

# A comparison of reversible jump MCMC algorithms for DNA sequence segmentation using hidden Markov models

Richard J. Boys     Daniel A. Henderson

Department of Statistics, University of Newcastle, U.K.

Richard.Boys@ncl.ac.uk     D.A.Henderson@open.ac.uk

## Abstract

This paper describes a Bayesian approach to determining the number of hidden states in a hidden Markov model (HMM) via reversible jump Markov chain Monte Carlo (MCMC) methods. Acceptance rates for these algorithms can be quite low, resulting in slow exploration of the posterior distribution. We consider a variety of reversible jump strategies which allow inferences to be made in discretely observed HMMs, with particular emphasis placed on the comparison of the competing strategies in terms of computational expense, algebraic complexity and performance. The methods are illustrated with an application to the segmentation of DNA sequences into compositionally homogeneous regions.

*Keywords:* Bayesian inference; Bioinformatics; discrete-valued time series; DNA; MCMC; model determination; Reversible jump.

## 1 Introduction

Since their introduction in a series of papers on probabilistic functions of Markov chains (e.g. Baum and Petrie (1966) and Baum *et al.* (1970)), *hidden Markov models* (HMMs) have become widely used for modelling latent structure in temporally dependent data. Furthermore, HMMs (also commonly referred to as Markov source models, Markov mixture models and Markov switching models) have been successfully applied to problems in a wide variety of disciplines; see, for example, the extensive list in Cappé (2001) of HMM related work applied to areas such as biology, climatology, econometrics, image processing and speech processing. Other useful background material includes a thorough introduction to HMMs by Rabiner (1989), placing particular emphasis on speech processing, and a general overview of HMMs in the context of modelling discrete-state time series by MacDonald and Zucchini (1997).

Put simply, in a hidden Markov model, an observed process  $\mathbf{Y} = (Y_1, Y_2, \dots, Y_n)$  is assumed to evolve independently, conditional on a *hidden* (unobserved) discrete state Markov chain  $\mathbf{S} = (S_1, S_2, \dots, S_n)$ . As such, the model can be defined in

terms of two sets of equations: the *observation equations*

$$\pi(Y_t|Y_1, \dots, Y_{t-1}, S_1, \dots, S_t) = \pi(Y_t|S_t), \quad Y_t \in \mathcal{Y}, \quad t = 1, 2, \dots, n$$

which detail the distribution of the observable process conditional on its hidden counterpart, and the *system or state equations*

$$\pi(S_t|S_1, \dots, S_{t-1}) = \pi(S_t|S_{t-1}), \quad S_t \in \mathcal{S} = \{1, 2, \dots, r\}, \quad t = 1, 2, \dots, n$$

describing the evolution of the hidden  $r$ -state Markov chain. Note that although the hidden process  $\mathbf{S}$  is discrete, the observed process  $\mathbf{Y}$  can be either discrete or continuous.

In some scenarios, the number of hidden states  $r$  in the HMM is assumed to be known *a priori*, usually as a result of explicit knowledge of the process under study. However, in many cases, an appropriate choice of  $r$  is far from clear. This paper focuses attention on such scenarios and describes this uncertainty through a probability distribution.

We adopt a Bayesian approach to inference as this allows us to incorporate information about all unknown quantities (including  $r$ ) in our model in a systematic way. Thus inferences about these unknown quantities are based on their joint posterior distribution. However, this distribution cannot be studied analytically due to the complexity of the model and therefore we approximate it using Markov chain Monte Carlo (MCMC) methods; see Brooks (1998). Essentially, these methods involve simulating a Markov chain with the (posterior) distribution of interest as its equilibrium distribution.

The task of discriminating between different values of  $r$  is, in essence, a model choice problem. Such problems are usually tackled within the Bayesian paradigm by analysing each model and comparing models using posterior model probabilities or Bayes factors. However, this requires the non-trivial calculation of the integrated likelihood (the normalising constant in Bayes' Theorem), for example, using the output of MCMC simulations; see, Chib and Jeliazkov (2001) and Newton and Raftery (1994) for more details. In this paper, we adopt an alternative approach which expands the model to include  $r$  as an additional unknown parameter and uses MCMC methods to approximate the marginal posterior for  $r$ . Specifically, we use reversible jump methods (Green, 1995) which allow samples to be drawn from the full model space. These methods have been used for mixture models by Richardson and Green (1997) and have recently been developed and applied in the context of HMMs with zero mean Gaussian output sequences by Robert *et al.* (2000).

We focus on the case where the observable process  $Y$  is discrete and, as such, the methods in this paper are applicable to general discrete-state time series. We motivate and illustrate these methods by studying the problem of segmenting deoxyribonucleic acid (DNA) sequences into homogeneous regions. Many DNA sequences display evidence of compositional heterogeneity in the form of *patches* or *domains* of similar structure; see the extensive bibliography by Li (2001). Knowledge of these compositional domains has proved useful in gene-finding algorithms such as Genscan (Burge and Karlin, 1997). HMMs have been a popular choice for modelling compositional heterogeneity in DNA sequences since they were first used by Churchill (1989). This work estimated the parameters in the HMM using the EM algorithm and employed the BIC to determine the most appropriate model. More recent papers taking a Bayesian approach to this problem include Muri (1998) and Boys *et al.* (2000).

The remainder of the paper is structured as follows. Section 2 describes the hidden Markov model followed, in Section 3, with an overview of the Bayesian method

and relevant simulation techniques. The main focus of Section 4 is the description of various reversible jump MCMC strategies whose aim is to produce a Markov chain with good mixing/convergence properties. The methods are applied, in Section 5, to the segmentation of the *lambda* bacteriophage, a benchmark sequence that is commonly used to compare segmentation algorithms. Section 6 contains some brief concluding remarks.

## 2 The hidden Markov model

We assume that the data  $\mathbf{y} = (y_1, y_2, \dots, y_n)$  are a realisation of the discrete (categorical) process  $\mathbf{Y}$ , with state space  $\mathcal{Y} = \{1, 2, \dots, b\}$ . The HMM we employ has observation and system equations

$$\begin{aligned} \pi(Y_t = y_t | S_t = k) &= p_{y_t}^{(k)}, & y_t \in \mathcal{Y}, & k \in \mathcal{S}, \\ \pi(S_t = j | S_{t-1} = i) &= \lambda_{ij}, & i, j \in \mathcal{S}, \end{aligned}$$

where  $\Lambda = (\lambda_{ij})$  is the matrix of hidden state transition probabilities and  $\mathcal{P} = \{\mathbf{p}^{(1)}, \dots, \mathbf{p}^{(r)}\}$  denotes the collection of observable state probabilities in the  $r$  hidden states with  $\mathbf{p}^{(k)} = (p_1^{(k)}, \dots, p_b^{(k)})$ . For notational convenience, we denote the collection of unknown transition probabilities for the hidden and observed states by  $\theta = (\Lambda, \mathcal{P})$ . Figure 1 shows the dependence structure of the model through its *directed acyclic graph* (DAG); unknown quantities are shown as circles and observed quantities as squares. Note that the node for  $r$  appears as a circle as our method assumes that it is unknown.

