Mathematical Medicine and Biology (2015) **32**, 79–98
doi:10.1093/imammb/dqt021
Advance Access publication on October 10, 2013

Efficient parameter estimation for models of healthcare-associated pathogen transmission in discrete and continuous time

Alun Thomas*
Division of Genetic Epidemiology, 391 Chipeta Way Suite D, Salt Lake City, UT 84108, USA
*Corresponding author. Email: alun@genepi.med.utah.edu

and

Andrew Redd, Karim Khader, Molly Leecaster, Tom Greene and Matthew Samore
Division of Epidemiology, University of Utah, Salt Lake City, UT, USA and VA Salt Lake City Health Care System, Salt Lake City, UT, USA

[Received on 4 March 2013; revised on 4 September 2013; accepted on 10 September 2013]

We describe two novel Markov chain Monte Carlo approaches to computing estimates of parameters concerned with healthcare-associated infections. The first approach frames the discrete time, patient level, hospital transmission model as a Bayesian network, and exploits this framework to improve greatly on the computational efficiency of estimation compared with existing programs. The second approach is in continuous time and shares the same computational advantages. Both methods have been implemented in programs that are available from the authors. We use these programs to show that time discretization can lead to statistical bias in the underestimation of the rate of transmission of pathogens. We show that the continuous implementation has similar running time to the discrete implementation, has better Markov chain mixing properties, and eliminates the potential statistical bias. We, therefore, recommend its use when continuous-time data are available.

Keywords: bacterial colonization; Bayesian networks; infectious disease transmission; Markov chain Monte Carlo integration; nosocomial infection; statistical bias; susceptible-infected models.

1. Introduction

Within the past 5 years, significant progress has been made using stochastic patient-level models to address important questions related to the sources of pathogens to better understand factors that affect their hospital associated transmission, and to measure the impact of intervention strategies to reduce their spread. Ideally, statistical analyses would be done straightforwardly using direct observations of transmission events. In reality, however, transmissions are inherently unobservable and must be inferred from clinical and surveillance testing done during a patient's hospital stay. Fortunately, many statistical methods have been developed to deal with such missing data problems, and making multiple imputations of the missing data using Markov chain Monte Carlo (MCMC) methods have been fruitful in this context with the works of Gibson *et al.* (2004), Forrester *et al.* (2007), Cooper *et al.* (2008) and Kypraios *et al.* (2010) being of particular note. The common element in these hidden Markov methods is the explicit representation of the history of admission, testing and discharge of each patient in a hospital unit, the hidden variable being the unobserved time of colonization. This contrasts with *compartmental* models in which only aggregate counts of patients and test results are considered (Cooper & Lipsitch, 2004; McBryde *et al.*, 2007; Drovandi & Pettitt, 2008; Christopher *et al.*, 2011). Which approach is appropriate will depend on any application and the need to balance the tractability of compartmental models against the detail and flexibility of patient-level hidden Markov models. For a more complete review, see O'Neill (2010).

The MCMC methods of Forrester *et al.* (2007), Cooper *et al.* (2008) and Kypraios *et al.* (2010) all use reversible jump MCMC (*RJMCMC*) (Green, 1995) and all update a single patient, or small proportion of the patients, at each iteration. Hence, a large number of iterations as required before convergence is achieved. Each study also analysed relatively small hospital units over short periods of time. If one is interested in analysing data from large hospitals over longer periods of time, the computational times required by any of these three approaches is substantial and possibly prohibitive. Each also assumes that a user has significant programming expertise.

In this work, we reframe the hospital transmission model as a Bayesian network and estimate its parameters using both discrete-time and continuous-time implementations. Because the focus of this work is on the method of estimation, rather than specific model extensions, we use a basic transmission model having a single transmission rate, importation probability and surveillance test sensitivity. This model was originally designed for pathogens that follow the susceptible-infected, or SI, pattern (Gibson *et al.*, 2004). In a hospital context, this implies that once patients become colonized by pathogens, such as methicillin-resistant Staphylococcus aureus (MRSA), they remain so for the duration of their hospitalization. Our approach and programs can be extended in the same ways that have been done by previous authors (Forrester *et al.*, 2007; Cooper *et al.*, 2008; Kypraios *et al.*, 2010).

In our restructured problem, the dimension of the state space is constant. Thus, we do not use reversible jump methods, but instead use more straightforward Gibbs (Geman & Geman, 1984) or Metropolis (Metropolis *et al.*, 1953) updates. In addition, we note that making these updates does not require complete calculation of all the terms in a likelihood, but only the, typically, small number of terms affected by a proposed perturbation. Thus, our implementations are considerably more efficient and faster than the previous ones. We have taken advantage of these computational enhancements to consider the statistical properties of parameter estimates from data for far larger hospital units and study duration than previously possible. In particular, using simulated data, we show that imposing time discretization on data that is reported continuously can lead to biases in the estimates of transmission rates and importation probabilities. These biases are avoided in our continuous-time implementation. We also show that the continuous-time program has similar computational requirements to the discrete-time program, and that the mixing properties of the Markov chain it induces are superior. We, therefore, recommend the continuous-time implementation for any further use.

2. Methods

In this section, we will describe two MCMC algorithms to estimate model parameters for healthcare-associated transmissions using patient histories of admission, discharge and test results from surveillance samples. The first algorithm is applicable to data that are either reported in discrete time or for which times have been rounded. If continuous-time data are available, our second algorithm can be used. We will use a similar notation to Cooper *et al.* (2008) but focus on one particular straightforward model. This simple model is chosen for purposes of clarity of exposition and illustration and should not be taken as a limitation of our approach. In particular, computational requirements are not impacted by any of the extensions considered by Cooper *et al.* (2008).

2.1 *Notation and model*

All times are recorded in days between 0 and T, the length of the observation period. We use the term *patient episode* to mean a contiguous episode of care within the unit under study for a particular patient defined by known admission and discharge times and during which test samples may be taken from the patient. The same patient may have multiple patient episodes, although this is not always accounted for in models. Patient episodes are indexed from 1 to N.

The observed data D consist of $\{t_i^a\}, \{t_i^d\}, \{s_{i,j}\}$ and $\{t_{i,j}^s\}$, where t_i^a and t_i^d are the admission and discharge times for the ith patient episode, $s_{i,j}$ is the result of the jth test done during patient episode i, with 0 and 1, respectively, indicating negative and positive results, and $t_{i,j}^s$ is the time at which the sample used for the test was taken.

The key element in the MCMC approach is the introduction of an augmented data set A. The augmented data A consist of $\{a_i\}, \{c_i\}$ and $\{t_i^c\}$, where a_i is a binary indicator set to 1 if the ith patient was colonized on admission and 0 otherwise, c_i is a binary indicator set to 1 if the ith patient was colonized during their stay and t_i^c is the time of this colonization event. When $c_i = 0$, t_i^c is undefined. It is assumed that once colonized, a patient remains so for the duration of their stay.

Let $h_i(t) = 1$ if the ith patient is on the ward at time t, and 0 otherwise, and let $m_i(t) = 1$ if the ith patient is colonized at time t, and 0 otherwise. Thus, at time t, $z(t) = \sum_i h_i(t)$ is the total number of patients on the ward and $y(t) = \sum_i h_i(t)m_i(t)$ is the number of colonized patients.

Let ν be the probability that a patient is colonized at admission. Let $\lambda > 0$ be the instantaneous rate at which patient to patient transmissions occur so that if a patient is uncolonized at time t and exposed to $y(t)$ colonized patients, the probability that they become colonized in the interval $(t, t + \Delta)$ is $1 - \mathrm{e}^{-\lambda y(t)\Delta}$, provided that the interval is short enough that $y(t)$ does not change. Let ξ be the test sensitivity, i.e. the probability that a test taken on a colonized patient gives a positive result. We will assume perfect specificity, i.e. that there are no false-positive results.

