View metadata, citation and similar papers at core.ac.uk
brought to you by CORE
provided by Open Research Exeter

Statistics and Computing manuscript No.
(will be inserted by the editor)

MCMC implementation for Bayesian hidden semi-Markov models with illustrative applications

Theodoros Economou · Trevor C. Bailey · Zoran Kapelan

Received: date / Accepted: date

Abstract Hidden Markov models (HMMs) are flexible, well-established models useful in a diverse range of applications. However, one potential limitation of such models lies in their inability to explicitly structure the holding times of each hidden state. Hidden semi-Markov models (HSMMs) are more useful in the latter respect as they incorporate additional temporal structure by explicit modelling of the holding times. However, HSMMs have generally received less attention in the literature, mainly due to their intensive computational requirements. Here a Bayesian implementation of HSMMs is presented. Recursive algorithms are proposed in conjunction with Metropolis-Hastings in such a way as to avoid sampling from the distribution of the hidden state sequence in the MCMC sampler. This provides a computationally tractable estimation framework for HSMMs avoiding the limitations associated with the conventional EM algorithm regarding model flexibility. Performance of the proposed implementation is demonstrated through simulation experiments as well as an illustrative application relating to recurrent failures in a network of underground water pipes where random effects are also included into the HSMM to allow for pipe heterogeneity.

Keywords HSMM · random effects · MCMC · recursive algorithms · Bayesian model · water pipes.

1 Introduction and Background

First introduced in speech recognition (see Rabiner (1989) for a review paper), hidden Markov models (HMMs) have found increasing use in various applications areas. However, despite the usefulness of HMMs, the state holding times are implicitly geometrically distributed, and this may constitute a potential limitation (see Guedon (2003) and Tokdar et al. (2010) for some examples).

A natural extension of the HMM is the hidden semi-Markov model (HSMM) where holding time distributions are defined explicitly while retaining the Markovian dependency structure. However, even though the transition between HMMs and HSMMs is mathematically straightforward, the complexity of the model increases considerably. Conceptually, one needs to consider all possible state sequences at the same time as all possible holding times for each state. This renders HSMMs computationally intensive and as a result, the literature on HSMM applications is considerably smaller than that relating to HMMs.

In this paper, a Bayesian formulation of HSMMs is considered, along with associated methods for MCMC sampling. The proposed approach provides a computationally efficient estimation framework for HSMMs at the same time as allowing for further flexibility, for instance the inclusion of random effects. In this section, background on HSMMs is provided while in Section 2 the model formulation is presented. In Section 3 a recursive method for likelihood calculation is presented as well as details on MCMC model estimation. Simulation results and an application to modelling recurrent failures in underground water pipes are presented in Section 4. Finally in Section 5, a summary is provided along with conclusions.

A few examples of applications that have found use for HMMs include: climate modelling where HMMs are used for downscaling precipitation forecasts (Bellone et al., 2000), in economics HMMs are used to capture non-stationarity in share price return series (Rydén et al., 1998), in medical applications to model disease progression in cancer studies (Kozumi, 2000; Jouyaux et al., 2000), in genetics (Yau et al.,

T. Economou, T. C. Bailey and Z. Kapelan
College of Engineering Mathematics and Physical Sciences
Uninevristy of Exeter
Tel.: +44-1392725280
E-mail: t.economou@ex.ac.uk

2011), in mechanical engineering (Jardine et al., 2006) and several others.

HSMMs were first introduced by Ferguson (1980) with an application in speech recognition (see Guedon (1992) for a thorough review of HMMs and HSMMs in speech processing applications). Since then, the popularity of HSMMs has increased in many disciples such as: computer science (e.g. Levinson (1986)); engineering (e.g. Dong and He (2007)); climate (e.g. Sansom and Thomson (2001)); finance (e.g. Bulla and Bulla (2006)); computational biology (e.g. Schmidler et al. (2000)) and many more (see Yu (2010) for a more detailed list).

The traditional tool for fitting HMMs and HSMMs is the EM algorithm where recursive (forward-backward) algorithms are used for calculating the otherwise computationally intensive likelihood of these models. These algorithms, discussed in more detail in Section 3, make use of the short term memory and discrete nature of the latent chains to efficiently evaluate the likelihood. Guedon (2003) uses these recursive algorithms along with the EM algorithm for estimating HSMMs and more recently, Bulla et al. (2010) have written an R package (R Development Core Team, 2012) which implements HSMMs using the approach in Guedon (2003). The limitation in using these techniques for estimating HMMs and HSMMs lies in model flexibility whereas a Bayesian approach to model fitting enables the full potential of these latent structure models.

HMMs and HSMMs fit naturally into the Bayesian context since these are essentially hierarchical models where the data are assumed to follow a suitably chosen probabilistic process (the conditional model) given the latent Markov chain. In particular, HMMs may be viewed as random effect models where the unobserved random quantities are instances of the hidden chain. This fact was used in Chib (1996) who considered a Gibbs sampler for estimating HMMs by deriving and directly sampling from the conditional distribution of the hidden state sequence, instead of updating the joint distribution of the chain state and the data at each time step. Other work involving Bayesian HMMs includes Scott (2002), Guha et al. (2008) and Yau et al. (2011). Because of the associated computational difficulties involved with MCMC, few authors have considered a Bayesian approach to HSMMs.

The method in Chib (1996) relies on the Markov structure of the model and is thus not applicable for HSMMs where the waiting times are not geometric. This point was addressed in Tokdar et al. (2010) where Chib's method was adapted and a Gibbs sampler was used to fit a two-state HSMM. However this approach requires derivation of the full conditionals for each unknown quantity given the data and all other unknown quantities including the hidden state sequence and in practice, this limits the degree of complexity that can be considered in the model formulation.

In the following sections in this paper, we instead advocate the use of recursive algorithms along with Metropolis-Hastings for estimation of HSMMs. This avoids sampling from the conditional distribution of the hidden state sequence by using the joint distribution of the hidden states and the data in the likelihood. This approach provides a computationally efficient estimation framework whilst also allowing for considerable flexibility in model formulation including for example, use of random effects. Note however, that there has been recent work by Dewar et al. (2012) and Johnson and Willsky (2012) on Bayesian HSMMs using computationally efficient Gibbs samplers. We refer to these papers later, in sections 3 and 5.

2 Model formulation

This section focuses on formulating the likelihood of a general HSMM in time. The latent semi-Markov chain is discrete (in time) and the conditional model is defined through a random variable $Y(t)$ given the state of the chain at t.

2.1 Hidden Markov and Hidden semi-Markov Models

In a Hidden Markov model (HMM), the conditional model assumed for the observed data depends on an underlying Markov chain with discrete state space $S \in \{1,\dots,M\}$, defined by an initial distribution $\pi = (\pi(1),\dots,\pi(M))$ and a transition matrix $P = \{p_{i,j}\}$ where $p_{i,j} = \Pr(S_{T_k} = j | S_{T_{k-1}} = i)$ and $\sum_j p_{i,j} = 1$. Note that $T_k, k = 0,1,2,\dots$ are the discrete time steps of the chain and S_{T_k} is the state of the chain at T_k. For a discrete Markov chain, the length of time τ that a state i remains in, is implicitly geometrically distributed: $h_i(\tau) = (p_{ii})^{\tau-1}(1-p_{ii})$ where $h_i(\tau)$ is the holding time distribution.

The hidden semi-Markov model (HSMM) allows explicit specification of the holding time distributions. A discrete semi-Markov chain can be defined by an initial distribution π, a transition matrix $P = \{p_{i,j}\}$ where $p_{j,j} = 0$ and $\sum_j p_{i,j} = 1$, and a set of holding time distributions for each state $\{h_1(\tau;\phi_1), h_2(\tau;\phi_2),\dots,h_M(\tau;\phi_M)\}$ with associated parameters $\phi = (\phi_1,\dots,\phi_M)$. Self transitions are not allowed ($p_{j,j}$=0) as this conflicts with the definition of holding times between state changes.

Suppose now that a semi-Markov chain has been observed in the interval $[T_0, T_{end}]$ and that $Q-1$ state changes have occurred with holding time intervals $(\tau_1,\tau_2,\dots,\tau_Q)$. The likelihood, assuming right censoring at T_{end} is:

$$L_{MC}(S_1,\dots,S_Q,\tau_1,\dots,\tau_Q;\pi,P,\phi) = \pi(S_1)\prod_{k=1}^{Q-1} h_{S_k}(\tau_k;\phi_k)p_{S_k,S_{k+1}} \times \Pr(\tau > \tau_Q;\phi_Q). \quad (1)$$

where $\Pr(\tau > \tau_Q; \phi_Q)$ is the survival function of $h_{S_Q}(\tau_Q; \phi_Q)$. In the non-censored case, i.e. a state change took place at T_{end}, then Q state changes have occurred and the likelihood is given by $\pi(S_1)\prod_{k=1}^{Q} h_{S_k}(\tau_k;\phi_k)p_{S_k,S_{k+1}}$. It is assumed that once the chain enters a state, it will stay there for at least one time step. This implies that any distribution chosen to characterise $h_S()$ must be zero-truncated.

For a more general semi-Markov chain, one could allow for $M(M-1)$ holding time distributions $h_{i,j}(\tau;\phi_{i,j})$ where $i \neq j$. Then the holding time distributions depend on the previous state. This generalisation is straightforward and the formulations presented subsequently can easily be adjusted to satisfy this.