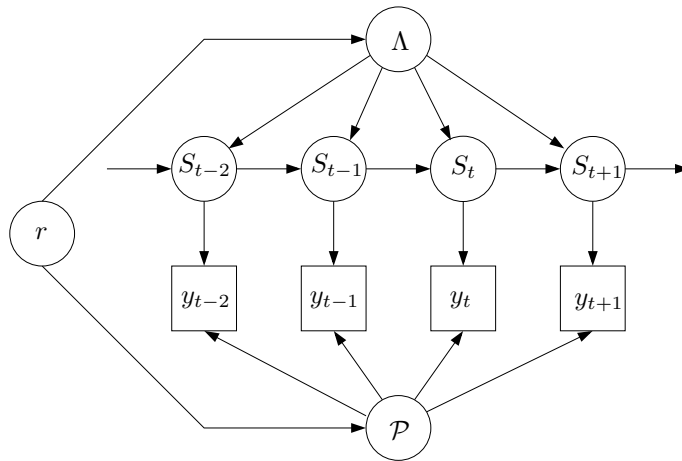


Figure 1: DAG of the HMM

Information from the data about the unknown parameters in the model is expressed through the likelihood. Generally, it is more convenient to work with the *complete-data* likelihood  $\pi(\mathbf{y}, \mathbf{s} | \theta, r)$  due to the missing data structure inherent in the HMM, though we note that other approaches are possible using the *observed-data* likelihood  $\pi(\mathbf{y} | \theta, r)$ ; see Celeux *et al.* (2000). For given  $r$ , the *complete-data* likelihood is simply a product of the observation and system equations, that is

$$\pi(\mathbf{y}, \mathbf{s} | \theta, r) \propto \prod_t p_{y_t}^{(s_t)} \lambda_{s_{t-1} s_t} = \prod_{i \in \mathcal{S}} \prod_{j \in \mathcal{S}} \lambda_{ij}^{m_{ij}} \prod_{k \in \mathcal{S}} \prod_{\ell \in \mathcal{Y}} \left( p_{\ell}^{(k)} \right)^{n_{\ell}^{(k)}}, \quad (1)$$

where

$$n_\ell^{(k)} = \sum_t \mathbb{I}(y_t = \ell, s_t = k) \quad \text{and} \quad m_{ij} = \sum_t \mathbb{I}(s_{t-1} = i, s_t = j) \quad (2)$$

are the relevant counts and transitions, and  $\mathbb{I}(x)$  is the indicator function which equals 1 if  $x$  is true and 0 otherwise.

### 3 Bayesian inference via MCMC

#### 3.1 Prior specification

A major benefit of using the Bayesian approach is that it permits the inclusion of *a priori* uncertainty about the unknown quantities into the analysis. We shall assume that this uncertainty is expressed through the prior distribution

$$\pi(r, \theta) = \pi(r)\pi(\Lambda|r)\pi(\mathcal{P}|r).$$

As in many reversible jump applications, we restrict the number of hidden states  $r$  to be at most  $r_{\max}$ . Further, we assume our (prior) uncertainty about  $r$  can be described by a (truncated) Poisson distribution:  $r \sim Po(a), r \in \{1, 2, \dots, r_{\max}\}$ . Note that if  $r_{\max}$  is reasonably large then the prior mean and variance of  $r$  is approximately  $a$ . Also, following the justification given in Boys *et al.* (2000), we use independent Dirichlet distributions for each  $\mathbf{p}^{(k)}$  and for the rows  $\boldsymbol{\lambda}_i = (\lambda_{ij})$  of the transition matrix  $\Lambda$  (given  $r$ ), that is

$$\mathbf{p}^{(k)} | r \sim \mathcal{D}(\mathbf{b}^{(k)}), \quad k \in \mathcal{S}, \quad (3)$$

$$\boldsymbol{\lambda}_i | r \sim \mathcal{D}(\mathbf{c}_i), \quad i, j \in \mathcal{S}, \quad (4)$$

where the hyperparameters  $\mathbf{b}$  and  $\mathbf{c}$  can be chosen to reflect the goal of the analysis; see Boys *et al.* (2000) for details in the fixed  $r$  case.

#### 3.2 Posterior inference

Information about the model parameters from the data and the prior distribution can be combined using Bayes' Theorem to obtain the posterior distribution

$$\pi(r, \theta, \mathbf{s}|\mathbf{y}) \propto \pi(r, \theta)\pi(\mathbf{y}|\mathbf{s}, r, \theta)\pi(\mathbf{s}|r, \theta). \quad (5)$$

The complexity of this distribution requires us to make inferences about the model parameters by simulating (dependent) posterior samples using a variable dimension MCMC scheme such as reversible jump.

A major inferential problem with HMMs is that their parameters are not identifiable in the likelihood (1). This is a consequence of it being invariant to permutations of the hidden state labels, that is

$$\prod_t p_{y_t}^{(s_t)} \lambda_{s_{t-1}s_t} = \prod_t p_{y_t}^{(\nu(s_t))} \lambda_{\nu(s_{t-1})\nu(s_t)}$$

for any permutation  $\nu(\cdot)$  of the integers  $\{1, 2, \dots, r\}$ . Consequently, the likelihood has  $r!$  symmetric modes, leading to inferential difficulties without strong identification of the parameters in the prior distribution. If a symmetric prior distribution is used (as we do later for our application), the posterior distribution is also invariant to permutations of the hidden state labels.

The impact of this invariance for our MCMC simulation is that it induces *label switching* in which the hidden state labels permute during the course of the simulation. A common way of dealing with this problem is to impose an ordering on the hidden states to prevent label switching and ensure the parameters are identifiable; see, for example, Richardson and Green (1997) and Robert *et al.* (2000). However, Celeux *et al.* (2000) and Stephens (2000b) have noted that employing such artificial ordering constraints can also result in inferential problems. One possible remedy is to adopt a variant of the relabelling algorithms described by Stephens (2000b) which permutes the MCMC output in order to focus on one of the  $r!$  modes. However, we shall focus on quantities which are unaffected by label switching.

### 3.3 MCMC scheme

Dependent samples from the posterior distribution (5) are generated by repeating the following three steps at each iteration,  $i = 1, 2, \dots, N$ :

- (a) update the number of hidden states  $r$  (and consequently update  $\theta$  and  $\mathbf{s}$ );
- (b) update the hidden states  $\mathbf{s} \sim \pi(\mathbf{s}|\mathbf{y}, \theta, r)$ ;
- (c) update the parameters  $\theta = (\Lambda, \mathcal{P}) \sim \pi(\theta|r, \mathbf{s}, \mathbf{y})$ .