The posterior distribution is then given by

$$p(A, \nu, \lambda, \xi \mid D) \propto p(D \mid A, \xi)p(A \mid \nu, \lambda)p(\nu, \lambda, \xi), \tag{1}$$

where

$$p(D \mid A, \xi) = \prod_{i,j:m_i(t_{i,j}^s)=1} \xi^{s_{i,j}}(1 - \xi)^{1-s_{i,j}}, \tag{2}$$

$$p(A \mid \nu, \lambda) = \prod_i \exp\left(-\int_0^T \lambda y(t)h_i(t)[1 - m_i(t)]\,\mathrm{d}t\right) \prod_{i:c_i=1} \lambda y(t_i^c - \delta) \prod_i \nu^{a_i}(1 - \nu)^{1-a_i}, \tag{3}$$

and $p(\nu, \lambda, \xi)$ is the parameter prior of choice. The value δ is infinitesimally small so that $y(t_i^c - \delta)$ is the count of colonized patients immediately before the ith patient is colonized.

2.2 *Discrete-time estimation*

2.2.1 *Formulation as a Bayesian network.* We assume that times are reported discretely in equal intervals of arbitrary length Δ days, e.g. full days, half days, hours, etc. The number of time intervals is T/Δ. In discrete time, it is not necessary to track $\{a_i\}$, $\{c_i\}$ and $\{t_i^c\}$ directly because $a_i = 1$ if and only if $m_i(t_i^a) = 1$, $c_i = 1$ only if $a_i = 0$ and $m_i(t_i^d) = 1$, and $t_i^c = \min(t_i^d + 1, \{t : m_i(t) = 1\})$. Hence, the augmented data are fully specified by $\{m_i(t) : h_i(t) = 1,\ i = 1 \ldots N, t = 1 \ldots T/\Delta\}$.

The posterior over the parameters and augmented variables is now the Bayesian network

$$p(\nu, \lambda, \xi, \{m_i(j)\} \mid D) \propto p(\nu, \lambda, \xi) \prod_i p(m_i(t_i^a) \mid \nu) \prod_i \prod_{t=t_i^a+1}^{t_i^d} p(m_i(t) \mid m_i(t-1), y(t-1), \lambda)$$
$$\times \prod_i \prod_j p(s_{i,j} \mid m_i(t_{i,j}^s), \xi), \tag{4}$$

where

$$p(m_i(t_i^a) \mid \nu) = \begin{cases} \nu & \text{if } m_i(t_i^a) = 1, \\ 1-\nu & \text{otherwise,} \end{cases} \tag{5}$$

gives the probability of importations,

$$p(m_i(t) \mid m_i(t-1), y(t-1), \lambda) = \begin{cases} \mathrm{e}^{-\lambda y(t-1)\Delta} & \text{if } m_i(t) = 0,\ m_i(t-1) = 0, \\ 1 - \mathrm{e}^{-\lambda y(t-1)\Delta} & \text{if } m_i(t) = 1,\ m_i(t-1) = 0, \\ 0 & \text{if } m_i(t) = 0,\ m_i(t-1) = 1, \\ 1 & \text{if } m_i(t) = 1,\ m_i(t-1) = 1, \end{cases} \tag{6}$$

gives the period to period transmission probabilities and

$$p(s_{i,j} \mid m_i(t_{i,j}^s), \xi) = \begin{cases} 1 & \text{if } s_{i,j} = 0,\ m_i(t_{i,j}^s) = 0, \\ 0 & \text{if } s_{i,j} = 1,\ m_i(t_{i,j}^s) = 0, \\ 1-\xi & \text{if } s_{i,j} = 0,\ m_i(t_{i,j}^s) = 1, \\ \xi & \text{if } s_{i,j} = 1,\ m_i(t_{i,j}^s) = 1, \end{cases} \tag{7}$$

specifies the test characteristics.

Figure 1 gives a graphical representation of part of a typical Bayesian network of the above form. Each vertex represents a variable and the arrows indicate which variables have immediate influence on others. The example shows the variables needed to represent two patient episodes. Episode 1 from period 18–21, with two tests, and episode 2 from period 20–22 with three tests including two taken during the same period. Note that the $\{y(t)\}$, the variables giving the by-period count of colonized individuals, form a central core for this structure of variables so that the patient episode variables interact directly only with this core and not with the variables for other patient episodes. This leads to many conditional independences between the variables that can be exploited when programming the MCMC algorithm.

As our examples involve units of large capacity observed over hundreds of days, the effect of the prior distributions is minimal and for convenience we chose Uniform $(0, 1)$ priors for ν and ξ, and an improper Uniform prior for $\log(\lambda)$.

2.2.2 *Monte Carlo Markov chain.* Our MCMC scheme is then a mixture of Gibbs updates (Geman & Geman, 1984) for ν and ξ and Metropolis updates (Metropolis *et al.*, 1953) for λ and the augmented data $\{m_i(t)\}$. More specifically, a single iteration of updates does the following:

- Update $\{m_i(t)\}$: For each patient episode i in some order update the episode history in the following way:

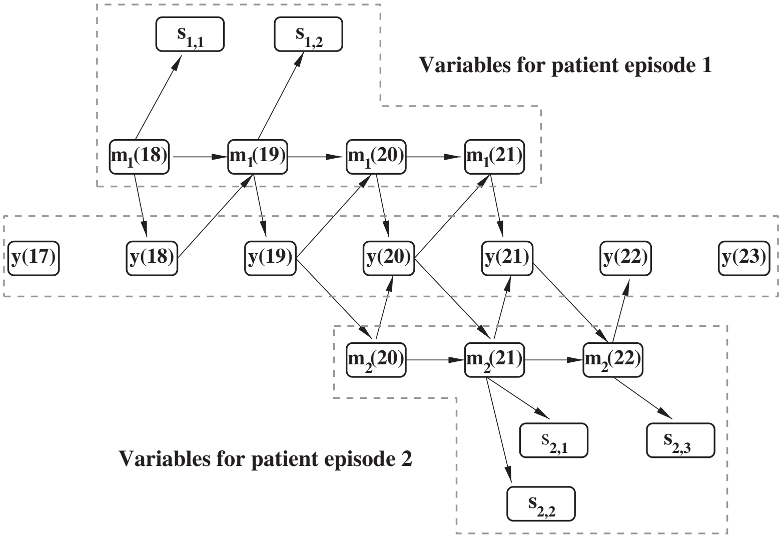

FIG. 1. A graphical representation of the Bayesian network used to implement the discrete-time model.

- Find t_i^c, i.e. the current time of colonization for the ith patient with the convention that $t_i^c = t_i^d + 1$ if the patient was not colonized during the episode.
- Propose a new value $t_i'^c$ uniformly at random from the range $t_i^a \leqslant t_i'^c \leqslant t_i^d + 1$ to be the new first colonization period.
- By flipping the states of the variables for patient episode i between t_i^c and $t_i'^c$, and by increasing or decreasing the appropriate $y(t)$ correspondingly, we generate a new patient event history. Let $\{m_i'(t)\}$ be the new history thus proposed.
- Accept this new history with probability

$$\min\left\{1, \frac{p(\nu, \lambda, \xi, \{m_i'(t)\} \mid D)}{p(\nu, \lambda, \xi, \{m_i(t)\} \mid D)}\right\}. \tag{8}$$

- Update ν: Calculate

$$u = \sum_i m_i(t_i^a) \quad \text{and} \quad v = \sum_i [1 - m_i(t_i^a)], \tag{9}$$

and generate a new value of ν from a Beta distribution with parameters u and v.

- Update ξ: Calculate

$$w = \sum_{i,j:m_i(t_{i,j}^s)=1} s_{i,j} \quad \text{and} \quad x = \sum_{i,j:m_i(t_{i,j}^s)=1} [1 - s_{i,j}], \tag{10}$$

and generate a new value of ξ from a Beta distribution with parameters w and x.

