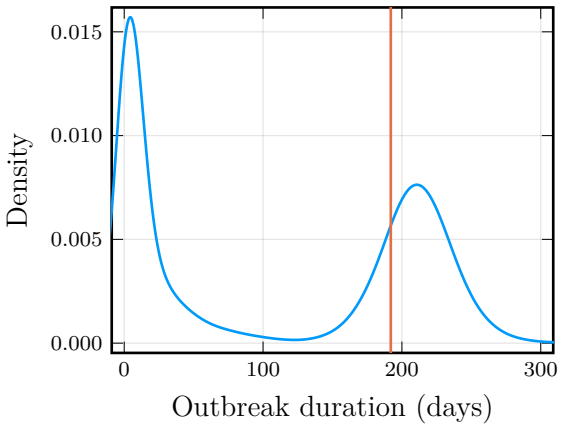

Figure 4.14: Posterior predictive distributions for the outbreak duration. The red lines in the plot corresponds to the observed time of final removal.

The posterior predictive distributions in Figures 4.13 indicate that the model is able to appropriately capture the outbreak dynamics. The mean (dashed line) appropriately matches the evolution of the observed data. These results are very similar to those obtained by McKinley *et al.* [51]. The posterior predictive distribution for the outbreak duration in Figure 4.14 shows that the outbreak is indeed a likely outbreak which did not fade out.

The effective reproduction number can be used to determine when R_{eff} drops below the threshold of one. This also provides us with a quantitative indication of how effective the interventions were. The time dependent term for $t > \tau$ is,

$$R_{\text{eff}}(t) = R_0 \exp\left(-q(t-\tau)\right).$$

We are interested in the first time that $R_{\text{eff}}(t) < 1$ and rearranging this we find,

$$t > -\frac{1}{q}\log\left(\frac{1}{R_0}\right) + \tau.$$

Solving this for the average values of the parameters, we find that the interventions reduce R_{eff} to less than 1, 3 days after interventions were introduced. This exponential decay can be seen in Figure 4.15 which shows the behaviour of R_{eff} for the average values of the parameters. This suggests the interventions were very effective in reducing the spread of Ebola. This is consistent with the results found in previous studies [45, 51].

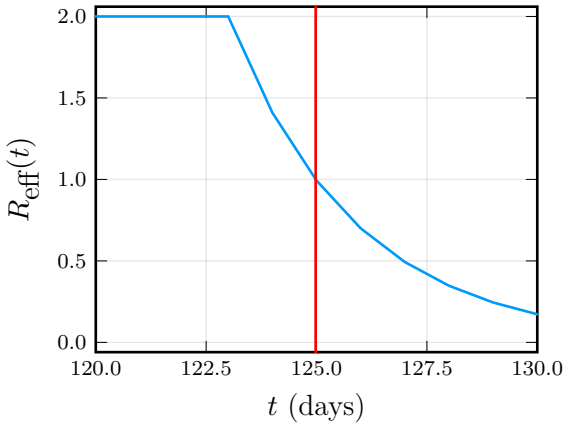

Figure 4.15: Plot of $R_{\text{eff}}(t)$ for the average parameters from the pmMH. The red line indicates the time at which $R_{\text{eff}}(t)$ drops below 1.

4.5 Summary

In this chapter we have detailed the development of a particle filtering methodology which uses importance sampling to produce realisations which are consistent with multiple datasets. The approach taken incorporates a binomial observation process into the model which enables the model to account for missing observations. We outline how the method

operates on a simple SIR model and then explore extensions necessary to fit the model to data from an outbreak of Ebola in the DRC.

The simulated example shows the ability for the method to marginalise over the missing data and recover the true parameter estimates. The posterior simulations show that the inference method works well. Inference results from the outbreak in the DRC are consistent with results reported in McKinley *et al.* [51]. This suggests the method appears to be able to be used to appropriately infer the parameters using two potentially censored time series. In the next chapter, we extend the model to a hierarchical context to fit to multiple outbreaks of Ebola.

Chapter 5

Hierarchical modelling of Ebola

This chapter builds on the work from Chapter 4, by using the particle filter to perform inference on four outbreaks of Ebola using a hierarchical model.

5.1 The data

There are six outbreaks of Ebola mentioned in Rosello *et al.* [61]; 1976 in Yambuku, 1995 in Kikwit, 2007 in Mweka, 2008 in Mweka, 2012 in Isiro, and 2014 in Boende. Each of these datasets are attached as supplementary material of [61] as a detailed line list. Rosello *et al.* [61] was the first paper to collate the data and make them all publicly available. This data provides records (where they exist) for dates of symptom onsets, hospitalisation, recoveries and death.

We make the assumption that the final sizes reported in [61] constitute the total number of infections over the course of each outbreak as Ebola has severe symptoms which are highly visible. We focus on only four of the six outbreaks as they are all outbreaks of the Zaire strain of ebolavirus [61]. We removed the 2007 outbreak as the data for this outbreak were very poor. There were no onset dates and the only proxy for this was the notification date. There was no detail on how these notifications were related to the actual date of onset and hence the methods struggled to fit to the data. The outbreak in Isiro was removed as it was identified to be of a different strain of ebolavirus, the less virulent, Bundibugyo strain [73, 61].

The data presented here is stratified into two time series for each outbreak. In order to construct this we used the data presented in Rosello *et al.* [61] and performed some pre-processing. The first point was to use the date of symptom onset and where this

was not available, to use the hospital admission date as a proxy. This inherently adds some variability to the potential estimation of latent and infectious periods but we note that more than 90% of cases were known with only their symptom onset for each of the considered outbreaks. For the removal time series we merged the date of recovery, death and end of disease dates. Merging the time series in this fashion means we lose a great deal of information on direct pathways individuals take throughout the outbreaks (such as those which progress into the hospital). That being said, there is a high degree of missing data across these more specific observations, particularly for the removal curves. This means that we are able to make better inferences on some simpler characteristics by combining information. However it should be noted that there is the potential to extend the methodologies to fit to a greater number of time series. We see in the outbreaks here, that documentation of cases—even when there are relatively few cases—is a difficult exercise.

Yambuku, 1976

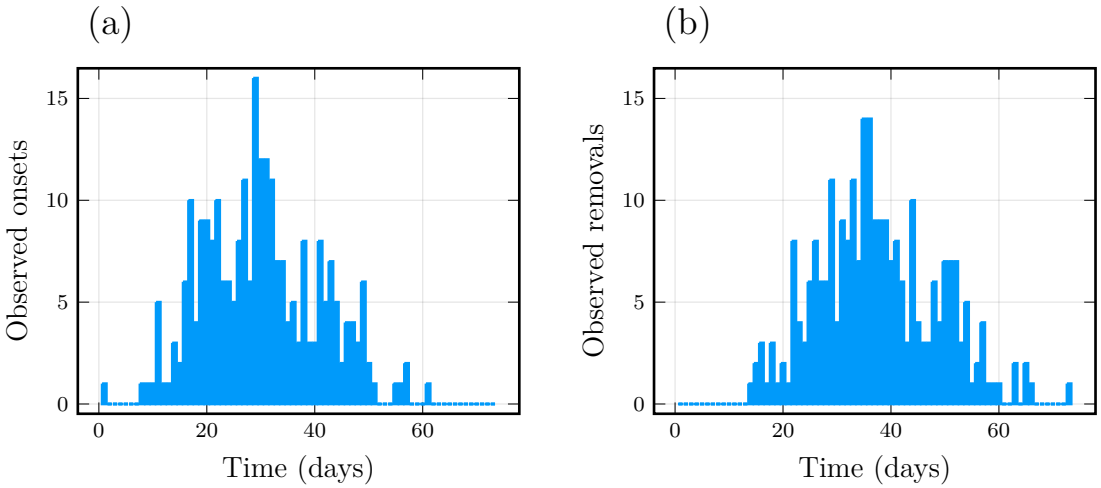

Figure 5.1: Daily incidence for (a) onsets and; (b) removals, during the 1976 outbreak of Ebola in Yambuku.

The 1976 outbreak in Yambuku began on the August 27 and was deemed over on November 6. Over this time the dates for 257 symptom onsets and 259 removals were known to daily precision. The time series are plotted in Figure 5.1. Overall there were 318 documented cases of Ebola. It is documented by Rosello *et al.* [61] and Camacho *et al.* [14] that a large proportion of the transmission was a result of needle sharing in the Yambuku Mission hospital. On September 30 (35 days after the start of the outbreak), the Yambuku

Mission hospital was closed and this heavily reduced the transmission[1]. Alongside this there was also a natural reported change of transmission as awareness of the ongoing outbreak increased [14, 61]. People started to reduce their contact during burial processes and attendance to homes of those in their village which were ill. It was estimated that close to 500 villages were potentially at risk during the outbreak and these villages were typically home to around 500 individuals [59, 61]. This estimates the population size at around 275, 000. This is different to the population used in other studies but is unlikely to have a great impact due to the low number of cases in proportion to the total population size.

Kikwit, 1995

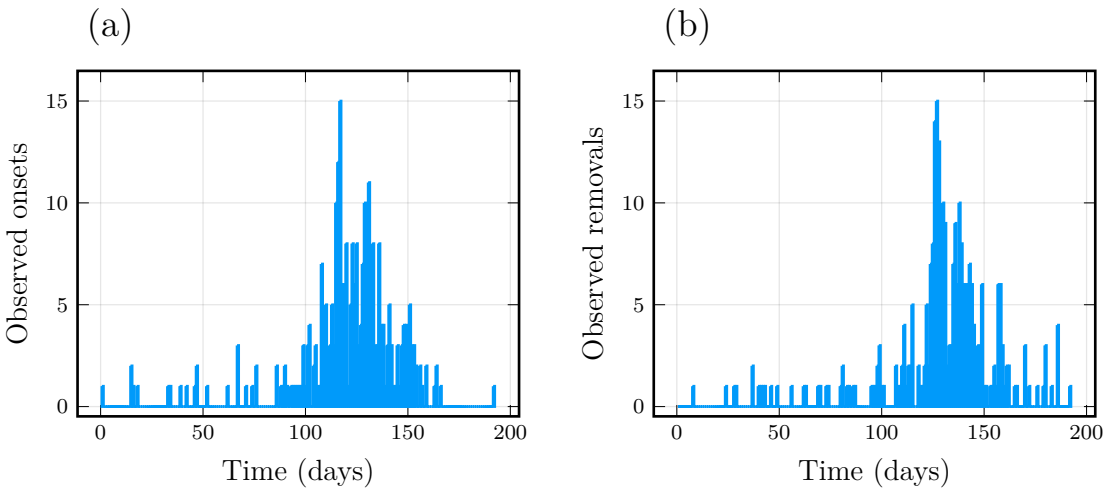

Figure 5.2: Daily incidence for (a) onsets and; (b) removals, during the 1995 outbreak of Ebola in Kikwit.

The 1995 outbreak in Kikwit began on January 6 and was deemed over on July 16. Over this time there were 317 reported cases and of these, the day of symptom onset was known for 293 of these and the day of removal through either recovery (248) or death (42) was known for 290 of these. The time series are plotted in Figure 5.2. This is more than the numbers reported in the Chapter 4, but the symptom onset was constructed using the known dates as well as hospitalisation dates which reasonably coincided with the onset of severe symptoms [51]. Interventions involved the use of PPE, public education and closure

[1]There is some uncertainty over the date reported in [61] as in [14] the date of closure is stated to be September 30. The models we consider for this outbreak of Ebola both assumed that this date was unknown and we attempt to infer the date of change in transmission as an additional parameter.

of the Kikwit hospital 123 days into the outbreak on May 9. We assume an estimated population of size 200, 000 which is much less than that used in the studies in Chapter 4 but appears to more appropriately reflect the actual population at risk [61].

Mweka, 2008

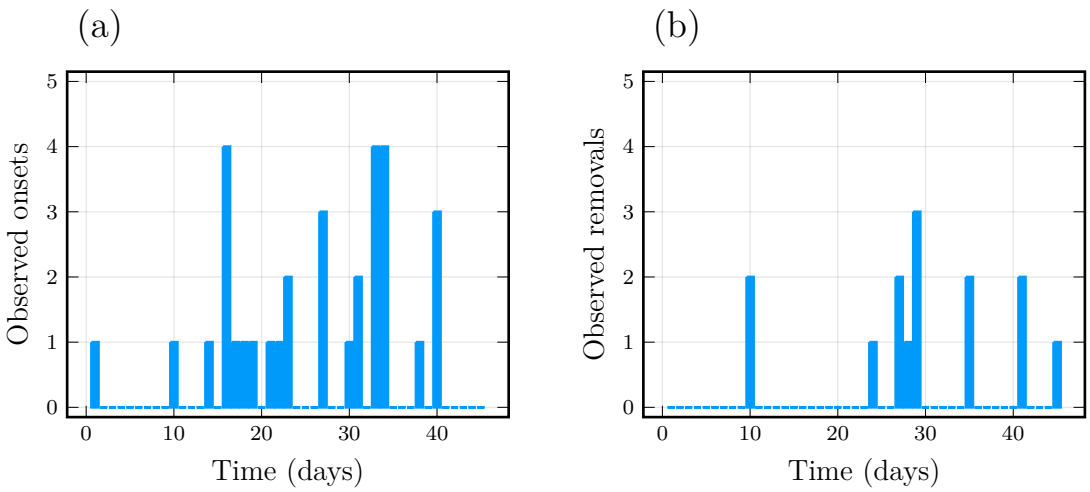

Figure 5.3: Daily incidence for (a) onsets and; (b) removals, during the 2008 outbreak of Ebola in Mweka.

The 2008 outbreak in Mweka was the second outbreak of Ebola in as many years. There were 32 cases from November 18, 2008 to January 1, 2009. Of these, the day of onset was known for all cases and the day of removal was known for 14. The time series are plotted in Figure 5.3. Intervention came in the form of the arrival of national and international support from the 18th to 25th of December, 2008. The first major intervention measure was the opening of an isolation centre by MSF Belgium in Kaluamba on 27 December. This was the last day a case was detected and hence it appears as though interventions were effective in stopping the spread at the end of the outbreak. We assume a population size of 170, 000 as given in [61].

Boende, 2014

The outbreak in Boende, 2014 saw 68 cases of Ebola identified. Of these there were 68 known days of onset and 43 known days of removal. The time series are plotted in Figure 5.4. The outbreak began on the 26th of July and was deemed over by October 5.

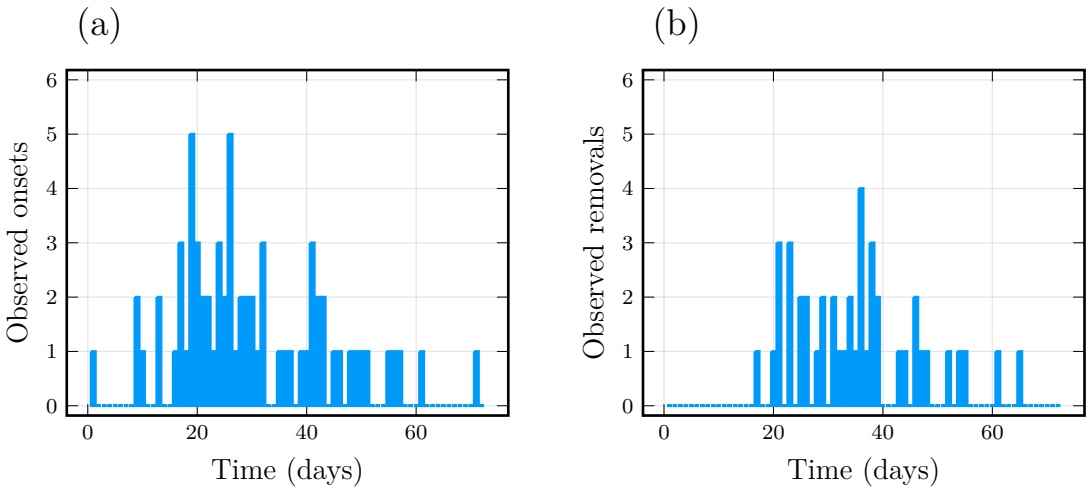

Figure 5.4: Daily incidence for (a) onsets and; (b) removals, during the 2014 outbreak of Ebola in Boende.

Interventions were introduced on September 14 in the form of two Ebola treatments being opened. One which was built from scratch in Lokolia and one in the General Reference hospital of Boende [61]. We assume an estimated population size of $200,000$ as given in [61].

5.2 Epidemic model development

The model used for each outbreak has the same SEIR structure as used in Chapter 4. This is done so as to promote a common model structure across the four outbreaks, and demonstrate the importance sampling method in a higher dimensional problem. The only difference between the model fitted to each outbreak in this chapter are that we assume a more complex time-dependent transmission term compared to Chapter 4.

Camacho *et al.* [14] model the outbreak in Yambuku using a complex eight compartment epidemic model, and fit to four time series which stratify individuals by their path of transmission. This enables much more information to be drawn from the data but also requires a methodology capable of dealing with the excessive missingness of each individual dataset [14]. One issue with the approach taken in Camacho *et al.* [14] is that most of these time series are heavily censored and information is missing for some of the processes for a range of the individuals. Stratifying the information increases the dimensionality of the problem and hence some parameters may become unidentifiable. While Camacho

et al. [14] are focused on attempting to quantify the contribution to R_0 from the different transmission routes, our aim for this study is slightly different and as such we choose to simplify the modelling processes and use the simpler SEIR model. This merges our particle filtering approach with a hierarchical Bayesian framework which will enable us to pool the information from each outbreak.

We attempted to fit a complex transmission parameter which directly accounts for the closure of the hospitals in the Yambuku and Kikwit outbreaks. Unfortunately, this proved to be difficult largely due to an inability to choose appropriate priors and a lack of stratification of the data according to route of transmission. Hence we consider a global reduction on the total transmission which takes into account changes in community behaviour and the effect of interventions. The infection rate is,

$$a_1 = \tilde{\beta}(t)\frac{IS}{N-1}, \tag{5.1}$$

where the new time dependent effective transmission term is,

$$\tilde{\beta}(t) = \beta\left(1 - \frac{\delta}{1+e^{-q(t-\tau)}}\right), \tag{5.2}$$

where β is the effective transmission parameter at the onset of the outbreak. The reduction in the transmission rate following the community change is quantified by δ, namely as $t \to \infty$, the effective transmission parameter reduces to $\beta(1-\delta)$. The parameter q characterises the rate of change in contact behaviour and explains how quickly the initial effective transmission parameter is reduced to $1-\delta$ of its value. The parameter τ is the date of the midpoint of the change in contact behaviour (the midpoint of the curve) and needs to be inferred.

The ways in which this parameterisation influences the transmission term can be determined by comparing the simpler form from Chapter 4,

$$\tilde{\beta}(t) = \begin{cases} \beta, \text{ for } t < \tau \\ \beta e^{-q(t-\tau)} \text{ for } t \geq \tau, \end{cases}$$

to the functional form presented here. Let $R_0 = 2, q = 0.2, \tau = 50$ in each of the parameterisations and let $\delta = 1$ in the new time dependent term to reflect similar dynamics to the more naive form. This means that the effective reproduction number $R_{\text{eff}}(t) = \tilde{\beta}(t)/\gamma$ will approach 0 as $t \to \infty$. Figure 5.5 shows the differences between the two approaches (red for the new functional form and blue for the functional form from Chapter 4).

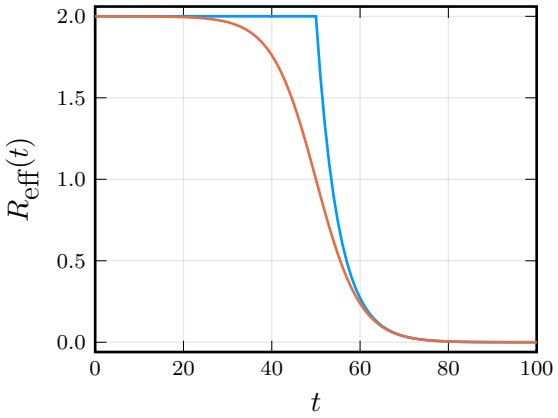

Figure 5.5: The functional forms of $R_{\text{eff}}(t)$ for each of the parameterisations for the parameters $R_0 = 2, q = 0.2, \tau = 50, \delta = 1$. The blue line indicates the basic form from Chapter 4 and the red line indicates the more complex term used here.