Move (a) entails a possible change in the dimension parameter  $r$  and is accomplished using the reversible jump moves described in Section 4. On its own this move is sufficient to ensure we have an irreducible Markov chain for exploring the posterior distribution. However, moves (b) and (c) are introduced to improve mixing and involve standard data-augmentation techniques, details of which can be found in Boys *et al.* (2000). Briefly, these fixed dimension moves (fixed  $r$ ) use standard Gibbs sampling procedures for hidden Markov models; see Robert *et al.* (1993) and Chib (1996).

In move (b), a new sequence of hidden states  $\mathbf{s}$  is generated from its full conditional distribution  $\pi(\mathbf{s}|\mathbf{y}, \theta, r)$  using a forward-backward simulation algorithm; see Boys *et al.* (2000). This algorithm results in improved rates of convergence when compared to a less computationally demanding Gibbs sampling algorithm which simulates each  $s_t$  from its (univariate) full conditional distribution. Move (c) generates new values for the parameters  $\theta$  from their full conditional distributions which, like their prior distributions, are Dirichlet. This conjugacy results from the multinomial form of the complete-data likelihood (1). Thus, we simulate from

$$\begin{aligned} \mathbf{p}^{(k)}|r, \mathbf{s}, \mathbf{y} &\sim \mathcal{D}(\mathbf{b}^{(k)} + \mathbf{n}^{(k)}), \quad k \in \mathcal{S}, \\ \boldsymbol{\lambda}_i|r, \mathbf{s}, \mathbf{y} &\sim \mathcal{D}(\mathbf{c}_i + \mathbf{m}_i), \quad i \in \mathcal{S}_r, \end{aligned}$$

where  $\mathbf{n}^{(k)} = (n_\ell^{(k)})$  and  $\mathbf{m}_i = (m_{ij})$  are as in (2).

## 4 Reversible jump MCMC

### 4.1 Overview

The reversible jump technique is, in essence, a generalisation of the Metropolis-Hastings algorithm to general, variable dimension, parameter spaces. The reversible jump nomenclature arises from the fact that the parameter space is explored by a countable number of move types which all attain detailed balance (are reversible)

and some of which allow jumps between subspaces of different dimensions which represent the different models under consideration. For our model, the variable dimension parameter is  $r$  and the move types we consider take the form of a “jump” from  $\mathbf{x} = (r, \theta, \mathbf{s})$  to  $\tilde{\mathbf{x}} = (\tilde{r}, \tilde{\theta}, \tilde{\mathbf{s}}) = f(\mathbf{x}, \mathbf{u})$  in a space of higher dimension, where  $f(\cdot)$  is an invertible deterministic function. Here, the current parameter values are augmented by supplementary random variates  $\mathbf{u}$  in such a way that the dimensions in the higher and lower dimensional parameter spaces are matched. The proposed value  $\tilde{\mathbf{x}}$  is accepted with probability  $\min(1, A)$  where

$$A = \text{‘likelihood’ ratio} \times \text{prior ratio} \times \text{proposal ratio} \times \text{Jacobian}$$

$$= \frac{\pi(\mathbf{y}, \tilde{\mathbf{s}}|\tilde{\theta}, r+1)}{\pi(\mathbf{y}, \mathbf{s}|\theta, r)} \times \frac{\pi(r+1)\pi(\tilde{\theta}|r+1)}{\pi(r)\pi(\theta|r)} \times \frac{r_m(\tilde{\mathbf{x}})}{r_m(\mathbf{x})q(\mathbf{u})} \times \left| \frac{\partial \tilde{\mathbf{x}}}{\partial(\mathbf{x}, \mathbf{u})} \right|,$$

$r_m(\mathbf{x})$  is the probability of choosing move  $m$  from the set of all variable dimension moves when the current state is  $\mathbf{x}$ ,  $q(\mathbf{u})$  is the density of the supplementary variables and the final term is the Jacobian determinant which is required because of the change of variables from  $\mathbf{x}$  to  $\tilde{\mathbf{x}}$ . The reverse move of jumping to a space of lower dimension is achieved by calculating the values of the supplementary variates  $\mathbf{u}$  that would be required to move from the lower dimension space to the higher one and then accepting the move with probability  $\min(1, A^{-1})$ .

The main criteria used to judge the performance of a proposed move is whether it achieves good mixing rates which allow the MCMC algorithm to explore the entire posterior distribution. Unfortunately, this is not always easy to achieve in the context of HMMs, as witnessed by Robert *et al.* (2000). We will also compare the moves we have constructed in terms of computational expense and algebraic complexity. The reversible jump moves we consider fall into the two categories (of pairs) of moves used by Richardson and Green (1997) in their application of reversible jump techniques to mixture distributions, namely the *split/merge* move and the *birth/death* move.

## 4.2 Split and merge

The basics of the split and merge moves are similar to those described in Robert *et al.* (2000). The moves are initiated with a random choice between splitting a hidden state into two, with probability  $b_r$ , and combining two hidden states into one, with probability  $d_r = 1 - b_r$ . Although it is possible to vary these probabilities for different  $r$ , we will implement the simple choice of  $b_r = 1/2$ ,  $r = 2, \dots, r_{\max} - 1$  with boundary values  $b_1 = 1$  and  $b_{r_{\max}} = 0$ .

### 4.2.1 Merge

Suppose that the current state of the MCMC sampler is  $\tilde{\mathbf{x}} = (r+1, \tilde{\theta}, \tilde{\mathbf{s}})$ . The aim of the move is to combine a randomly chosen pair of hidden states,  $j_1$  and  $j_2$ , to obtain a new hidden state  $j^*$  such that the new state  $\mathbf{x} = (r, \theta, \mathbf{s})$  can be thought of as a realisation from a model with similar properties to the higher dimensional model. We will choose a move which preserves the stationary characteristics of the hidden process  $\mathcal{S}$ . It should be noted that the move does not assume that the pair of hidden states which are merged are ‘adjacent’ in some sense, though we do consider using an adjacency condition to improve acceptance rates in Section 4.2.3.