- Update λ: Propose a new value

$$\lambda' = \exp[\log(\lambda) + Z] \quad \text{where } Z \sim N(0, \sigma^2), \tag{11}$$

and accept the proposed value with probability

$$\min\left\{1, \frac{p(\nu, \lambda', \xi, \{m_i(j)\}|D)}{p(\nu, \lambda, \xi, \{m_i(j)\}|D)}\right\}, \tag{12}$$

where σ^2 is a variance parameter whose value may need to be tuned depending on the application. We found that $\sigma^2 = 1$ worked well over all the applications shown here.

2.2.3 *Computational complexity.* Recall that the total number of patient episodes is N, and that the study times goes from 0 to T days so that the total number of time periods in the discretization is T/Δ. Let M be the mean length of stay of a patient and let the capacity of the unit be C. Thus, NM is roughly proportional to, and no larger than TC, so $O(NM) = O(TC)$.

The Gibbs updates are straightforward and quick to execute. Updating ν requires only checking one variable for each patient and hence is an $O(N)$ operation. Updating ξ requires only checking one variable for each test. If we assume that the number of tests is roughly proportional to the number of patient episodes, this is also an $O(N)$ process.

The Metropolis updates for λ and $\{m_i(t)\}$ are more computationally demanding. While the proposal schemes are very simple, the acceptance probabilities appear to require complete posterior probability computations, each of which requires a time of $O(TC/\Delta)$ since each $m_i(t)$ needs to be queried. Note, however, that many of the terms cancel from the ratios of posterior probabilities. In order to exploit this in our programs, we make a list of each term of the posterior (4) that each variable and parameter appear in. Then, when the values of a set of variables or parameters are changed, it is straightforward to calculate the acceptance ratios efficiently using only non-cancelling terms. Nonetheless, updating λ is still an $O(TC/\Delta)$ operation. However, when updating each of the N patient episodes, only the terms of (4) that involve concurrent episodes need to be considered, which can be done in a time of $O(MC/\Delta)$ for each episode, instead of $O(TC/\Delta)$. Thus, the time required for a complete update of the parameters and each patient history is

$$O(N) + O(N) + O\left(\frac{TC}{\Delta}\right) + O\left(\frac{NMC}{\Delta}\right) = O\left(\frac{TC}{M}\right) + O\left(\frac{TC}{\Delta}\right) + O\left(\frac{TC^2}{\Delta}\right), \tag{13}$$

which is clearly dominated by the final term $O(TC^2/\Delta)$. The storage requirement for the discrete implementation is dominated by the need to store the $\{m_i(t)\}$ binary indicators and is, hence, $O(TC/\Delta)$. Thus, as we illustrate below, finer time discretizations and longer study periods will increase the time and storage required, as will larger study units.

2.3 *Continuous-time estimation*

2.3.1 *Formulation as a list of events.* In order to implement a continuous-time version, we restate the problem in terms of *events*. An event is an admission, discharge, colonization or test result, and each is associated with its time of occurrence. Admission, discharge and test events are part of the observed data D, while the colonization events are the augmented data A. Again, we do not have to directly track

a_i and c_i, but use the convention that a patient was colonized on admission if and only if $t_i^c = t_i^a$, and that the patient was not colonized during the episode if and only if $t_i^c = t_i^d$.

For any particular value of the augmented data, we can order the events in time from first to last: $(e_1, e_2, \ldots, e_E)$ with associated times $t_1 \leqslant t_2 \leqslant \cdots \leqslant t_E$. The posterior distribution (1) can then be rewritten as

$$p(\nu, \lambda, \xi) \prod_{k=1}^{E} g(e_k) \exp\left(- \int_{t_{k-1}}^{t_k} \sum_i \lambda y(t) h_i(t) [1 - m_i(t)]\, \mathrm{d}t \right) \tag{14}$$

$$= p(\nu, \lambda, \xi) \prod_{k=1}^{E} g(e_k) \exp\{-\lambda y(t_{k-1})[z(t_{k-1}) - y(t_{k-1})][t_k - t_{k-1}]\}, \tag{15}$$

where $g(e_k)$ is a factor depending on the type and time of the event as follows:

- If e_k is an admission or a discharge, $g(e_k) = 1$.
- If e_k is the test result $s_{i,j}$ at time $t_k = t_{i,j}^s$,

$$g(e_k) = \begin{cases} 1 & \text{if } s_{i,j} = 0,\ t_k < t_i^c, \\ 0 & \text{if } s_{i,j} = 1,\ t_k < t_i^c, \\ 1 - \xi & \text{if } s_{i,j} = 0,\ t_k > t_i^c, \\ \xi & \text{if } s_{i,j} = 1,\ t_k > t_i^c. \end{cases} \tag{16}$$

- If e_k is the colonization event for patient i at time $t_k = t_i^c$,

$$g(e_k) = \begin{cases} \nu & \text{if } t_k = t_i^a, \\ (1 - \nu)\lambda y(t_{k-1}) & \text{if } t_i^a < t_k < t_i^d, \\ 1 - \nu & \text{if } t_k = t_i^d. \end{cases} \tag{17}$$

Roughly speaking, we can think of $g(e_k)$ as the probability of what happened at event e_k and the associated exponentiated integral as the probability that nothing happened in the time since the previous event. With this representation, if an event time changes, the only terms in (14) that need to be recalculated to account for this are those for events that occur between the old and new times.

2.3.2 *Monte Carlo Markov chain.* In the MCMC scheme, we update ν and ξ as before. To update λ, we note that the terms of the likelihood (15) that involve λ are proportional to $\lambda^\alpha \mathrm{e}^{-\lambda\beta}$, where α is the count of transmission events that occur in the augmented data, and

$$\beta = \sum_{k=1}^{E} y(t_{k-1})[z(t_{k-1}) - y(t_{k-1})][t_k - t_{k-1}]. \tag{18}$$

If we assume an improper Uniform prior for λ, it follows that its conditional distribution given the current state of the augmented data and the other parameters is a Gamma distribution from which it is straightforward to sample using a Gibbs update instead of the less efficient Metropolis update used in the discrete case.

For the colonization times t_i^c, we again use a Metropolis scheme, as follows:

- Propose a new colonization time $t_i'^c$ from the mixture distribution that puts discrete probabilities of $1/(2+t_i^d-t_i^a)$ at t_i^a and t_i^d and distributes the remaining probability uniformly between these two points. That is, generate $t_i'^c$ from

$$f(x)=\frac{1}{2+t_i^d-t_i^a} \quad \text{for } t_i^a \leqslant x \leqslant t_i^d. \tag{19}$$

Note that, although this proposal distribution is a mixture, the proposals are symmetrical.

- Accept t_i^c with probability

$$\min\left\{1, \frac{p(\nu,\lambda,\xi,t_i'^c,\{t_k^c : k \neq i\} \mid D)}{p(\nu,\lambda,\xi,t_i^c,\{t_k^c : k \neq i\} \mid D)}\right\}. \tag{20}$$

Most of the terms in the ratio of (20) cancel. If e_f is the event immediately preceding the earlier of t_i^c and $t_i'^c$, and e_l is the event immediately following the later of the times, the ratio can be calculated as

$$\prod_{j=f+1}^{l} \frac{g(e_j')\exp\{-\lambda y'(t_{j-1}')[z(t_{j-1}')-y'(t_{j-1}')][t_j'-t_{j-1}']\}}{g(e_j)\exp\{-\lambda y(t_{j-1})[z(t_{j-1})-y(t_{j-1})][t_j-t_{j-1}]\}}, \tag{21}$$

where $\{e_j'\}$ and $\{t_j'\}$ are the new, proposed, events and times. These will differ from the incumbent events only in that:

- the time $t_i = t_i^c$ for the colonization event e_i is changed to $t_i'^c$ and the event is moved to the appropriate position in the list of events;
- the indexes of the events between the t_i^c and $t_i'^c$ are increased by 1 if $t_i'^c < t_i^c$ and decreased by 1 if $t_i'^c > t_i^c$;
- the values of $y(t_j)$ are changed to $y'(t_j')$ to reflect an increase in the count of colonized patients in the unit in the time interval $(t_i'^c, t_i^c)$ if $t_i'^c < t_i^c$ or decrease in the count in $(t_i^c, t_i'^c)$ if $t_i'^c > t_i^c$.

