

Biometrika Trust

Population-Based Reversible Jump Markov Chain Monte Carlo

Author(s): Ajay Jasra, David A. Stephens and Christopher C. Holmes

Source: *Biometrika*, Vol. 94, No. 4 (Dec., 2007), pp. 787-807

Published by: Oxford University Press on behalf of Biometrika Trust

Stable URL: <http://www.jstor.org/stable/20441418>

Accessed: 26-02-2018 18:47 UTC

REFERENCES

Linked references are available on JSTOR for this article:

http://www.jstor.org/stable/20441418?seq=1&cid=pdf-reference#references_tab_contents

You may need to log in to JSTOR to access the linked references.

JSTOR is a not-for-profit service that helps scholars, researchers, and students discover, use, and build upon a wide range of content in a trusted digital archive. We use information technology and tools to increase productivity and facilitate new forms of scholarship. For more information about JSTOR, please contact support@jstor.org.

Your use of the JSTOR archive indicates your acceptance of the Terms & Conditions of Use, available at <http://about.jstor.org/terms>



Biometrika Trust, Oxford University Press are collaborating with JSTOR to digitize, preserve and extend access to *Biometrika*

Population-based reversible jump Markov chain Monte Carlo

By AJAY JASRA

Department of Mathematics, Imperial College London, London SW7 2AZ, U.K.
ajay.jasra@imperial.ac.uk

DAVID A. STEPHENS

Department of Mathematics and Statistics, McGill University, Montreal H3A 2K6, Canada
dstephens@math.mcgill.ca

AND CHRISTOPHER C. HOLMES

Department of Statistics, University of Oxford, Oxford OX1 3TG, U.K.
cholmes@stats.ox.ac.uk

SUMMARY

We present an extension of population-based Markov chain Monte Carlo to the transdimensional case. A major challenge is that of simulating from high- and transdimensional target measures. In such cases, Markov chain Monte Carlo methods may not adequately traverse the support of the target; the simulation results will be unreliable. We develop population methods to deal with such problems, and give a result proving the uniform ergodicity of these population algorithms, under mild assumptions. This result is used to demonstrate the superiority, in terms of convergence rate, of a population transition kernel over a reversible jump sampler for a Bayesian variable selection problem. We also give an example of a population algorithm for a Bayesian multivariate mixture model with an unknown number of components. This is applied to gene expression data of 1000 data points in six dimensions and it is demonstrated that our algorithm outperforms some competing Markov chain samplers. In this example, we show how to combine the methods of parallel chains (Geyer, 1991), tempering (Geyer & Thompson, 1995), snooker algorithms (Gilks et al., 1994), constrained sampling and delayed rejection (Green & Mira, 2001).

Some key words: Bayesian variable selection; Gene expression data; Mixture model; Reversible jump Markov chain Monte Carlo; Uniform ergodicity.

1. INTRODUCTION

We consider the problem of simulating from a probability measure $\pi(x)\lambda(dx)$ defined on measurable space $(E, \mathcal{E})$, where λ is a σ -finite measure on $\mathcal{E}$, with π known pointwise, at least up to a normalizing constant. This may be achieved by simulating an ergodic Markov chain $\{X_n\}_{n \geq 0}$, with kernel $K : E \times \mathcal{E} \rightarrow [0, 1]$, of stationary distribution π . The Markov chain can be used, for example, to estimate expectations of π -integrable functions. In this paper, we focus on the reversible jump Markov chain Monte Carlo (Green, 1995), where the state space is a union of subspaces of differing dimension, that is, where $E = \bigcup_{k \in \mathcal{K}} (\{k\} \times E^k)$, $\mathcal{K} \subseteq \mathbb{N}$, $E^k \subseteq \mathbb{R}^k$.

In statistical terms, π will often be a Bayesian posterior distribution, in contexts such as classification and regression (Denison et al., 2002) and mixture modelling (Richardson & Green,

1997). However, in many examples, the naive samplers fail to move around the support of the target in a feasible computation time; see Brooks et al. (2003) for examples.

To deal with these problems, several Markov chain Monte Carlo approaches have been suggested, including auxiliary variable methods (Brooks et al., 2003) and tempered transitions (Jennison et al., 2003); Green (2003b) provides a review. One approach used for difficult sampling problems in fixed-dimensional spaces relies on population-based Markov chain Monte Carlo methods (Liang & Wong, 2001; Liu, 2001). It is straightforward, conceptually, to extend this approach to the variable-dimension case. Such an extension is the focus of this paper; although we remark that methods other than Markov chain Monte Carlo may be used for difficult simulation problems, such as sequential Monte Carlo (Del Moral et al., 2006), but such methods are not the focus of this paper.