In Figure 5.5 we can see that the change occurs much earlier than the date of intervention measures in this toy example. It is also more gradual in that there is not a sudden drop in R_{eff}. This may or may not be appropriate and would be heavily dependent on the intervention measures which were applied. An example where the original functional form may be more appropriate is in the case of a major lockdown whereby individuals are not allowed to interact (similar to what has been seen for COVID-19). In this instance a sudden reduction in transmission would be reasonable and expected. The new functional form allows for conclusions to be drawn about reductions due to overall changes arising from interventions and changes to community interaction. It does however prove difficult to quantify individual effects from direct intervention versus natural community change as per the Chapter 4 analysis, but this proves useful in quantifying whether intervention measures were the main cause for the reduction in transmission during outbreaks of Ebola.

The advantage of this formulation as opposed to the one used previously is that it accounts for reduction in the transmission prior to the implementation of major interventions (such as hospital closures) and steadily sees this reduction increase up until τ. Following the mid-point the reduction in the effective transmission parameter tapers off in an exponential decay fashion to a potentially non-zero quantity, which is a more realistic model of the true dynamics. Individuals in the community are expected to grow increasingly cautious following deaths of individuals and widespread infection. The idea that interventions are typically most useful at their initial deployment and cannot guarantee a complete elimination of transmission is also reflected in this term which appears more appropriate in comparison to the model in [45] but does not suffer from identifiability

issues as the formulation in [18].

In this model we again assume that the average exposure period and infectious period are $1/\sigma$ and $1/\gamma$ respectively. We also incorporate the observation probabilities p_k for each outbreak where $k = 1$ denotes the probability of observing an onset of symptoms and $k = 2$ denotes the probability of observing a removal. For the Kikwit outbreak we then split these probabilities between the first stage of the outbreak when active surveillance was not being conducted. This is denoted by $p_{k,0}$ and the second phase is represented by $p_{k,1}$. The other parameters are as defined in this Section. Let $\boldsymbol{\theta}_i$ denote the parameters for each outbreak, where, Yambuku ($i = 1$), Kikwit ($i = 2$), Mweka ($i = 3$) and Boende ($i = 4$). The parameters of each model are given as,

$$\boldsymbol{\theta}_i = \left(R_0^{(i)}, q^{(i)}, \frac{1}{\sigma}^{(i)}, \frac{1}{\gamma}^{(i)}, p_1^{(i)}, p_2^{(i)}, \delta^{(i)}, \tau^{(i)} \right), \text{ for } i = 1, 3, 4,$$

$$\boldsymbol{\theta}_i = \left(R_0^{(i)}, q^{(i)}, \frac{1}{\sigma}^{(i)}, \frac{1}{\gamma}^{(i)}, p_{1,0}^{(i)}, p_{2,0}^{(i)}, p_{1,1}^{(i)}, p_{2,1}^{(i)}, \delta^{(i)}, \tau^{(i)} \right), \text{ for } i = 2.$$

5.3 Independent inferences

In this section we seek to conduct inference on the outbreaks independently. This lays the groundwork for making more informative choices of parameters in the hierarchical model and enables us to identify any issues with fitting the models. This section also provides us with a comparison point for the hierarchical model. We should see some similar results for the hierarchical model compared to the independent inferences, but with a clear influence of the pooling effect on shared parameters.

The time dependent transmission term is justified for both the 1976 and 1995 outbreaks as we can see a clear peaking to the spread and subsequent drop off around the time interventions are introduced. Conversely, for both the 2008 and 2014 outbreaks we see that interventions arrived late in the outbreak, following the peak of observations. This suggests that there may have been natural changes to transmission chains as a result of community awareness over Ebola or that the outbreaks faded out naturally [61]. To capture this steady change we assume that the change in contact behaviour parameter, q follows a Gamma$(2, 0.1)$ prior in all cases. This makes sure that if the change does occur, we expect it to be a gradual reduction—as would be adopted in the community—rather than an unrealistic instantaneous change in transmission. In order to facilitate the choice of prior on q we allow uninformative priors on both the midpoint to change of transmission reduction τ and reduction in transmission δ. We assume τ is equally likely to occur over the duration of the outbreak. We then assume that $\delta \sim \text{U}(0, 1)$ and hence we allow the data to determine the strength of the reduction based on the assumption of a gradual

change.

The prior on R_0 is chosen to relay the average values found in previous studies on the Zaire ebolavirus [14, 18, 73, 45, 51]. This is that R_0 estimates have been found in the range from $(1.4, 4.7)$. Reported values have been averaged to see an estimate of around 2.1 as the most likely value. To facilitate this we choose a prior of Gamma$(5, 0.5)$ which feasibly allows for this range of values and has a mode of 2, which is consistent with majority of reported estimates.

We assume a tighter prior on the incubation period than in Chapter 4. The majority of estimated incubation periods were found to be between 4 and 12 days with most reported values being close 6 days [73]. We use this to enforce a prior distribution centred on 6 days, $1/\sigma \sim \text{Normal}(6, 1)$.

The infectious period is more difficult to define because of some dynamics of Ebola which we are neglecting here (like individuals going to hospital and traditional burials). We have simplified the process somewhat to enable characterisation of some key epidemiological quantities across outbreaks of Ebola. In order to facilitate this there is the sole removal compartment in the model and all individuals transition to this compartment following removal, either through recovery or death. The average infectious period is thus defined across all individuals, and not stratified by whether they were in hospital or not. As such we expect slightly higher variability in this estimate. Reported estimates of the overall average time an individual spent infectious was around 9 days [73, 45, 51]. As such we take the prior $1/\gamma \sim \text{Normal}(9, 1)$.

The priors on the observation densities are loosely chosen in accordance with the ratio of detected cases against the total number of cases found from later observations. This naturally is captured by Beta distributions which are assigned to reflect the empirical proportion of observed cases. The priors on the observation probabilities, midpoints for change in transmission are featured in Table 5.1. Note that the wide uniform priors on the midpoint parameters $\tau^{(i)}$ reflect that the change in transmission is equally likely to occur at any point over the course of the outbreak.

Out of simplicity we assume initial conditions of $\boldsymbol{Z}(0) = (1, 1, 0, 0, 0)$ for all outbreaks other than Yambuku. This corresponds to a single infectious individual which has progressed onto having symptoms. This corresponds to the event on the first day in each context. For the Yambuku outbreak we assume that there was a secondary initial infectious case, $\boldsymbol{Z}(0) = (2, 1, 0, 1, 0)$ [12, 14].

Parameter	Description	Prior / Definition
$q^{(i)}$	Shape of change of contact behaviour	Gamma$(2, 0.1)$
$p_1^{(i)}$	Proportion of onsets observed for outbreak i	Beta$(7, 2)$
$p_2^{(i)}$	Proportion of removals observed for outbreak i	Beta$(4, 2)$
$p_{1,0}$	Midpoint date for the change of person-to-person contact behaviour for Kikwit outbreak	Beta$(2, 20)$
$p_{2,0}$	Midpoint date for the change of person-to-person contact behaviour for Kikwit outbreak	Beta$(2, 20)$
$p_{1,1}$	Midpoint date for the change of person-to-person contact behaviour for Kikwit outbreak	Beta$(7, 3)$
$p_{2,1}$	Midpoint date for the change of person-to-person contact behaviour for Kikwit outbreak	Beta$(7, 3)$
$\tau^{(1)}$	Midpoint date for the change of person-to-person contact behaviour for Yambuku outbreak	$U(1, 72)$
$\delta^{(1)}$	Reduction of the person-to-person transmission rate following change of contact behaviour for Yambuku outbreak	$U(0, 1)$
$\tau^{(2)}$	Midpoint date for the change of person-to-person contact behaviour for Kikwit outbreak	$U(1, 191)$
$\delta^{(2)}$	Reduction of the person-to-person transmission rate following change of contact behaviour for Kikwit outbreak	$U(0, 1)$
$\tau^{(3)}$	Midpoint date for the change of person-to-person contact behaviour for Mweka outbreak	$U(1, 44)$
$\delta^{(3)}$	Reduction of the person-to-person transmission rate following change of contact behaviour for Mweka outbreak	$U(0, 1)$
$\tau^{(4)}$	Midpoint date for the change of person-to-person contact behaviour for Boende outbreak	$U(1, 72)$
$\delta^{(4)}$	Reduction of the person-to-person transmission rate following change of contact behaviour for Boende outbreak	$U(0, 1)$

Table 5.1: Full set of model parameters for the independent inferences. Note that Normal distributions are truncated where appropriate.

All particle filters were run in parallel on 8 cores. There were 200 particles used for the Yambuku and Boende inferences, 100 for the Mweka inference and 400 for the Kikwit inference. We are able to use slightly less particles for the Kikwit model than in the previous chapter as there is more certainty to the evolution of the outbreak over the first 55 days. The numbers of particles were chosen to obtain reasonable mixing. We run independent pmMH routines for each outbreak. The inferences on the Yambuku, Kikwit, Mweka and Boende data were run for 4, 12, 2 and 2 hours respectively. This provided comparable minimum ESS of above 500 for each chain. Marginal posterior distributions

are shown in Figures 5.6–5.9 and summaries of the posterior distributions (means and 95% HPD intervals) are shown in Table 5.2.

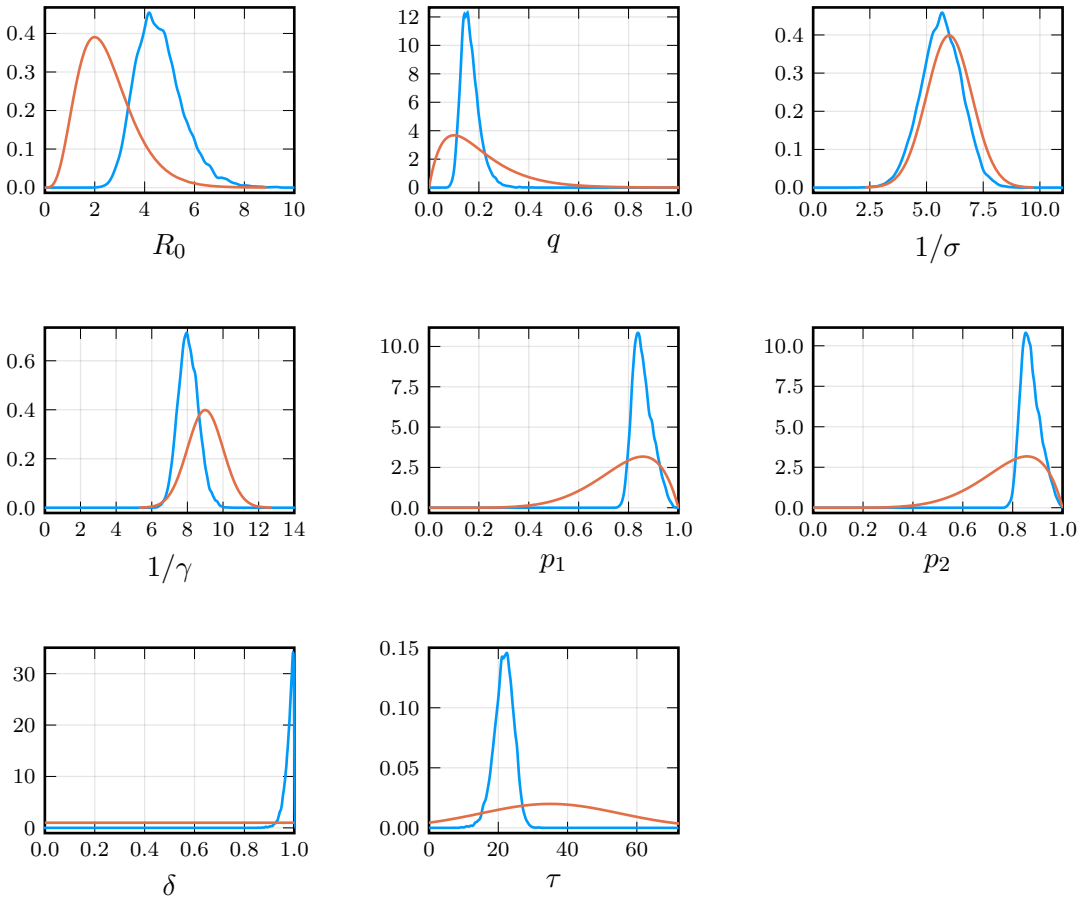

Figure 5.6: Marginal posterior distributions (blue) and priors (red) for the parameters of the model fitted to the Yambuku outbreak.

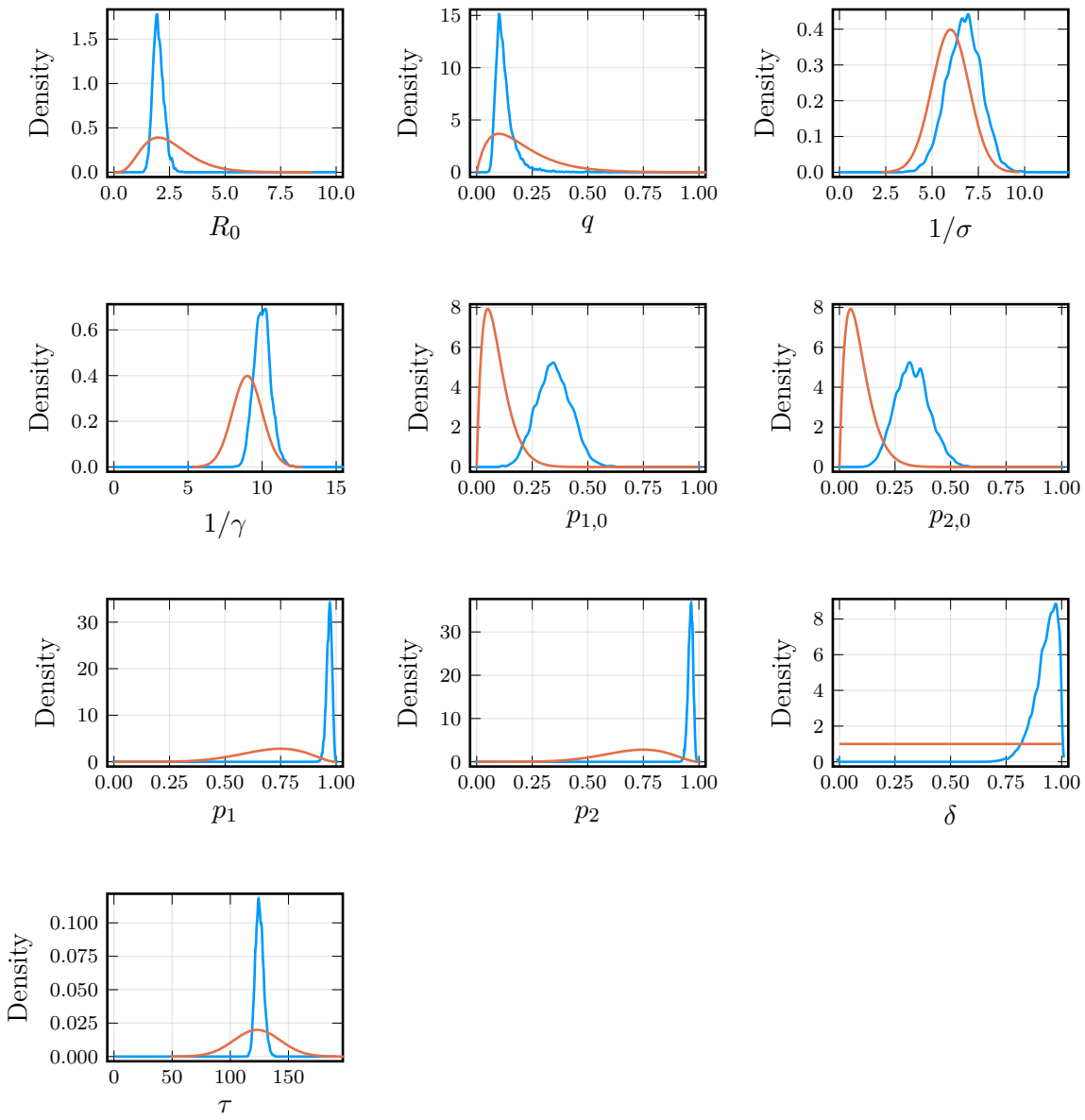

Figure 5.7: Marginal posterior distributions (blue) and priors (red) for the parameters of the model fitted to the Kikwit outbreak.

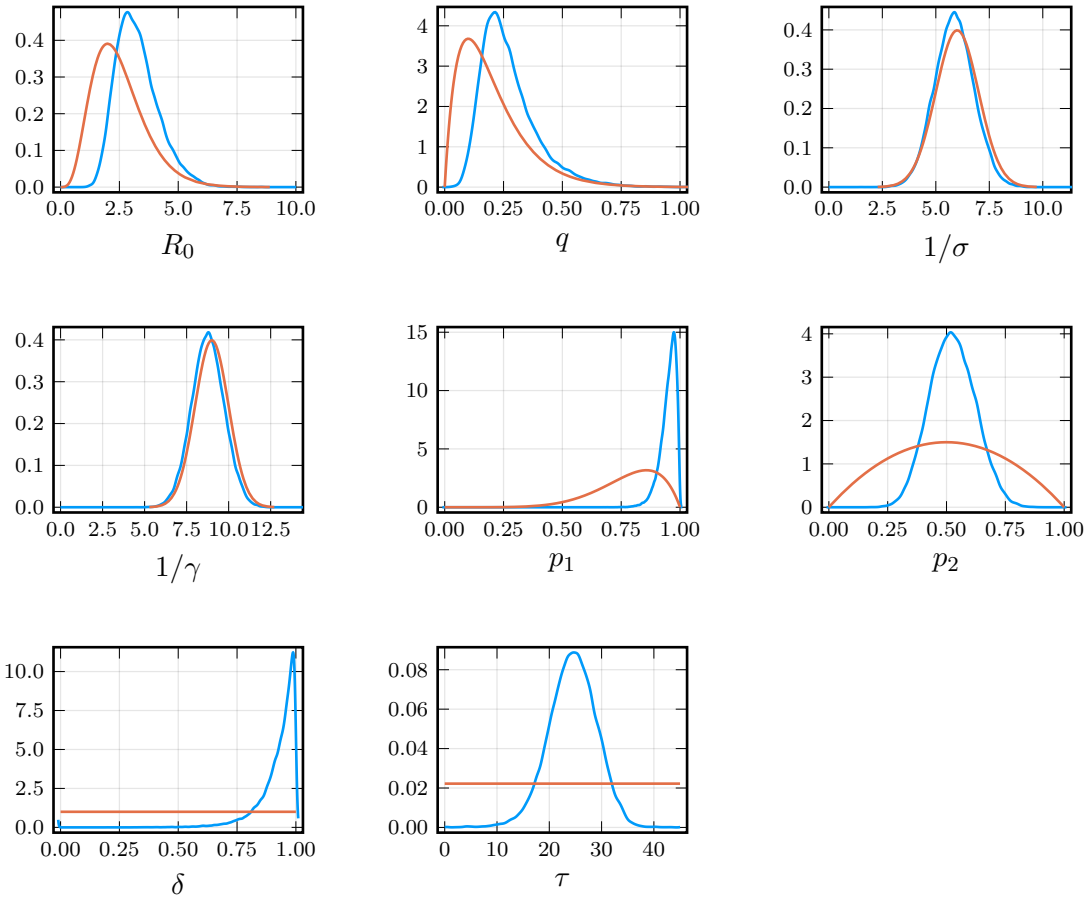

Figure 5.8: Marginal posterior distributions (blue) and priors (red) for the parameters of the model fitted to the Mweka outbreak.