2.3.3 *Computational complexity.* As before, updating ν and ξ are $O(N)$ computations. Updating λ requires a single scan through all of the events to calculate α and β which takes a time of $O(E)$. Each patient episode involves three events, the admission, colonization and discharge events, plus any test result events. Thus, if we again assume that the number of tests is proportional to the number of patient events, $O(E)=O(N)$.

As in the discrete case, at each iteration we update the t_i^c for every patient i, hence the dominating calculation is again the updating of the augmented data. The time required to evaluate the acceptance probability for the update of a single patient episode is proportional to the number of events that occur during the episode which, on average, is $O((M/T)N)=O(C)$. Hence, a complete scan of updates can be done in a time of $O(NC)=O(TC^2/M)$. The storage requirement for the continuous implementation is simply to maintain the list of events and so is $O(N)=O(TC/M)$. The computational requirements for the discrete and continuous implementations, therefore, scale in similar fashion with respect to the length of the study period and unit capacity. As we show below, the actual time required for the continuous version is similar to that required for a discretization by day, and compares favourably with the requirements for finer discretizations.

Whether or not the model that we use requires an RJMCMC depends on how the sampling is formulated. Cooper *et al.* (2008) chose to handle the possibility of importation and never infected individuals by using binary variables, with a separate variable indicating the time of colonization. Then, e.g. if a binary variable indicates an importation, the associated colonization time is undefined. Hence, different states have different numbers of parameters which is a situation that can be addressed by an RJMCMC. In contrast, we model this with a single variable indicating the time of colonization using the convention that a colonization time exactly equal to the admission time is an importation, and a colonization time exactly equal to the discharge time indicates a never-colonized patient. While this is clearly equivalent to the previous formulation, the number of variables is now fixed. Although these colonization time variables have mixed distributions, with finite probabilities at the end points but distributed continuously in between, the standard Metropolis proposal–acceptance machinery still applies, and so we do not need to resort to an RJMCMC.

2.4 *Simulation*

We evaluated the computational and statistical performance of both discrete- and continuous-time methods on data simulated in continuous time. We used a method similar to that given by Ripley (1987) for generating random birth–death processes. The various possible events—admissions, discharges, colonizations and tests—are assumed to occur at random with rates that can depend on the current state of the system. The time to the next event is generated from an Exponential distribution with rate equal to the sum of the individual event rates. The type of the event is then chosen with probability proportional to the event rates. The simulation program allows specification of the capacity of the unit, the length in days of the study period, the mean length of stay for a patient, the mean occupancy rate of the unit and the rate at which a patient is subjected to a random test in addition to the model parameters ν, λ and ξ.

The output is a list of events specified by the time, the patient involved and the type of event. Although the colonization events are simulated, they are not usually output to the analysis program. There is, however, an option to force the output of the colonization events which we used to compare the estimates made with our methods with the estimates that would be possible if colonizations were observable and, hence, evaluate the information lost due to their unobservability.

We made several sets of simulations to evaluate four aspects of our methods: how the computational time requirements scale with the size of the problem; how well the Markov chains mix; how correlated the simulations from the ergodic distribution are and how discretization can lead to biased estimation. These results cover a range of parameter options, but, given the large number of combinations possible, we have chosen to give an illustrative rather than comprehensive presentation; however, the characteristics shown here are typical of what we have seen in more extensive testing.

For the first three issues, we ran three analysis programs: the discrete-time program discretized by day; the discrete-time program discretized by quarter day and the continuous-time program. In the analysis of statistical bias, we also included a program that used the simulated transmission events directly in order to compare our methods with the ideal, unobtainable, situation. In each of the data sets simulated, we set the mean length of stay for patients to 5 days, the mean occupancy rate to be 80% and the mean number of tests per patient to be 1 every 3 days.

In all of what follows, we define a single iteration to be one round of Metropolis updates to all the augmented data and Gibbs or Metropolis updates, as appropriate, to the parameters. The values of the parameters were output after each iteration.

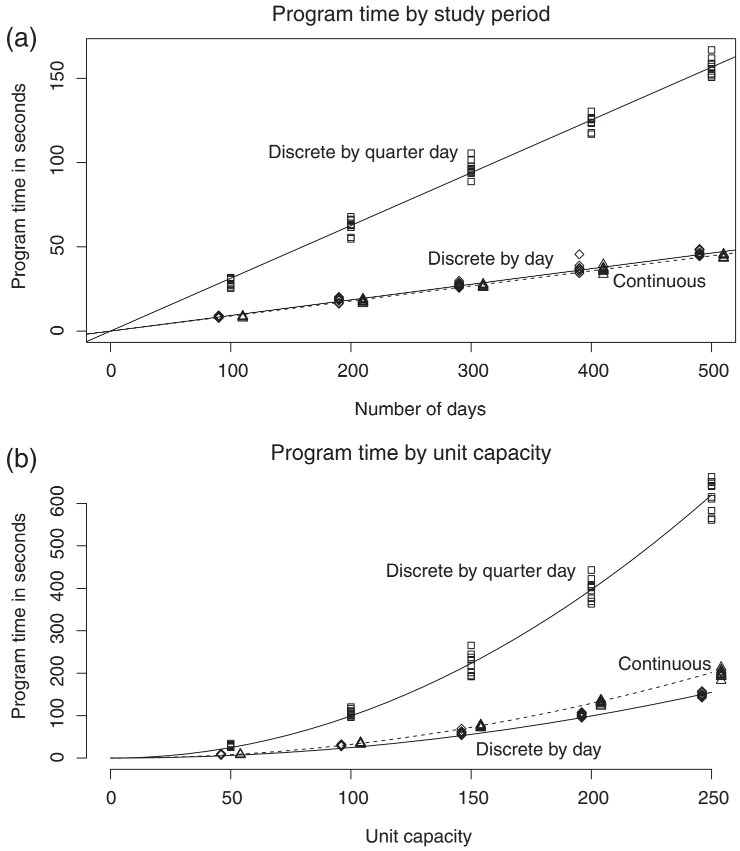

FIG. 2. How the time taken for 1000 iterations of each of 3 analysis programs increases as (a) the study period increases and (b) the unit capacity increases. The best linear and quadratic fits, respectively, to the observed times are also shown with the fits for the continuous-time implementation as dashed lines. The points for the discrete by day and continuous cases are slightly offset horizontally for clarity.

All the simulations and analyses were run on a laptop computer with a 2.4-GHz CPU.

3. Results

3.1 *Simulation study*

3.1.1 *Computational time.* To evaluate the computational time requirements, we first simulated 10 random data sets from a baseline model where $(\lambda, \nu, \xi) = (0.001, 0.1, 0.2)$ for a unit with a 50-bed capacity and a 100-day study period. The times taken by our programs to make 1000 iterations were recorded. We then simulated larger data sets first by increasing the study period in 100-day increments to 500 days, and then by increasing the unit size in 50-bed increments to 250 beds. Each scenario was simulated and analysed 10 times. The analysis times are presented in Fig. 2, and show the expected linear growth with respect to time and quadratic growth with respect to capacity.