2. POPULATION-BASED MARKOV CHAIN MONTE CARLO

Population-based Markov chain Monte Carlo operates by embedding the target into a sequence of related probability measures $\{\pi_i\}_{i \in \mathbb{T}_N}$, $\mathbb{T}_N := \{1, \dots, N\}$, and simulating the N parallel chains, the population, as in parallel tempering (Geyer, 1991; Hukushima & Nemoto, 1996). In addition, the chains are allowed to interact via various crossover moves; a summary is given in §3 but see Liu (2001) for an extensive review.

The main advantage of population-based simulation over other methods is the fact that the population simultaneously represents many properties of the target distribution. This is particularly useful in transdimensional simulation, where it can be difficult to construct efficient dimension-changing proposals. Green (2003b) notes that some Markov chain Monte Carlo methods retain information about which states have been visited, as in the product space approach (Carlin & Chib, 1995; Godsill, 2001), whilst the standard reversible jump Markov chain Monte Carlo does not retain this information. There are advantages to both approaches. The first approach can provide an improved mode-jumping property, whereas the standard approach has greater capacity to discover new states that are consistent with the target. The objective of this paper is to develop an algorithm which combines these properties.

We shall use the following notation. A measurable space is denoted by $(E, \mathcal{E})$: throughout this article $\mathcal{E}$ is a countably generated σ -algebra. The N -fold product σ -algebra is written $\mathcal{E}^N$ and $\delta_x(dx')$ is used to represent the Dirac measure.

For two probability measures λ_1 and λ_2 on $\mathcal{E}$, the total variation distance is written as $\|\lambda_1 - \lambda_2\|_{\text{tv}} := \sup_{A \in \mathcal{E}} |\lambda_1(A) - \lambda_2(A)|$. The Dobrushin coefficient (Dobrushin, 1956) of a Markov kernel K is denoted by $\beta(K) := \sup_{(x,y) \in E^2} \|K(x, \cdot) - K(y, \cdot)\|_{\text{tv}}$. The composition of two Markov kernels, K and P , is written as $K \circ P(x, dy) := \int_E K(x, du)P(u, dy)$, except if $K = P$ in which case K^2 is used. Particular Markov kernels of interest are the product and mixture kernels that combine component kernels K_i in a multiplicative and additive fashion; these will be denoted by

$$\prod_i K_i, \quad \sum_i \tau_i K_i,$$

for mixture weights τ_i . Given a probability measure λ and Markov kernel K , the standard notation $\lambda K(dy) := \int_E \lambda(dx)K(x, dy)$ is adopted. For a bounded measurable function f , the oscillations are written $\text{osc}(f) = \sup_{(x,y) \in E^2} |f(x) - f(y)|$.

Finally, we use the vector notation $x_{1:N} := (x_1, \dots, x_N)$ and denote a vector with its i th element missing as x_{-i} and with only its i th and j th elements by $x_{i,j}$. Also, $\mathbb{T}_{l:p} := \{l, \dots, p\}$ for any $l \leq p$, $(l, p) \in \mathbb{Z}^2$.

3. AN ILLUSTRATIVE EXAMPLE: FINITE MIXTURE MODELLING

3.1. The model

Previous work on the Bayesian analysis using mixtures with an unknown number of components is described in Richardson & Green (1997), Stephens (2000) and Dellaportas & Papageorgiou (2006).

Let $y_{1:n}$ denote observed data with support $y_i \in Y \subseteq \mathbb{R}^r$, $i = 1, \dots, n$. We assume that the y_i are independent and identically distributed with density

$$p(y_i | \eta_{1:k}, w_{1:k}) = \sum_{j=1}^k w_j f(y_i; \eta_j),$$

where $\eta_{1:k}$ are component-specific parameters, the weights $w_{1:k}$ are such that $\sum_{j=1}^k w_j = 1$, $w_j \geq 0$ for $j \in \mathbb{T}_k$, p denotes an arbitrary probability density function and f is the component density. For our model, we restrict ourselves to the case of multivariate t , $T_r(\mu, \Lambda, s)$, where (μ, Λ) are the location and covariance parameters and s is the degrees of freedom.