Robert *et al.* (2000) give a procedure for updating the hidden state transition

matrix from  $\tilde{\Lambda}$  to  $\Lambda$  which takes

$$\lambda_{j^*j} = \frac{\tilde{\pi}_{j_1}}{\tilde{\pi}_{j_1} + \tilde{\pi}_{j_2}} \tilde{\lambda}_{j_1j} + \frac{\tilde{\pi}_{j_2}}{\tilde{\pi}_{j_1} + \tilde{\pi}_{j_2}} \tilde{\lambda}_{j_2j}, \quad \lambda_{ij^*} = \tilde{\lambda}_{ij_1} + \tilde{\lambda}_{ij_2},$$

for  $i, j \neq j^*$  and copies the remaining elements. This procedure preserves the stationary behaviour of the  $\mathbf{S}$ -process since  $\pi_i = \tilde{\pi}_i$  for  $i \neq j^*$  and  $\pi_{j^*} = \tilde{\pi}_{j_1} + \tilde{\pi}_{j_2}$  where  $\pi = \pi\Lambda$  and  $\tilde{\pi} = \tilde{\pi}\tilde{\Lambda}$ . The observable state probabilities for the new hidden state  $j^*$  are obtained by taking  $\mathbf{p}^{(k)} = \tilde{\mathbf{p}}^{(k)}$  for  $k \neq j^*$  and

$$\mathbf{p}^{(j^*)} = (\tilde{\pi}_{j_1} \tilde{\mathbf{p}}^{(j_1)} + \tilde{\pi}_{j_2} \tilde{\mathbf{p}}^{(j_2)}) / \pi_{j^*}.$$

The merge move concludes with a relabelling of the hidden states in which we take  $s_t = \tilde{s}_t$  unless  $\tilde{s}_t = j_1$  or  $j_2$ , in which case we take  $s_t = j^*$ .

#### 4.2.2 Split

The split move is the reverse of the merge move in that the aim is to go from  $\mathbf{x}$  to  $\tilde{\mathbf{x}}$  whilst preserving the stationary characteristics of the hidden process; see Robert *et al.* (2000). First a hidden state  $j^*$  is chosen at random and split into two new hidden states  $j_1$  and  $j_2$ . The stationary probabilities for these new states are obtained by splitting that for  $j^*$  using a supplementary random variate  $u_0$ , and the remaining stationary probabilities are left unchanged. Thus  $\tilde{\pi}_i = \pi_i$  for  $i \neq j_1, j_2$  and

$$\tilde{\pi}_{j_1} = u_0 \pi_{j^*} \quad \text{and} \quad \tilde{\pi}_{j_2} = (1 - u_0) \pi_{j^*}.$$

Next the  $r \times r$  hidden state transition matrix  $\Lambda$  is augmented using supplementary variates  $v_0, u_j, v_i$  for  $i, j \neq j_1, j_2$  to obtain a new matrix  $\tilde{\Lambda}$  with stationary probabilities  $\tilde{\pi}$ . This is done by copying all transition probabilities that are neither from, nor to,  $j_1$  or  $j_2$ , that is,  $\tilde{\lambda}_{ij} = \lambda_{ij}$  for  $i, j \neq j_1, j_2$ , and taking

$$\begin{aligned} \tilde{\lambda}_{j_1j} &= u_j \lambda_{j^*j} / u_0, & \tilde{\lambda}_{j_2j} &= (1 - u_j) \lambda_{j^*j} / (1 - u_0), & \text{for } j \neq j_1, j_2, \\ \tilde{\lambda}_{ij_1} &= v_i \lambda_{ij^*}, & \tilde{\lambda}_{ij_2} &= (1 - v_i) \lambda_{ij^*}, & \text{for } i \neq j_1, j_2, \\ \tilde{\lambda}_{j_1j_2} &= v_0 \left( 1 - \sum_{j \neq j^*} u_j \lambda_{j^*j} / u_0 \right), \\ \tilde{\lambda}_{j_2j_1} &= \left\{ (1 - v_0) \sum_{j \neq j^*} u_j \lambda_{j^*j} + u_0 v_0 - \sum_{i \neq j^*} \kappa_i v_i \lambda_{ij^*} \right\} / (1 - u_0), \end{aligned}$$

where  $\kappa_i = \pi_i / \pi_{j^*}$ . Finally, the remaining elements,  $\tilde{\lambda}_{j_1j_1}$  and  $\tilde{\lambda}_{j_2j_2}$ , are set so that the rows of the transition matrix sum to unity.

The supplementary variates  $u_0, v_0, u_j, v_i$  for  $i, j \neq j_1, j_2$  can be simulated (independently) from any distribution with the appropriate support. We follow the choice of Robert *et al.* (2000) by taking

$$u_0 \sim \text{Beta}(\alpha, \beta), \quad v_0 \sim U(v_0^L, v_0^U), \quad u_j, v_i \sim \text{Beta}(\gamma, \delta).$$

This choice of distribution does not necessarily achieve optimal acceptance rates for the move though there is scope for tuning the parameters of the beta distributions. Also guidance on which distributions do produce optimal acceptance rates is now available in a recent unpublished paper by Brooks *et al.* (2000). The choice of parameters for the distribution of  $v_0$  needs to be specified carefully so that the resulting matrix  $\tilde{\Lambda}$  is stochastic. Given the other supplementary variates, these

upper and lower limits are

$$v_0^L = \max \left\{ 0, 1 - \frac{1 - \sum_{i \neq j^*} \kappa_i \tilde{\lambda}_{ij_1} / u_0}{1 - \sum_{j \neq j^*} \tilde{\lambda}_{j_1 j}} \right\},$$

$$v_0^U = \min \left\{ 1 - \frac{1 - \sum_{i \neq j^*} \kappa_i \tilde{\lambda}_{ij_1} / u_0 - (1 - u_0) \left( 1 - \sum_{j \neq j^*} \tilde{\lambda}_{j_2 j} \right) / u_0}{1 - \sum_{j \neq j^*} \tilde{\lambda}_{j_1 j}}, 1 \right\}.$$

If  $v_0^L > v_0^U$ , then the split move is rejected immediately as there is no valid  $v_0$ . It can be readily verified that this procedure results in a transition matrix  $\tilde{\Lambda}$  corresponding to a Markov chain with the desired stationary distribution  $\tilde{\pi}$ .

The observable state probabilities  $\tilde{\mathcal{P}}$  are obtained by taking  $\tilde{\mathbf{p}}^{(j)} = \mathbf{p}^{(j)}$  for  $j \neq j_1, j_2$ . Various strategies are possible for determining the corresponding values for the new states  $j_1$  and  $j_2$ . We shall split the elements of  $\mathbf{p}^{(j^*)}$  using  $b - 1$  supplementary variates  $w_i$  as follows:

$$\tilde{p}_i^{(j_1)} = \frac{w_i}{u_0} p_i^{(j^*)}, \quad \tilde{p}_i^{(j_2)} = \frac{(1 - w_i)}{(1 - u_0)} p_i^{(j^*)}, \quad i \in \mathcal{Y}_{b-1},$$

where  $\mathcal{Y}_x$  denotes the first  $x$  elements of  $\mathcal{Y}$ . Taking  $w_i \sim \mathcal{U}(w_i^L, w_i^U)$ ,  $i \in \mathcal{Y}_{b-1}$ , where

$$w_i^L = \max \left\{ 0, 1 - \frac{(1 - u_0)}{p_i^{(j^*)}} \rho_i^{(j_2)} \right\} \quad w_i^U = \min \left\{ \frac{u_0}{p_i^{(j^*)}} \rho_i^{(j_1)}, 1 \right\},$$

with  $\rho_i^{(j)} = 1 - \sum_{k \in \mathcal{Y}_{i-1}} \tilde{p}_k^{(j)}$ , ensures that  $\tilde{\mathbf{p}}^{(j_1)}$  and  $\tilde{\mathbf{p}}^{(j_2)}$  sum to unity. The remaining elements are obtained by ensuring that the vectors sum to unity.