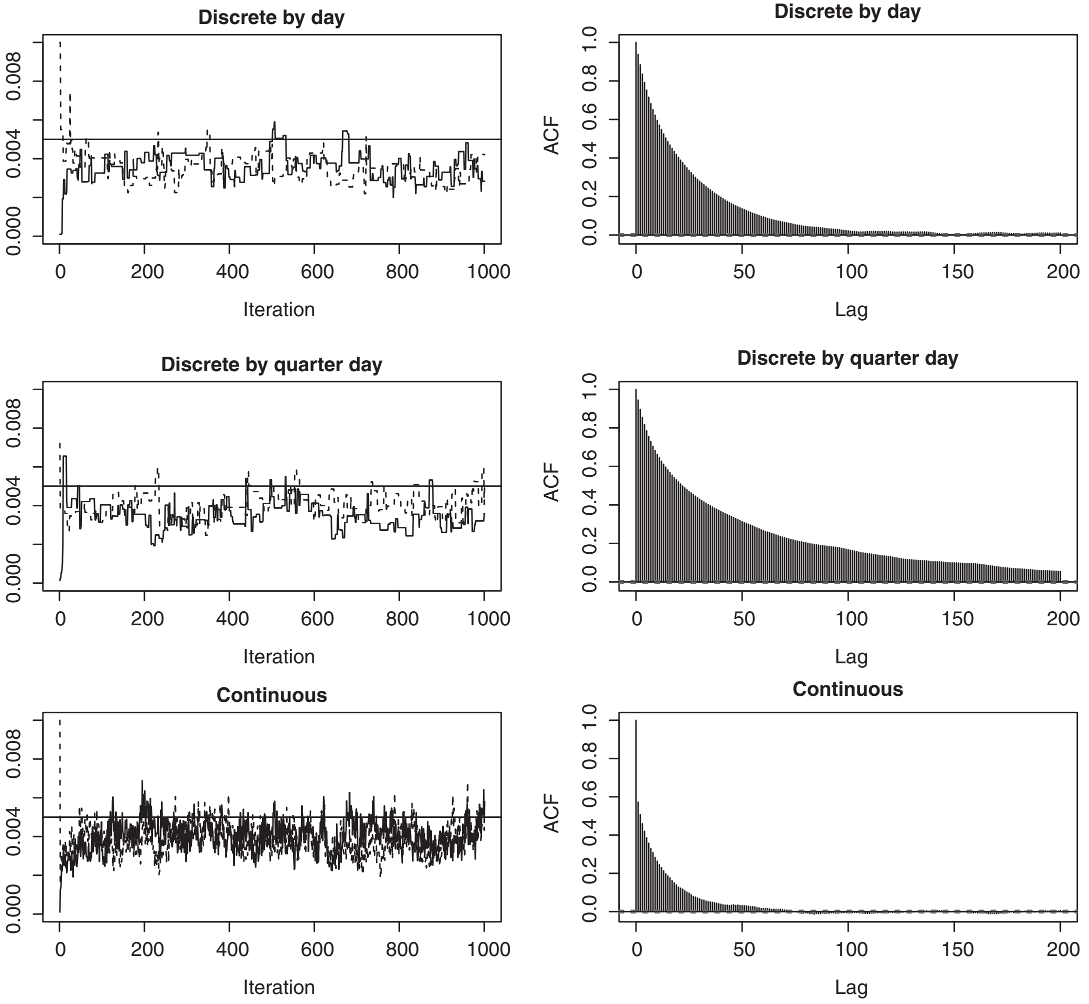

FIG. 3. Markov chain mixing properties. The first column shows the first 1000 sampled transmission rates from three MCMC simulation schemes started from low (solid line) and high (dashed line) parameter values. The horizontal line shows the value used to generate the data set. The second column shows autocorrelation functions to lag 200 for the transmission rates sampled by the MCMC processes estimated from 100,000 samples each.

3.1.2 *MCMC mixing.* In order to illustrate the MCMC mixing properties, we generated a single data set with $(\lambda, \nu, \xi) = (0.005, 0.05, 0.15)$ for a 50-bed unit for 100 days. We then ran 1000 iterations of each analysis method with two different starting points in parameter space: one with high parameter values, $(\lambda, \nu, \xi) = (0.01, 0.5, 0.5)$, and one with low values $(0.0001, 0.005, 0.005)$. Figure 3 shows plots of the 1000 values of the transmission rate sampled from the posterior distribution for each method and each starting point. Each case shows good mixing with rapid convergence regardless of the starting point. The mixing properties for the other parameters were similar.

TABLE 1 *The ESSs for runs of 100,000 correlated samples for parameter values sampled by three MCMC processes*

	Transmission	Importation	False negative
Discrete by day	2067.7	3717.1	13704.2
Discrete by quarter day	1059.2	1202.0	5582.2
Continuous	6296.6	5548.3	16622.0

3.1.3 *Autocorrelation.* The same simulated data set used to examine mixing was also used to compute the autocorrelations between simulations from the ergodic distribution. In this case, we made a single long run of 100,000 iterations from each analysis program from which we estimated autocorrelations out to a lag of 200. The autocorrelations for transmission rate are also given in Fig. 3 and show that the continuous implementation gives lower autocorrelations than the discrete methods. In order to compare relative efficiency from correlated samples, we note that, for a correlated sample of n observations, each with variance σ^2, the variance of the sample mean is given by

$$\frac{\sigma^2}{n^2}\left\{n+2\sum_{i=1}^{n-1}(n-i)f_i\right\},$$

where f_i is the ith autocorrelation. Hence, we take the effective sample size (ESS) as the estimated equivalent number of independent samples given by

$$\mathrm{ESS}=\frac{n^2}{n+2\sum_{i=1}^{l}(n-i)\hat{f}_i}, \tag{22}$$

where $\hat{f}_i$ is the estimated ith autocorrelation and l is the longest lag considered. Table 1 gives the ESS for each of these runs of 100,000 iterations. Note that the ESS differs for each parameter being estimated.

3.1.4 *Statistical bias.* In the course of development and testing, we noted that the discrete by day method tended to underestimate transmission rates and overestimate the importation probabilities. In order to evaluate this more systematically, we simulated 500 separate data sets with $(\lambda, \nu, \xi) = (0.005, 0.05, 0.15)$. We ran each of our estimation programs on each data set for 2100 iterations. The first 100 iterations were in each case discarded and the remaining 2000 were used to estimate the posterior mean for the three parameters. Box and whisker plots for the sampling distributions of these posterior mean parameter estimates are given in Fig. 4. The scenario of sampling distribution for the ideal transmissions observed is also included here for comparison.

3.2 *Comparison with RJMCMC*

We compared our novel continuous MCMC methods with RJMCMC using an RJMCMC program provided by Dr Ben Cooper (B Cooper, personal communication). The RJMCMC program implements a discrete-time approximation to the continuous-time model described above. This particular implementation estimates the transmission rate and importation probability under the assumption that there are no false test results. We used a data set on a 150-bed unit observed for 548 days. The total number

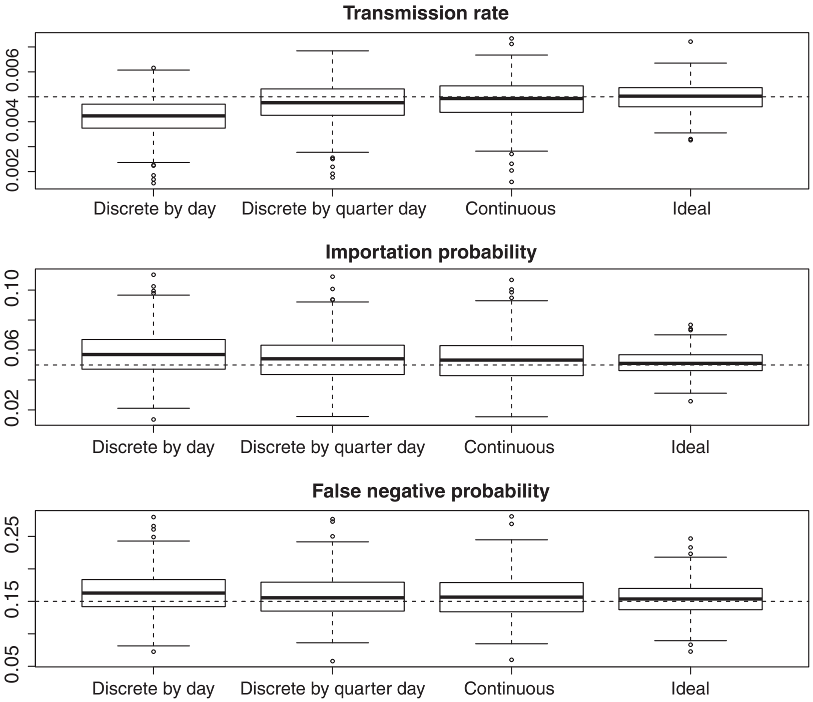

FIG. 4. Box and whisker plots showing the sampling distributions of the mean posterior parameter estimates based on 500 randomly generated data sets, under four analysis models. The 'Ideal' case is when the simulated transmission data are used to obtain the posterior distributions, and is not generally feasible in reality. The dashed horizontal lines show the values used to generate the data in each case.

of patients was 2940. This data set was generated by extending an example data set also provided by Dr Cooper. The data were recorded discretely by day. Both the RJMCMC and our continuous MCMC programs were run for 10,000 s and each printed the time taken, the current value of the transmission rate and the current value of the importation probability after each set of updates.