2.2 Semi-Markov modulated models

Consider a general random variable $Y(t)$, where t is time, observed in some arbitrary time period $[T_0, T_{end}]$ at times $T_0 \leq t_1, t_2, \ldots, t_n \leq T_{end}$ to obtain observations $y(t_1), \ldots, y(t_n)$. Let $f(Y(t)|\theta_{S_t}, S_t)$ be the probability model for $Y(t)$ with parameters θ_{S_t}, where for the moment we assume that S_t is not random. Assuming independence between each $Y(t)$, the joint likelihood of the observations is $\prod_{i=1}^{n} f(y(t_i)|\theta_{S_{t_i}}, S_{t_i})$. For notational convenience, suppose that for time interval τ, $\mathbf{y}(\tau) = \{y(t_j)\}$ for all $t_j \in \tau$, so that

$$L(\mathbf{y}(\tau)|\theta_{S_t}, S_t) = \prod_{j:t_j \in \tau} f\left(y(t_j)|\theta_{S_{t_j}}, S_{t_j}\right) \tag{2}$$

is the likelihood contribution of the data in time interval τ.

Now suppose that a semi-Markov chain S_t is the underlying process driving the conditional model $f(Y(t)|\theta_{S_t}, S_t)$. Given that the chain was observed, the likelihood of this semi-Markov modulated process is formulated by combining the (now) conditional likelihood of the observations in Eqn (2) and the likelihood of the chain in Eqn (1):

$$\begin{aligned} L_{SMM}(D;\Theta) = \pi(S_1) \prod_{k=1}^{Q-1} h_{S_k}(\tau_k;\phi_k) L(\mathbf{y}(\tau_k)|\theta_{S_k}, S_k) p_{S_k,S_{k+1}} \\ \times \Pr(\tau > \tau_Q;\phi_Q) L(\mathbf{y}(\tau_Q)|\theta_{S_Q}, S_Q) \end{aligned} \tag{3}$$

where $D = (T_0, y(t_1), \ldots, y(t_n), T_{end}; S_1, \ldots, S_Q, \tau_1, \ldots, \tau_Q)$ and $\Theta = (\theta_S, \pi, P, \phi)$. Note that we formulate the likelihood for the (more realistic) case of right-censored data and that the modification to the likelihood is trivial (see section 2.1) for the case where a state change occurred at T_{end}.

Because the likelihood in Eqn (3) depends on having observed the chain, to formulate the HSMM likelihood, Eqn (3) needs to be summed over all possible states $S \in (1, \ldots, M)$ and all possible time intervals $\tau \in (1, 2, \ldots, T_{end})$:

$$\begin{aligned} &L_{HSMM}(T_0, y(t_1), \ldots, y(t_n), T_{end}; \Theta) = \\ &\sum_{\tau_1 + \cdots + \tau_Q = T_{end}} \sum_{S_1=1}^{M} \cdots \sum_{S_Q=1}^{M} L_{SMM}(D;\Theta). \end{aligned} \tag{4}$$

The observed data are just the observed values at each occurrence $y(t_j)$ and the bounds of the observation period $[T_0, T_{end}]$.

The model presented here is one which allows jumps to and from a number of parallel ongoing processes $Y(t)$, with the jumps being controlled by the hidden chain. The Markovian structure implicitly introduces correlation between observations. Note that a specific model for the $Y(t)$ has not been specified other than making the assumption of independence. Note also that an upper limit has not been imposed on the holding time distributions so it is theoretically possible for a particular state to occupy the whole observation period.

3 Bayesian model implementation

3.1 Discretisation

The evaluation of the likelihood in Eqn (4) is computationally prohibitive for any reasonable length of observation period $[T_0, T_{end}]$ and number of latent states. We show in this section how recursive algorithms analogous to those used in HMMs (Baum et al., 1970) may be used to overcome that problem.

The recursive algorithms in HMMs depend on the latent chain being discrete in time, where in each time step the joint distribution of chain and data is calculated. To use these algorithms in HSMMs, we need to conceptually 'discretise' time in steps, rather than work with holding time intervals. The holding time distributions of HSMMs considered here are discrete, so it is easy to do so for the latent process. However, it is important that the conditional model $f(Y(t)|\theta_s, S)$ for the observations has conditionally orthogonal increments given the state S. This is possible where the random variable $Y(t)$ is either independent of time or it is a stochastic temporal process with independent increments (i.e. for any $t_1 < t_2 < \cdots < t_n$, $Y(t_2) - Y(t_1), Y(t_3) - Y(t_2), \ldots, Y(t_n) - Y(t_{n-1})$ are independent).

Suppose the observation period is divided in equal time steps $T_0, T_1, \ldots, T_{end}$. The 'discretised' version of a HSMM is a process which starts at T_0 with probability $\pi(S_{T_0})$, then S_{T_0} is held for at least one time step and the observed data at T_1 are the events that occurred in $(T_0, T_1]$. The process will either keep holding S_{T_0} until T_2 or enter another state with some probability taken from the appropriate entry of P, and hold that for at least one time step. The rest follows accordingly. Note that the discrete time steps of the chain $T_0, T_1, \ldots$ are deliberately different from the time steps $t_1, t_2, \ldots$ of the observations for the sake of generality. The two may often be the same, however there may be cases where the time steps of the chain are larger, e.g. hidden chain captures monthly effects on data recorded daily.

3.2 Recursive algorithms - Forward

A forward variable $v_T(j)$ is considered sequentially at each discrete time step $T = T_1, T_2, \ldots, T_{end}$ (Rabiner, 1989) where

$$v_T(j) = \Pr(\text{data up to } T \text{ and chain exits } S_T = j),$$

i.e. the joint probability of the data up to T and the chain transitioning out of state j at time step T, meaning that the chain occupies state j at T but jumps to state $i \neq j$ at $T+1$. As with HMMs, $v_T(j)$ can be computed recursively:

$$\begin{aligned} v_{T_1}(j) &= \pi(j) h_j(1;\phi_j) L(\mathbf{y}(\tau_{0,1})|\theta_j, S=j) \\ v_{T_k}(j) &= \pi(j) h_j(k;\phi_j) L(\mathbf{y}(\tau_{0,k})|\theta_j, S=j) \\ &\quad + \sum_{m=1}^{k-1} \sum_{i=1}^{M} v_{T_m}(i) p_{i,j} h_j(k-m;\phi_j) L(\mathbf{y}(\tau_{m,k})|\theta_j, S=j), \end{aligned} \tag{5}$$

where $\tau_{i,j} = [T_i, T_j]$. Summing the last variable $v_{T_{end}}(j)$ over all states gives the likelihood in Eqn (4). Note that the complexity of the forward algorithm for HSMMs is $O(T_{end}^2)$ which is significantly more than the $O(T_{end})$ complexity for HMMs. In practice, it may be sensible to restrict the support of the holding time distributions (Yu, 2010) especially if there are physical arguments as to how long a state can be occupied. The holding time distributions are then truncated which in turn reduces the complexity of the forward algorithm but restricts flexibility. Dewar et al. (2012) propose a Bayesian implementation of HSMMs where they introduce a method which dynamically truncates the holding time distributions to increase efficiency. The method theoretically allows a state to occupy the whole observation period but in practice it is more efficient, using carefully chosen auxiliary variables to restrict the outermost summation in Eqn (5).

Forward variables present an elegant way of evaluating the joint distribution of the data and the hidden chain. However, multiplications of probabilities are involved and these rapidly get smaller, leading to potential computer underflow. An efficient way to prevent underflow (see Devijver (1985) and references therein) is by scaling the forward probabilities at each time step. The idea is to work with the conditional distribution of the states (which sums to 1) instead of the joint distribution of the states and the data. However, care is needed if the intention is to evaluate the likelihood and one needs to keep track of the scaling factors until the last time step.

In HMMs, one may scale at each time step and simply "re-scale" at the last time step to obtain the likelihood. In HSMMs this is not possible since at each new time step, an extra term is introduced which does not depend on $v_{T-1}(j)$. This term, given in Eqn (5), is the probability that a state has been held from the start of the observation period. Because these extra terms do not contain any scaled components, re-scaling simply at the last time step to obtain the likelihood is invalid. It is therefore necessary to "re-scale before scaling" at each step in order to calculate the HSMM likelihood.

Scott (2002) gives a matrix representation of the forward (and backward) algorithm in HMMs which aids in better understanding these algorithms. A forward matrix A_T is defined at each step T whose $(i,j)^{th}$ element is the probability of occupying state j at T given that: state i was occupied at $T-1$ and the data up to T. Summing the rows of these matrices provides the necessary terms for recursion. Each A_T is scaled so that all elements sum to 1 where the scaling factor is the joint likelihood (Eqn 4) up to time T. However, to keep track of the likelihood until T_{end}, at each new step T the scaling factor at $T-1$ needs to be multiplied back in A_T before scaling it. This is what we mean by "re-scale before scaling". The re-scaling occurs after summations to avoid underflow. Here, we formulate the HSMM forward recursion similarly performing as many summations as possible before re-scaling.