In specifying the prior distributions, we follow Stephens (2000); the component mean vectors are taken to be independently $N_r(\xi, \kappa^{-1})$, and the Λ_j are independently $\text{IW}_r(2\alpha, 2\Psi)$, where $\text{IW}_r(\cdot, \cdot)$ is the inverse Wishart distribution. The following hierarchical structure is adopted:

$$\begin{aligned} \Psi &\sim W_r\{2g, (2h)^{-1}\}, & \text{where } W_r(\cdot, \cdot) \text{ is the Wishart distribution;} \\ w_{1:k-1} | k &\sim D(\delta), & \text{where } D(\cdot), \text{ the symmetric Dirichlet distribution;} \\ k &\sim U_{\{1, \dots, k_{\max}\}}, & \text{where } U_S \text{ is the discrete uniform distribution on the countable set } S. \end{aligned}$$

When we are using the multivariate t -distribution, the degrees of freedom are assumed to be known. Thus, our prior is

$$p(w_{1:k-1}, \mu_{1:k}, \Lambda_{1:k}, \Psi, k) = \left\{ \prod_{j=1}^k p(\mu_j) p(\Lambda_j | \Psi) \right\} p(\Psi) p(w_{1:k-1} | k) p(k).$$

3.2. Data processing and prior distributions

For our example, we consider the problem of clustering gene expression data (Heard et al., 2006). The data consist of the relative levels of gene expression, a measure of genetic activity, for $n = 4221$ genes of the parasite *Plasmodium* measured at $r = 46$ time-points across a 48-h portion of the parasite life cycle. The data are discussed in detail in Bozdech et al. (2003). Finding meaningful subgroupings of the data is an important task for biological investigators.

Even with modern computing power, it is not feasible to apply a fully Bayesian analysis to such data. Therefore, the data are preprocessed to reduce the $(n \times r)$ -dimensional data to $l \times q$ dimensions. This is achieved by adopting a K -means partitioning approach to reduce n to l , and then principal components to reduce r to q ; we use $l = 1000$ and $q = 6$ here.

Prior hyperparameters are set in a similar way to Stephens (2000); we set ξ to be the midpoint of the observed data in its corresponding dimension, and κ is taken to be $\text{diag}(1/R_1^2, \dots, 1/R_q^2)$ where R_d is the range of the data in dimension $d \in \mathbb{T}_q$. Also, $g = q/2$, $\delta = 1$ and $\alpha = \alpha' + (q + 1)/2$, where $\alpha' = 3$. Finally, h is $\text{diag}\{100q/(2\alpha R_1^2), \dots, 100q/(2\alpha R_q^2)\}$. For illustration, a t -distribution with four degrees of freedom is used as the component density in our mixture model and we set $k_{\max} = 20$.

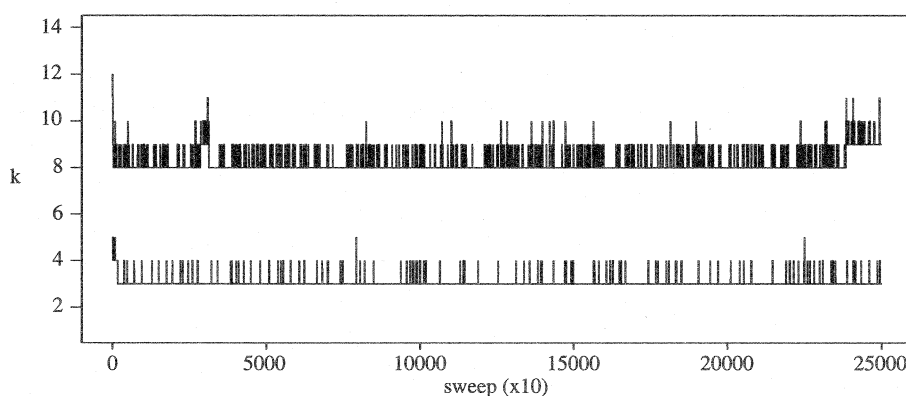


Fig. 1. Gene expression data. Sampled values of k using the standard reversible jump algorithm, for two chains started at different points. The algorithm was run for 250 000 sweeps and every 10th sample was taken.

3.3. Performance of standard sampler

We ran the reversible jump algorithm, outlined in Appendix 1, for 250 000 sweeps from two different starting points, which had different initial k . The C program was run on a Pentium 4, 3 GHz machine and took approximately three hours. The sampled values of k can be seen in Fig. 1; we can observe extremely poor mixing, with all variable-dimension acceptance rates below 1%. There appears to be the support for $k \in \mathbb{T}_{3:5}$ components, but also for $k \in \mathbb{T}_{8:11}$. The main problem is that the standard sampler cannot easily jump between these two modes.

To jump from a k - to a $(k + 1)$ -component mixture model, we need to draw $1 + q + q(q + 1)/2 = 28$ random variables, so we will need proposals/jump functions that are more tailored than those used in the basic sampler. In addition, if between-model jumps are made infrequently, a within-model chain spends a long time in a high-density region of the state space specific to that model which is consistent with the data. This renders a jump between models even less likely, as high-density regions in different models do not necessarily correspond to each other in a straightforward fashion. This heuristic argument applies to local moves, and, whilst more global moves might be constructed, and as noted by Green (2003a) they are more likely to produce better mixing than the local moves attempted here, dimension matching dictates that it will be difficult to achieve efficiently.