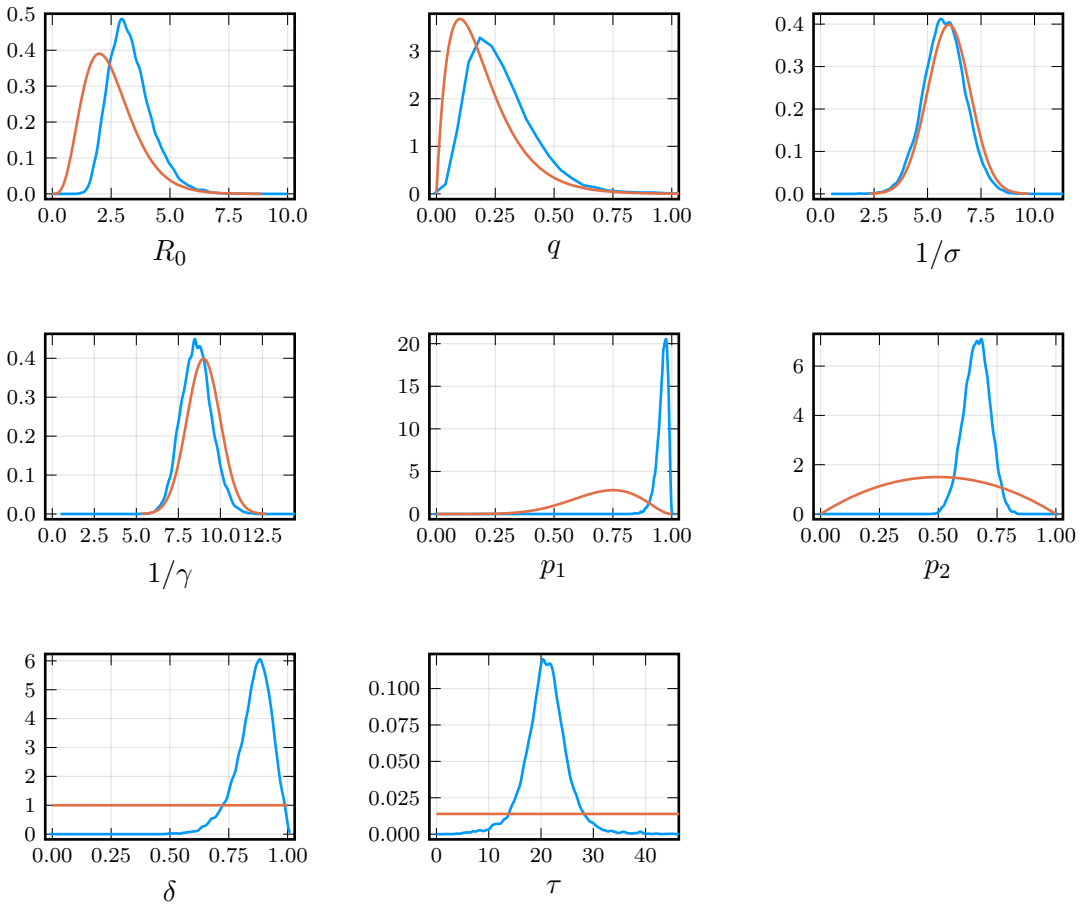

Figure 5.9: Marginal posterior distributions (blue) and priors (red) for the parameters of the model fitted to the Boende outbreak.

Parameter	Yambuku	Kikwit	Mweka	Boende
R_0	4.60 (2.87, 6.53)	2.00 (1.55, 2.49)	3.26 (1.64, 5.04)	3.33 (1.73, 5.15)
q	0.17 (0.10, 0.24)	0.13 (0.07, 0.22)	0.27 (0.08, 0.52)	0.28 (0.06, 0.55)
$1/\sigma$	5.65 (3.92, 7.42)	6.73 (4.90, 8.45)	5.87 (4.11, 7.66)	5.79 (3.88, 7.67)
$1/\gamma$	8.06 (6.94, 9.16)	10.02 (8.94, 11.13)	8.77 (6.88, 10.69)	8.58 (6.86, 10.41)
p_1	0.86 (0.79, 0.94)	–	0.95 (0.89, 1.00)	0.96 (0.92, 1.00)
p_2	0.87 (0.81, 0.95)	–	0.53 (0.34, 0.72)	0.66 (0.56, 0.77)
$p_{1,0}$	–	0.35 (0.20, 0.50)	–	–
$p_{2,0}$	–	0.33 (0.18, 0.47)	–	–
$p_{1,1}$	–	0.97 (0.94, 0.99)	–	–
$p_{2,1}$	–	0.96 (0.94, 0.98)	–	–
δ	0.98 (0.94, 1.00)	0.93 (0.83, 1.00)	0.92 (0.77, 1.00)	0.86 (0.71, 0.99)
τ	21.63 (15.85, 26.80)	125.12 (118.49, 131.76)	24.42 (15.71, 33.58)	21.01 (12.68, 29.72)

Table 5.2: Posterior means and 95% HPD (in parentheses) from routines fitted to the four outbreaks of Zaire Ebolavirus.

We see that across the four outbreaks the initial value of R_0 is estimated above 1. These estimates are roughly consistent with one another and the values reported in literature. The estimate for the Yambuku outbreak is inflated—relative to the other outbreaks—as we would expect given the increased hospital transmission and this being the first outbreak of Ebola. The estimate for the Kikwit outbreak is consistent with our results form the previous analysis (assuming the simpler time dependent transmission term) and other reported values of R_0 [73].

The change in contact behaviour q is shown to be slightly higher in the Mweka and Boende outbreaks compared to the Yambuku and Kikwit outbreaks. This appears reasonable as the counts were much lower over the course of the outbreak and these outbreaks were short in comparison. The reduction from changing community behaviour δ show that the measures taken in the Yambuku outbreak were most effective. This is likely due to the severe measures taken in closing of the hospitals and how this would have influenced individuals perceptions of Ebola. The reductions are also quite strong in the other three outbreaks. The latent and infectious periods are all roughly consistent with one another. There is a larger degree of variability in the infectious period and this is likely due to mis-reporting of removals. This is something that is unavoidable and a product of data collection.

It is interesting that all changes to transmission occurred before the known dates of intervention. This suggests that the change in community behaviour was a major contributing factor to the reduction in transmission. We can quantify the change in contact behaviour by solving for the effective reproduction number and seeing when this drops below 1. For the models in this section, the effective reproduction number is,

$$R_{\text{eff}}(t) = R_0 \left(1 - \frac{\delta}{1 + \exp\left(-q\left(t - \tau\right)\right)}\right).$$

Solving this for $R_{\text{eff}}(t) \leq 1$ we gain some insight into when the outbreaks were under control. The solution to this is,

$$t^\star = -\frac{1}{q} \log\left(\frac{\delta R_0}{R_0 - 1} - 1\right) + \tau.$$

The Yambuku outbreak saw R_{eff} drop to below 1 after 30.2 days with 95% HPD $(28.1, 32.1)$ days. Major interventions were introduced 35 days into the outbreak suggesting that the community began adopting changes to their mixing behaviour before this. The interventions in this outbreak arrived slightly after the peak incidence and hence appears likely these measures aided in eliminating Ebola in the community as opposed to controlling it.

The Kikwit outbreak had R_{eff} drop to below 1 after 126.2 days with 95% HPD $(122.6, 130.4)$ days. The major intervention was the closure of the hospital and this occurred 123 days into the outbreak. It seems clear as a result of the lower bound of the HPD interval that the closure of the hospital was a contributing factor to the reduction in transmission but that there was some change prior to the closure.

For both the Mweka and Boende outbreaks R_{eff} did not necessarily drop to below 1. This is likely a result of the limited data available. It is also probable that as both these outbreaks had all onsets detected but a large portion of missing removals (half removals missing for Mweka and a third missing for Boende) that the model may incorrectly account for this. It is also possible that each of these outbreaks did not take off when we compare them to the Yambuku and Kikwit outbreaks. This may be more likely for Ebola as the offspring distribution has a large variance [43]. In these two outbreaks it is clear that changes did occur and that due to the low number of cases, it is likely that R_{eff} did not need to drop to below 1 in order for the epidemic to fadeout.

To validate our models we can assess posterior predictive distributions of the detected incidences. The simulations and their conditions are provided in Appendix B and demonstrate that all the models appear to fit well. The mean incidence from each posterior simulation closely matches the observed incidences of each of the outbreaks.

5.4 Hierarchical modelling process

A major advantage of a hierarchical approach is that information contained in the larger, more well documented outbreaks can be used to help fit the models to outbreaks which were less informative. Parameters like R_0, the latent period, $1/\sigma$, and infectious period, $1/\gamma$, are characteristics of outbreaks of Ebola and/or African communities. As such we expect there to be similarities between outbreaks.

Let $\mathcal{D}_i, i = 1, \ldots, 4$ correspond to the data for Yambuku ($i = 1$), Kikwit ($i = 2$), Mweka ($i = 3$) and Boende ($i = 4$) outbreaks. The data $\mathcal{D} = \{\boldsymbol{y}_{1,1:T}, \boldsymbol{y}_{2,1:T}\}$ are vectors with elements $y_{1,t}$ and $y_{2,t}$ which correspond to the number of detected symptom onsets and detected removals respectively on day t. For simplicity of notation let $\mathcal{D} = \{\mathcal{D}_1, \ldots, \mathcal{D}_4\}$ denote all the data sets. Let $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_4)$ where $\boldsymbol{\theta}_i$ are the parameters for outbreak i, and let $\boldsymbol{\phi}$ be the hyper-parameters for the hierarchical model.

The hierarchical nature is evident when considering that the quantities $R_0, 1/\sigma$ and $1/\gamma$ are related to outbreaks of Ebola. This enables us to quantify the feasible values of these parameters across the outbreaks as well as the between outbreak variability. An approach for doing this is to assume that this top level of the hierarchy has parameters corresponding to the mean (or mode) and variance in the outbreak level parameters. In order to account for the known differences in R_0 between outbreaks and attempt to capture this in the model, we can use a non-centred approach to representing R_0. That is we assume a normal hierarchical structure whereby,

$$R_0 | \mu_{R_0}, \nu_{R_0} \sim N(\mu_{R_0}, \nu_{R_0}),$$

but we write,

$$R_0 = \mu_{R_0} + \nu_{R_0} \eta_{R_0}, \tag{5.3}$$

where we set priors on the population mean and variance of R_0 to capture the uncertainty in this baseline value. The overall value of R_0 can be truncated by setting bounds on the transformed parameter. We assume $\eta_{R_0} \sim N(0, 1)$ which can be seen to ensure that these two parameterisations are equivalent but that the latter enables us to appropriately infer the difference in R_0 for each outbreak which may arise due to various environmental or temporal factors, such as base knowledge of Ebola in the community and the year in which the outbreak occurred. This parameterisation essentially lets us consider μ_{R_0} as the expected R_0 in an average population. The parameter ν_{R_0} then approximately captures the variability in R_0 across the outbreaks. The value of η_{R_0} for each outbreak is a measure of how different that outbreaks basic reproduction number is from the average across the outbreaks.

For the latent and infectious period the scenario is simpler as each of these are unlikely to be influenced by other factors and are a property of the disease itself. We assign the following priors,

$$\frac{1}{\sigma}|m_\sigma, \nu_\sigma \sim \text{Gamma}(a_\sigma, b_\sigma),$$
$$\frac{1}{\gamma}|m_\gamma, \nu_\gamma \sim \text{Gamma}(a_\gamma, b_\gamma),$$

where m_σ, m_γ are the modes of prior distributions and ν_σ, ν_γ are the variances. We choose to work with modes here instead of means as the mode ensures more well behaved posterior distributions. As the hyper-parameters correspond to the mode and variance of the prior distribution so we transform them back to the shape and scale, a, b of the Gamma distributions by noting that $X \sim \text{Gamma}(a, b)$ has mode $m = (a-1)b$ and variance $\nu = ab^2$. Solving this system of equations yields,

$$b = -\frac{1}{2}(m \pm \sqrt{m^2 + 4\nu}),$$
$$a = \frac{\nu}{b^2}.$$

One can check that provided with a mode and variance which are both non-zero, the negative branch is the solution which provides $a, b > 0$ as is required for the Gamma distribution. Hence, the transformations are,

$$b = -\frac{1}{2}(m - \sqrt{m^2 + 4\nu}),$$
$$a = \frac{\nu}{b^2}.$$

The parameters for the hierarchical model are,

$$\boldsymbol{\theta}_i = \left(\eta_{R_0}^{(i)}, q^{(i)}, \frac{1}{\sigma}^{(i)}, \frac{1}{\gamma}^{(i)}, p_1^{(i)}, p_2^{(i)}, \delta^{(i)}, \tau^{(i)}\right), \text{ for } i = 1, 3, 4,$$
$$\boldsymbol{\theta}_i = \left(\eta_{R_0}^{(i)}, q^{(i)}, \frac{1}{\sigma}^{(i)}, \frac{1}{\gamma}^{(i)}, p_{1,0}^{(i)}, p_{2,0}^{(i)}, p_{1,1}^{(i)}, p_{2,1}^{(i)}, \delta^{(i)}, \tau^{(i)}\right), \text{ for } i = 2.$$

and the hyper-parameters are,

$$\boldsymbol{\phi} = (\mu_{R_0}, \nu_{R_0}, m_\sigma, \nu_\sigma, m_\gamma, \nu_\gamma).$$

The hierarchical structure is shown in Figure 5.10.

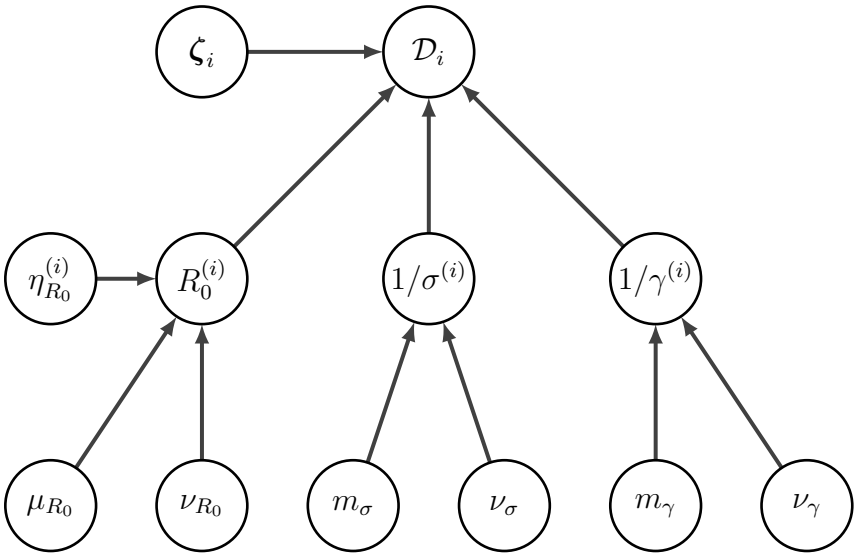

Figure 5.10: The hierarchical structure across the outbreaks. This shows the structure for a given outbreak but the general trend can be extended by noting the presence of i indicates outbreak i. We denote the outbreak specific quantities without any dependency on the hierarchical structure by $\boldsymbol{\zeta}_i$.

Our aim is to infer the full posterior distribution of $(\boldsymbol{\theta}, \boldsymbol{\phi})$ given the four outbreaks of Ebola. In order to sample from the posterior we use a pmMH procedure with a Gaussian proposal distribution. To do this effectively we need to consider how we can evaluate the unnormalised log-posterior density at a given value. The full log-posterior distribution can be written as,

$$\log p(\boldsymbol{\theta}, \boldsymbol{\phi}|\mathcal{D}) \propto \log \Big(L(\boldsymbol{\theta}, \boldsymbol{\phi}) p(\boldsymbol{\theta}, \boldsymbol{\phi}) \Big).$$

The prior distribution can be factored as follows,

$$p(\boldsymbol{\theta}, \boldsymbol{\phi}) = p(\boldsymbol{\phi}) p(\boldsymbol{\theta}|\boldsymbol{\phi}),$$

and as the hyper parameters are shared across the outbreaks this can be factored further, as conditional on the hyper-parameters, the parameters for each outbreak are independent,

$$p(\boldsymbol{\theta}, \boldsymbol{\phi}) = p(\boldsymbol{\phi}) \prod_{i=1}^{4} p(\boldsymbol{\theta}_i|\boldsymbol{\phi}).$$

The outbreaks we consider here are assumed independent from one another and hence we can write the likelihood function as the following,

$$L(\boldsymbol{\theta}, \boldsymbol{\phi}) = \prod_{i=1}^{4} L_i(\boldsymbol{\theta}_i, \boldsymbol{\phi}),$$

where $L_i(\cdot)$ denotes the likelihood function for outbreak i. Note the dependency on the hyper parameters in the likelihood, this is to reconstruct the epidemiological parameters through the deterministic relationship in Eq. (5.3). Using these results the unnormalised log-posterior distribution is,

$$\begin{aligned}
\log p(\boldsymbol{\theta}, \boldsymbol{\phi}) &\propto \log \Big(p(\boldsymbol{\phi}) p(\boldsymbol{\theta}|\boldsymbol{\phi}) L(\boldsymbol{\theta}, \boldsymbol{\phi}) \Big) \\
&= \log p(\boldsymbol{\phi}) + \log p(\boldsymbol{\theta}|\boldsymbol{\phi}) + \log L(\boldsymbol{\theta}, \boldsymbol{\phi}) \\
&= \log p(\boldsymbol{\phi}) + \log \prod_{i=1}^{4} p(\boldsymbol{\theta}_i|\boldsymbol{\phi}) + \log \prod_{i=1}^{4} L_i(\boldsymbol{\theta}_i, \boldsymbol{\phi}) \\
&= \underbrace{\log p(\boldsymbol{\phi}) + \sum_{i=1}^{4} \log p(\boldsymbol{\theta}_i|\boldsymbol{\phi})}_{\text{prior contribution}} + \underbrace{\sum_{i=1}^{4} \log L_i(\boldsymbol{\theta}_i, \boldsymbol{\phi})}_{\text{likelihood contribution}} .
\end{aligned}$$

In order to estimate the log-posterior density at a given point we simply evaluate the prior and hyper prior densities and then run the particle filters for each outbreak to obtain unbiased estimates of the log-likelihood (per outbreak) and sum these to estimate the total log-likelihood. In order to run the particle filters we must choose the number of particles for each simulation. Before identifying the number of particles to use, we can consider the use of parallelism to reduce the variance of the likelihood estimates. The first approach to parallelism relies on calculating estimates of the total likelihood M_c times, which is the number of available cores, and averaging the results,

$$\hat{L}(\boldsymbol{\theta}, \boldsymbol{\phi}) = \frac{1}{M_c} \sum_{j=1}^{M_c} \hat{L}^{(j)}(\boldsymbol{\theta}, \boldsymbol{\phi}),$$

where $\hat{L}^{(j)}(\boldsymbol{\theta}, \boldsymbol{\phi})$ is an estimate of the total likelihood which is obtained by running each particle filter once. An alternative approach is to instead calculate this total estimate as,

$$\hat{L}(\boldsymbol{\theta}, \boldsymbol{\phi}) = \prod_{i=1}^{4} \hat{L}_i(\boldsymbol{\theta}_i, \boldsymbol{\phi}),$$

where the likelihood estimates for each outbreak are,

$$\hat{L}_i(\boldsymbol{\theta}_i, \boldsymbol{\phi}) = \frac{1}{M_c} \sum_{j=1}^{M_c} \hat{L}_i^{(j)}(\boldsymbol{\theta}_i, \boldsymbol{\phi}).$$

This can be done as each outbreak is independent and so an estimate of the likelihood can be produced per outbreak and the variance reduced on a per outbreak basis.