To describe the forward recursion, we first define several quantities. Define forward matrices $A_T = \{a_{T,i,j}\}$ where

$$a_{T,i,j} = \Pr(S_{T-1} = i \text{ and chain exits } S_T = j | \text{data up to } T)$$

is the probability of exiting state j at T given being in state i at $T-1$, conditional on the data up to T. Summing the rows of this matrix gives vectors

$$\alpha_T(j) = \Pr(\text{chain exits } S_T = j | \text{data up to T}).$$

Note that $v_T(j)$ and $\alpha_T(j)$ are related - the latter is a scaled version of the former, so that its elements add to 1. Clearly, the off-diagonal terms of A_T are 'easier' to work with since they indicate a state change and their calculation is similar as in HMMs with the exception of having to include the holding time probability of 1 time step (from $T-1$ to T). The diagonal entries however, are more complicated since they reflect exiting state j at T given state j at $T-1$.

Define quantity $\delta_{i,j,T}$ where $\delta_{i,i,T} = 0$ and:

$$\delta_{i,j,T} = \Pr(S_{T-1} = i \text{ and chain exits } S_T = j | \text{data up to } T),$$

which are the off-diagonal $(i,j)^{\text{th}}$ entries of A_T. $\delta_{i,j,T}$ may be calculated recursively using $\alpha_T(j)$ as in HMMs.

Further, define quantities ξ_j and γ_j which make up the diagonal entries of A_T:

$$\begin{aligned} \xi_T(j) &= \Pr(S_{T-1} = j \text{ and chain exits } S_T = j \text{ and at least one} \\ &\qquad \text{state change has occurred before } T-1 | \text{data up to } T), \\ \gamma_T(j) &= \Pr(S_{T-1} = j \text{ and chain exits } S_T = j \text{ and no state} \\ &\qquad \text{change occurred before } T-1 | \text{data up to } T). \end{aligned}$$

$\gamma_T(j)$ relates to holding state j from the start (i.e. from T_0) and cannot be calculated sequentially like $\xi_T(j)$.

Given the time-independence (or independent increments) assumption of the conditional model $f(Y(t)|\theta_S, S)$, we can increase computational efficiency by pre-calculating vectors:

$$F_j(k) = L(\mathbf{y}(\tau_{k-1,k})|\theta_j, S=j) \quad \text{for } k = 1, 2, \ldots, T_{end},$$

relating to the conditional likelihood contributions at each time step. In addition, we pre-calculate vectors whose elements correspond to ratios of probabilities from holding time distributions:

$$H_j = \left(\frac{h_j(2;\phi_j)}{h_j(1;\phi_j)}, \frac{h_j(3;\phi_j)}{h_j(2;\phi_j)}, \ldots, \frac{h_j(T_{end}-T_0;\phi_j)}{h_j(T_{end}-T_0-1;\phi_j)} \right)$$

for $j = 1,2,\ldots,M$, which will increase efficiency when calculating the diagonals of A_T. Note that each H_j has length $(T_{end}-T_0-1)$. At each time step T, each $\xi_{T-T_a}(j)$ with $T_a = 1,2,\ldots,T-1$ can be multiplied with the appropriate entries of F_j and H_j, to accumulate information on the probability of holding state j up to T given that at least one state change has occurred.

Let the likelihood of the HSMM at time T be denoted by ℓ_T. Then, the forward algorithm for $T = T_1,\ldots,T_{end}$ is as follows:

$$\begin{aligned}
T_1: \quad \gamma_{T_1}(j) &= v_{T_1}(j) = \pi(j)h_j(1;\phi_j)F_j(1),\\
\xi_{T_1}(j) &= \delta_{i,j,T_1} = 0,\\
\ell_{T_1} &= \sum_{j=1}^{M} \gamma_{T_1}(j), \quad \alpha_{T_1}(j) = \gamma_{T_1}(j)/\ell_{T_1},\\
\{a_{T_1,j,j}\} &= \alpha_{T_1}(j) \text{ and } a_{T_1,i,j} = 0.\\
T_2: \quad \gamma_{T_2}(j) &= \gamma_{T_1}(j)H_j(1)F_j(2),\\
\delta_{i,j,T_2} &= \alpha_{T_1}(i)p_{i,j}h_j(1;\phi_j)F_j(2),\\
\xi_{T_2}(j) &= \left[\sum_{i\neq j} \delta_{i,j,T_2}\right] \times \ell_{T_1} \text{ (re-scale)},\\
v_{T_2}(j) &= \gamma_{T_2}(j) + \xi_{T_2}(j),\\
\ell_{T_2} &= \sum_{j=1}^{M} v_{T_2}(j), \quad \alpha_{T_2}(j) = v_{T_2}(j)/\ell_{T_2} \text{ (scale)},\\
\{a_{T_2,j,j}\} &= \gamma_{T_2}(j) \text{ and } a_{T_1,i,j} = \ell_{T_1}\delta_{i,j,T_2}.\\
T_N \geq T_3: \quad \gamma_{T_N}(j) &= \gamma_{T_{N-1}}(j)H_j(N-1)F_j(N),\\
\delta_{i,j,T_N} &= \alpha_{T_{N-1}}(i)p_{i,j}h_j(1;\phi_j)F_j(N),\\
\xi_{T_k}(j) &= \xi_{T_k}(j)H_j(N-k)F_j(N) \text{ for } k = 2,\ldots,N-1,\\
\xi_{T_N}(j) &= \left[\sum_{i\neq j} \delta_{i,j,T_N}\right] \times \ell_{T_{N-1}},\\
v_{T_N}(j) &= \gamma_{T_N}(j) + \sum_{u=2}^{T_{N-1}} \xi_u(j) + \xi_{T_N}(j),\\
\ell_{T_N} &= \sum_{j=1}^{M} v_{T_N}(j), \alpha_{T_N}(j) = v_{T_N}(j)/\ell_{T_N},\\
\{a_{T_2,j,j}\} &= \frac{1}{\ell_{T_N}}\left(\gamma_{T_N}(j) + \sum_{u=2}^{T_{N-1}} \xi_u(j)\right)\\
\text{and } a_{T_1,i,j} &= \frac{1}{\ell_{T_N}}\left(\ell_{T_{N-1}}\delta_{i,j,T_N}\right).
\end{aligned}$$

At the last time step $T = T_{end}$, the survivor function of the holding times should be used unless the assumption of forcing a state change at the last time step is appropriate. The HSMM likelihood in Eqn (4) is given by $\ell_{T_{end}}$.

Note that the conditional model $f(Y(t)|\theta_S,S)$ is arbitrary, as long as either $Y(t)$ is independent if time or it is a process with independent increments. Combinations of different models for $Y(t)$ are also possible where for example each hidden state relates to a different conditional model.

Underflow can still present problems so working on the log scale is sensible. The only problem is then how to evaluate $\log(X+Y)$ from $\log(X)$ and $\log(Y)$. One possibility would be to let $M = \max(\log(X),\log(Y))$ so that:

$$\log(X+Y) = \log\left(e^{\log(X)-M} + e^{\log(Y)-M}\right) + M,$$

which is a method immune to underflow since in the worst case $\log(X+Y) = M$.

3.3 Recursive algorithms - Backward

The backward algorithm, as presented in Scott (2002), constructs backward matrices $B_T = \{b_{T,i,j}\}$ such that:

$$b_{T,i,j} = \Pr(S_{T-1} = i \text{ and chain occupies } S_T = j | \text{data up to } T_{end}).$$

There are two differences between the elements $a_{T,i,j}$ of the forward matrices and $b_{T,i,j}$: first, the latter depend on all observed data instead of just data up to T and second, the $b_{T,i,j}$ relate to the probability of the chain occupying state j at T but not necessarily transitioning out of j in the next time step. So in order to implement the backward algorithm, we need to alter the forward algorithm by effectively replacing the holding time distributions $h_j(\tau;\phi_j)$ with the corresponding survival functions $\Pr(\tau \geq t)$ to obtain forward matrices $A'_T = \{a'_{T,i,j}\}$ where

$$a'_{T,i,j} = \Pr(S_{T-1} = i \text{ and chain occupies } S_T = j | \text{data up to } T).$$

Once the forward matrices A'_T have been calculated, backward matrices are obtained by

$$\begin{aligned}
b_{T,i,j} &= \frac{a'_{T,i,j}}{\alpha'_T(j)} \times \beta_{T+1}(j)\\
&= \Pr(S_{T-1} = i | \text{chain occupies } S_T = j, \text{data up to } T)\\
&\times \Pr(\text{chain occupies } S_T = j | \text{data up to } T_{end}),
\end{aligned}$$

where as before, the variables $\alpha'_T(j)$ are calculated by summing the rows of A'_T and are defined as:

$$\alpha'_T(j) = \Pr(\text{chain occupies } S_T = j | \text{data up to } T).$$

The variables $\beta_{T+1}(j)$ are calculated by summing the columns of B_{T+1} and are defined as:

$$\beta_{T+1}(j) = \Pr(\text{chain occupies } S_T = j | \text{data up to } T_{end})$$

and note that $B_{T_{end}} = A'_{T_{end}}$. Simply put, this algorithm starting from T_{end} modifies (or updates) the elements of A'_T so that the sum of its rows is equal to the sum of the columns of B_{T+1}.

The backward matrices B_T are by definition scaled so that their elements sum to 1. Summing the rows of each B_T, gives a vector corresponding to the distribution of states $p(S_T)$ at time step T given all the data, which is effectively

the marginal distribution of each state at each time step. The distribution $p(S_T)$ can be used to summarise the distribution $p(S)$ of the whole state sequence. Unlike in HMMs, the backward algorithm cannot be implemented during estimation as the forward algorithm needs to be modified.