Figure 5 shows the parameter values outputted in the first 1000 s of each run. As can be seen from the plot for the transmission rate, the RJMCMC takes longer to get to the ergodic distribution and exhibits more long-term correlation. Table 2 gives the posterior mean and standard deviation for each parameter and for each method as estimated from the simulation runs after discarding the values generated in the first 200 s. While the posterior standard deviations are very similar, there are differences in the mean estimates between methods. We attribute this effect again to discretization issues. Inspection of the RJMCMC program code shows that in implementing their continuous model in discrete time, the Cooper group allowed the possibility that an individual could become colonized on their admission day. This contrasts with the approximation that we made in our discrete-time implementation, and hence we observe the opposite effect to that shown in Section 3.1, i.e. compared with continuous analysis, this RJMCMC program slightly favours the transmission rate at the expense of the importation probability.

Again discarding the values generated in the first 200 s, we estimated the autocorrelation functions for each method and each parameter. These are given in Fig. 6. For our method, the autocorrelations for lag greater than about 200 become small. For the RJMCMC the autocorrelations remain high throughout

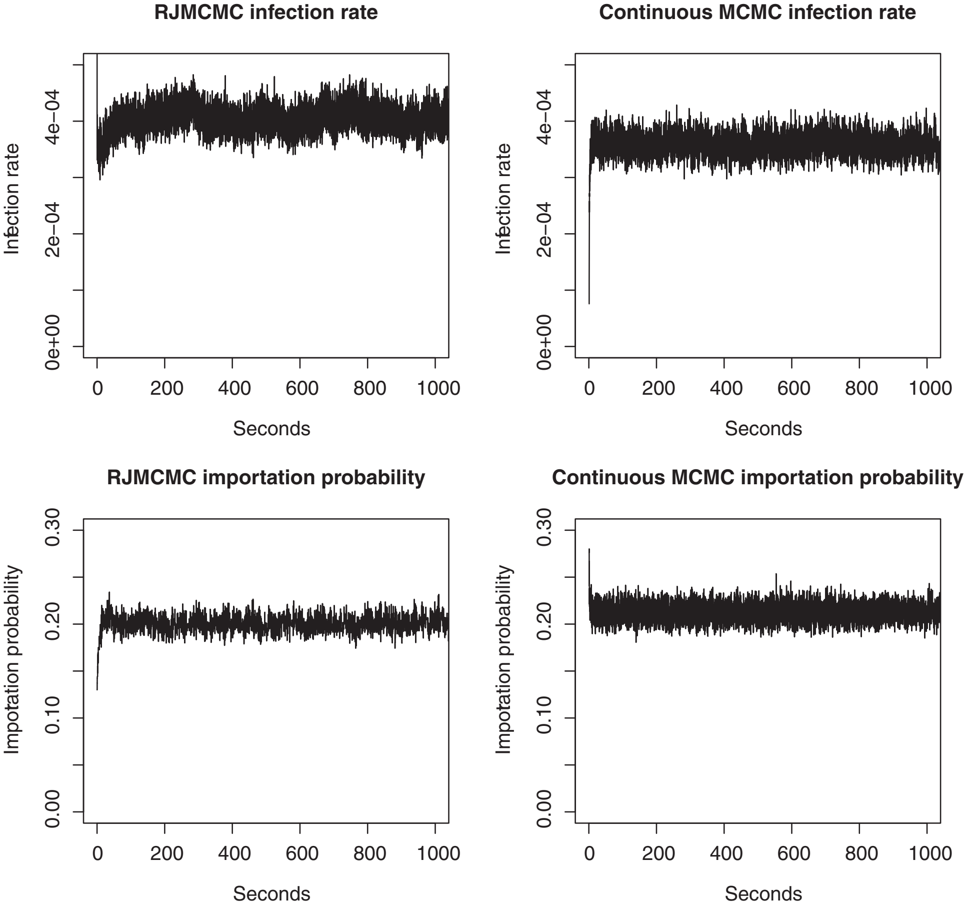

FIG. 5. The sampled transmission rate parameters and importation probabilities for both our continuous MCMC scheme and RJMCMC plotted against the computational time required.

TABLE 2 *First- and second-order properties of the posterior distributions for parameters estimated by RJMCMC and continuous MCMC methods*

	Transmission rate		Importation probability	
	Mean	Standard deviation	Mean	Standard deviation
RJMCMC	0.000407	0.000019	0.2004	0.0082
Continuous MCMC	0.000359	0.000018	0.2127	0.0086

the values shown in the figure and do not become small until approximately lag 15,000 for transmission rate and a lag of 5000 for importation probability. However, the RJMCMC produced approximately 10 times as many samples in the given time: 1,053,675 compared with 111,587. In order to compare the efficiency, therefore, we again use the ESS formula given in (22). Using a maximum lag of 15,000, the

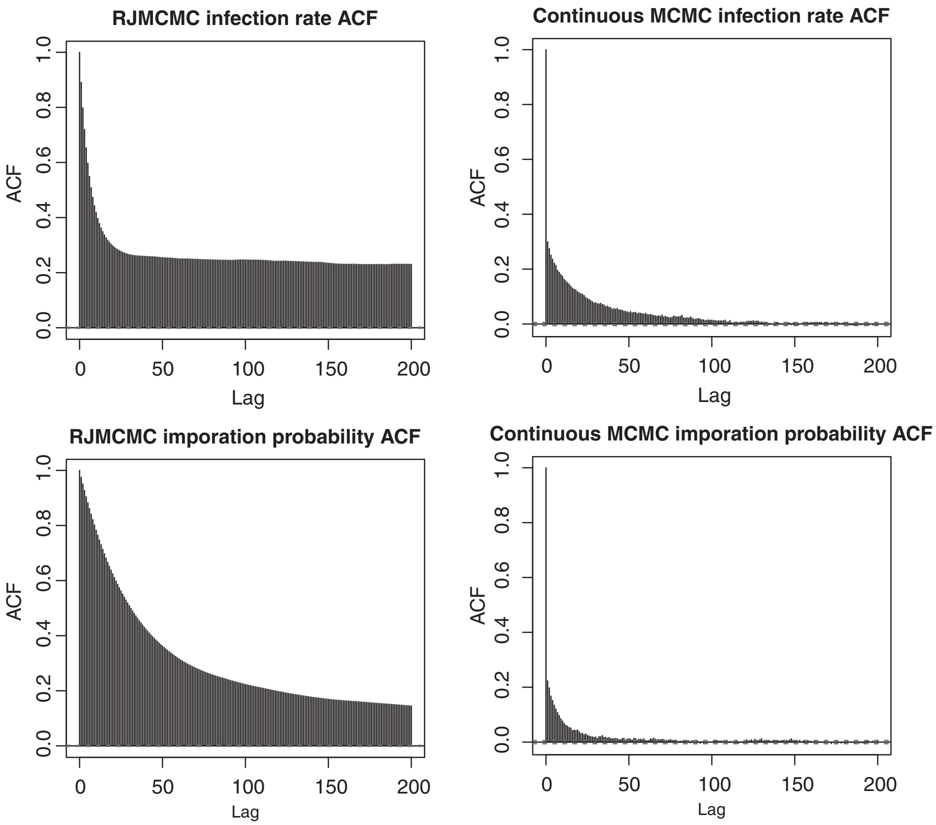

FIG. 6. The autocorrelation profiles for output from both our continuous MCMC methods and the RJMCMC. These were estimated from runs of 10,000 s for each method with the values from the first 200 s being discarded.