For an efficient running of the pmMH routine we still require that the variance in the log-likelihood estimate be less than 3. There is an issue here as the particle filter used for each outbreak requires a different number of particles. Let $\hat{L}(\cdot)$ denote the estimate of the total likelihood contribution and let $\hat{L}_i(\cdot)$ denote the estimate of the likelihood from the ith particle filter. As the outbreaks are independent, we have the following,

$$\text{Var}\left(\log\left(\hat{L}\left(\boldsymbol{\theta},\boldsymbol{\phi}\right)\right)\right) = \sum_{i=1}^{4} \text{Var}\left(\log\left(\hat{L}_i(\boldsymbol{\theta}_i,\boldsymbol{\phi})\right)\right).$$

Hence in order to reduce the variance of the full log-likelihood estimate, we need to ensure that the sum of the variance of the individual likelihood estimates is small. This can be loosely ascertained by using the independent inferences. We find that at the MAP estimate for each of the datasets when fitted individually that we can use 300 particles for the Yambuku, Mweka and Boende datasets but that due to the length of the time series for the Kikwit dataset we need slightly more and so use 400. We also choose to run the particle filters sequentially, on 16 cores to further reduce the likelihood estimate. The parallelism and running of the particle filters in this sequential manner, results in lower variance estimates in the likelihood obtained through each particle filter, which leads to a lower variance in the total likelihood estimate. This approach also ensures that the variance in a single filter cannot dominate the total variance and lead to poor mixing of the overall Markov chain.

Prior distributions for non-shared parameters are assigned in the same manner as per the independent outbreaks (and are featured later in this section in Table 5.4). For the shared parameters we assign hyper-prior distributions on the parameters of the prior distributions to capture the variability between outbreaks. In order to test our assumptions we can perform a prior predictive check. A prior predictive check relies on sampling hyper-parameters from the hyper priors and using this sample to construct what is effectively the prior distributions. Namely, for a parameter ζ which may depend on a parameter ω then, the target distribution is $p\left(\zeta\right)$,

$$p\left(\zeta\right) = \int p\left(\zeta|\omega\right) p\left(\omega\right) d\omega.$$

We can then simply draw a sample of hyper-parameters $\omega_i \sim p\left(\omega\right)$ for $i = 1, \ldots, n$ and then a Monte Carlo estimate of the prior density at a given ζ is,

$$\hat{p}\left(\zeta\right) = \frac{1}{n}\sum_{i=1}^{N} p\left(\zeta|\omega_i\right).$$

To estimate the density over the support of ζ we can define a grid and evaluate this density for the sampled hyper-parameters ω by noting that $p\left(\zeta|\omega_i\right)$ is a known density.

We can use the independent inferences from Section 5.3 to determine how to set the hyper-priors. We expect more variability in the R_0 values for each outbreak as these are dependent on specific transmission structures. For this reason we can centre the mean at the average of R_0 from previous studies [14, 18, 45, 51],

$$\mu_{R_0} \sim N(2.1, 0.5).$$

We can then assign a hyper-prior which allows for relatively large variation between outbreaks on ν_{R_0},

$$\nu_{R_0} \sim \text{Half-Normal}(0, 1).$$

For the hyper-priors on m_σ and ν_σ we want to allow for less variability between outbreaks and focus on inferring a value of the true parameters. The prior predictive check is much more useful here. We want priors which use the information from the independent studies and leverages the extremes from that. We begin by fitting hyper-priors to the latent period. The minimum average latent period from the independent inferences was 5.65 days and the maximum was 6.98 days. Choosing hyper priors of,

$$\begin{aligned} m_\sigma &\sim N\left(6.3, 1\right) \\ \nu_\sigma &\sim \text{Half-Normal}\left(0, 0.1\right) \end{aligned}$$

induces a prior which has high density where these minimum and maximum periods are. Forcing a tight prior on the variance is useful as this ensures there should be minimal variability between outbreaks. This leaves the mode as the definitive parameter we are trying to infer.

We follow this same process for the hyper-priors m_γ and ν_γ as these values are also likely to be less varied between outbreaks. The minimum average infectious period from the independent inferences was 8.06 days and the maximum was 10.09 days. Hyper priors of

$$\begin{aligned} m_\gamma &\sim N\left(9, 0.4\right) \\ \nu_\gamma &\sim \text{Half-Normal}\left(0, 0.1\right) \end{aligned}$$

induce priors which again feature these two values in the region of high density. Note that for each of the modes m_σ and m_γ, we truncate the distributions at 1 day. This is done so as to enforce feasible latent and infectious periods and also ensures that we do not run into any issues with the particle filters as the rates are not too large [8].

Table 5.3 shows the relationship between the hyper-parameters and the shared parameters. This table includes priors on each of the parameters and shows how the outbreak specific quantities which are shared are related. Table 5.4 shows the outbreak specific

parameters and the priors on these. A superscript (i) across the two tables indicates a parameter for outbreak i. The Prior/Definition column expresses the prior (or hyper-prior in the case of the hyper-parameters) distributions where appropriate. The definition for the transformed hyper-parameters are provided for the shape and scale parameters.

Supports are indicated for each parameter where not inherently specified by the prior (such as for the Gamma distribution needing shape and scale greater than 0). All midpoint dates, τ and reduction parameters δ are outbreak specific and have there priors centred on the date of known interventions as this is likely to promote individuals to change their contact behaviour. The large variances allow for the midpoint to occur earlier as it is likely that individuals expressed some level of caution prior to obvious interventions. The proportions correspond loosely to the probability of observing an onset or removal each day. This can be empirically calculated by estimating the proportion of cases with known dates out of the total number of cases.

Parameter	Description	Prior / Definition	Support
$R_0^{(i)}$	Basic reproductive number of outbreak i	$\mu_{R_0} + \nu_{R_0}\eta^{(i)}$	$(0.1, 10)$
$\eta_{R_0}^{(i)}$	Relative deviation of outbreak i from baseline R_0	$N(0, 1)$	$(-5, 5)$
μ_{R_0}	Average R_0 across outbreaks	$N(2.1, 0.5)$	$(0.1, 10)$
ν_{R_0}	Between outbreak variation in R_0	Half-Normal$(0, 1)$	$(0.01, 10)$
$1/\sigma^{(i)}$	Latent period in outbreak i	Gamma(a_σ, b_σ)	$(0.5, 21)$
b_σ	Scale parameter for the prior of $1/\sigma$	$-(m_\sigma - \sqrt{m_\sigma^2 + 4\nu_\sigma})/2$	-
a_σ	Shape parameter for the prior of $1/\sigma$	ν_σ / b_σ^2	-
m_σ	Mode of $1/\sigma$ between outbreaks	$N(6, 0.5)$	$(0.5, 21)$
ν_σ	Between outbreak variation in $1/\sigma$	Half-Normal$(0, 1)$	$(0.01, 10)$
$1/\gamma^{(i)}$	Latent period in outbreak i	Gamma(a_γ, b_γ)	$(0.5, 21)$
b_γ	Scale parameter for the prior of $1/\gamma$	$-(m_\gamma - \sqrt{m_\gamma^2 + 4\nu_\gamma})/2$	-
a_γ	Shape parameter for the prior of $1/\gamma$	ν_γ / b_γ^2	-
m_γ	Mode of $1/\gamma$ between outbreaks	$N(9, 0.5)$	$(0.5, 21)$
ν_γ	Between outbreak variation in $1/\gamma$	Half-Normal$(0, 1)$	$(0.01, 10)$

Table 5.3: Parameters which rely on the hierarchical structure. The transformations and related variables are also shown. Note that Normal distributions are truncated where appropriate.

Parameter	Description	Prior / Definition
$q^{(i)}$	Shape of change of contact behaviour	Gamma$(2, 0.1)$
$p_1^{(i)}$	Proportion of onsets observed for outbreak i	Beta$(7, 2)$
$p_2^{(i)}$	Proportion of removals observed for outbreak i	Beta$(4, 2)$
$p_{1,0}$	Midpoint date for the change of person-to-person contact behaviour for Kikwit outbreak	Beta$(2, 20)$
$p_{2,0}$	Midpoint date for the change of person-to-person contact behaviour for Kikwit outbreak	Beta$(2, 20)$
$p_{1,1}$	Midpoint date for the change of person-to-person contact behaviour for Kikwit outbreak	Beta$(7, 3)$
$p_{2,1}$	Midpoint date for the change of person-to-person contact behaviour for Kikwit outbreak	Beta$(7, 3)$
$\tau^{(1)}$	Midpoint date for the change of person-to-person contact behaviour for Yambuku outbreak	$U(1, 72)$
$\delta^{(1)}$	Reduction of the person-to-person transmission rate following change of contact behaviour for Yambuku outbreak	$U(0, 1)$
$\tau^{(2)}$	Midpoint date for the change of person-to-person contact behaviour for Kikwit outbreak	$U(1, 191)$
$\delta^{(2)}$	Reduction of the person-to-person transmission rate following change of contact behaviour for Kikwit outbreak	$U(0, 1)$
$\tau^{(3)}$	Midpoint date for the change of person-to-person contact behaviour for Mweka outbreak	$U(1, 44)$
$\delta^{(3)}$	Reduction of the person-to-person transmission rate following change of contact behaviour for Mweka outbreak	$U(0, 1)$
$\tau^{(4)}$	Midpoint date for the change of person-to-person contact behaviour for Boende outbreak	$U(1, 72)$
$\delta^{(4)}$	Reduction of the person-to-person transmission rate following change of contact behaviour for Boende outbreak	$U(0, 1)$

Table 5.4: Outbreak specific parameters for the hierarchical model. Note that Normal distributions are truncated where appropriate.

5.5 Results

In this section we show the results from fitting a Hierarchical model to the four outbreaks of Ebola. We run a pmMH targeting the 40-dimensional posterior distribution. The filters are run sequentially as mentioned in Section 5.4. We run each filter in parallel on 16 cores using 300 particles for the Yambuku, Mweka and Boende outbreaks and 400 for the Kikwit outbreak. This number of particles appeared to provide estimates of the

total log-likelihood within the tolerances specified in Section 3.2. The pmMH routine was run for two days and the resultant minimum ESS was above 700 with more than half the parameters having ESS over 1000. Marginal posterior distributions are presented in Figures 5.11–5.14. Note that in this section we drop the superscripts on each of the parameters (i) for clarity.

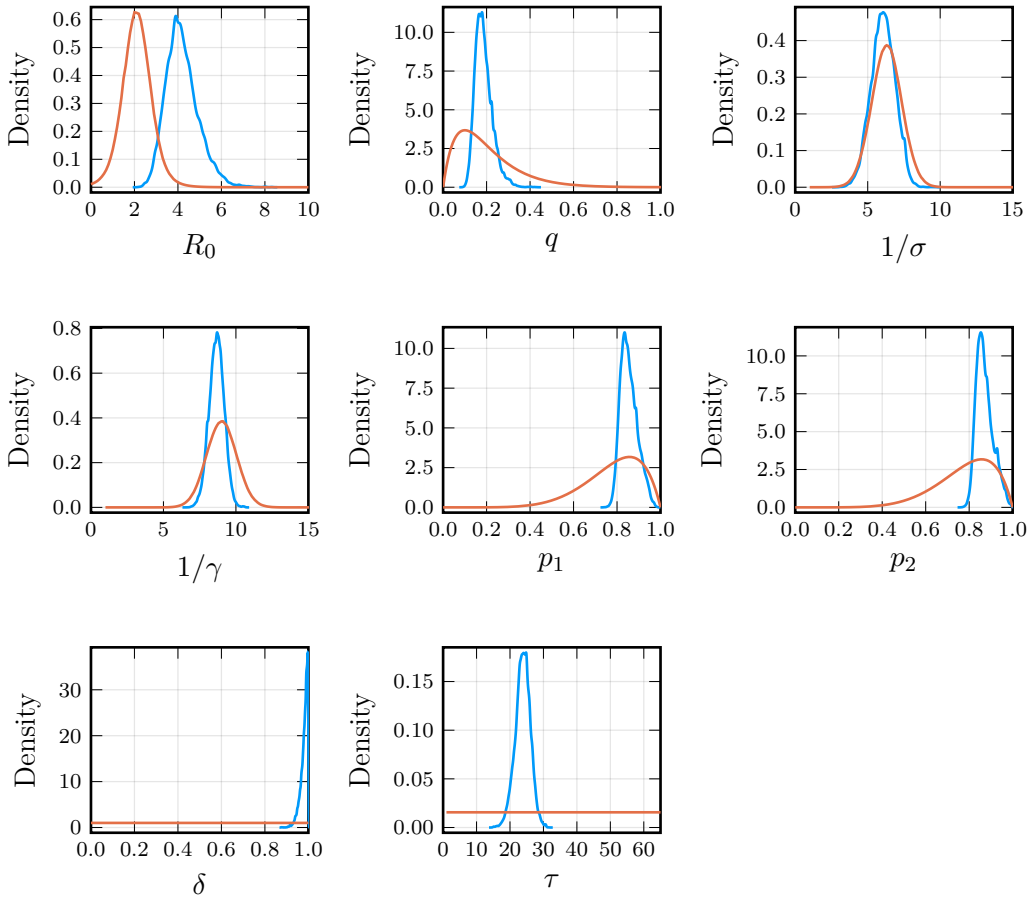

Figure 5.11: Marginal posterior distributions (blue) and priors (red) for the parameters of the (hierarchical) model fitted to the Yambuku outbreak.

In Figures 5.13 and 5.14 we see that the posteriors for $1/\sigma$ and $1/\gamma$ are dominated by the prior distributions. This demonstrates the advantage of the hierarchical structure. In Figures 5.11 and 5.12 we see that the posteriors for $1/\sigma$ and $1/\gamma$ are shifted from the induced prior. The result of this is that the information in the Yambuku and Kikwit outbreaks allows for improved inference of these terms in the Mweka and Boende outbreaks where there is less information. This lack of information is likely a result of the number of missing removal times in each of the latter datasets.

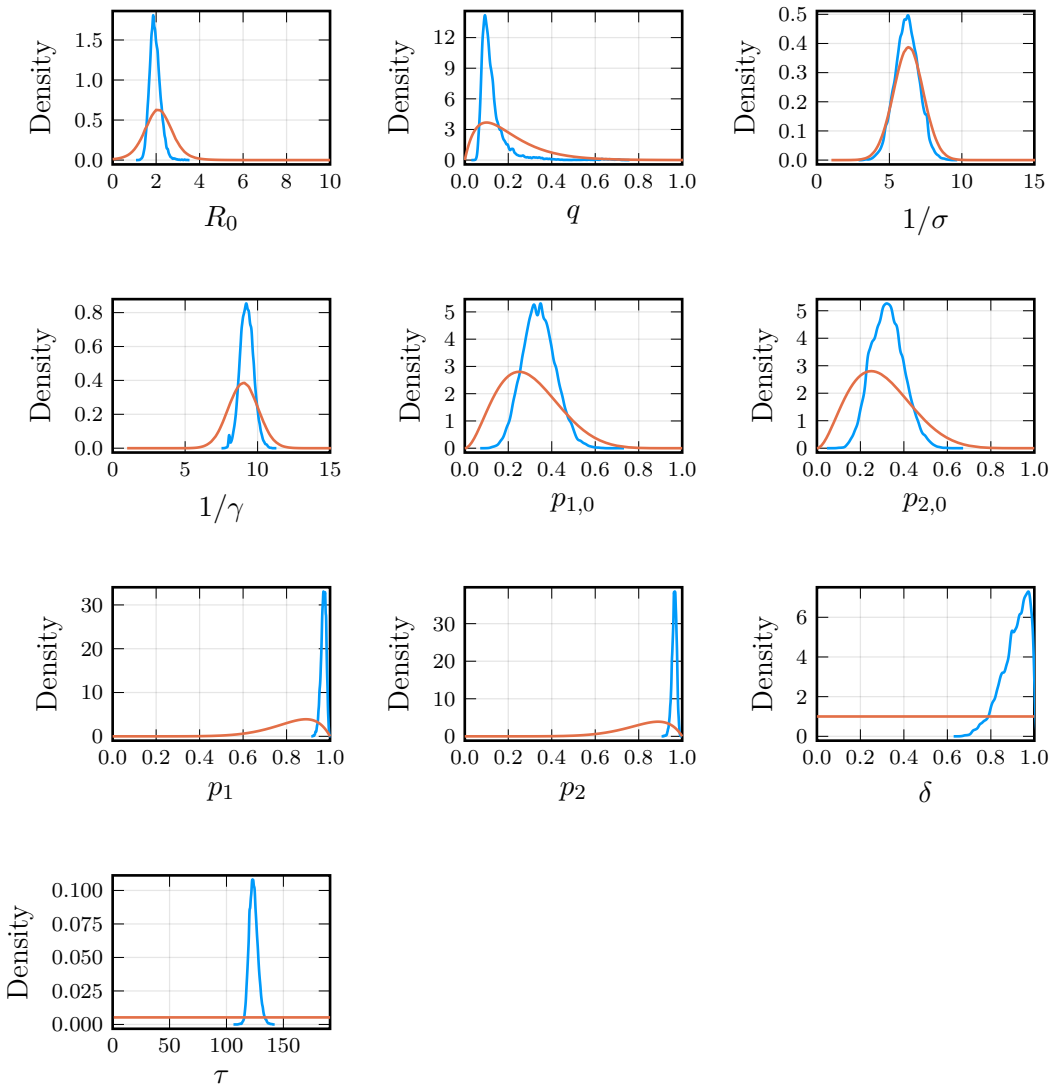

Figure 5.12: Marginal posterior distributions (blue) and priors (red) for the parameters of the (hierarchical) model fitted to the Kikwit outbreak.

The model appears to be able to estimate the midpoint $\tau^{(i)}$ in the change of mixing behaviour in the communities quite well for each outbreak. The posteriors for the reduction parameters $\delta^{(i)}$ lie close to 1 and show that all changes were quite strong in reducing transmission. This is known to be the case for Ebola in that it is primarily transmitted through secretions and so as community awareness increased, individuals were most likely distancing themselves. We summarise the posterior distributions through posterior means and 95% HPD for each parameter shown in Table 5.5.

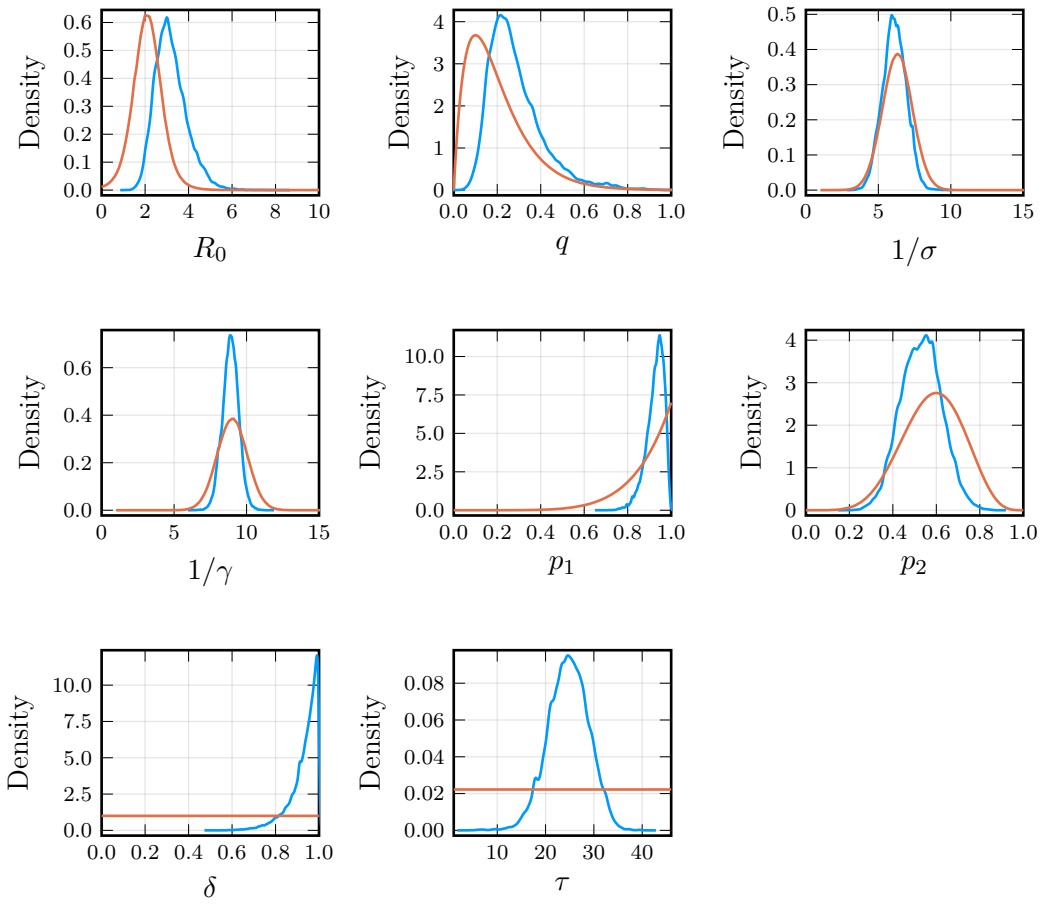

Figure 5.13: Marginal posterior distributions (blue) and priors (red) for the parameters of the (hierarchical) model fitted to the Mweka outbreak.