However, in some applications such as gene sequencing, it is of interest to estimate the most likely state trajectory rather than the marginal distribution of each state separately. Maximum a posteriori (MAP) estimation can be used, which finds the mode of the posterior distribution for the state trajectory. In other words, MAP is used to find the state trajectory that maximises the joint posterior of the hidden states (see Scott (2002) for details).

3.4 Metropolis-Hastings

Upon evaluation of the likelihood using the forward algorithm, the Metropolis-Hastings (MH) algorithm can be used directly for parameter estimation. A Gibbs sampler is also possible and Scott (2002) presents a feasible implementation in the case of HMMs where the forward-backward algorithm is used to obtain samples of the hidden chain sequence and utilise them to perform Gibbs sampling on the conditional model. As mentioned earlier, sampling the state sequence can be avoided in each MCMC step by integrating out the state sequence and this is effectively done here by simply evaluating the likelihood and using it to evaluate the acceptance probability in MH.

The difficulty in using MH often lies in the choice and tuning of the proposal distribution $q(\theta^*|\theta)$. For HSMMs, the dimension of the parameter space can be large with each parameter having different support. For instance, parameters in π and rows of P take values on $[0,1]$ and in general are not independent. An easy choice for those parameters is to use an independence sampler where $q(\pi_1) = \mathrm{U}(0,1)$ and $q(\pi_k|\pi_1,\ldots,\pi_{k-1}) = \mathrm{U}(0, 1-\sum_{i=1}^{k-1}\pi_i)$ for $k = 1,\ldots,M-1$ so that using Bayes' theorem,

$$q(\pi_1,\ldots,\pi_{M-1}) = q(\pi_1)q(\pi_2|\pi_1)\cdots q(\pi_{M-1}|\pi_1,\ldots,\pi_{M-2}). \quad (6)$$

An alternative approach is to use an independence sampler for π where the proposal is a Dirichlet distribution: $\pi \sim \mathrm{Dir}(1)$. The independence sampler is easy to use, however it typically leads to slow convergence since information embedded in the existing location of the chain is ignored. Marin and Robert (1997) propose a random walk sampler where the proposal $q(\pi^*|\pi)$ for π is given by:

$$q(\pi^*|\pi) = \mathrm{Dir}(\alpha\pi_1,\ldots,\alpha\pi_M), \quad (7)$$

where $\mathrm{E}\left[\pi_j^*\right] = \pi_j$, giving a proposal centred at the previous value of the chain. Large values of α produce 'moves' that are more local, leading to higher acceptance rate. Marin and Robert (1997) suggest to either choose α at random from a predetermined set of values for each MCMC iteration or perform prior small runs to determine a reasonable value.

The Dirichlet proposal, although elegant, has only one 'tuning' parameter (α) controlling the variance of the proposal. A more flexible approach would be to use a multivariate Gaussian proposal after a logistic transformation. More specifically, let $\pi_{-1} = (\pi_2,\ldots,\pi_M)$ and use the proposal

$$q\left(\log(\pi_{-1}^*/\pi_1^*) \mid \log(\pi_{-1}/\pi_1)\right) \sim \mathrm{N}\left(\log(\pi_{-1}/\pi_1), \Sigma\right) \quad (8)$$

where Σ is a diagonal matrix and entries in the diagonal are variances controlling the acceptance rate for each element. Transforming back is trivial, $\pi_i^* = 1/(1+e^{-x})$ where x is a draw from (8). More control over the proposal can lead to smaller rejection rates and improved mixing relative to (7).

In addition to the parameters of the hidden chain, the parameters of both the conditional model and holding time distributions may be sampled using a Normal random walk sampler. Suitable transformations may be necessary to accommodate certain parameters. For instance, if arbitrary parameter $\phi \in (0,\infty)$ then $\phi^* = \exp(X^*)$ is a suitable candidate where $X^*|X \sim \mathrm{N}(X = \log(\phi), \sigma^2)$. In the case of the Normal distribution the proposal for ϕ, using general transformation theory, is $q(\phi^*|\phi) \sim \mathrm{N}\left(\log(\phi), \sigma^2\right) \times (\phi^*)^{-1}$ meaning that the ratio of proposals in the acceptance probability calculation simplifies to $q(\phi|\phi^*)/q(\phi^*|\phi) = \phi^*/\phi$.

Proposal distributions from different samplers may be combined, such as the random walk and the independence sampler. The drawback of such a convoluted proposal is loss of control in trying to achieve a desired acceptance rate. Componentwise MH can be used where a block of parameters is updated at each MCMC iteration. Gelman et al. (1996) concluded that for high dimensional problems, optimal acceptance rates lie around 24% whereas other authors point towards acceptance rates in the range of 20% to 50% (Gamerman, 1997).

3.5 Label switching

As in the case of mixture models, the problem of label switching may arise when using MCMC to fit HMMs and HSMMs. The issue relates to identifiability and is due to the invariance of the likelihood under relabelling of the hidden states (Richardson and Green, 1997; Celeux et al., 2000). The likelihood invariance directly affects the (joint) posterior distribution of the parameters. Stephens (2000) states that label switching may lead to a highly symmetric and multi-modal posterior making it difficult to summarise, especially by using marginal distributions as these are likely to be inappropriate. Label switching can be diagnosed from time series plots of MCMC parameter samples as well as density estimation plots. Signs of jumps in the former coupled with

associated multi-modality in the latter will indicate label switching issues. A possible prevention technique lies in ordering the states in some way. This is difficult to determine from the data due to the latent nature of the model. A sensible way is to constrain parameters so that if label switching occurs, these constraints are violated. A typical example would be to constrain parameters of the conditional model which relate to the mean.

Although constraints may be imposed by using appropriate proposals, it is also possible to use appropriate priors which have a density of zero in the areas where violations occur (Scott, 2002). This will disrupt the symmetry in the posterior by breaking the symmetry in the prior, thus providing a solution to label switching (Stephens, 2000). In the context of MH, using such priors will result in rejection of any proposed candidates outside the range of the constraints. However, this effectively implies using priors that are informative. Other authors have also considered imposing constraints by re-parametrising the model, see for example Robert and Titterington (1998).

Using constraints such as parameter ordering helps with the identifiability issue, but it is not a 'perfect' solution. Celeux et al. (2000) argue that the effects of parameter ordering are less benign than thought since the design and performance of the MCMC sampler are directly affected. The authors also stress that the true posterior has $M!$ modes ($M =$ number of hidden states) and that a constrained model typically concentrates on a single mode, but may not result in the same inference if the constraints were changed. Celeux et al. (2000) propose a method using MCMC to sample from the true multi-modal posterior and develop ways to reorder samples as if they all came from a single mode. Similarly, Stephens (2000) consider relabelling MCMC output based upon minimising the posterior expected loss.

In addition to these methods of coping with label switching, any natural ordering that may be implied by knowledge of the behaviour hidden chain are useful since restrictions on the posterior are based on physical understanding. Prior information on the latent part of the model may be also helpful in model design where, for instance, absorbing states may be included in the transition matrix.

4 Model application

4.1 Simulation experiments

We begin with a relatively simple model to illustrate the proposed mechanisms for fitting HSMMs. The conditional model is Gaussian where the mean depends on a hidden chain with two states,

$$Y(t)|S \sim \mathrm{N}(\mu_S, \sigma^2) \qquad S = 1,2. \tag{9}$$

The parameters of the hidden chain are the initial distribution π and the parameters ϕ_S of the holding time distributions which we assume to be zero-truncated Poisson:

$$h_S(\tau|\tau > 0; \phi_S) = \frac{e^{-\phi_S}\phi_S^{\tau}}{\tau!(1-e^{-\phi_S})} \qquad S = 1,2. \tag{10}$$

Because there are just two states, the 2×2 transition matrix $P = \{p_{i,j}\}$ is defined as $p_{i,i} = 0$ and $p_{i,j} = 1$.

Data from the model in Eqn (9) were simulated once for $t = 1, \ldots, 500$ time steps, with parameter values given in Table 1. The first 400 time steps were used for model fitting while the last 100 were kept for out-of-sample prediction purposes. Two MCMC chains were run, each for 10000 samples and thinned by 5. After thinning, 500 samples were used as burn-in for each chain resulting in 3000 samples in total. The model was coded in R and each chain took around 35 minutes to run on a reasonably fast machine (Intel Q6850 3GHz with 4GB RAM). Four random walk samplers were used, one for each μ_S, σ^2, π_S and ϕ_S. For parameter $\pi = \{\pi_S\}$, the Dirichlet proposal in Eqn (7) was used with $\alpha = 10$. For this and subsequent model implementations, the acceptance rate for each sampler was adjusted to be in the range $[0.2, 0.5]$.