ESS for the RJMCMC when estimating the transmission rate was 640.2 while, for our MCMC method, using a maximum lag of 200, it was 7089.5. For importation probability, the difference was less dramatic with the RJMCMC providing an ESS of 4071.2, with a maximum lag of 5000, while our approach gave an ESS of 16475.1, with a maximum lag of 200.

3.3 *Analysis of data from a Veterans Health Administration ICU*

In order to evaluate our programs in a real application, we analysed the incidence of MRSA colonization in a Veterans Health Administration (VA) surgical ICU. Over a 4-year period, the number of patients in the unit ranged from 1 to 18. The VA has a policy of surveillance testing on admission and transfer for every patient which yielded 3171 tests for 3200 patient episodes. Note that this differs from our simulation model which assumes that tests are administered randomly; however, as we condition on the test times, the parameters can be estimated in the same way. Tests administered to the patients when they were not in the ICU were not considered. A limitation of our current model is that patient episodes involving the same person were treated as independent episodes.

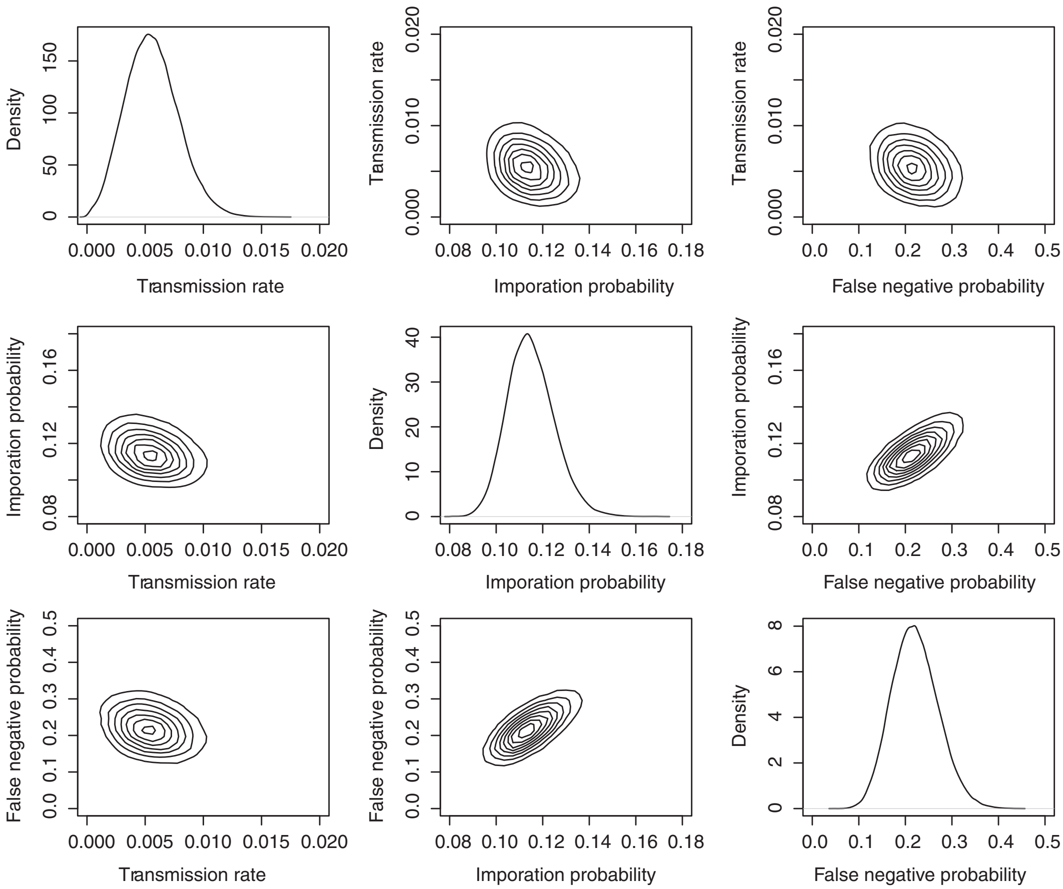

FIG. 7. A matrix of univariate and bivariate parameter marginal posterior distributions obtained from continuous-time analysis of the VA ICU data. Univariate kernel density estimates are shown on the diagonal with pairwise bivariate marginals, shown as contour plots, given in the off diagonal positions.

The data were recorded in continuous time and we used only the continuous-time program in this case. We made a single run of 100,000 iterations to evaluate the parameter posterior distributions. We used an R interface installed on the VA informatic computing infrastructure to access our programs. The run took under 17 min to complete.

The posterior means and 95% credible intervals for the parameters were: transmission rate: 0.0056, (0.0016, 0.0104); importation probability: 0.115, (0.097,0.137); false-negative probability: 0.221, (0.130,0.324). Figure 7 shows the univariate and bivariate posterior marginals for these parameters.

4. Discussion

In this paper, we report the development and evaluation of novel methods to model healthcare-associated transmission of pathogens and, hence, estimate transmission, importation and test sensitivity parameters. We first described a new, more efficient, implementation of a discrete-time approach that allows discretization by arbitrary time interval. We then described a novel method to implement the same underlying model in continuous time. Previous work in this field has been restricted by computational constraints to analyse only small patient populations such as individual ICUs. In our study, we analysed hospital units of up to 250 beds and for up to 500 days using a laptop computer. We showed, both by analysis of the algorithms and by the empirical confirmation in Fig. 2, that the methods scale reliably as a linear function of the study time period and as the square of the unit capacity; so, clearly, analysis of larger data sets is possible.

Section 3.2 suggests that our use of simple Gibbs and Metropolis updates results in far better MCMC mixing properties than the RJMCMC. Roughly speaking, in the example shown, in the same amount of time, the RJMCMC provides 10 times more observations, but our approach provides 10 times more information for the transmission parameter. Perhaps fortuitously, this corresponds closely with the Cooper group's practice of thinning their simulations and using only every 100th iteration made (Cooper *et al.*, 2008). However, to what extent this improvement is directly due to avoiding a reversible jump is not clear. A major factor in making our approach feasible is the extensive exploitation of factorization of terms from the Metropolis acceptance probabilities. For the discrete implementation, the terms remaining in the acceptance probability shown in (8) can be found by expansion from (4). For the continuous case, (21) shows the remaining terms. It is this trick that allows us to make a complete sweep of Metropolis updates to each patient history, conditional on current parameter values and all other patient histories, at each iteration. In contrast, in one iteration the implementation by the Cooper group either proposes an update to a single patient history, or a group of patient histories, and makes complete evaluations of the likelihood to determine the acceptance probability. Even if multiple changes are proposed, these are accepted or rejected as a single block. Exploiting factorization is probably the most significant source of improvement in our methods. It is very likely that the RJMCMC approach could also exploit factorization and individual history updates, but this has not been shown for any specific scheme. It seems that there might well be some advantage in future work that combines the best elements of both existing approaches. For instance, it might be beneficial to tune the proposal distribution to favour colonization events early or late in a patient's hospital stay or to make importation or non-colonization more probable. Such asymmetric proposals can be assessed using Hastings's methods (Hastings, 1970).