From the hierarchical fit we find that the R_0 values for each outbreak are above 1. For the Yambuku, Mweka and Boende outbreaks this initial value is above 3 and is again the largest for the Yambuku outbreak. This is consistent with previous studies [14, 73]. The average latent and infectious periods are largest for the Kikwit outbreak which is consistent with the independent study.

We find all midpoints in the change in transmission occur before the known major interventions. This supports the concept that individuals began changing their mixing behaviour prior to implementation of these measures.

An advantage of the hierarchical approach is the pooling effect. We can assess the influence of this by looking at boxplots of the marginal posteriors of $R_0, 1/\sigma$ and $1/\gamma$ across the outbreaks between the independent approach and the hierarchical approach.

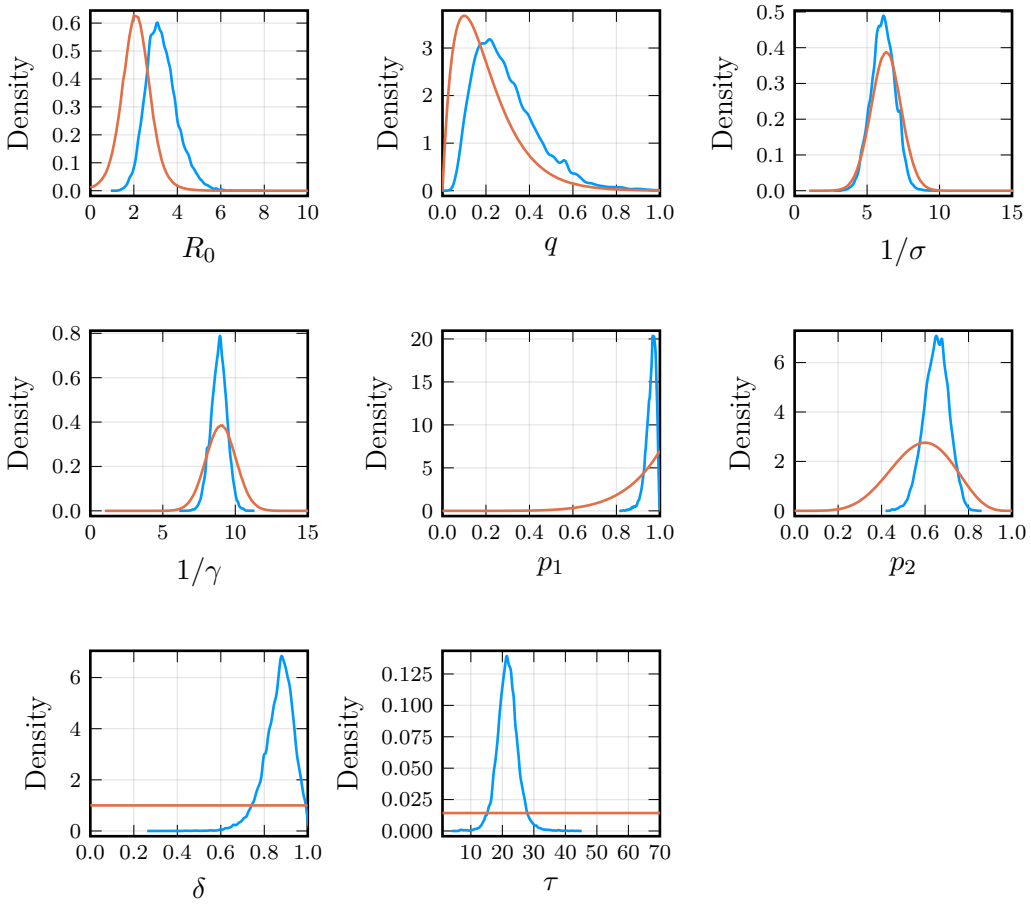

Figure 5.14: Marginal posterior distributions (blue) and priors (red) for the parameters of the (hierarchical) model fitted to the Boende outbreak.

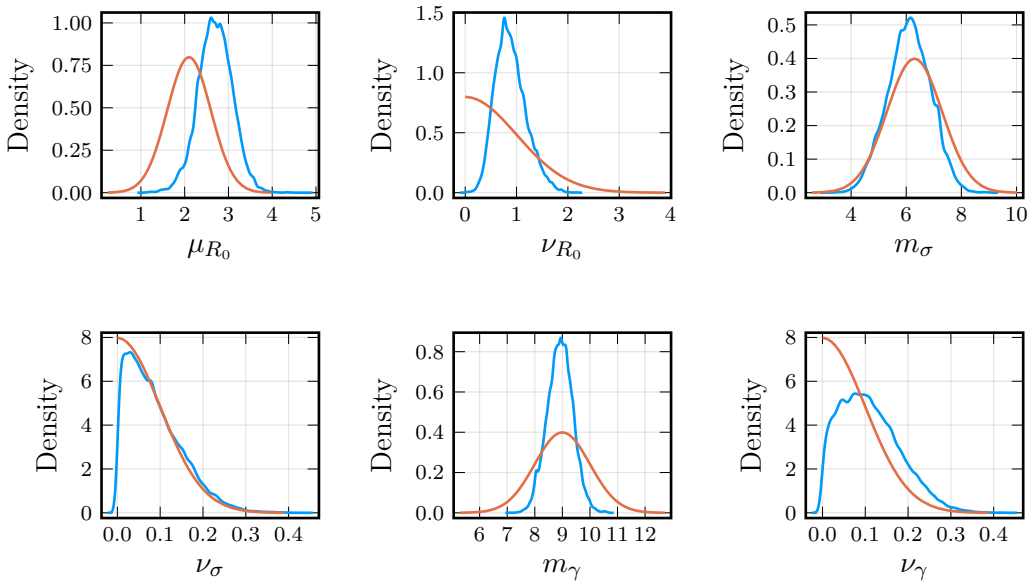

Figure 5.15: Marginal posterior distributions (blue) and hyper-priors (red) for the hyper-parameters of the hierarchical model.

Parameter	Yambuku	Kikwit	Mweka	Boende
R_0	4.19 (2.90, 5.77)	1.95 (1.50, 2.44)	3.19 (1.90, 4.68)	3.28 (2.03, 4.81)
q	0.19 (0.11, 0.26)	0.13 (0.06, 0.25)	0.29 (0.09, 0.55)	0.30 (0.06, 0.58)
$1/\sigma$	6.06 (4.47, 7.63)	6.21 (4.59, 7.77)	6.11 (4.56, 7.67)	6.11 (4.54, 7.68)
$1/\gamma$	8.67 (7.65, 9.67)	9.26 (8.35, 10.18)	8.89 (7.77, 9.97)	8.87 (7.76, 9.91)
p_1	0.85 (0.79, 0.94)	–	0.93 (0.85, 0.99)	0.96 (0.92, 1.00)
p_2	0.87 (0.81, 0.95)	–	0.52 (0.35, 0.71)	0.65 (0.54, 0.76)
$p_{1,0}$	–	0.34 (0.20, 0.49)	–	–
$p_{2,0}$	–	0.32 (0.18, 0.47)	–	–
$p_{1,1}$	–	0.97 (0.95, 0.99)	–	–
$p_{2,1}$	–	0.96 (0.94, 0.98)	–	–
δ	0.98 (0.95, 1.00)	0.91 (0.79, 1.00)	0.93 (0.79, 1.00)	0.87 (0.74, 1.00)
τ	23.82 (19.30, 28.19)	123.77 (116.61, 131.23)	24.70 (16.76, 32.88)	21.64 (14.67, 27.94)

Table 5.5: Posterior means and 95% HPD (in parentheses) of the parameters for the hierarchical model fitted to the four outbreaks of Zaire ebolavirus.

Parameter	Mean (95% HPD)
μ_{R_0}	2.70 (1.92, 3.46)
ν_{R_0}	0.89 (0.35, 1.50)
m_σ	6.11 (4.56, 7.58)
ν_σ	0.09 (0.001, 0.209)
m_γ	8.92 (8.02, 9.83)
ν_γ	0.11 (0.0004, 0.24)

Table 5.6: Posterior means and 95% HPD (in parentheses) of the hyper-parameters for the hierarchical model fitted to the four outbreaks of Zaire ebolavirus.

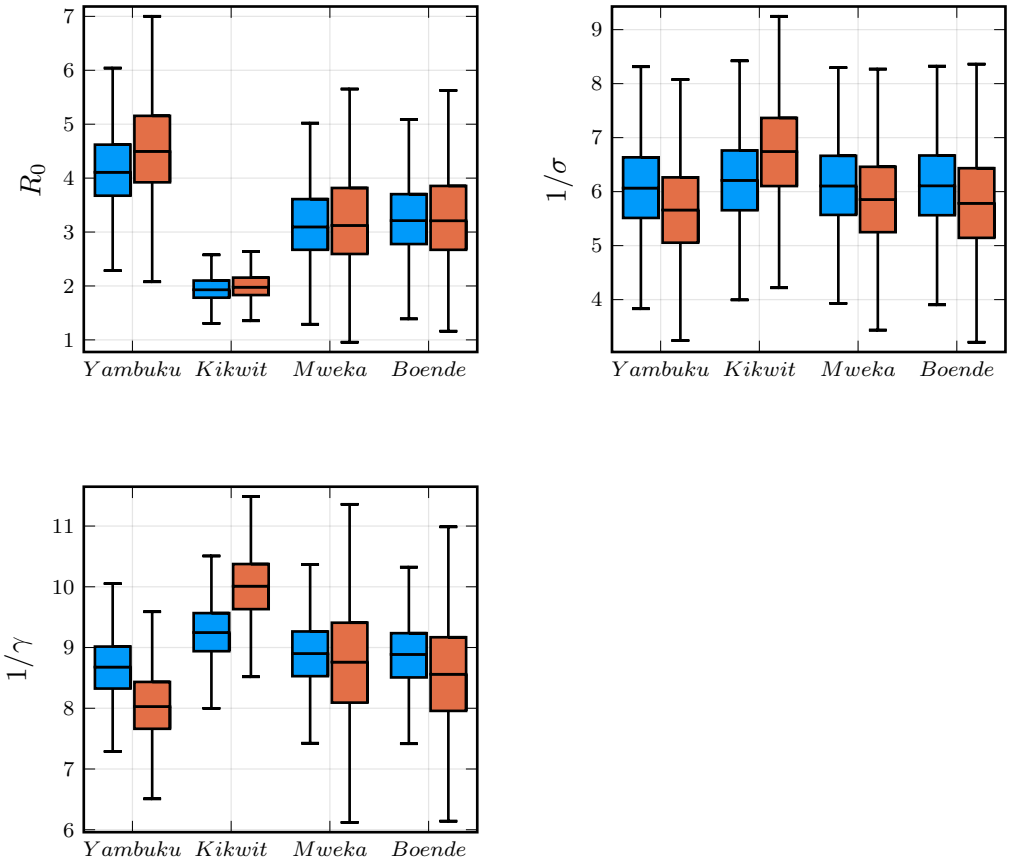

Figure 5.16: Marginal posterior distributions for the shared parameters. The blue boxes represent the marginal posterior distributions from the hierarchical approach and the red boxes represent marginal posterior distributions from the independent inferences.

In Figure 5.16 as we have allowed relatively large between-outbreak variation in R_0, this allows the marginal distributions of R_0 to roughly match the distributions from the independent inferences. The slight change to R_0 in the hierarchical model is a result of the assumption of some common mixing behaviour in different provinces of the DRC. The power of the hierarchical approach is more obvious in the boxplots for $1/\sigma$ and $1/\gamma$. The marginal distributions of $1/\sigma$ and $1/\gamma$ (shown in blue) are much more similar between outbreaks. This is a result of the pooling process where the information contained in the more informative datasets is dominating the shape of the posterior for the shared parameters. We see that the hierarchical model pools the estimates to the average of the independent studies. We estimated the posterior mode of the latent period to be 6.1 days which is slightly less than the initially assumed value of 6.3 days taken as the average across the independent outbreaks. This is consistent with the average value reported across previous studies [73]. The average infectious period across outbreaks is estimated to be 8.9 days and this is consistent with what is known about infectious period of Ebola [14, 73].

In Figure 5.15 we see that there is strong information in the data to refine the mode for $1/\gamma$. The distribution becomes less spread about the 9 day period. The same level of strength cannot be seen in the mode for $1/\sigma$. The hyper-prior density appears to dominate this term. This is rather unsurprising if we consider the independent inferences. In Figures 5.6, 5.8 and 5.9 we see that the prior dominates the marginal posterior density of $1/\sigma$. Only the Kikwit data contains enough information for the likelihood to influence the $1/\sigma$ distribution. Fortunately this is the advantage of a Bayesian framework whereby the induced prior assigned to $1/\sigma$ captures the results we obtained in the independent study. As the Kikwit model dominated that density, it is clear in the boxplot of the marginal posterior densities for $1/\sigma$ that the results for the other outbreaks are pulled towards the result of the Kikwit model.

In order to validate the fit of the hierarchical model, we can assess the posterior predictive distributions of the incidences for each outbreak. Figures 5.17–5.20 show the results of 10000 posterior simulations (per model). For the Yambuku outbreak we condition the simulations on seeing at least 150 detections. For the Kikwit outbreak, due to the low number of reports over the first 55 days we condition the simulations on matching the detected number of onsets and removals over the first 55 days where there was no surveillance and then an additional 10 days to account for the start of the second phase of the outbreak. The additional days help the simulations to match the evolution at the beginning of the outbreak. For the Mweka and Boende outbreaks we simply require a minimum of 10 and 30 detections, respectively, to be observed over the course of the outbreak. These conditions enable the posterior simulations to more accurately capture the evolution of the individual outbreaks by producing potential trajectories that are more consistent with the real epidemic data. We see in all cases that the average detections

for onsets and removals follows the trend of the outbreaks. The data lies within the 95% credible intervals in all cases and this suggests that the model is able to capture the key dynamics of each outbreak. There are some instances of the observed data lying outside the credible intervals (mainly in the Mweka and Boende posterior simulations) however this is likely due to not appropriately capturing the variance in the offspring distribution. A multi-type model which encorporated some probability of heightened transmission for an individual may more appropriately capture the days where there is larger incidence. Considering the simplicity of this model, the posterior simulations appropriately capture the true epidemic trajectory in the range of feasible trajectories.

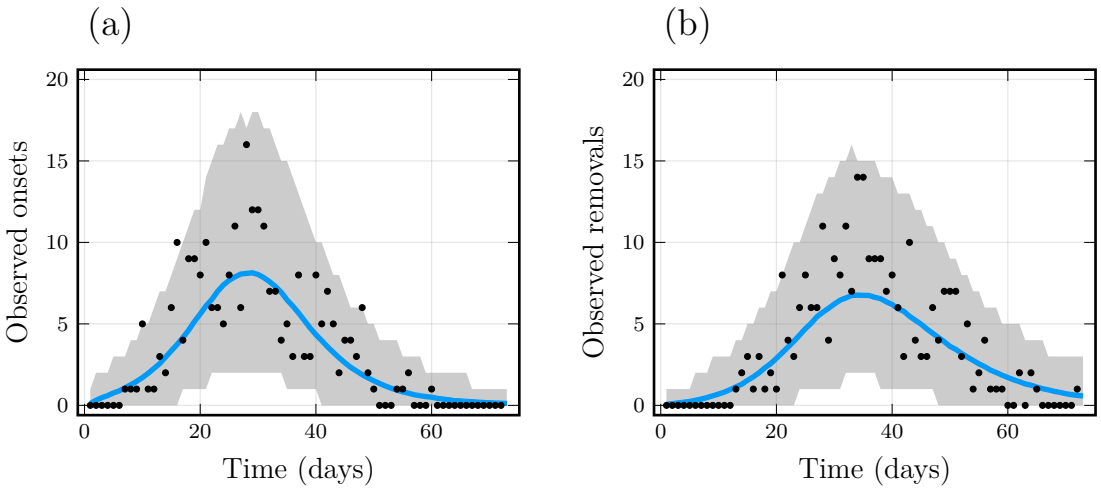

Figure 5.17: Yambuku data versus the posterior simulations. Results are shown for 10000 outbreaks simulated according to the SEIR model with partial detection, using the posterior sample: (a) the incidence of symptom onsets; (b) the incidence of removals. The black curve indicates the daily average incidence across the simulations. The grey area corresponds to the 95% credible interval of the simulations. The blue line indicates the detected events for the Yambuku outbreak.

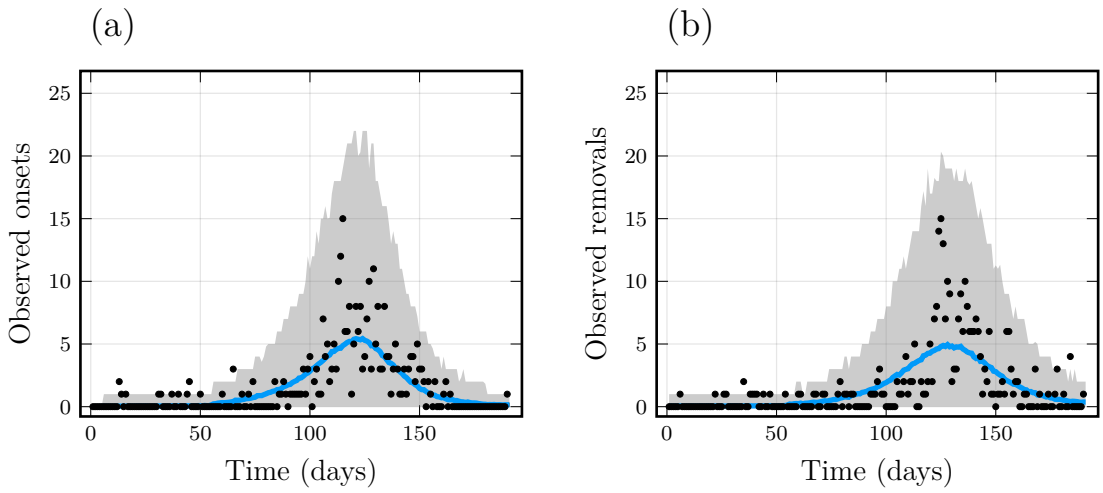

Figure 5.18: Kikwit data versus the posterior simulations. See caption of Figure 5.17 for more details.