Table 1 Priors, inputs and estimates, where $\kappa = 0.5$ and $\eta = 0.005$

Param.	Prior	Input Values	Posterior Mean (s.e.)	95% Cr.I.
μ_1	$\mathrm{N}(0, 1000)$	3	2.98 (0.06)	[2.87,3.09]
μ_2	$\mathrm{N}(0, 1000)$	5	4.88 (0.13)	[4.62,5.13]
$1/\sigma^2$	$\mathrm{Gam}(\kappa, \eta)$	1	1.03 (0.04)	[0.96,1.12]
π_1	$\mathrm{Dir}(1,1,1)$	0.3	0.28 (0.24)	[0.01,0.85]
π_2	$\mathrm{Dir}(1,1,1)$	0.7	0.72 (0.24)	[0.15,0.99]
ϕ_1	$\mathrm{Gam}(\kappa, \eta)$	30	29.13 (1.84)	[25.70,32.80]
ϕ_2	$\mathrm{Gam}(\kappa, \eta)$	5	7.66 (0.99)	[5.64,9.75]

Table 1 shows the input values for each parameter as well as prior distributions, estimates (posterior means), standard errors and credible intervals. For π (and later for rows of P), we use a flat $\mathrm{Dir}(\mathbf{1})$ prior distribution. Such a prior is flexible in the sense that the marginal priors for each π_S have Beta distributions with parameters depending on the parameters of the Dirichlet which here we set equal to 1. Throughout this section, we use a Gamma distributed prior with large mean and variance for strictly positive parameters and a zero mean, large variance Gaussian prior for parameters with infinite support.

The estimates in Table 2 show good agreement with the input values. The top plot in Fig 1, shows the simulated values of $Y(t)$ as well as the simulated chain sequence. The backward algorithm in section 3.3 was implemented to obtain the posterior distribution of $p(S_T)$, the probability distribution of each state at each time step. The middle plot in

Fig 1 shows the the posterior mean of $p(S_T = 2)$ from the HSMM in Eqn (9). For completeness, the bottom plot in Fig 1 shows the posterior mean of $p(S_T = 2)$ from an HMM implemented to the same data. The clusters of high probability are more 'sharp' for the HMM in the observation period ($T = 1,\ldots,400$) and this is due to the the exponentially decaying tails of the Geometric distribution. The HSMM also recovers the high probability clusters while allowing for holding time distributions with fatter tails, which explains the lack of 'sharpness'. However, the predicted chain sequence does depend on the observed data so to see the underlying difference between the HMM and HSMM we compare the predicted state sequence in the prediction period ($T = 401,\ldots,500$). It is difficult for the HMM to reproduce the true state sequence as highlighted in the bottom plot of Fig 1 (in the long-run, the state probabilities come from the stationary distribution of the hidden Markov chain which is constant in time).

A further simulation experiment involves simulated data where the conditional model is a non-homogeneous Poisson process (NHPP); we use the acronym HSMM-NHPP for the joint model. The NHPP is a counting process, describing occurrences in time where the intensity function (occurrence rate) is temporally varying. Specifically, we simulate J instances from the HSMM-NHPP where the intensity function $\lambda(t|S)$ depends on a hidden semi-Markov chain S (events may be thought of as failures in $i = 1,\ldots,J$ components and the intensity function as the failure rate). A NHPP whose intensity function varies according to a Markov process is sometimes known in the literature as a Markov modulated Poisson process (MMPP) (Scott and Smyth, 2003; Fearnhead and Sherlock, 2006). The Markov modulation indirectly introduces correlation between successive inter-arrival times, a property that the NHPP alone does not possess; see Fig 2 for a particular realisation of this process.

With slight change of notation, we introduce the random variable $Y(t_a,t_b)$, the number of events in the interval $(t_a,t_b]$ where $0 \leq t_a < t_b$, which for the NHPP is Poisson distributed. The (conditional) likelihood contribution $L(y[t_a,t_b|S])$ is of a Poisson random variable with mean $\int_{t_a}^{t_b} \lambda(t|S)$. Suppose that each component i has failed n_i times at $t_{i,1},\ldots,t_{i,n_i}$ and that the intensity function obeys the power law such that:

$$\lambda(t;x_i|S) = \theta_S t^{\theta_S - 1} \exp\{\beta_0 + \beta_1 x_i\}, \qquad (11)$$

where the shape parameter $\theta_S > 0$ is different for each state S of the hidden chain. $Y(t_a,t_b)$ and $\lambda(t;x_i|S)$ are related by the fact that $\mathrm{E}(Y(t_a,t_b)) = \int_{t_a}^{t_b} \lambda(t;x_i|S)$. The single covariate x_i has an associated parameter β_1, and β_0 is the intercept which also relates to the scale parameter of the power law. Depending on values of θ_S, the rate is either increasing/decreasing non-linearly with time, or it is constant if $\theta_S = 1$. Note that

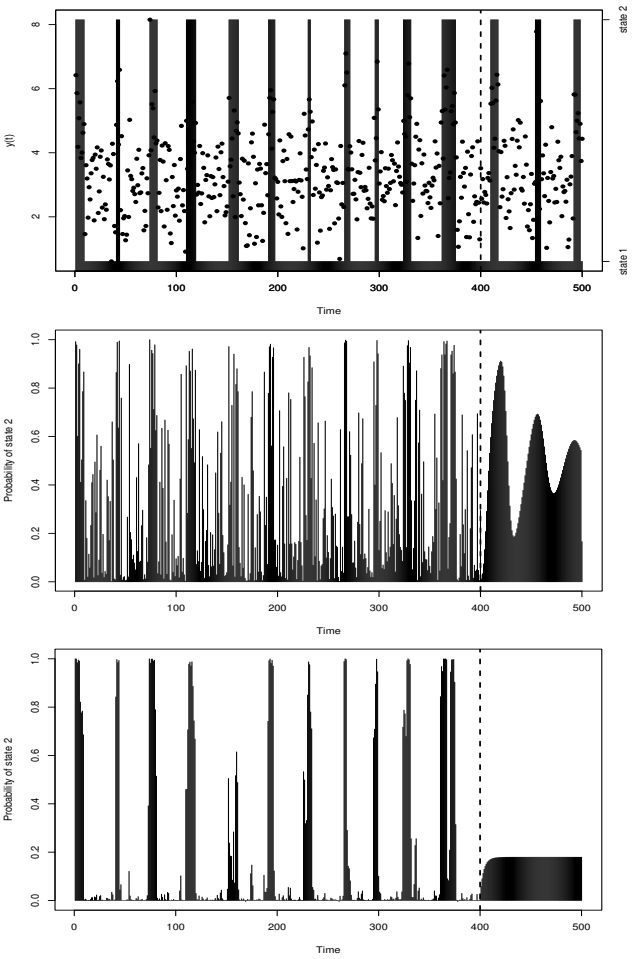

Fig. 1 Top: Simulated values and state sequence from the two state Gaussian model. Middle: Posterior mean of $p(S_T = 2)$, the HSMM probability of state 2 at each time step given the data. Bottom: Same as middle but for a HMM. The dashed vertical line corresponds to the start of the prediction period.

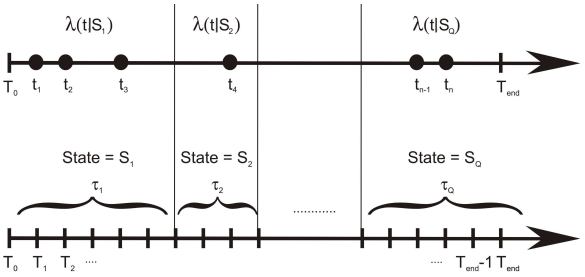

Fig. 2 Semi-Markov modulated counting process.

although we assume a single underlying semi-Markov process, for simulation purposes each imaginary component is affected by a different realisation of this latent process.

We assume 3 hidden states with initial distribution π, transition matrix P and zero-truncated Poisson holding times same as Eqn (10) with $S = 1,2,3$. $J = 50$ objects were simulated from the 3 state NHPP-HSMM where the time period

for each ranged from 80 to 150 time steps. Table 2 shows the priors for each parameter as well as the input values used to simulate the data. Two chains were run for 6000 iterations with a burn-in of 1000 and by thinning of 5 this resulted in 2000 samples from the posterior of each parameter. Each chain took around 5 hours to run, which we consider acceptable given the computational intensity involved in calculating the HSMM likelihood for 50 objects.

Table 2 Priors, inputs and estimates where, $\kappa = 0.5$ and $\eta = 0.005$

Param.	Prior	Input values	Posterior mean (s.e.)	95% Cr.I.
θ_1	Gam(κ,η)	0.5	0.50 (0.002)	[0.49,0.51]
θ_2	Gam(κ,η)	1	0.99 (0.003)	[0.99,1.01]
θ_3	Gam(κ,η)	2	1.99 (0.002)	[1.99,2.00]
β_0	N$(0,1000)$	-1	-0.99 (0.01)	[-1.01,-0.96]
β_1	N$(0,1000)$	0.03	0.03 (0.0001)	[0.029,0.031]
π_1	Dir$(1,1,1)$	0.1	0.06 (0.04)	[0.002,0.152]
π_2	Dir$(1,1,1)$	0.7	0.72 (0.07)	[0.58,0.84]
π_3	Dir$(1,1,1)$	0.3	0.23 (0.07)	[0.10,0.37]
$p_{1,2}$	Dir$(1,1)$	0.7	0.72 (0.03)	[0.64,0.78]
$p_{1,3}$	Dir$(1,1)$	0.3	0.29 (0.03)	[0.22,0.36]
$p_{2,1}$	Dir$(1,1)$	0.2	0.19 (0.03)	[0.15,0.25]
$p_{2,3}$	Dir$(1,1)$	0.8	0.81 (0.026)	[0.75,0.85]
$p_{3,1}$	Dir$(1,1)$	0.65	0.68 (0.030)	[0.61,0.73]
$p_{3,2}$	Dir$(1,1)$	0.35	0.33 (0.030)	[0.27,0.39]
ϕ_1	Gam(κ,η)	15	14.38 (0.27)	[13.8,14.9]
ϕ_2	Gam(κ,η)	10	9.8 (0.20)	[9.4,10.2]
ϕ_3	Gam(κ,η)	2	1.9 (0.09)	[1.7,2.1]

To cope with potential label switching, the posterior was zero unless $\theta_1 < \theta_2 < \theta_3$ or equivalently $\lambda(t|S_1 = 1) < \lambda(t|S = 2) < \lambda(t|S = 3)$. The model was implemented using a combination of 3 random walk samplers for each of $(\theta_S, \beta_0, \beta_1)$, ϕ_S and (P, π). For P and π, the Dirichlet proposal in Eqn (7) was utilised with $\alpha = 150$ for P and $\alpha = 75$ for π (these were chosen by performing small runs prior to the full MCMC).