The procedure for the reallocation of the hidden states requires only the reallocation of those currently allocated to  $j^*$ . This relabelling is achieved using a restricted form of the forward-backward algorithm. We denote the probability of this reallocation by  $P_{\text{alloc}}$ .

The acceptance probability for the split move is  $\min(1, A_S)$ , where

$$A_S = \frac{\pi(\mathbf{y}, \tilde{\mathbf{s}} | \tilde{\boldsymbol{\theta}}, r + 1)}{\pi(\mathbf{y}, \mathbf{s} | \boldsymbol{\theta}, r)} \times \frac{\pi(r + 1)}{\pi(r)} \frac{\prod_{i=1}^{r+1} \mathcal{D}(\tilde{\boldsymbol{\lambda}}_i | \tilde{\mathbf{c}}_i)}{\prod_{i=1}^r \mathcal{D}(\boldsymbol{\lambda}_i | \mathbf{c}_i)} \frac{\prod_{j=j_1, j_2} \mathcal{D}(\tilde{\mathbf{p}}^{(j)} | \tilde{\mathbf{b}}^{(j)})}{\mathcal{D}(\mathbf{p}^{(j^*)} | \mathbf{b}^{(j^*)})} \times (r + 1)$$

$$\times \frac{d_{r+1} \times 2 / (r + 1)}{b_r P_{\text{alloc}}} \left\{ \mathcal{B}(u_0 | \alpha, \beta) \prod_{j \neq j_1, j_2} \mathcal{B}(u_j | \gamma, \delta) \prod_{i \neq j_1, j_2} \mathcal{B}(v_i | \gamma, \delta) \right.$$

$$\left. \times \frac{1}{v_0^U - v_0^L} \mathcal{B} \left( \frac{v_0 - v_0^L}{v_0^U - v_0^L} \middle| 1, 1 \right) \prod_{i \in \mathcal{Y}_{b-1}} \frac{1}{w_i^U - w_i^L} \mathcal{B} \left( \frac{w_i - w_i^L}{w_i^U - w_i^L} \middle| 1, 1 \right) \right\}^{-1} \times J_S, \quad (6)$$

$\mathcal{D}(\cdot | \boldsymbol{\theta})$  is the  $\mathcal{D}(\boldsymbol{\theta})$  density and  $\mathcal{B}(\cdot | a, b)$  is the Beta( $a, b$ ) density. Note that the acceptance probability for the merge move is  $\min(1, A_S^{-1})$  because of the reversibility.

Equation (6) gives full details of all terms so that generalisations to other proposal distributions are clear. The last factor of the first row  $(r + 1)$  is present as,

according to Cappé *et al.* (2001), even though no ordering of the hidden states is made, we are working on a quotient space. Note that this term would still be present even if an ordering were imposed, due to a  $(r+1) = (r+1)!/r!$  contribution. The factor  $2/(r+1)$  is present as there are  $r$  possible hidden states to choose in the split move and  $r(r+1)/2$  possible pairs to combine in the merge move. The Jacobian for this move  $J_S$  is somewhat complicated to evaluate and must be determined numerically by the solution of a linear system of equations; for further details, see Appendix B in Robert *et al.* (2000).

### 4.2.3 Adjacency condition

Restricting the merge move to combining adjacent hidden states, as in Richardson and Green (1997), may lead to more successful moves. There are many (symmetric) distance measures  $D(\mathbf{p}^{(i)}, \mathbf{p}^{(j)})$  between the hidden state dependent probabilities that can be used, for instance, Euclidean distance. The effect of using an adjacency condition on the merge move is to choose component  $j_1$  at random, take

$$j_2 = \arg \min_{j \in \mathcal{S}} D(\mathbf{p}^{(j_1)}, \mathbf{p}^{(j)})$$

and then combine them into the new hidden state  $j^*$ . The split move is as before except that the adjacency condition needs to be checked: the new hidden states  $j_1$  and  $j_2$  are deemed to be adjacent if

$$D(\mathbf{p}^{(j_1)}, \mathbf{p}^{(j_2)}) < \max \left\{ \min_{i \neq j_1, j_2} D(\mathbf{p}^{(j_1)}, \mathbf{p}^{(i)}), \min_{i \neq j_1, j_2} D(\mathbf{p}^{(i)}, \mathbf{p}^{(j_2)}) \right\}.$$

If  $j_1$  and  $j_2$  do not satisfy this condition, the move is terminated immediately as it is not reversible.

The acceptance probability for this modified split move is given by  $\min(1, A_S \times r/2)$ . This results from the fact that there are still  $r$  choices of hidden states in the split move, but only  $r+1$  to choose from in the merge move.

## 4.3 Birth and Death

We now consider birth and death moves which, as with split/merge moves, begin with a random choice between creating and deleting hidden states with probabilities  $b_r$  and  $d_r$  respectively.

### 4.3.1 Death

Suppose that the current state of the MCMC sampler is  $\tilde{\mathbf{x}} = (r+1, \tilde{\boldsymbol{\theta}}, \tilde{\mathbf{s}})$ . The death move is very simple. It consists of deleting a hidden state  $j^*$  at random and adjusting the other parameters to allow for this deletion. First, we delete  $\tilde{\mathbf{p}}^{(j^*)}$  and take  $\tilde{\mathbf{p}}^{(k)} = \mathbf{p}^{(k)}$  for  $k \neq j^*$ . Then row and column  $j^*$  of  $\tilde{\Lambda}$  are deleted before rescaling it to obtain the stochastic matrix  $\Lambda$ . Finally, the data are reallocated to the new hidden states using the forward-backward algorithm to obtain  $\mathbf{s}$ .