If the times of admissions, discharges and test samples are collected and reported in continuous time, then we have substantial reasons for preferring the use of the continuous analysis over a discretized one. First, Fig. 2 shows that the computational requirements of the continuous program are very similar to those needed in an analysis that discretizes by day, and less than those required by finer time discretizations. Second, the form of the likelihood in the continuous model allows MCMC updates to the transmission rate using the Gibbs samples from a Gamma distribution instead of Metropolis updates. Using Metropolis updates generally requires tuning a perturbation parameter in the proposal scheme, and may require making many proposals before a new state is accepted. The state generated by the Gibbs sampler, on the other hand, depends only on the states of the other parameters and not on the previous state of the parameter being updated and so generally allows for more rapid movement between states. This improved mixing behaviour is clearly shown in Fig. 3. Third, Fig. 3 also shows that the correlations between successive simulations is smaller for the continuous model. From Table 1, we conclude that, as a rule of thumb in planning the number of MCMC iterations to make, 10 simulations from the continuous model chain give information equivalent to one independent observation from the same posterior.

Finally, in addition to these computational considerations, the continuous model eliminates the statistical biases in parameter estimates that is seen when using time discretization, as illustrated in Fig. 4.

We have concluded that this bias is a direct result of time discretization for the following reason: if, in an imputation of the complete data, an individual is colonized on the first day of their stay, this is interpreted in our model as an importation and so contributes to the estimate of ν rather than to λ. In circumstances where the probability of a colonization on the first day is small, the effect of this interpretation is negligible, however, if the transmission rate is high, or if the unit is large, there is a significant chance of the occurrence of a transmission which is misinterpreted as an importation. Empirical evaluations we have made (data not shown) support the observation that the bias is seen to be greatest when λ and the unit size are large. Figure 4 also supports this conclusion as the bias is mitigated when a finer time discretization is used, and eliminated in the continuous-time model. If data are not collected in continuous time but is, e.g. given as daily reports, whether it is better to analyse such data using discrete methods or, by assigning events and arbitrary time of day, continuous methods is an open question. It may also be possible to address discretization bias with different modelling assumptions, but since we anticipate that electronic health records will facilitate continuous collection of data, we have not pursued either of these issues. We also note in passing that rules of thumb to distinguish between importations and hospital-associated transmissions based on a positive test in, say, the first 2 days of hospitalization will also result in a similar underestimation of transmission rates.

The above evaluation of the statistical properties of our estimates was made possible by the speed improvements in our programs compared with previous implementations. These programs are straightforward to use and available for general use. Their inclusion in an R package may be a particularly helpful aid to accessibility, and we intend to develop this package to include further extensions.

The results of the VA ICU data analysis yields very plausible parameter estimates, and illustrates correlations between them. For instance, since in any reconstruction of the augmented data the observation of a colonized individual will be attributed to either an importation or a transmission, we see a negative correlation in the posterior bivariate marginal for the two corresponding parameters in Fig. 7. Similarly, because of the surveillance testing of new entries to the unit, a high false-negative probability allows more colonizations to be attributed to importation, and hence we see a positive correlation between these parameters.

There are numerous possible model extensions, including those previously implemented (Forrester *et al.*, 2007; Cooper *et al.*, 2008; Kypraios *et al.*, 2010). Because transmission events are thought to be facilitated by nurses, who generally stay on one unit, and doctors, who travel across units, it might be of interest to model separate transmission rates between and within units. Additionally, a model that explicitly incorporates the test type, surveillance versus clinical culture, could be developed to better understand the implications of these testing events and a related model could consider modelling symptomatic and asymptomatic colonization.

One of the limitations in this work is that we are explicitly interested in pathogens that typically fit an SI framework. That is, we have assumed that once patients become colonized, they remain colonized for the duration of their hospitalization. This model is not well suited for situations in which there is recovery from the infection. Extensions to other pathogens when there is a latent period, when there is a recovery time or when there is a period of immunity following recovery, are likely to be more complex and computationally more demanding as the number of possible states, and transitions between states, increases. However, provided the Markov assumption, that future events depend only on the current state and not the previous history, is reasonable, these extensions should be tractable by our methods.

We anticipate that the consequences of these contributions, together with the adoption of electronic health records, will drive the field of hospital transmission modelling forwards at an increased pace and in new directions, including possible use by hospital policy makers. Augmented data transmission models for hospitals attempt to mirror most closely the true transmission process behind the observed data. Basic and even complex hierarchical statistical models make assumptions that overly simplify the process behind the data. Estimates of parameters and intervention effects, as well as their standard errors, from models that make such simplifying assumptions may be inefficient and biased.

4.1 *Software*

The programs for analysis and simulation were written in Java and C++ and are available as stand-alone programs or as R packages available from the Comprehensive R Archive Network (CRAN). The discrete-time analysis program is called `AnInfection` and is available as the R package `transnet`. The simulation program in Java is called `SimInfection` and is included in `transnet`. The continuous-time model was implemented in C++ and is available as the `transmission` package on CRAN; a stand-alone version is also available from the authors. A C++ version of the simulation program is also included within the `transmission` package.

Acknowledgements

We would like to thank Dr Makoto Jones, MD. MS. for his expert opinion on the plausibility of the model and parameter estimates. We would also like to thank Dr Ben Cooper for allowing us to use the programs developed by his group.

Funding

The research reported here was supported by the Department of Veterans Affairs, Veterans Health Administration, Health Services Research and Development Service project REA-08-264. The views expressed in this article are those of the authors and do not necessarily represent the views of the Department of Veterans Affairs.

This work was supported using resources and facilities at the VA Salt Lake City Health Care System with funding support from the Centers for Disease Control and Prevention's Epicenters Program 07FED706506.

This investigation was supported by the University of Utah Study Design and Biostatistics Center, with funding in part from the National Center for Research Resources and the National Center for Advancing Translational Sciences, National Institutes of Health, through Grant 8UL1TR000105 (formerly UL1RR025764).

References

Christopher, S., Verghis, R. M., Antonisamy, B., Sowmyanarayanan, T. V., Brahmadathan, K. N., Kang, G. & Cooper, B. S. (2011) Transmission dynamics of Methicillin-resistant Staphylococcus Aureus in a medical intensive care unit in India. *Public Libr. Sci., One*, **6**, 7.

Cooper, B. & Lipsitch, M. (2004) The analysis of hospital infection using hidden Markov models. *Biostatistics*, **5**, 223–237.

Cooper, B. S., Medley, G. F., Bradley, S. J. & Scott, G. M. (2008) An augmented data method for the analysis of nosocomial infection data. *Am. J. Epidemiol.*, **168**, 548–557.

Drovandi, C. C. & Pettitt, A. N. (2008) Multivariate Markov process models for the transmission of Methicillin-resistant *Staphylococcus aureus* in a hospital ward. *Biometrics*, **64**, 851–859.

Forrester, M. L., Pettitt, A. N. & Gibson, G. J. (2007) Bayesian inference of hospital-acquired infectious diseases and control measures given imperfect surveillance data. *Biostatistics*, **8**, 383–401.

Geman, S. & Geman, D. (1984) Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Trans. Pattern Anal. Mach. Intell.*, **45**, 721–741.

Gibson, G. J., Kleczkowski, A. & Gilligan, C. A. (2004) Bayesian analysis of botanical epidemics using stochastic compartmental models. *Proc. Natl Acad Sci. USA*, **101**, 12120–12124.

Green, P. J. (1995) Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, **82**, 711–732.

Hastings, W. K. (1970) Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, **57**, 97–109.

Kypraios, T., O'Neill, P. D., Huang, S. S., Rifas-Shiman, S. L. & Cooper, B. S. (2010) Assessing the role of undetected colonization and isolation precautions in reducing methicillin-resistant staphylococcus aureus transmission in intensive care units. *BMC Infect. Dis.*, **10**, 29.

McBryde, E. S., Pettitt, A. N., Cooper, B. S. & McElwain, D. L. S. (2007) Characterizing an outbreak of vancomycin-resistent enterococci using hidden Markov models. *J. R. Soc. Interface*, **4**, 745–754.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N. & Teller, A. H. (1953) Equations of state calculations by fast computing machines. *J. Chem. Phys.*, **21**, 1087–1091.

O'Neill, P. D. (2010) Introduction and snapshot review: relating infectious disease transmission models to data. *Stat. Med.*, **29**, 2069–2077.

Ripley, B. D. (1987) *Stochastic Simulation*. New York: John Wiley and Sons.