Table 2 also shows the posterior mean for each fitted parameter along with standard errors and the 95% credible interval (Cr.I). The approach we have adopted as proposed in Section 3 leads to posterior means for the parameters which recover the associated input values used to simulate the data convincingly with relatively narrow 95% Cr.I.

4.2 Application to pipe failures

Predicting pipe failures or aspects of pipe condition in water distribution systems is an essential planning tool for water companies. Traditionally, occurrences of pipe failures are modelled using counting process models such as the NHPP (Kleiner and Rajani, 2001). However, the complex processes that give rise to the occurrence of failures are not fully understood nor always observed which is why conventional counting process models are often unable to perform adequately. Factors affecting failures can be related to pipe characteristics (e.g., pipe material, diameter, past failure history, etc.), the environment (e.g., soil characteristics, weather data, corrosivity, traffic conditions, etc.) and the service itself (e.g., water pressure, level/type of maintenance, etc.). Unfortunately, measurements on such factors are rarely available in data sets making the task of modelling quite challenging.

The problem is a common one: how to model complex (temporal) processes with limited amounts of data. The unobserved processes affecting the underlying failure rate of each pipe are temporal which makes the use of an HSMM an appealing adjustment to the conventional modelling of pipe failures. The available data we use to illustrate this here consists of yearly number of failures (bursts) in each pipe in a group of 30 interconnected water pipes forming part of an underground water distribution network in North America. For confidentiality reasons, data location and origin cannot be disclosed but a full description can be given as to the content of the data. The installation date of each pipe is different (the earliest being 1947 and the latest being 1959) but they were all observed until 2003. The only available covariate is pipe length in metres.

Pipe failures are generally rare events over the lifetime of a pipe (here the total number of failures in all 30 pipes over the whole observation period was only 110) so most data consist of a large number of 'zeros' and a small number of 'ones' at each recording time step. Following the approach in Economou et al. (2012) we therefore consider a zero-inflated model where each pipe is modelled by a mixture of a zero-generating process and a NHPP. In Economou et al. (2012) the mixing probability was constant in time but here we extend this by allowing the mixing probability to be driven by a latent semi-Markov chain. The parameters of the chain are assumed to be the same for each pipe because the pipe group comes from the same network and each pipe is likely to be affected by the same external unobserved processes. Note however that each pipe may experience a different realisation of the chain. In addition, the failure rate of the NHPP is assumed to depend on a pipe specific random effect to allow for heterogeneity in the failure mechanism for each pipe.

Specifically, the model is a HSMM-NHPP with two hidden states. One state corresponds to a NHPP with a power law intensity function whereas the other state relates to a zero process. The model may be formulated in the following way: given the hidden chain S, the model is NHPP with failure rate

$$\lambda(t|S,\theta_i) = \theta_i t^{\theta_i - 1} e^{\{\beta_0 + \beta_1 x_i\}}, \quad \text{if} \quad S = 2 \tag{12}$$

$$\theta_i \sim \text{Gamma}(\alpha, \delta), \qquad i = 1, \ldots, 30 \tag{13}$$

and $\lambda(t|S,\theta_i) = 0$ if $S = 1$. The shape parameter θ_i is different for each pipe and x_i is the pipe length. Furthermore,

Table 3 Posterior summary; $\kappa = 0.5$ and $\eta = 0.005$

Param.	Prior	Posterior Mean (s.e.)	95% Cr.I.	$\hat{R}$
β_0	$N(0,1000)$	-3.89 (0.70)	[-5.38,-2.50]	1.02
β_1	$N(0,1000)$	3.29 (0.94)	[1.52,5.23]	1.01
ϕ_1	$Gam(\kappa,\eta)$	0.52 (0.45)	[0.002,1.63]	1.01
ϕ_2	$Gam\kappa,\eta$	0.94 (0.29)	[0.471,1.63]	1.01
π_1	$Dir(1,1)$	0.61 (0.27)	[0.06,0.982]	1.03
π_2	$Dir(1,1)$	0.39 (0.27)	[0.02,0.939]	1.03
α	$Gam(\kappa,\eta)$	26.73 (13.42)	[6.23,59.92]	1.23
δ	$Gam(\kappa,\eta)$	23.01 (10.87)	[5.93,48.85]	1.26

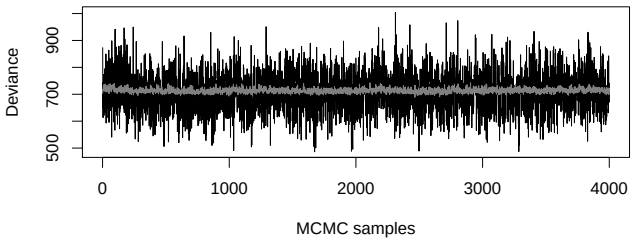

Fig. 3 Deviance samples - actual (grey) and simulated data (black).

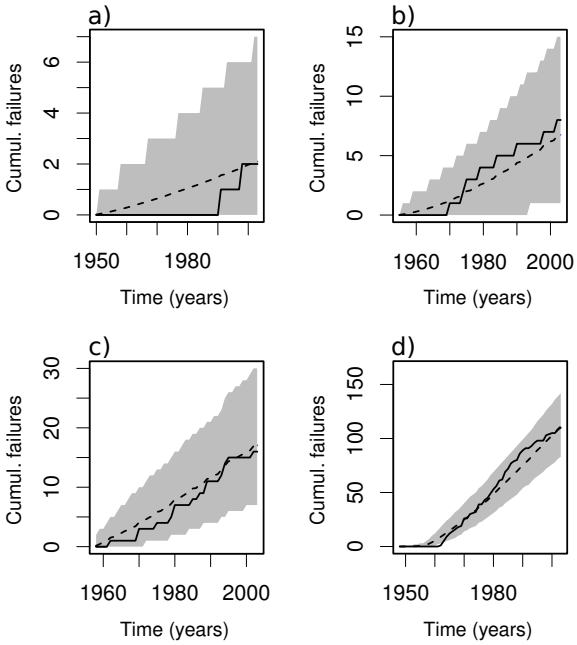

Fig. 4 Observed and predicted cumulative number of failures. Panels a), b) and c) are for individual randomly selected pipes whereas d) is for the whole pipe group. Solid line: observed; Dashed line: predicted; Grey shaded: 95% prediction intervals.

we assume truncated Poisson distributions for the holding times of each state as in Eqn (10) and note that the 2×2 transition matrix is given by $p_{i,i} = 0$ and $p_{i,j} = 1$. The only other unknowns in the model are the parameters of the initial distribution $\pi = (\pi_1, \pi_2)$.

A component-wise random walk MH sampler similar to the one used for the simulations earlier was used and each θ_i was sampled individually. Two MCMC chains were run and after a burn-in of 1000, we collected 10000 samples from each one and thinning by 5 resulted in 4000 samples in total. Convergence was assessed using the Gelman and Rubin $\hat{R}$ values (Gamerman, 1997) (given in Table 3) indicating convergence (values close to 1). This was also true for each θ_i. Note that label switching was not an issue here as the zero-process relating to $S = 1$ has no parameters and indeed none of the trace plots of posterior samples showed any signs of label switching (jumps or multi-modality).

Estimates for global model parameters are given in Table 3 along with standard errors and 95% Cr.I. The interval for β_1 does not include zero indicating that the positive effect of pipe length on the failure rate is significant. Estimates for ϕ_1 and ϕ_2 imply that the holding times for the state relating to the zero-process are somewhat smaller on average but in general the 'persistence' of each state is similar and short term.

Fig 3 plots deviance samples for both data predicted from the fitted model and for the actual data. This is a posterior predictive diagnostic model fit test (Gelman et al., 2004) and a p-value can be estimated to quantify any discrepancies between model and data according to a quantity $T(y,\theta)$ (y for data and θ for parameters) which here is chosen to be the deviance (i.e. minus twice the log-likelihood). The p-value here was estimated as 0.44, by $(1/N)\sum_{i=1}^{N} I(T(y^{(rep)},\theta_i) > (y,\theta_i))$ implying adequate model fit (an extreme p-value implies that the observed data are extreme in relation to data simulated from the model). Here, $I()$ is the indicator function which is equal to 1 or 0 if the argument is either true or false respectively. Also, $y^{(rep)}$ represents data simulated from the posterior predictive distribution and N is the number of MCMC samples.