### 4.3.2 Birth

In the birth move, we propose the creation of a new segment type  $j^*$  using supplementary variates  $\mathbf{u}$ ,  $\mathbf{v}$  and  $w_i$ ,  $i \neq j^*$  generated from the prior distribution. The new hidden state dependent probabilities are determined by first simulating  $\mathbf{v}$  from the prior distribution (3), taking  $\tilde{\mathbf{p}}^{(j^*)} = \mathbf{v}$  and then copying over the remaining

probabilities. The  $j^*$  row of the new transition matrix  $\tilde{\Lambda}$  is formed by simulating a row vector  $\mathbf{u}$  from the prior (4) and taking  $\tilde{\lambda}_{j^*} = \mathbf{u}$ . The  $j^*$  column is filled by taking  $\tilde{\lambda}_{ij^*} = w_i$ ,  $i \neq j^*$ , where the supplementary variates  $w_i$  are simulated from the appropriate marginal distribution:  $w_i \sim \text{Beta}(\tilde{c}_{ij^*}, \sum_{j \neq j^*} \tilde{c}_{ij})$ . The remaining elements of  $\tilde{\Lambda}$  are filled with scaled copies of the elements of  $\Lambda$ , where the scaling is chosen to ensure that  $\tilde{\Lambda}$  is stochastic. Finally, the data are reallocated using the forward-backward algorithm to obtain  $\tilde{\mathbf{s}}$ .

The acceptance probability for the birth move is  $\min(1, A_B)$ , where

$$A_B = \frac{\pi(\mathbf{y}, \tilde{\mathbf{s}}|\tilde{\theta}, r+1)}{\pi(\mathbf{y}, \mathbf{s}|\theta, r)} \times \frac{\pi(r+1)}{\pi(r)} \frac{\prod_{i=1}^{r+1} \mathcal{D}(\tilde{\lambda}_i|\tilde{c}_i)}{\prod_{i=1}^r \mathcal{D}(\lambda_i|c_i)} \frac{\prod_{j=1}^{r+1} \mathcal{D}(\tilde{\mathbf{p}}^{(j)}|\tilde{\mathbf{b}}^{(j)})}{\prod_{j=1}^r \mathcal{D}(\mathbf{p}^{(j)}|\mathbf{b}^{(j)})} \times (r+1) \\ \times \frac{d_{r+1}}{b_r(r+1)} \frac{\pi(\mathbf{s}|\mathbf{y}, \theta, r)}{\pi(\tilde{\mathbf{s}}|\mathbf{y}, \tilde{\theta}, r+1)} \\ \times \left\{ \mathcal{D}(\mathbf{u}|\tilde{c}_{j^*}) \mathcal{D}(\mathbf{v}|\tilde{\mathbf{b}}^{(j^*)}) \prod_{i \neq j^*} \mathcal{B}\left(w_i \mid \tilde{c}_{ij^*}, \sum_{j \neq j^*} \tilde{c}_{ij}\right) \right\}^{-1} \prod_{i \neq j^*} (1-w_i)^{r-1}.$$

The acceptance probability for the death move is  $\min(1, A_B^{-1})$ . It is clear from the following simple rearrangement of Bayes' Theorem

$$\frac{\pi(\mathbf{y}, \mathbf{s}|\theta, r)}{\pi(\mathbf{s}|\mathbf{y}, \theta, r)} = \pi(\mathbf{y}|\theta, r),$$

that  $A_B$  is independent of  $\mathbf{s}$  and  $\tilde{\mathbf{s}}$ , and so the two moves are reversible.

Viallefont *et al.* (2001) used a similar birth/death move to analyse Poisson mixtures and found that it compared favourably with more complicated split/merge strategies. They also reported that a drawback of their birth/death move was that it was computationally more expensive than the split/merge alternatives because the reallocation step involves all the data. However, there is no need to update the hidden states in such moves if the *observed-data* likelihood  $\pi(\mathbf{y}|\theta, r)$  can be computed directly, as in our case, from the forward sweep of the forward-backward algorithm; see, for example, MacDonald and Zucchini (1997). This simplified move has a shorter execution time and is, in some sense, a discrete-time version of the birth-death process alternative to reversible jumps of Stephens (2000a).

### 4.3.3 Birth and death of empty hidden states

Richardson and Green (1997) first introduced a birth/death move to act solely on dormant (empty) components in an attempt to improve the mixing of their split/merge-based reversible jump algorithm. Empty components are components that (currently) have no data allocated to them. Robert *et al.* (2000) employed a similar move but, as it performed poorly in their application, they removed it from their algorithm.

These special birth/death moves are essentially the same as those described above but, because they act on empty hidden states,  $\mathbf{s}$  remains unchanged (apart from the necessary relabelling). This results in a slightly different acceptance prob-

ability for the birth of an empty hidden state, namely,  $\min(1, A_E)$ , where

$$A_E = \frac{\pi(\tilde{\mathbf{s}}|\tilde{\theta}, r+1)}{\pi(\mathbf{s}|\theta, r)} \times \frac{\pi(r+1)}{\pi(r)} \frac{\prod_{i=1}^{r+1} \mathcal{D}(\tilde{\lambda}_i|\tilde{\mathbf{c}}_i)}{\prod_{i=1}^r \mathcal{D}(\lambda_i|\mathbf{c}_i)} \frac{\prod_{j=1}^{r+1} \mathcal{D}(\tilde{\mathbf{p}}^{(j)}|\tilde{\mathbf{b}}^{(j)})}{\prod_{j=1}^r \mathcal{D}(\mathbf{p}^{(j)}|\mathbf{b}^{(j)})} \times (r+1) \times \frac{d_{r+1}}{b_r(r_0+1)} \\ \times \left\{ \mathcal{D}(\mathbf{u}|\tilde{\mathbf{c}}_{j^*}) \mathcal{D}(\mathbf{v}|\tilde{\mathbf{b}}^{(j^*)}) \prod_{i \neq j^*} \mathcal{B}\left(w_i \left| \tilde{\mathbf{c}}_{ij^*}, \sum_{j \neq j^*} \tilde{\mathbf{c}}_{ij} \right.\right) \right\}^{-1} \prod_{i \neq j^*} (1-w_i)^{r-1}$$

and  $r_0$  is the number of empty hidden states prior to the birth move. The acceptance probability for the death of an empty hidden state is  $\min(1, A_E^{-1})$  as the two moves are reversible.

## 5 Application to DNA sequence segmentation

### 5.1 DNA sequence data

We now apply our methods to the problem of detecting homogeneous segments in a heterogeneous DNA sequence. Briefly, a DNA sequence consists of a string of nucleic acids (or bases) attached to a sugar-phosphate backbone in such a way that the sequence possesses an implicit temporal ordering (from the 5' to the 3' end). Each position in the sequence consists of one of four bases: adenine (A), cytosine (C), guanine (G) and thymine (T). Thus a DNA sequence  $\mathbf{y} = (y_1, y_2, \dots, y_n)$  can be considered as a string of letters from the  $b = 4$  letter alphabet  $\mathcal{Y} = \{\mathbf{A}, \mathbf{C}, \mathbf{G}, \mathbf{T}\}$ . In this context, the hidden states describe the latent segment structure in the sequence in which different segment types have different base probabilities  $\mathbf{p}^{(k)} = (p_A^{(k)}, p_C^{(k)}, p_G^{(k)}, p_T^{(k)})$ ; see Figure 2. Although this HMM does not capture the full complexity of the dependence structure in a DNA sequence, it will serve to illustrate many of the computational aspects of this work. Extensions to more realistic models which permit inferences about the order of Markov dependence between bases can be found in Henderson and Boys (2001).