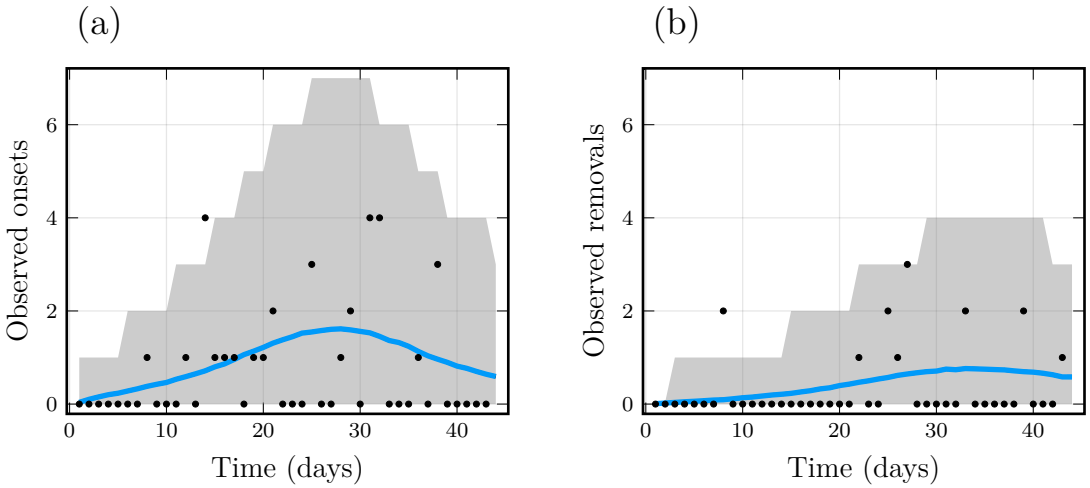

Figure 5.19: Mweka data versus the posterior simulations. See caption of Figure 5.17 for more details.

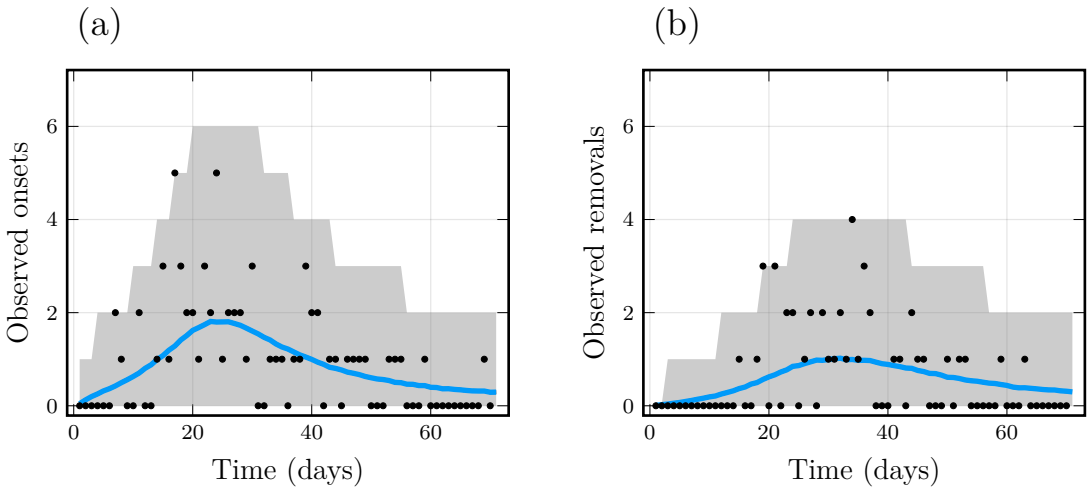

Figure 5.20: Boende data versus the posterior simulations. See caption of Figure 5.17 for more details.

5.6 Summary

In this chapter we have used the particle filtering methodology from Chapter 4 to fit a hierarchical model for inference on multiple outbreaks of the Zaire strain of ebolavirus. The hierarchical model enables for some common characteristics to be inferred across the outbreaks and enables us to capture this uncertainty.

We determined that in all cases the intervention measures curbed the remaining transmission but were not the cause of the reductions. We found that all four outbreaks had mean R_0 greater than 2 and that the mean estimates were all in the range found in previous studies 1.4–4.7.

The power of this approach is not completely evident as only two of the four datasets were strongly informative (Yambuku and Kikwit). While this does lead to change in the distributions of the hyper-parameters, more datasets would lead to improved parameter estimates. The average latent and infectious periods were identified to be 6.12 days and 8.84 days respectively. The analysis here provides a good foundation for the study of multiple Ebola outbreaks, however more work will be required to directly quantify the effectiveness of different intervention regimes.

Chapter 6

Discussion and further research

6.1 SPSA and MAP estimation

In Chapter 3 we have shown how the SPSA algorithm can be modified to be used to estimate the MAP of an epidemic model when an estimate of the likelihood is obtained through a particle filter. While it does require the selection of somewhat abstract parameters, the performance is directly controlled by the performance of the SMC algorithm used alongside it. In the SIR example we saw that parameter values for the search are relatively straightforward to choose. The learnings from that simple example were extended to be applied to the more complicated SEIAR example and the main differences are to consider the support of individual model parameters. Chapter 4 showed that the algorithm is able to be applied to more complicated examples which use real world datasets. We see across all these examples that the method is robust and able to obtain good estimates of the MAP at low computational cost.

SPSA has the advantage of being able to be run without prior knowledge of the properties of the posterior distribution. Not necessarily needing this prior knowledge means that to implement SPSA effectively relies only on a few points. SPSA requires only consideration of the support of the parameters and a way to estimate the likelihood. It should be noted however that knowing the properties of the posterior—such as the variance in certain parameters and their correlations—can improve the choice of the search parameters. This knowledge can be used for blocking conditions based on movement in the different parameters and to make informed choices of the number of particles to use.

The SPSA search can also be rerun with increased precision by adjusting the parameters $\boldsymbol{\delta}$ and $\boldsymbol{s}$ of the gain sequences. These parameters essentially control the magnitude of the update and perturbation steps respectively. We can initially use larger parameters in

the gain sequences to first converge in on the highest density region, and increasingly better estimates of the MAP can be found by reducing the size of the parameters in the gain sequences. This process of rerunning the algorithm is equivalent to increasing the number of iterations and as shown in Chapter 3 this increase in iterations leads to more precise estimates. The increased number of iterations alters the parameter B which features in Eq. (3.9) and is chosen to be equal to 10% or less than the total number of iterations. This parameter influences the finite sample performance of the algorithm which is how well the algorithm performs over a practical number of iterations [69]. For the analysis in this thesis we have opted primarily to focus on finite sample performance and as such, the searches were run once over a large but not excessive number of iterations. Furthermore, our analysis was primarily focused on demonstrating a strong level of convergence, regardless of starting point. In a realistic implementation, more care can be taken in choosing the parameters of the search and allowing more computational resources, in the form of larger numbers of particles and iterations. This approach tends to be more appropriate in specific implementations as we require only a single estimate of the MAP.

Finite sample performance is a good measure of how well an optimisation algorithm works in a practical setting. For our problems, we care about quickly obtaining an estimate which we can then use to tune the particle filter and then begin our inference procedure. Spall [69] demonstrates the convergence of SPSA becomes more likely as the number of iterations increases but also makes mention of the importance of considering the practical application and that accuracy may need to be sacrificed for the benefit of efficient and accessible results. This latter point is particularly relevant when providing information towards aiding public health. Quick and reliable estimates of parameters like R_0 and the average infectious period can be extremely useful in identifying early intervention strategies. Often an estimate to within 1-2 decimal places will suffice. Furthermore, the performance of particle filters is not too sensitive to how precise the MAP estimate is [8, 22, 57, 65]. The main item of importance is the performance of the particle filters near the MAP so for that purpose the algorithm is sufficient.

The computational cost of the search is dominated by the runtime of the particle filter. As such, it would appear that in order to minimise the overall runtime, it would be beneficial to reduce the number of particles. While this approach can be considered for some implementations with certain datasets, for the majority of problems, this would not provide good estimates. For most problems it is ideal to increase the number of particles such that estimates of the log-likelihood at random points (near the starting point) have relatively low variance in the range of 0–10. The number of particles to use varies between problems but typically we want to know that once the search is not being dominated by the variance in the log-likelihood estimates. This requires some additional understanding of the problems at hand, potentially including some pilot runs of the SPSA search from various starting points. That being said, this will still prove to be simpler than other

algorithms.

Averaging of multiple estimates of the function value and gradients can be useful in producing better estimates within the search [69]. This averaging approach more accurately represents the shape of the posterior density and hence the algorithm explores the parameter space more carefully. The key issue with this idea is that with our problems we are primarily interested in obtaining an estimate of the MAP at minimal computational expense. Using multiple function evaluations requires multiple runs of the particle filter and the runtime of the search grows linearly with the number of particle filter runs. Parallelism could be used here to estimate multiple gradients simultaneously, which could then be averaged over. That being said, ensuring that the gradient estimates are appropriately averaged would likely prove difficult as the noise could result in averaging multiple, poor estimates. This would prove to be most problematic in regions of the posterior where the log-likelihood estimates have high variance. In contrast to this, the swarm style approach used in Chapter 3 takes advantage of parallelism to average across multiple independent searches from the same starting point. This stops the noise of any individual run of the particle filter dominating the gradient estimate.

An issue we have so far not discussed is that SPSA is a stochastic algorithm and AKDE relies on the stochastic process of pmMH (in this thesis) [11, 69]. The AKDE method has the potential to yield different estimates of the MAP as a result of the precision of the grid supplied as well as the mixing of the chains. The AKDE approach requires consideration of the coarseness of the grid at which to evaluate an estimate of the multi-variate kernel density. A more coarse grid requires a longer runtime and can even result in some large computation issues which arise due to storage. This mainly arises when the size of the parameter space is large and the grid is chosen to be too large. The point about the mixing of the chains is related to how well the parameter space is explored. If the chain supplied to the kernel density estimator is poorly mixed, estimates will inevitably be flawed. This typically requires multiple or long pmMH runs to obtain reliable estimates through this approach. In non-simulated problems it is not a trivial thing to develop efficient pmMH procedures without repeated runs and tuning. To obtain a good estimate through this approach, we need to go through the tuning process needed for efficient pmMH procedures.

MAP estimates obtained through the SPSA algorithm do not rely on the tuning of the pmMH procedure. This brings with it a improvement in performance relative to going through the tuning process of implementing an optimal pmMH routine. The SPSA algorithm also enables a more methodological tuning process whereby it is the first step in the tuning process. Initially the SPSA algorithm can be run to obtain an estimate of the MAP, which can then be used to tune the number of particles to use. Then one can focus on pilot runs of the pmMH procedure to tune the proposal distribution and hence more quickly obtain an efficient inference routine. This methodological process means

that we do not need to be tuning the particle filter and proposal distribution at the same time which inherently influence one another.

One point we have so far neglected is why we focused on this specific algorithm. Simulated annealing is another method for solving this type of problem but suffers from requiring the same tuning as MCMC methods while introducing some additional considerations [69]. The issue of variability in the likelihood estimates is also a contributing factor whereby the annealing process can get stuck at the non-global optimum [69]. While this is still an issue in SPSA, it typically requires much less runtime. This difference in runtime is primarily due to the probabilistic search through the parameter space in simulated annealing. This search depends on the choice of appropriate proposal distributions and temperature function [69]. The temperature function in particular is crucial in obtaining good estimates by allowing the search to not get stuck on local optimum. This is guaranteed by cooling the system at a sufficiently low rate which is equivalent to allowing a larger number of iterations in SPSA [69]. In our limited testing we saw that there was a large amount of complexity in tuning the simulated annealing algorithm when using a particle filter and that typically it would be more worthwhile to go through the steps for the AKDE approach–which we have already outlined the potential issues with. Algorithms like simulated annealing and that of the AKDE approach are effective when we are interested in refined estimates of the MAP but both can prove to be more time consuming to work with. This is where the SPSA method becomes more effective as the reduced runtimes ensure we can reliably provide estimates with a quick turn around.

In the SIR example we explained how the search can get stuck if we reuse the estimates of the minimum loss to compare against current loss estimates. We can instead evaluate the loss function a secondary time on each iteration but this leads to a doubling of the necessary particle filter runs at each step of the search. The SEIAR example uses this approach as it does tend to provide improved estimates as the search no longer gets stuck due to an underestimated value of the loss function. Although this is the case, the estimates for the SIR example show that this reevaluation is not always necessary. The reevaluation of the loss will be more important in problems where there is a much higher degree of noise in the log-likelihood estimates. This will be the case when we have unlikely events or high dimensional problems and hence more particles are needed to reduce the variance in the log-likelihood estimates. As a result of the SPSA algorithm itself being rather simple, typically we require around only a few minutes of runtime per search and so running multiple searches is not overly time consuming. This allows us to configure the search based on different problems in the ways we have outlined here and in Chapter 3.

Since particle filters are sequential, we can parallelise them in order to reduce the variance in the log-likelihood estimates in the search [8, 24]. This can be particularly useful in reducing the noise in the function estimates and lead to improved convergence. This form of parallelism needs to be applied with caution however as otherwise the search

may converge to a local minimum (rather than global) due to the lack of noise. An approach for solving this is to inject some noise into the update step and this enables the search to operate as a global optimisation algorithm [7, 50, 69].

In the SIR and SEIAR examples in Chapter 3 we sampled starting points from the prior such that there was some initial log-posterior density. It should also be the case that enough particles are used such that the variance in the function evaluations is not too large. What constitutes too large is problem dependent but typically a variance of 0–10 in the log-likelihood estimate at the starting point can be considered reasonable. A point we have so far ignored in how we obtained starting points is that we did not take into consideration known information beforehand. In real world applications we are likely to be able to have an informed guess of a decent starting point based on expert knowledge. This is something which can reduce the runtime of the search algorithm as typically priors will express this domain expertise surrounding likely parameters and should mean we start closer to the posterior.

Future research

The SPSA algorithm has been applied to an extensive set of problems in the literature [7, 34, 49]. However, the details of the search parameters and tuning processes are not well documented. Most problems appear to be solved by arbitrarily altering the algorithm to increase performance on the given problem [34]. Relatively minor adjustments have been made to the algorithm and while results reflect that the algorithm does appear to perform reasonably, there is room for improvement regarding the process of choosing parameters. The work in this thesis hoped to provide a simplified approach in the parameter choices for the search but there are a great deal of interactions between them. As such it is likely that similar results could be obtained for other combinations and so more sophisticated study is likely needed. Regardless, it should be the case that increasing the number of iterations should improve long term accuracy.

Another possible extension which would be worthwhile considering is to determine a more sophisticated stopping criterion. The current stopping condition is based on the number of iterations but it would be worth considering a methodology whereby if we have not improved the value of the loss estimate after some number of steps, then we stop the search. For the analysis here we have not gone into assessing stopping criteria as the runtimes we observed were low enough to merit using the iterations as the only criterion. However we note that it would be necessary to test the search on a higher dimensional problem with a suitable stopping condition. It is also simple to adjust the algorithm to operate on the number of function evaluations as opposed to a number of iterations. This might potentially enable more precise connections between the runtime of the algorithm

and number of function evaluations allowed.

An area of future work would be to assess the sensitivity in the variance of the log-likelihood estimates at the MAP found through the SPSA search. This could be done by measuring the variance at increasingly better estimates of the MAP and seeing how sensitive the variance estimates are. This would help in providing a more justified choice of the number of iterations and particles to use. We would likely find that the bounds on the variance are not very sensitive to the range we looked at here.

Another worthwhile extension would be to explore higher order versions of the SPSA method. These methods require approximations to the Hessian matrix and have been shown to account for constrained spaces in a more robust way [69]. The reason we chose to appeal to a simplified finite-difference scheme was that the main source of computational expense is the particle filters. Requiring a larger number of runs of a particle filter on any given iteration will drastically increase the runtime of the search. This is also why we appealed to using only a single perturbation at each time step as opposed to a symmetric perturbation. While this is a key issue, it could be the case that a higher-order technique could lead to a substantial reduction in the required number of iterations and could lead to a reduced total number of loss evaluations.

6.2 Particle filtering and Ebola

The proposed particle filtering approach to inference in Chapters 4 and 5 extends upon the idea of using importance sampling in a particle filter to produce realisations which are consistent with two distinct observed time series of the same process. The results show that the algorithm effectively enables inference to be carried out on real outbreaks of Ebola. Our results in Chapter 5 are largely consistent with the *data-augmented MCMC* (DA-MCMC) approach presented in McKinley *et al.* [51]. The advantage of this approach is that it marginalises over the missing data rather than integrating over it as per DA-MCMC. This allows for more efficient inference methods and allows the algorithm to be scaled for conducting inference on more than a single outbreak.

Previous studies of the outbreak in Kikwit relied on DA-MCMC [45, 51]. This infers the missing event times as well as the parameters [58, 56]. This is considered the gold standard approach for dealing with missing data but it does suffer from several prominent issues. DA-MCMC methods perform worse as the amount of missing data increases which is a result of issues with convergence and mixing of the chains [52, 58]. Our proposed method does not suffer from these same issues as we instead marginalise over the missing data. The weaknesses of DA-MCMC are not too prominent if there is only a small number of missing event times as the dimension of the parameter space and missing event times

will not be that much larger than the dimension of the parameter space. For a single outbreak, the DA-MCMC approach can work quite well, but the methodology does not scale well, particularly in instances where we attempt inference on multiple datasets. In Chapter 5 we fit a hierarchical model to multiple outbreaks of the Zaire Ebolavirus with missing event times. The dimension of the parameter space alone in that model is 40 and we also need to infer the entirely latent curves as well as all the missing onset and removal times for each outbreak. While our methodology may make some assumptions on the observation process, it enables efficient inference which makes fitting such high dimensional models possible.

The particle filter produces realisations which exactly match the observed state of the system. While this is promising, some potential issues arise over the modelling decisions to allow an extended version of the SEIR to be fitted to the Ebola data. One obvious concern is regarding the binomial observation process. There is no prior reason for this to be an appropriate modelling assumption but was one tested out of simplicity. While it does appear that this was a reasonable approach for this problem (in comparison to [51]) there is concern over its appropriateness for datasets with higher amounts of missing data. It is likely that a better approach would be to incorporate time-dependency into the observation probability.

Another issue with the modelling approach is that we cannot guarantee exactly seeing N_F infections. It is possible there were more infections than this–particularly in older outbreaks–as monitoring would not have been as strict as modern day standards [61]. Further to this point is that there is some evidence over further missed cases (up to 17%) based on serological studies conducted following the Yambuku outbreak [12]. It has been discussed whether these cases were asymptomatically infectious which could further influence the evolution of the outbreak [12]. This is a limitation of our model in that it incorporates only the most basic dynamics to fit the models using the two time series. This is a simple area of extension for the model, we can add compartments for asymptomatic infectiousness and even high or low risk individuals. This would enable more questions to be asked surrounding the dynamics of Ebola. Furthermore, accounting for additional cases could be included by relaxing the observed final size N_F and bounding the total number of undetected onsets and removals by $N - N_1$ and $N - N_2$, where N_1 and N_2 are the total number of detected events of the two observed events, respectively.

There is also a limitation in how the model proposed in Chapter 4 accounts for the missing data over the first phase of the outbreak. It is clear that many of the simulations end up in states with a higher number of infected cases (from the posterior simulations) than is feasible based on future observations and without an appropriate number of particles, this can lead to high variance estimates of the log-likelihood which drastically worsens mixing of the resultant chains. Unfortunately this is an issue with forward simulation techniques and tends to be exacerbated by longer time series [3]. This arises due

to an inability in using the states at later time points to improve the distributions over the states during earlier time steps. An improved approach could be to infer or limit the number of missing cases which occurred during the different phases of the outbreak. This would potentially offer performance improvements as we would not need as many particles to approximate the state of the process during the early phase. We could justify this modelling choice by acknowledging that it would be unlikely that there were a large number of cases over the missing reporting phase. If there were, then it would be almost guaranteed that authorities would be alerted to the outbreak earlier than they were. This suggests that cases were sporadic over this first phase and this appears to be the case based on the data in Rosello *et al.* [61] which recovers some of the missing cases.