Plots of observed and predicted cumulative number of failures per year are shown in Fig 4. The bottom right plot reflects the whole group of pipes whereas the other three refer to individual pipes. The predicted line is taken as the posterior predictive distribution mean of the yearly cumulative failure count and the backward algorithm in Section 3.3 was used to calculate the posterior distribution of the most likely state sequence. It is clear the model has performed well in fitting the data both at the individual pipe level as well at the group level.

For completeness, an HMM has also been implemented to the pipe data where the conditional model is the same as Eqns (12) and (13) but the underlying latent model is a two-state Markov chain. We used the Deviance Information Criterion (DIC) to compare the HMM and HSMM. The DIC is an estimate of the expected mean squared predictive er-

ror and may be used to compare models implemented with MCMC (Spiegelhalter et al., 2002). The measure penalises model fit for the number of parameters and a smaller DIC indicates better fit. In this application, the DIC was 585.53 for the HSMM and 711.49 for the HMM, implying that the HSMM fitted considerably better.

5 Discussion

Bayesian HSMMs have received relatively little attention in the literature largely because of their computational intractability. In this paper, recursive algorithms were developed in order to provide a computationally feasible Bayesian implementation of HSMMs. The implementation was illustrated using simulation experiments and a real-life application. The model formulation we have presented here is very general and extremely flexible. Unlike previous uses of HSMMs, the model may be applied to more that one component as demonstrated in the water pipe network application (see section 4) where each pipe depends on a different realisation of the same hidden process. A feasible generalisation of that would be to allow each component to be driven by a different latent chain.

Nevertheless, we did make the assumption in formulating the likelihood that either $Y(t)$ are independent in time or that $Y(t)$ is a stochastic temporal process with independent increments. Despite these assumptions, the formulation here allows for a wide range for models, for instance Economou et al. (2009) have applied a Poisson GLM with time dependent covariates where the intercept is dependent on a latent semi-Markov chain. In addition, any stochastic process in time with independent increments may be used as a conditional model, for example Poisson process (including NHPP), Gaussian (or Wiener) process and more generally Lèvy processes or even a mixture of such processes. Furthermore, no constraints are imposed on non-linearity or where covariates may be incorporated in the model, including parameters of the latent chain such as the transition matrix or the distribution of the holding times.

One aspect that was not discussed in the paper concerns the choice of the number of hidden states. This is analogous to the choice of numbers of components in a mixture modelling and can be addressed using reversible jump MCMC as was done in Robert et al. (2000) for HMMs. We anticipate this would be quite computationally intensive for networks where many hidden chains are involved. Of course any physical justification is ideal as in the case of Hughes et al. (1999) and Bellone et al. (2000) who model precipitation occurrences and the hidden states are linked to unobserved weather states which link synoptic-scale atmospheric patterns to local-scale precipitation.

An 'empirical' way of deciding the number of states simplifies to just look at the model output. From practical experience in simulations (for instance a two-state NHPP-HSMM was applied to a simple NHPP, i.e. one too many states), we found that the means of holding time distributions for many states tend to zero with very small credible intervals implying there are too many states in the model. In addition, non-identifiability of state holding times after constraints have been put in to account for label switching is also a sign of having more states than necessary.

Johnson and Willsky (2012) proposed a Bayesian implementation of HSMMs which incorporates the number of latent states in the estimation and uses Gibbs sampling to provide an efficient estimation mechanism. Specifically, Johnson and Willsky (2012) extend the Hierarchical Dirichlet Process (HDP) HMM to a HDP-HSMM; the main idea behind HPD-based models is to use a HDP prior over an infinite state space, enabling inference on both model parameters and state complexity. The HDP is a prior over an infinite transition matrix, however each row of the matrix is linked hierarchically through a hyper-prior so that the transition distributions tend to have mass concentrated around a typical set of states; in other words such a prior provides a tendency to re-use and re-occupy a particular set of states (from the infinite set). Johnson and Willsky (2012) manage to maintain the conjugacy of the HDP-HMM for the HDP-HSMM and use a Gibbs sampler which is potentially faster than MH with better mixing properties.

Lastly, the primary tool for MCMC implementation in this paper is Metropolis-Hastings and as presented here, it may involve many likelihood evaluations per MCMC iteration (component-wise MH). A referee pointed out that data augmentation is another MCMC implementation technique which can be computationally more efficient. Data augmentation may be described as the Bayesian analogue of the EM algorithm (Gilks et al., 1996) and it essentially introduces auxiliary variables or latent data which are also sampled in the MCMC. For HMMs and HSMMs, the latent state sequence constitutes the auxiliary variables allowing for the likelihood in Eqn (3) to be used instead of Eqn (4), in sampling from the posterior. However, the likelihood in Eqn (4) still needs to be evaluated, albeit only once, in order to sample the auxiliary variables (see Yau et al. (2011) for a recent application of data augmentation in HMMs). Although MH can lead to better mixing if proposal distributions are well chosen, data augmentation will in general be computationally more efficient.

Appendix A: Pseudo code for simulation experiment with Gaussian observations

Simulate data from 2-state Gaussian HSMM, Eqn (9):

1. Set values for μ_S, σ, ϕ_S and π_S for $S = 1, 2$.
2. Set $i = 1$ and sample initial state S_i from $\pi = \{\pi_S\}$.

3. Sample duration τ_i of S_i, from zero-truncated Poisson with parameter ϕ_{S_i}.
4. Sample state transition from distribution corresponding to S_1^{th} row of P.
5. Repeat steps 3 and 4 until $\sum_i \tau_i \geq 250$.
6. If $\sum_i \tau_i > 250$, truncate so that $\sum_i \tau_i = 250$.
7. For each $T = 1,\ldots,250$, sample from $\mathrm{N}(\mu_S, \sigma^2)$ according to which state S the chain is at time step T.

Metropolis-Hastings:

- Set $i = 0$ and initialise parameters $\mu_S^{(i)}, \sigma^{(i)}, \phi_S^{(i)}$ and $\pi_S^{(i)}$.
- Calculate $\ell^{(i)}$, the log-likelihood using the forward algorithm in section 3.2.
- Calculate $P^{(i)}$, the log-posterior by

$$P^{(i)} = \ell^{(i)} + \log\left(p\left(\mu_S^{(i)}\right)\right) + \log\left(p\left(1/\sigma^{(i)}\right)\right) + \log\left(p\left(\phi_S^{(i)}\right)\right) + \log\left(p\left(\pi_S^{(i)}\right)\right) \quad (14)$$

  where $p(\cdot)$ is the prior for each parameter (Table 1).

Do for $i = 1,\ldots,M$ where M is the required number of MCMC iterations:

μ_S: $\mu_S^* = \mu_S^{(i-1)} + \varepsilon$, $\varepsilon \sim \mathrm{N}(0, \sigma_\mu^2)$
Calculate ℓ^* and hence P^* using Eqn (14)
Calculate acceptance probability as $\eta = \min(1, \Omega)$ where $\Omega = \exp\{P^* - P^{(i-1)}\}$
Sample U from $U(0,1)$
If $U \leq \eta$, set $\mu_S^{(i)} = \mu_S^*$, $\ell^{(i)} = \ell^*$ and $P^{(i)} = P^*$.

σ: $\sigma^* = \exp\{\log(\sigma^{(i-1)}) + \varepsilon\}$, $\varepsilon \sim \mathrm{N}(0, \sigma_\sigma^2)$
Calculate ℓ^* and hence P^* using Eqn (14)
Calculate acceptance probability as $\eta = \min(1, \Omega)$ where $\Omega = \exp\left\{P^* + \log(\sigma^*) - [P^{(i-1)} + \log(\sigma^{(i-1)})]\right\}$
Sample U from $U(0,1)$
If $U \leq \eta$, set $\sigma^{(i)} = \sigma^*$, $\ell^{(i)} = \ell^*$ and $P^{(i)} = P^*$.

ϕ_S Same as σ, replacing ϕ_S with σ.

π_S Sample $\pi^* \sim \mathrm{Dir}\left(\alpha\pi^{(i-1)}\right)$
Calculate ℓ^* and hence P^* using Eqn (14)
Calculate acceptance probability as $\eta = \min(1, \Omega)$ where

$$\Omega = \exp\{P^* + \log(d(\pi^{(i-1)}; \alpha\pi^*)) - [P^{(i-1)} + \log(d(\pi^*; \alpha\pi^{(i-1)}))]\}$$

where $d(\pi;\theta)$ is a Dirichlet density with parameter θ
Sample U from $U(0,1)$
If $U \leq \eta$, set $\pi_S^{(i)} = \pi_S^*$, $\ell^{(i)} = \ell^*$ and $P^{(i)} = P^*$.

Adjust $\sigma_\mu^2, \sigma_\sigma^2, \sigma_\phi^2$ and α to achieve desired acceptance rates.

Note that R code for implementing the HSMMs and HMMs from both simulation experiments is available both as supplementary material to the paper and on `http://empslocal.ex.ac.uk/people/staff/te201/HSMM/`.