We illustrate our methods by studying the genome of the bacteriophage *lambda*, a virus affecting the intestinal bacterium *Escherichia coli*. This relatively small genome ( $n = 48502$ ) has become a benchmark sequence for the comparison of statistical segmentation algorithms; see Braun and Müller (1998) for a recent overview. Its complete sequence is stored in the GenBank® sequence database (Benson *et al.*, 2000) under Accession No. J02459 and can be obtained from the National Center for Biotechnology Information (NCBI) web pages at <http://www.ncbi.nlm.nih.gov/>.

$\mathbf{y}$	5' ...	A	A	G	G	G	A	T	A	...	C	C	T	C	C	A	...	3'
$\mathbf{s}$		1	1	1	1	1	1	3	3	...	3	2	2	2	2	2		
$\mathcal{P}$		← $\mathbf{p}^{(1)}$ →						← $\mathbf{p}^{(3)}$ →				← $\mathbf{p}^{(2)}$ →						

Figure 2: Representation of segment structure in DNA. The different colours represent different compositional regions which are labelled by the hidden states.

## 5.2 Results

We have chosen to describe our prior uncertainty about  $r$  using a Poisson distribution with parameter  $a = 3$  and to limit the number of segment types to  $r_{\max} = 9$ . Also, as very little is known *a priori* about the base probabilities in each segment, we have made the exchangeable choice  $\mathbf{b}^{(k)} = (1, 1, 1, 1)$ . Finally, we have inputted prior information for the hidden state transition matrix using the  $\mathbf{c}_i$  which suggests that, on average, segments will be 1000 bases in length ( $E(\lambda_{jj}) = 0.999$ ). We also assume that the off-diagonal elements of  $\Lambda$  are exchangeable within rows and that the precision of this distribution has the information content of a sequence with  $1000r_{\max}$  transitions.

We report here the performance of three different implementations of the MCMC scheme described in Section 3.3. Each implementation differed only in step (a) by using different combinations of the following move types: split/merge ( $SM$ ), split/merge with adjacency ( $SM_a$ ), birth/death ( $BD$ ), and birth/death of empty hidden states ( $BD_e$ ). The combinations used were (i)  $SM$  and  $BD_e$  moves, (ii)  $SM_a$  and  $BD_e$  moves, and (iii)  $BD$  and  $BD_e$  moves. In each case, the choice within each pair during step (a) of the MCMC scheme was made at random.

Each MCMC scheme was run for  $N = 100000$  iterations after a burn-in of 100000 iterations, taking every 10th iterate. These long runs were chosen to enable effective monitoring of the convergence of the sampler and to get an accurate assessment of the performance of the different move types.

All three implementations took similar execution times, with that using only  $BD$  and  $BD_e$  moves being slightly quicker. Figure 3 displays various plots of the output from the sampler for implementation (iii). The left-hand plots show the

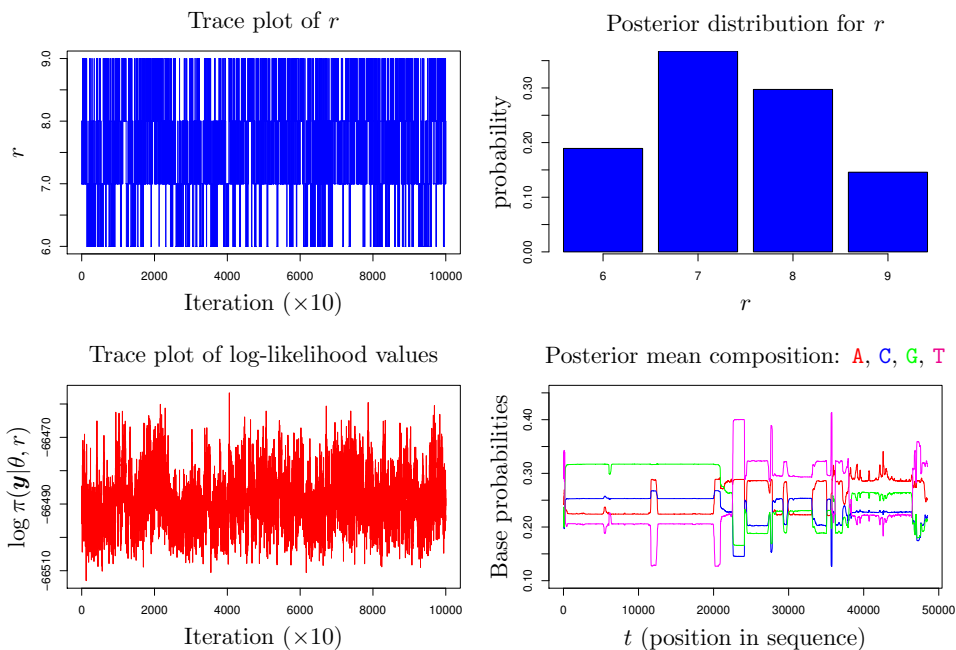


Figure 3: MCMC output using reversible jump moves  $BD$  and  $BD_e$

trace plots for  $r$  and the log-likelihood and indicate that the burn-in has been long enough to achieve convergence. The plots for the other implementations gave similar conclusions. Table 1 shows the performance of the different move types. Clearly, split/merge move types are rarely accepted, with little observed gain being made using the adjacency condition. The  $BD$  move performs much better and attains an acceptance rate similar to the split/merge moves in Robert *et al.* (2000). However, the best performance is achieved by the birth/death move on empty components. Overall, the strategy which uses only birth/death moves produces the best mixing chain, with the value of  $r$  changing in around 15% of the iterations. Moreover, these moves are preferable to the split/merge moves as they are conceptually much simpler and require less computational effort.

Strategy	Proportion of accepted moves				Overall mixing rate
	$SM$	$SM_a$	$BD$	$BD_e$	
$SM$ and $BD_e$	2e-05	–	–	0.108	0.108
$SM_a$ and $BD_e$	–	2e-05	–	0.107	0.107
$BD$ and $BD_e$	–	–	0.037	0.109	0.146

Table 1: Performance of three reversible jump strategies

Figure 3 also shows the (marginal) posterior probabilities for  $r$  estimated from the MCMC output of implementation (iii), and suggests that the bacteriophage *lambda* genome has around seven or eight different segment types. The values for this distribution and those obtained from the other implementations are given in Table 2. The distributions obtained using split/merge moves are very similar, suggesting that  $r$  is six or seven, but differ from that obtained using the better mixing chain consisting of only birth/death moves. Both conclusions are not inconsistent with an analysis by Churchill (1992) which suggested that  $r = 4$  for this model after taking  $r_{\max} = 4$ .