Another consideration would be to extend the model used in Chapters 4 and 5 into a multi-type model. The SEIR model used could be extended into a mutli-type model which separates into two branches, depending on whether the infection is detected or not. This would use more information about particular individuals but would enable us to reflect the increased chances of detecting a death if we have detected the infection. This would extend the complexity in the particle filter even further as we would then need to ensure the forcing of events is feasible between the two branches of the model, however this would provide an interesting future area of study. Khan *et al.* [41] also provides data for whether the infected individual was a health care worker or not and this has recently been studied in [60]. This analysis is similar in construction to that of [14] and so would be easily extended to that study also. A multi-type model could be used for all these purposes and the importance sampling methodology could be used with these models, which would offer reduced computational expense in comparison to standard particle filtering approaches.

The algorithm could be extended to produce realisations consistent with more than two time series but there is issue in the quality of data when trying to do so. Historically only one to two parts of an epidemic could be observed to a high standard [4, 45]. This was why the algorithm was designed in terms of two observed event types. In light of the current state of the world, it is likely to be the case that future outbreaks may be surveilled to a higher level and the methods here can be extended for this case. In the models used in Chapters 4 and 5 we did not account for individuals to progress to different compartments through multiple routes. Models which incorporate more complex transitions and a greater number of compartments are likely to require much greater consideration in the forcing of events such that realisations are consistent with future states of the system.

As per the discussion in Black [8] the methods here are not black box. The modification of rates and the appropriate dealing with the modified process is something that will need to be assessed for each model. This requires a great deal of care to ensure that the simulations are consistent with not only the current observations but also future observations. The level of complexity with decisions like this can be coded into the

algorithms in sensible ways like that used in the filters in this thesis. In Chapters 4 and 5 we allowed fadeout once the total number of detections across both datasets had been met and the total number of infections had occurred. This ensures we still have a pool of individuals available for future events to occur. This approach can become quite complex as the number of compartments and possible transitions increases.

A key property of particle filters is the ability to run them in parallel [8, 24, 52]. This is an easy process in most programming languages and offers improvements in the runtime of inference procedures. This improvement is a by-product of lower variance log-likelihood estimates which then allow fewer particles to be used. Parallelism was easily applicable to the problems in Chapters 4 and 5 and led to much improved runtimes. This is something which cannot be done using DA-MCMC as the algorithm is serial in nature and hence we can see dramatic speed-ups in computation by relying on a particle filtering approach [8, 52, 58].

There was a slight sense of uncertainty over what data was used in previous studies in Chapter 4. It appears likely that the analyses in Chowell *et al.* [18] and Lekone and Finkenstädt [45] took March 1st as the initial day of the outbreak. This suggests that of the undetected cases, they must have strictly come from that secondary phase of the outbreak. This is not the approach taken in this thesis or in McKinley *et al.* [51]. Our approach fits to the outbreak beginning on January 6th, the date of the first suspected case and this introduces a greater deal of missingness over the almost two month period with no reports which needs to be accounted for. This leads to some greater uncertainty in some of the parameters compared to Chowell *et al.* [18] and Lekone and Finkenstädt [45]. While that is the case, results from our inference methods are consistent with McKinley *et al.* [51] and this suggests that the method is performing well.

Fitting the hierarchical model in Chapter 5 was facilitated due to the particle filtering methodology. While the results of the independent models and hierarchical model are consistent, the interesting results lie in the details. We used the results of the independent inferences to inform better prior distributions on the shared parameters in the hierarchical model. This leads to less movement in the marginal posterior distributions than we saw in the independent inferences. Essentially this means that the prior in the hierarchical model reflects the posterior density (for $1/\sigma$) found during the inference on the parameters of the Kikwit model. Similar results would be able to be obtained by using the information from the different inferences to inform better priors in the independent section but this is something which is directly accounted for in the hierarchical model.

Hierarchical models increase in power as more datasets are added in and you increase the strength of the pooling by reducing the allowable heterogeneity between the data. It is likely that the four datasets we used were not informative enough to take full advantage of the framework. It is clear that this approach still offered some benefit as results for each

outbreak were slightly different to independent inferences and it did enable some insight into the between outbreak parameters. We found that the mean latent period across these outbreaks is close to 6.1 days, which is consistent with previous studies and slightly different to what we initially assumed of 6.3 days [14, 19, 73]. The average infectious period is found to be around 8.8 days which is consistent with previously reported estimates [14, 73]. We were also able to capture the variability in R_0 between outbreaks reflecting what we found in the independent studies as well as being consistent with previous analyses [14, 19, 73].

The runtimes for the independent inferences in Chapter 5 were lower than the hierarchical model however they required choices of priors which may not reflect the true range of values in particular parameters. A hierarchical model alleviates this issue by accounting for this uncertainty through the hyper-parameters and hyper-priors. This can mean that choosing prior distributions can be simplified as the hierarchical structure is able to draw on the more informative datasets to refine the induced prior densities. This in turn aids in fitting less informative datasets while not assuming too much dependency on the chosen prior. It should be noted that it is possible to avoid fitting a hierarchical model by using the inferences from the more informative datasets as priors for the less informative data which is an advantage of a Bayesian framework. This could potentially lead to improved inferences but would require a great deal of care to be implemented appropriately. It is also likely that this approach would be much more time consuming and could lead to incorrect conclusions to be drawn. This is apparent if we consider the (independent) inference results for the Yambuku data where R_0 was estimated to be 4.6. If we centre a prior on this for the Mweka and Boende datasets it is possible that we may overestimate R_0 for these models. The hierarchical framework minimises issues like this as all datasets influence the induced prior distributions and those with more information dominate the posterior density.

Several assumptions made in the fitting of the models in Chapter 5 could be relaxed. It would be more appropriate to infer the initial conditions for all the outbreaks rather than assuming a condition out of simplicity. While the models appear to fit the observed data reasonably it seems likely that for the Mweka and Boende outbreaks, the value of R_0 may be slightly high as there is no evidence to suggest this should be much higher than the Kikwit outbreak. This could be down to the small number of cases over the course of the outbreak. It could also simply be due to the initial number of infected individuals which may have been slightly higher. A tighter prior on R_0 would assume that there is less variability between outbreaks and may prove to yield better inferences.

Assigning tighter priors on the variances in the hierarchical model leads to parameters which are more similar across the outbreaks. This approach is one we adopted and aimed to infer the mean (and modes) of the induced priors for $1/\sigma$ and $1/\gamma$. This assumes there is limited heterogeneity in these values between outbreaks. Relaxing this assumption leads

essentially to the independent model fits. It may be possible that there is a greater deal of heterogeneity in these holding times between the outbreaks and so it would be worth investigating the influences of priors. Another point regarding the priors is that a proper sensitivity analysis could be useful. There appears to be limited power in the model to infer the $1/\sigma$ in all outbreaks besides Kikwit. We have assessed the use of a non-centred prior on the modes m_σ and m_γ and found that the model inferred the infectious period to be the same. The latent period was also inferred to be close to the values reported in Chapter 5 but there was an obvious influence of the hyper-prior and priors on the marginal posterior. More data and more complex dynamics are likely to increase the ability of the model to infer smaller changes to the latent period.

Future research

The particle filtering methodology developed in Chapter 4 could be extended for use on different datasets from other diseases. This method is something which is also applicable in other fields of ecology modelling such as the Lokta-Volterra model which often assumes there are observations of two population numbers. The methods here could also be extended to apply to the problem presented in Camacho *et al.* [14] where the source of infection is stratified by syringe or person to person. Such problems would require more study into more complex observation processes.

As mentioned in Black [8], certain models can be represented in different ways. The SEIR model in our work is one such model which could have been represented using binomial observation processes which would enable use of the bootstrap filter however the methods proposed here would still produce lower variance estimates through the use of importance sampling. An approach which could enable incorporation of these observation processes would be to potentially derive functional forms of the observation probabilities each day based on the suspected distributional form of the observation process. This could for example allow daily or weekly fluctuations in observation probabilities. An importance sampling based simulation routine could then produce simulations which are more likely to be consistent with this approach. Another potential alternative could be to invert the observation density and sample exact counts from this to use as matching criterion for the simulations. This is something which would require care so as to not be too computationally expensive as opposed to the simpler implementation of the bootstrap filter.

The particle filter here could be extended to be used for more complex models. This would allow for more complex dynamics to be accounted for, potentially leading to better inferences. This is something which would be especially beneficial for accounting for latent or infectious periods which are distributed according to Gamma distributions which tend

to be more realistic. The algorithm can be extended for models like this by ensuring that events are forced in the appropriate manner. This forcing is something which would need to be carried out with care.

The main limitation of the models in this thesis are that we have not directly accounted for some key dynamics of Ebola such as hospitalisation of individuals or traditional burial processes [14, 45, 61]. Accounting for the separation of individuals into a hospital setting (such as the models in Camacho *et al.* [14] and Legrand *et al.* [44]) would allow for different transmission rates and infectious periods to be applied. We could also separate the individuals in the population into low-risk (healthy) and high-risk (children, elderly) categories and seek to determine the transmissibility and susceptibility in these two groups. Furthermore, it may be the case that a further level of stratification according to age of stature in the community may account for the heterogeneity in the processes in a better way.

Extension to the models is not just limited to the individual inferences but also to the full hierarchical model. While the model developed through Chapter 5 has been shown to fit reasonably to the data, there are several key improvements which could be made. The first is the inclusion of more datasets, using other strains of Ebola. This could lead to increased inference capabilities and hence allow for more complex models to be fit. This increased number of datasets would offer more strength in inferring the hyper-parameters and could lead to improved insights about the average behaviour and variability across outbreaks. The inclusion of additional strains could be facilitated through adding additional parameters which indicate the strain. This would enable us to characterise potential differences in transmissibility, case fatality ratios and severity between strains, something which has not been studied in detail [73]. The framework we have developed here enables such models to be fit. That being said, the hierarchical model may not provide benefits over fitting to the different strains separately and may actually lead to more uncertain inferences.

The second main extension would be to consider more complex and unique models for each of the major outbreaks which more appropriately capture the dynamics of those specific outbreaks. This is particularly prominent issue in the Yambuku and Kikwit outbreaks. The Yambuku outbreak is known to have a distinctive and high contribution to transmission as a result of the sharing of syringes [14, 61]. This is something our model was not able to distinguish between as the infectious compartment was not separated by whether the individual was hospitalised or not. This separation could potentially allow for increased inference of the contribution from the hospital as we could fit the community contribution of R_0 under the hierarchical structure. This could be accounted for in each of these outbreaks by modelling R_0 as a linear combination of contributions from the different infectious compartments (hospitalised or not). Each form of transmission could then have separate time-dependent transmission terms which could lead to a more appropriate

model.

The hierarchical structure in Chapter 5 could be extended to incorporate intervention measures with similar characteristics in the hierarchical structure. We could assign hyper-priors on the intervention parameters which fall under certain categories which could potentially increase inference capabilities. This could enable insights into the influences of hospital closures as opposed to isolation centres and increased public awareness. This idea could also be applied to current COVID-19 studies whereby we could model the efficacy of different forms of interventions from different data sources.

An extension which would drastically increase widespread use of the methods would be to automate the development of importance sampling particle filters. The methods in this work and in [8] are all developed on a problem-by-problem basis. This can be very time consuming, especially during model selection as is necessary in the early stages of analysis. An approach for developing such methodology would be to separate the problem into the key steps:

1. Provide compartments, propensities from a given state and observed state(s);
2. Define information for filter, population size, time series, final sizes, limiting factors and number of particles;
3. Formulate forcing conditions for the detected events; and,
4. Return particle filter.

This sounds relatively straightforward but the complexity arises in choosing the forcing conditions and ensuring the particle filter appropriately accounts for edge cases. This is something which is unfortunately a major limitation in this kind of method and determining a solution to this would increase ease of use of the methods. This would also require a solution to the inclusion of different observation processes as this would further increase general applicability of the methods.

A major limitation with the analyses in this work is that Ebola is known to exhibit over-dispersed transmission [41, 43]. This is something which our models do not account for as we assume a homogeneously mixing population. The methodologies developed in this work could be used in models which enable these events to occur. This is a similar idea to separating the model into low- and high-risk individuals and possibly by age. We could seed a small number of highly infectious individuals into the model and account for these using a more complex compartmental structure. This would be particularly useful in capturing the heterogeneity in the behaviour during the Kikwit outbreak whereby there are two large peaks which could in theory correspond to super-spreader events. In line with extensions to the models is to add in parameters which could quantify the fear in the community of Ebola, and the effect on transmission. The current model uses a time-dependent transmission term which captures changes from community and

any interventions in the same term. The addition of a fear profile could allow for the effectiveness of interventions to be determined independently of the natural change in the community. Issues would arise in attempting to assign priors to values on the fear parameter and this would need to be thought out appropriately to gain any relevant insights from the inference. This idea would likely require more complex model structures to facilitate identification of individuals in hospitals or quarantine centres so that we are directly able to assess the influence of closing these facilities.

An additional further area of research would be to consider a formal comparison between the hierarchical and independent model fits from Chapter 5. A recent method proposed in [74] uses importance sampling for Bayesian model selection, for a broad class of continuous-time Markov chain models. This method could potentially be applied here to assess which method fits the data better and whether the hierarchical model is a better alternative to fitting four CTMC models independently. We could also consider the use of information criterion, like the way Deviance Information Criterion is used in [14] to choose between models. It may prove difficult to apply each of these methods in there current forms correctly as we are comparing four models to one. However, it is likely that these model selection approaches could be adapted to apply to the problem here.

An extension to the particle filtering methodology could be to consider more refined proposal and subsequent choice of the next observed event. The current approach chooses an index j in proportion to its rate under the modified process,

$$\Pr(J = j) = \frac{c_j}{c_0}.$$

Provided the original process is left unmodified, we can adjust the modified process to increase the probability of consistent realisations. In the analysis of the Kikwit outbreak in Chapter 4 we saw different variances in the log-likelihood contributions on particular days depending on the ordering of the events over that day. This arises due to the need to propose the event times over the observation window and that the type of the next observed event is chosen in proportion to its rate. Some realisations over a given day will be inherently more likely than others. It would be worth trying to incorporate some level of knowledge about the number of each event remaining—over either the current day or in total—relative to one another into the selection methodology. This could potentially be done by choosing the next event based on,

$$\Pr(J = j) \propto \kappa_j \frac{c_j}{c_0},$$

where κ_j could for example be the ratio of relative events. The renormalisation would be simple as there are only two events (divide by the total unnormalised probabilities). If we assume (without loss of generality) y_1 of event 1 and y_2 of event 2 remain to be observed

over the observation period then we could assume,

$$\kappa_j = \frac{y_j}{y_1 + y_2}.$$

This approach increases the probability of proposing the particular event with a larger number of observations remaining and may lead to improved consistency when there are many events to be simulated over a day. This is one potential idea of additional information and it may be the case that an alternative could lead to further reduction in the variance of the log-likelihood estimates. An approach for determining how well this idea works would be to compare the posterior distributions from adopting this approach against the results of the method proposed in this thesis. The distribution over the log-likelihood estimates could be compared to see which method leads to reduced variance in the log-likelihood estimates. This could also be applied to the modified rates of the unforced events and increase the probability of events where there are larger numbers remaining. A major consideration with implementing this approach would be to ensure that this additional calculation—however it is carried out—does not introduce an increase to the computational expense which outweighs the benefit.

An additional area of further study would be to use the importance sampling particle filters presented here in conjunction with SMC^2 algorithms. SMC^2 algorithms use sequential Monte Carlo to target the parameter posterior and states of the system at the same time [25, 32]. These methods aim to reduce the issues with tuning particle filtering approaches and it would be worthwhile assessing whether improved performance can be attained. It would be interesting to investigate the methods presented here being used in conjunction with that approach to see if more optimal pmMH procedures can be developed.

Appendix A

Algorithms

Algorithm 7 Bootstrap Particle Filter

Inputs Data $\boldsymbol{y}_{1:T}$, number of particles N_{part}, initial distribution $p(x_0)$

Set $t = 0$
for $i = 1 : N_{\text{part}}$ **do**
    $x_0^{(i)} \sim p(x_0)$
end for
for $t = 1 : T$ **do**
    for $i = 1 : N_{\text{part}}$ **do**
        Using SSA simulate,

$$x_t^{(i)} \sim p(x_t | x_{t-1}^{(i)})$$

        Calculate weights,

$$w_t^{(i)} = \Pr(y_t | x_t^{(i)})$$

    end for
    if $t = 1$ **then**
        Calculate initial estimate of likelihood,

$$p_1(y_1) = \frac{1}{N_{\text{part}}} \sum_{i=1}^{N_{\text{part}}} w_t^{(i)}$$

12: **else**
13: Calculate estimate of likelihood,

$$p_t(y_t|\boldsymbol{y}_{1:t-1}) = \frac{1}{N_{\text{part}}} \sum_{i=1}^{N_{\text{part}}} w_t^{(i)}$$

14: **end if**
15: Normalise weights,

$$\tilde{w}_t^{(i)} = \frac{w_t^{(i)}}{\sum_{i=1}^{N_{\text{part}}} w_t^{(i)}}$$

16: Resample particles with replacement using the renormalised weights
17: **end for**
18: Calculate estimate of likelihood,

$$\hat{L}(\theta) = p_1(y_1) \prod_{t=2}^{T} p_t(y_t|\boldsymbol{y}_{1:t-1})$$

19: **return** Estimate of the likelihood $\hat{L}(\theta)$

Algorithm 8 Alive Bootstrap Particle Filter

Inputs Data $\boldsymbol{y}_{1:T}$, number of particles N_{part}

Set $x_1^{(i)} = y_1, i = 1, 2, \ldots, N_{\text{part}}$

Set $p_1(y_1) = 1$

for $t = 2 : T$ **do**

  for $i = 1 : n_t$ until $\sum_{i=1}^{n_t} I(x_t^{(i)}, y_t) = N_{\text{part}} + 1$ **do**

    Sample $a_{t-1}^{(i)}$ uniformly from $\{1, 2, \ldots, N_{\text{part}}\}$

    Using the SSA, simulate,

$$x_t^{(i)} \sim p(x_t | x_{t-1}^{a_{t-1}^{(i)}})$$

    Set $n_t \leftarrow n_t + 1$

  end for

  Calculate estimate of the likelihood at time t,

$$p_t(y_t | \boldsymbol{y}_{1:t-1}) = \frac{N_{\text{part}}}{n_t - 1}$$

end for

Calculate estimate of the likelihood,

$$\hat{L}(\theta) = \prod_{t=2}^{T} p_t(y_t | \boldsymbol{y}_{1:t-1})$$

return Estimate of the likelihood $\hat{L}(\theta)$

Appendix B

Posterior simulations for independent inferences

The posterior simulations are all simulated using the inference results. The Yambuku, Mweka and Boende simulations are all conditioned on seeing at least 150, 10 and 30 detected onsets respectively. The Kikwit outbreak is conditioned on not exceeding the number of detected onsets and removals over the first 65 days. This accounts for the period with low surveillance and then 10 days afterwards, when the outbreak begins the second phase. This results in simulations with similar characteristics to the true outbreaks.