Appendix B: Pseudo code for simulation experiment with NHPP observations

Simulate data from 3-state NHPP-HSMM with intensity function $\lambda(t;x|S)$ as in Eqn (11), for 50 hypothetical objects:

1. Set values for $\theta_S, \beta_0, \beta_1, \phi_S, \pi_S$ and P for $S = 1,2,3$.
2. Sample $B_j \sim U(80,150)$, for $j = 1,\ldots,50$, to decide the observation period in discrete time steps for each object.
3. Sample $x_j \sim U(50,150)$ to set values for the covariate x.
4. Set $j = 1$, $i = 1$ and sample initial state S_i from $\pi = \{\pi_S\}$.
5. Sample duration τ_i of S_i, from zero-truncated Poisson with parameter ϕ_{S_i}.
6. Sample state transition from distribution corresponding to S_1^{th} row of P.
7. Repeat steps 3 and 4 until $\sum_i \tau_i \geq B_j$.
8. If $\sum_i \tau_i > B_j$, truncate so that $\sum_i \tau_i = B_j$.
9. For each $T = 1,\ldots,B_j$, sample from $\mathrm{Pois}(\Lambda([T-1,T]|S))$ according to which state S the chain is at time step T, where $\Lambda([T-1,T]|S) = \int_{T-1}^{T} \lambda(t;x|S)dt$.
10. Repeat steps 4–9 jor $j = 2,\ldots,50$.

Metropolis-Hastings:

- Very similar to the one in Appendix A.
- The forward algorithm needs to be ran for each of the $j = 1,\ldots,50$ objects so that the log-likelihood $\ell^{(i)} = \sum_{j=1}^{50} \ell_j^{(i)}$ (assuming independence of the objects).
- The log-posterior is obtained essentially as in Eqn (14). However, we restrict the posterior to be zero in areas of the parameter space other than $\theta_1 < \theta_2 < \theta_3$ to impose the label-switching measure.
- Strictly positive parameters such as θ_S and ϕ_S are updated the same as σ and ϕ_S in Appendix A.
- Parameter π and vectors made up of rows from transition matrix P, excluding zero elements, are updated exactly like π in Appendix A.

Acknowledgements The pipe dataset used in this paper was provided by Dr Yehuda Kleiner whom the authors gratefully acknowledge.

References

Baum, L. E., Petrie, T., Soules, G., and Weiss, N. (1970). A maximization technique in the statistical analysis of probabilistic functions of Markov chains. *Annals of Mathematical Statistics*, 41:164–171.

Bellone, E., Hughes, J. P., and Guttorp, P. (2000). A hidden Markov model for downscaling synoptic atmospheric patterns to precipitation amounts. *Climate Research*, 15:1–12.

Bulla, J. and Bulla, I. (2006). Stylized facts of financial time series and hidden semi-Markov models. *Computational Statistics and Data Analysis*, 51:2192–2209.

Bulla, J., Bulla, I., and Nenadic, O. (2010). HSMM - an R package for analyzing hidden semi-Markov models. *Computational Statistics and Data Analysis*, 54:611–619.

Celeux, G., Hurn, M., and Robert, C. P. (2000). Computational and inferential difficulties with mixture posterior distributions. *Journal of the American Statistical Association*, 95(451):957–970.

Chib, S. (1996). Calculating posterior distributions and modal estimates in Markov mixture models. *Journal of Econometrics*, 75(1):79–97.

Devijver, P. A. (1985). Baum's forward-backward algorithm revisited. *Pattern Recognition Letters*, 3(6):369–373.

Dewar, M., Wiggins, C., and Wood, F. (2012). Inference in hidden Markov models with explicit state duration distributions. *IEEE Signal Processing Letters*, 19(4):235–238.

Dong, M. and He, D. (2007). A segmental hidden semi-Markov model (HSMM)-based diagnostics and prognostics framework and methodology. *Mechanical systems and signal processing*, 21:2248–2266.

Economou, T., Kapelan, Z., and Bailey, T. C. (2012). On the prediction of underground water pipe failures: Zero-in ation and pipe specific effects. *Journal of Hydroinformatics*, 14(4):872–883.

Economou, T., Vitolo, R., Bailey, T. C., Kapelan, Z., and Waterhouse, E. K. (2009). A latent structure model for high river flows. In *Proceedings of the 24th International Workshop on Statistical Modelling*, pages 125–129.

Fearnhead, P. and Sherlock, C. (2006). An exact gibbs sampler for the Markov-modulated Poisson process. *Journal of the Royal Statistical Society, Series B (Statistical methodology)*, 68(5):767–784.

Ferguson, J. D. (1980). Variable duration models for speech. In Ferguson, J. D., editor, *Proceedings of the Symposium on the Applications of Hidden Markov Models to Text and Speech*, pages 143–179, Princeton, NJ.

Gamerman, D. (1997). *Markov Chain Monte Carlo: Stochastic Simulation for Bayesian Inference*. Chapman and Hall.

Gelman, A., Carlin, J. B., Stern, H. S., and Rubin, D. B. (2004). *Bayesian data analysis*. Chapman and Hall/CRC.

Gelman, A., Roberts, G. O., and Gilks, W. R. (1996). Efficient Metropolis jumping rules. *Bayesian Statistics 5*, pages 599–607.

Gilks, W. R., Richardson, S., and Spiegelhalter, D. J. (1996). *Markov Chain Monte Carlo in practice*. Chapman and Hall/CRC.

Guedon, Y. (1992). Review of several stochastic speech unit models. *Computer Speech and Language*, 6:377–402.

Guedon, Y. (2003). Estimating hidden semi-Markov chains from discrete sequences. *Journal of Computational and Graphical Statistics*, 12(3):604–639.

Guha, S., Li, Y., and Neuberg, D. (2008). Bayesian hidden Markov modeling of array CGH data. *Journal of the American Statistical Association*, 103(482):485–497.

Hughes, J. P., Guttorp, P., and Charles, S. P. (1999). A non-homogeneous hidden Markov model for precipitation occurrence. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 48(1):15–30.

Jardine, A. K., Lin, D., and Banjevic, D. (2006). A review on machinery diagnostics and prognostics implementing condition-based maintenance. *Mechanical Systems and Signal Processing*, 20(7):1483–1510.

Johnson, M. J. and Willsky, A. S. (2012). Bayesian Nonparametric Hidden Semi-Markov Models. *ArXiv e-prints*, (http://arxiv.org/abs/1203.1365v2).

Jouyaux, C., Richardson, S., and Longini, I. (2000). Modeling markers of disease progression by a hidden Markov process: Application to characterizing cd4 cell decline. *Biometrics*, 56(3):733–741.

Kleiner, Y. and Rajani, B. (2001). Comprehensive review of structural deterioration of water mains: statistical models. *Urban Water*, 3:131–150.

Kozumi, H. (2000). Bayesian analysis of discrete survival data with a hidden Markov chain. *Biometrics*, 56(4):1002–1006.

Levinson, S. E. (1986). Continuously variable duration hidden Markov models for automatic speech recognition. *Computer Speech and Language*, 1:29–45.

Marin, J.-M. and Robert, C. P. (1997). *Bayesian Core: A Practical Approach to Computational Bayesian Statistics*. Springer.

R Development Core Team (2012). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Rabiner, L. (1989). A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–285.

Richardson, S. and Green, P. J. (1997). On Bayesian analysis of mixtures with an unknown number of components. *Journal of the Royal Statistical Society, Series B*, 59(4):731–792.

Robert, C. P., Rydén, T., and Titterington, D. M. (2000). Bayesian inference in hidden Markov models through the reversible jump Markov chain Monte Carlo method. *Journal of the Royal Statistical Society, Series B*, 62(1):57–65.

Robert, C. P. and Titterington, D. M. (1998). Reparameterization strategies for hidden Markov models and Bayesian approaches to maximum likelihood estimation. *Statistics and Computing*, 8:145–158.

Rydén, T., Terasvirta, T., and Asbrink, S. (1998). Stylized facts of daily return series and the hidden Markov model. *Journal of Applied Econometrics*, 13:217–244.

Sansom, J. and Thomson, P. (2001). Fitting hidden semi-Markov models to breakpoint rainfall data. *Journal of Applied Probability*, 38A:142–157.

Schmidler, S. C., Liu, J. S., and Brutlag, D. L. (2000). Bayesian segmentation of protein secondary structure. *Journal of Computational Biology*, 7(1-2):233–248.

Scott, S. (2002). Bayesian methods for hidden Markov models: recursive computing in the 21st century. *Journal of the American Statistical Association*, 97:337–351.

Scott, S. and Smyth, P. (2003). The Markov modulated Poisson process and Markov Poisson cascade with applications to web traffic modelling. *Bayesian Statistics*, 7:671–680.

Spiegelhalter, D. J., Best, N. G., Carlin, B. P., and Van Der Linde, A. (2002). Bayesian measures of model complexity and fit. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 64(4):583–639.

Stephens, M. (2000). Dealing with label switching in mixture models. *Journal of the Royal Statistical Society, Series B*, 62(4):795–809.

Tokdar, S., Xi, P., Kelly, R., and Kass, R. (2010). Detection of bursts in extracellular spike trains using hidden semi-Markov point process models. *Journal of Computational Neuroscience*, 29:203–212.

Yau, C., Papaspiliopoulos, O., Roberts, G. O., and Holmes, C. (2011). Bayesian non-parametric hidden Markov models with applications in genomics. *Journal of the Royal Statistical Society, Series B*, 73(1):37–57.

Yu, S.-Z. (2010). Hidden semi-Markov models. *Artificial Intelligence*, 174:215–243.