Strategy	Posterior probabilities for $r$						
	$\leq 3$	4	5	6	7	8	9
$SM$ and $BD_e$	0.000	0.002	0.066	0.332	0.356	0.170	0.074
$SM_a$ and $BD_e$	0.000	0.006	0.081	0.359	0.322	0.159	0.073
$BD$ and $BD_e$	0.000	0.000	0.000	0.189	0.368	0.297	0.146

Table 2: Estimated marginal posterior distributions for  $r$

In many cases, the composition of the genome is determined by first choosing an appropriate number of segment types (here, seven or eight) and then using the segmentation  $\mathbf{s}$  conditional on this choice. However, as we have an estimate for the distribution of the number of segment types  $r$ , we can “integrate out” the uncertainty due to  $r$  and obtain the (posterior) mean base probabilities at each point along the genome. The lower right-hand plot in Figure 3 graphs these probabilities using the output from scheme (iii). It is reassuring that, even though the other schemes suggested a possibly different value for  $r$ , they produce almost identical mean base probability plots. Also, these plots exhibit a compositional structure which is similar to those produced in previous analyses of this genome: the first half of the sequence is reasonably homogeneous whereas the second half displays considerable heterogeneity.

## 6 Conclusion

Reversible jump techniques provide a useful method for gaining insight to the number of hidden states in a discrete-valued HMM. However, care is needed to ensure that appropriate moves are used to obtain a well-mixing MCMC sampler. Simple birth/death moves have been shown to work well in our application to DNA sequence analysis, but in other applications more complex split/merge moves can perform better.

## Acknowledgements

The work of DAH was supported by a Lord Adams Fellowship from the University of Newcastle.

## References

- Baum, L. E. and Petrie, T. (1966) Statistical inference for probabilistic functions of finite state Markov chains. *Annals of Mathematical Statistics*, **37**, 1554–1563.
- Baum, L. E., Petrie, T., Soules, G. and Weiss, N. (1970) A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. *Annals of Mathematical Statistics*, **41**, 164–171.
- Benson, D. A., Karsh-Mizrachi, I., Lipman, D. J., Ostell, J., Rapp, B. A. and Wheeler, D. L. (2000) GenBank. *Nucleic Acids Research*, **28**, 15–18.
- Boys, R. J., Henderson, D. A. and Wilkinson, D. J. (2000) Detecting homogeneous segments in DNA sequences by using hidden Markov models. *Applied Statistics*, **49**, 269–285.
- Braun, J. V. and Müller, H.-G. (1998) Statistical methods for DNA sequence segmentation. *Statistical Science*, **13**, 142–162.
- Brooks, S. P. (1998) Markov chain Monte Carlo method and its application. *Statistician*, **47**, 69–100.
- Brooks, S. P., Giudici, P. and Roberts, G. O. (2000) Efficient construction of reversible jump MCMC proposal distributions. Technical report 2000-10, University of Cambridge.
- Burge, C. and Karlin, S. (1997) Prediction of complete gene structures in human genomic DNA. *Journal of Molecular Biology*, **268**, 78–94.
- Cappé, O. (2001) *Ten years of HMMs*. On-line bibliography available from <http://tsi.enst.fr/~cappe/docs/hmmbib.html>.
- Cappé, O., Robert, C. P. and Rydén, T. (2001) Reversible jump MCMC converging to birth-and-death MCMC and more general continuous time samplers. Available from <http://www.ceremade.dauphine.fr/~xian/ctrjcmc.ps.gz>.
- Celeux, G., Hurn, M. and Robert, C. P. (2000) Computational and inferential difficulties with mixture posterior distributions. *Journal of the American Statistical Association*, **95**, 957–970.

- Chib, S. (1996) Calculating posterior distributions and modal estimates in Markov mixture models. *Journal of Econometrics*, **75**, 79–97.
- Chib, S. and Jeliazkov, I. (2001) Marginal likelihood from the Metropolis-Hastings output. *Journal of the American Statistical Association*, **96**, 270–281.
- Churchill, G. A. (1989) Stochastic models for heterogeneous DNA sequences. *Bulletin of Mathematical Biology*, **51**, 79–94.
- (1992) Hidden Markov chains and the analysis of genome structure. *Computers and Chemistry*, **16**, 107–115.
- Green, P. J. (1995) Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, **82**, 711–732.
- Henderson, D. A. and Boys, R. J. (2001) A Bayesian approach to DNA sequence segmentation using reversible jump MCMC methods. Statistics Preprint STA01.3, University of Newcastle, U.K.
- Li, W. (2001) *Bibliography on Features, Patterns, Correlations in DNA and Protein Sequences*. On-line bibliography available from [http://linkage.rockefeller.edu/wli/dna\\_corr/](http://linkage.rockefeller.edu/wli/dna_corr/).
- MacDonald, I. L. and Zucchini, W. (1997) *Hidden Markov and Other Models for Discrete-valued Time Series*. London: Chapman and Hall.
- Muri, F. (1998) Modelling bacterial genomes using hidden Markov models. In *COMPSTAT '98 Proceedings in Computational Statistics* (eds R. W. Payne and P. J. Green), pp. 89–100. Heidelberg: Physica-Verlag.
- Newton, M. A. and Raftery, A. E. (1994) Approximate Bayesian inference with the weighted likelihood bootstrap (with discussion). *Journal of the Royal Statistical Society, Series B*, **56**, 3–48.
- Rabiner, L. R. (1989) A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, **77**, 257–286.
- Richardson, S. and Green, P. J. (1997) On Bayesian analysis of mixtures with an unknown number of components (with discussion). *Journal of the Royal Statistical Society, Series B*, **59**, 731–792.
- Robert, C. P., Celeux, G. and Diebolt, J. (1993) Bayesian estimation of hidden Markov chains: A stochastic implementation. *Statistics and Probability Letters*, **16**, 77–83.
- Robert, C. P., Rydén, T. and Titterton, 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**, 57–75.
- Stephens, M. (2000a) Bayesian analysis of mixtures with an unknown number of components – an alternative to reversible jump methods. *Annals of Statistics*, **28**, 40–74.
- (2000b) Dealing with label switching in mixture models. *Journal of the Royal Statistical Society, Series B*, **62**, 795–809.
- Viallefont, V., Richardson, S. and Green, P. J. (2001) Bayesian analysis of Poisson mixtures. *Journal of Nonparametric Statistics*. To appear.