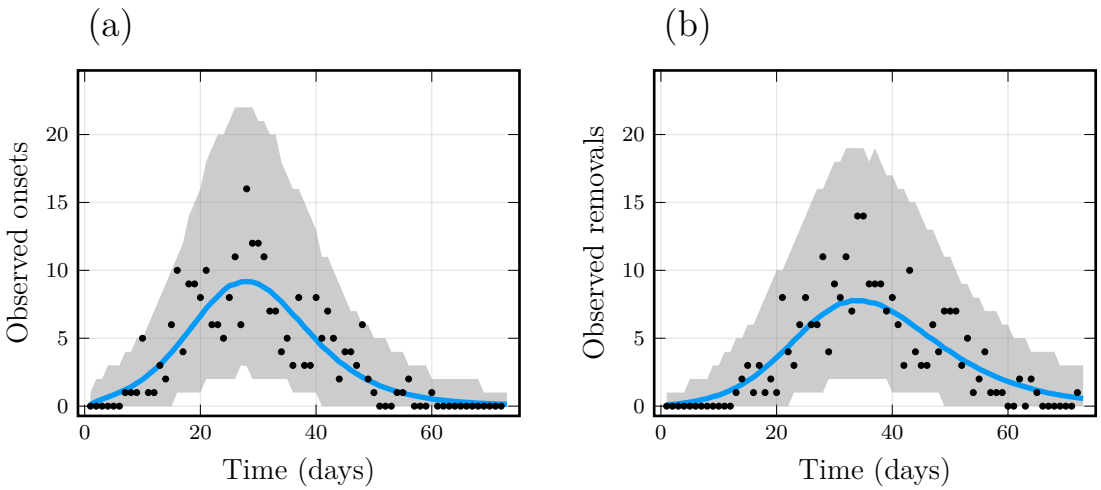

Figure B.1: Yambuku data versus the posterior simulations. Results are shown for 10000 outbreaks simulated according to the SEIR model with partial detection, using the posterior sample: (a) the incidence of symptom onsets; (b) the incidence of removals. The blue line indicates the daily average incidence across the simulations. The grey area corresponds to the 95% credible interval of the simulations. The black dots indicates the detected events for the Yambuku outbreak.

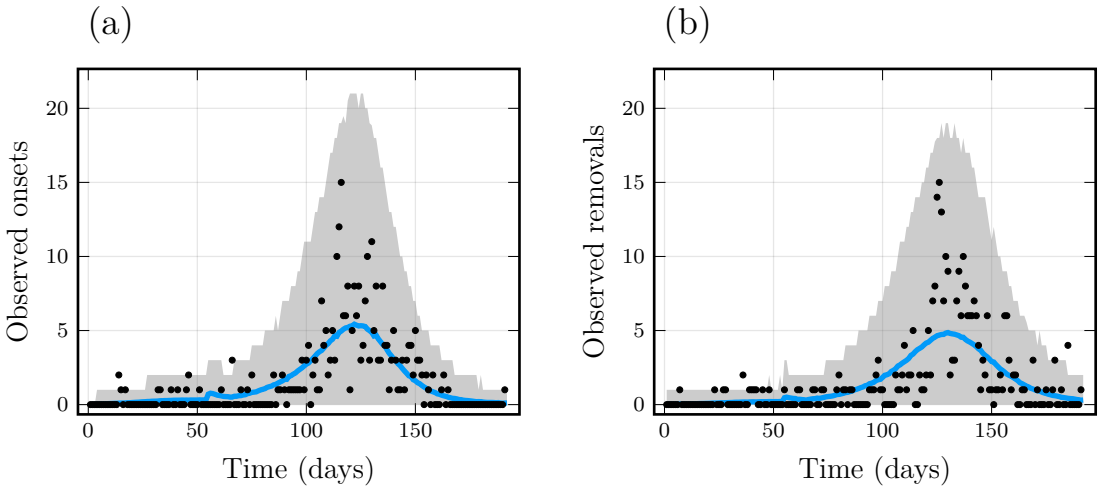

Figure B.2: Kikwit data versus the posterior simulations. See caption of Figure B.1 for more details.

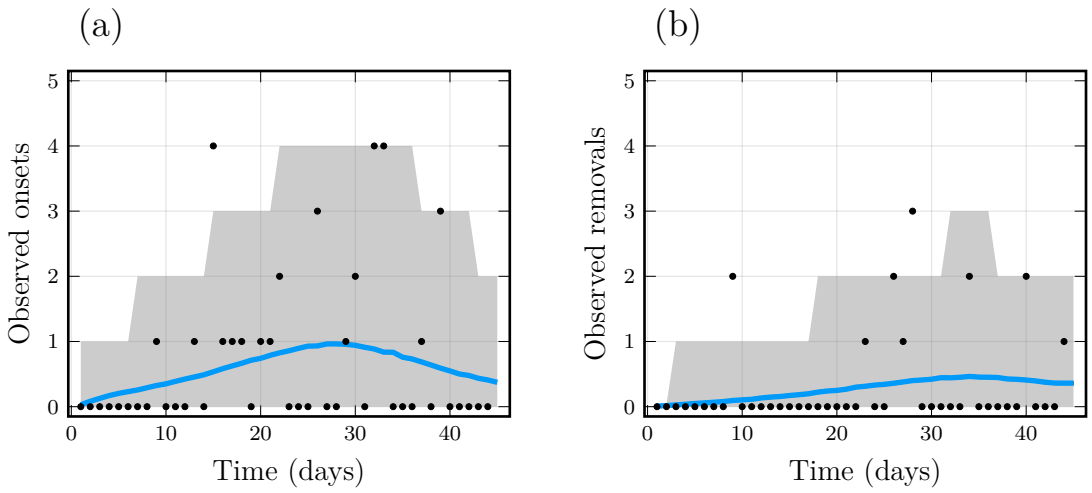

Figure B.3: Mweka data versus the posterior simulations. See caption of Figure B.1 for more details.

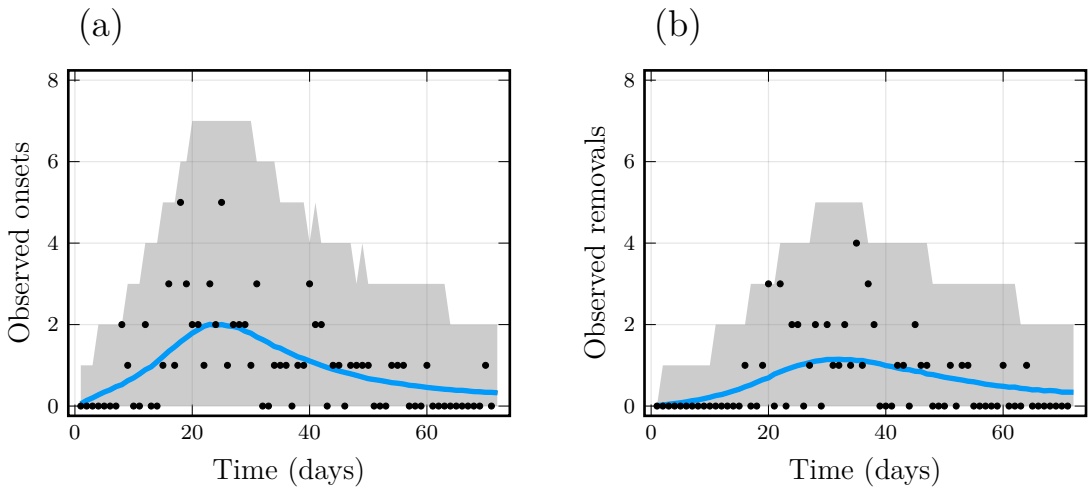

Figure B.4: Boende data versus the posterior simulations. See caption of Figure B.1 for more details.

Bibliography

[1] Allen, L. J., Brauer, F., Van den Driessche, P. and Wu, J., 2008. *Mathematical Epidemiology*, Springer Berlin Heidelberg.

[2] Andersson, H. and Britton, T., 2000. *Stochastic Epidemic Models and Their Statistical Analysis*, Springer New York.

[3] Andrieu, C., Doucet, A. and Holenstein, R., 2010. 'Particle Markov chain Monte Carlo methods', *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **72**(3), 269–342.

[4] Angelis, D. D., Presanis, A. M., Birrell, P. J., Tomba, G. S. and House, T., 2015. 'Four key challenges in infectious disease modelling using data from multiple sources', *Epidemics* **10**, 83–87.

[5] Bédard, M., 2008. 'Optimal acceptance rates for metropolis algorithms: Moving beyond 0.234', *Stochastic Processes and their Applications* **118**(12), 2198–2222.

[6] Begon, M., Bennett, M., Bowers, R., French, N., Hazel, S. and Turner, J., 2002. 'A clarification of transmission terms in host-microparasite models: numbers, densities and areas', *Epidemiology and Infection* **129**(1), 147–153.

[7] Bhatnagar, S., Prasad, H. L. and Prashanth, L. A., 2012. *Stochastic Recursive Algorithms for Optimization*, Springer-Verlag GmbH.

[8] Black, A. J., 2019. 'Importance sampling for partially observed temporal epidemic models', *Statistics and Computing* **29**(4), 617–630.

[9] Black, A. J. and Ross, J. V., 2013. 'Estimating a Markovian epidemic model using household serial interval data from the early phase of an epidemic', *PLoS ONE* **8**(8), e73420.

[10] Blanchard, P., Higham, D. J. and Higham, N. J., 2019. 'Accurate computation of the log-sum-exp and softmax functions'.

[11] Botev, Z. I., Grotowski, J. F. and Kroese, D. P., 2010. 'Kernel density estimation via diffusion', *The Annals of Statistics* **38**(5), 2916–2957.

[12] Breman, J. *et al.*, 1978. The epidemiology of Ebola haemorrhagic fever in Zaire, Elsevier/North Holland, pp. 103–124.

[13] Bwaka, M. A. *et al.*, 1999. 'Ebola hemorrhagic fever in Kikwit, Democratic Republic of the Congo: Clinical observations in 103 patients', **179**(s1), S1–S7.

[14] Camacho, A., Kucharski, A., Funk, S., Breman, J., Piot, P. and Edmunds, W., 2014. 'Potential for large outbreaks of Ebola virus disease', *Epidemics* **9**, 70–78.

[15] Choi, M. J. *et al.*, 2021. 'Use of Ebola vaccine: Recommendations of the advisory committee on immunization practices, United States, 2020', **70**(1), 1–12.

[16] Chong, E. K. P. and Zak, S. H., 2001. *An Introduction to Optimization, 2nd Edition*, Wiley-Interscience.

[17] Chopin, N., 2020. *An introduction to Sequential Monte Carlo*, Springer.

[18] Chowell, G., Hengartner, N., Castillo-Chavez, C., Fenimore, P. and Hyman, J., 2004. 'The basic reproductive number of Ebola and the effects of public health measures: the cases of Congo and Uganda', *Journal of Theoretical Biology* **229**(1), 119–126.

[19] Chowell, G. and Nishiura, H., 2014. 'Transmission dynamics and control of Ebola virus disease (EVD): a review', **12**(1).

[20] Domachowske, J., 2020. Ebola, *in* 'Vaccines', Springer International Publishing, pp. 143–149.

[21] Doucet, A., Freitas, N. and Gordon, N., eds, 2001. *Sequential Monte Carlo Methods in Practice*, Springer New York.

[22] Doucet, A., Pitt, M. K., Deligiannidis, G. and Kohn, R., 2015. 'Efficient implementation of Markov chain Monte Carlo when using an unbiased likelihood estimator', *Biometrika* **102**(2), 295–313.

[23] Drescher, M., Perera, A. H., Johnson, C. J., Buse, L. J., Drew, C. A. and Burgman, M. A., 2013. 'Toward rigorous use of expert knowledge in ecological research', **4**(7), art83.

[24] Drovandi, C., 2014. Pseudo-marginal algorithms with multiple CPUs, Technical report.
URL: *http://eprints.qut.edu.au/61505*

[25] Drovandi, C. C., Pettitt, A. N. and McCutchan, R. A., 2016. 'Exact and Approximate Bayesian Inference for Low Integer-Valued Time Series Models with Intractable Likelihoods', *Bayesian Analysis* **11**(2), 325–352.

[26] Gamerman, D., 1997. *Markov Chain Monte Carlo*, Chapman & Hall.

[27] Gelman, A., 2013. *Bayesian Data Analysis*, CRC Press.

[28] Gibson, G., 1998. 'Estimating parameters in stochastic compartmental models using Markov chain methods', *Mathematical Medicine and Biology* **15**(1), 19–40.

[29] Gilks, W. R., Richardson, S. and Spiegelhalter, D. J., 1995. *Markov Chain Monte Carlo in Practice*, Taylor & Francis Ltd.

[30] Gillespie, D. T., 1977. 'Exact stochastic simulation of coupled chemical reactions', *The Journal of Physical Chemistry* **81**(25), 2340–2361.

[31] Godsill, S., 2019. Particle filtering: the first 25 years and beyond, *in* 'ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)', IEEE.

[32] Golightly, A. and Kypraios, T., 2017. 'Efficient SMC^2 schemes for stochastic kinetic models', *Statistics and Computing* **28**(6), 1215–1230.

[33] Gordon, N., Salmond, D. and Smith, A., 1993. 'Novel approach to nonlinear/non-gaussian Bayesian state estimation', *IEE Proceedings F Radar and Signal Processing* **140**(2), 107.

[34] Gosavi, A., 2014. *Simulation-based optimization : parametric optimization techniques and reinforcement learning*, Springer, Boston.

[35] Hastings, W. K., 1970. 'Monte carlo sampling methods using Markov chains and their applications', *Biometrika* **57**(1), 97–109.

[36] Jasra, A., Lee, A., Yau, C. and Zhang, X., 2013. 'The alive particle filter'.

[37] Jenkinson, G. and Goutsias, J., 2012. 'Numerical integration of the master equation in some models of stochastic epidemiology', *PLoS ONE* **7**(5), e36160.

[38] Joint United Nations Programme on HIVAIDS, 2020. '2020 Global AIDS Update – Seizing the moment – Tackling entrenched inequalities to end epidemics'.

[39] Keeling, M. J. and Rohani, P., 2007. *Modeling Infectious Diseases in Humans and Animals*, Princeton University Press, New Jersey.

[40] Kermack, W. O. and McKendrick, A. G., 1927. 'A contribution to the mathematical theory of epidemics', *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* **115**(772), 700–721.

[41] Khan, A. S. *et al.*, 1999. 'The Reemergence of Ebola Hemorrhagic Fever, Democratic Republic of the Congo, 1995', *The Journal of Infectious Diseases* **179**(s1), S76–S86.

[42] Kroese, D. P., Taimre, T. and I. Botev, Z., 2011. *MCM Handbook*, Wiley.

[43] Lau, M. S. Y., Dalziel, B. D., Funk, S., McClelland, A., Tiffany, A., Riley, S., Metcalf, C. J. E. and Grenfell, B. T., 2017. 'Spatial and temporal dynamics of superspreading events in the 2014–2015 West Africa Ebola epidemic', **114**(9), 2337–2342.

[44] Legrand, J., Grais, R. F., Boelle, P. Y., Valleron, A. J. and Flahault, A., 2006. 'Understanding the dynamics of Ebola epidemics', **135**(4), 610–621.

[45] Lekone, P. E. and Finkenstädt, B. F., 2006. 'Statistical inference in a stochastic epidemic SEIR model with control intervention: Ebola as a case study', *Biometrics* **62**(4), 1170–1177.

[46] Li, T., Bolic, M. and Djuric, P. M., 2015. 'Resampling methods for particle filtering: Classification, implementation, and strategies', **32**(3), 70–86.

[47] Mantle, J. and Tyrrell, D. A. J., 1973. 'An epidemic of influenza on Tristan da Cunha', *Journal of Hygiene* **71**(1), 89–95.

[48] Marin, J. and Robert, C., 2007. *Bayesian core: a practical approach to computational Bayesian statistics*, Springer, New York.

[49] Maryak, J. and Chin, D., 2001. Global random optimization by simultaneous perturbation stochastic approximation, *in* 'Proceedings of the 2001 American Control Conference. (Cat. No.01CH37148)', IEEE.

[50] Maryak, J. L. and Chin, D. C., 2008. 'Global random optimization by simultaneous perturbation stochastic approximation', *IEEE Transactions on Automatic Control* **53**(3), 780–783.

[51] McKinley, T. J., Cook, A. R. and Deardon, R., 2009. 'Inference in epidemic models without likelihoods', *The International Journal of Biostatistics* **5**(1).

[52] McKinley, T. J., Ross, J. V., Deardon, R. and Cook, A. R., 2014. 'Simulation-based Bayesian inference for epidemic models', *Computational Statistics & Data Analysis* **71**, 434–447.

[53] Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H. and Teller, E., 1953. 'Equation of state calculations by fast computing machines', **21**(6), 1087–1092.

[54] Moral, P. D., Doucet, A. and Jasra, A., 2006. 'Sequential Monte Carlo samplers', *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **68**(3), 411–436.

[55] Ndanguza, D., Tchuenche, J. M. and Haario, H., 2011. 'Statistical data analysis of the 1995 Ebola outbreak in the Democratic Republic of Congo', **24**(1), 55–68.

[56] O'Neill, P. D. and Roberts, G. O., 1999. 'Bayesian inference for partially observed stochastic epidemics', **162**(1), 121–129.

[57] Pitt, M. K., dos Santos Silva, R., Giordani, P. and Kohn, R., 2012. 'On some properties of Markov chain Monte Carlo simulation methods based on the particle filter', *Journal of Econometrics* **171**(2), 134–151.

[58] Pooley, C. M., Bishop, S. C. and Marion, G., 2015. 'Using model-based proposals for fast parameter inference on discrete state space, continuous-time Markov processes', **12**(107), 20150225.

[59] Report of an International Commission, 1978. 'Ebola haemorrhagic fever in Zaire, 1976', *Bulletin of the World Health Organization* **56**(2), 271 – 293.

[60] Robert, A., Camacho, A., Edmunds, W. J., Baguelin, M., Tamfum, J.-J. M., Rosello, A., Kéïta, S. and Eggo, R. M., n.d.. 'Control of ebola virus disease outbreaks: Comparison of health care worker-targeted and community vaccination strategies', **27**, 106–114.

[61] Rosello, A. *et al.*, 2015. 'Ebola virus disease in the Democratic Republic of the Congo, 1976-2014', *eLife* **4**.

[62] Ross, S. M., 2014. *Introduction to Probability Models*, Academic Press.

[63] Sarkka, S., 2013. *Bayesian Filtering and Smoothing*, Cambridge University Press.

[64] Schön, T. B., Svensson, A., Murray, L. and Lindsten, F., 2018. 'Probabilistic learning of nonlinear dynamical systems using sequential Monte Carlo', *Mechanical Systems and Signal Processing* **104**, 866–883.

[65] Sherlock, C., Thiery, A. H., Roberts, G. O. and Rosenthal, J. S., 2015. 'On the efficiency of pseudo-marginal random walk Metropolis algorithms', *The Annals of Statistics* **43**(1), 238–275.

[66] Sidje, R. B., 1998. 'Expokit: A software package for computing matrix exponentials', *ACM Trans. Math. Softw.* **24**(1), 130–156.
URL: *https://doi.org/10.1145/285861.285868*

[67] Silvia, D. S., 1996. *Data Analysis A Bayesian Tutorial*, Oxford University Press.

[68] Spall, J., 1992. 'Multivariate stochastic approximation using a simultaneous perturbation gradient approximation', *IEEE Transactions on Automatic Control* **37**(3), 332–341.

[69] Spall, J. C., 2003. *Introduction to Stochastic Search and Optimization*, John Wiley & Sons, Inc.

[70] Touloupou, P., Alzahrani, N., Neal, P., Spencer, S. E. F. and McKinley, T. J., 2018. 'Efficient model comparison techniques for models requiring large scale data augmentation', **13**(2), 437–459.

[71] Tumpey, T. M., 2005. 'Characterization of the reconstructed 1918 Spanish influenza pandemic virus', *Science* **310**(5745), 77–80.

[72] Turkman, M. A. A., Paulino, C. D. and Müller, P., 2019. *Computational Bayesian Statistics*, Cambridge University Press.

[73] Van Kerkhove, M. D., Bento, A. I., Mills, H. L., Ferguson, N. M. and Donnelly, C. A., 2015. 'A review of epidemiological parameters from Ebola outbreaks to inform early public health decision-making', *Scientific Data* **2**(1).

[74] Walker, J. N., Black, A. J. and Ross, J. V., 2019. 'Bayesian model discrimination for partially-observed epidemic models', **317**, 108266.