3.4. Alternative algorithms

Possible Markov chain Monte Carlo methods that might be used to deal with the problems encountered here include the auxiliary variable method of Brooks et al. (2003), but this will require a reasonable movement of the chain in the first place. Constructing proposal distributions by creating an approximation of the target in each dimension using fixed-dimensional Markov chain Monte Carlo, as described in D. Hastie's 2005 University of Bristol Ph.D. thesis, is complicated by the label-switching problem; see Jasra et al. (2005) for a discussion. Delayed rejection (Green & Mira, 2001) and tempered transitions (Jennison et al., 2003) often do not provide a general solution to the problems highlighted by this example. For the former method, insufficient information is learnt at the first-stage rejection to provide a significantly improved second-stage proposal, whereas, for the latter, it often takes a large number of intermediate simulations, such as 100, to provide a reasonable proposal, but even so the performance gain is not always substantial.

4. POPULATION-BASED REVERSIBLE JUMP

4.1. The population Markov chain Monte Carlo method

Consider a sequence of probability measures $\{\pi_i\}_{i \in \mathbb{T}_N}$, each assumed to admit a density, also denoted by π_i , with respect to λ on $(E, \mathcal{E})$. Denote by E_i the support of the i th density $E_i := \{(x_k, k) \in E : \pi_i(x_k, k) > 0\}$, $i \in \mathbb{T}_N$. Define the probability measure $\pi^*\{d(x_{1:k_1}, k_1, \dots, x_{1:k_N}, k_N)\}$ on $(E^N, \mathcal{E}^N)$ as

$$\pi^*\{d(x_{1:k_1}, k_1, \dots, x_{1:k_N}, k_N)\} = \left[\prod_{i=1}^N \pi_i(x_{1:k_i}, k_i) \lambda\{d(x_{1:k_i}, k_i)\} \right].$$

Our objective is now to construct an ergodic Markov kernel $\tilde{K} : E^N \times \mathcal{E}^N \rightarrow [0, 1]$ with stationary distribution π^* .

4.2. The structure of the population

In the population approach, we generate N parallel chains in order to explore the target correctly. For the remainder of the paper $\pi_1 \equiv \pi$ and the sequence of densities are taken to be $\pi_i \propto \pi^{\zeta_i}$, $1 = \zeta_1 > \dots > \zeta_N > 0$, where $\{\zeta_i\}_{i \in \mathbb{T}_N}$ are inverse temperature parameters; the collection $\{\zeta_i\}$ is referred to as the temperature ladder. This is the approach in Liang & Wong (2001), but other settings include taking $\zeta_i = 1$ for each i ; see Del Moral et al. (2006) for further discussion. We seek to use the extra information contained in N chains at different temperatures to allow large moves in dimension of the chain of interest as well as allowing improved performance in more local moves, within and between dimensions.

One of the main problems of parallel tempering (Geyer, 1991; Hukushima & Nemoto, 1996) is that only minimal interactions between the chains are allowed. Our approach differs as we will allow $\tilde{K}$ to include moves which use the entire population, other than merely the exchange move; see §4.4. Thus, we seek a more population-based approach to justify the increased cost in computation.

4.3. Some theory for Markov chains

As our results are not confined to the transdimensional case, k is dropped from the notation; that is, E is a general state space associated with countably generated σ -algebra $\mathcal{E}$.

We will concentrate on the concept of uniform ergodicity; see Roberts & Rosenthal (2004) and the references therein. The following property of uniformly ergodic Markov kernels K is used; the total variation distance between the n -step kernel and its stationary measure π ($\|K^n(x, \cdot) - \pi(\cdot)\|_{\text{TV}}$) is bounded above by

$$R = (1 - \epsilon)^{\lfloor n/n_0 \rfloor},$$

where n_0 and ϵ are parameters in the following minorization condition for $C = E$: there exists $\epsilon \in (0, 1)$ such that, for ν some nontrivial probability measure and integer $n_0 > 0$ and for all $(x, A) \in C \times \mathcal{E}$,

$$K^{n_0}(x, A) \geq \epsilon \nu(A). \quad (1)$$

If $C \in \mathcal{E}$ is such that (1) is satisfied, then we term it (n_0, ϵ, ν) -small.

It follows that, to compare convergence rates of two uniformly ergodic Markov kernels, we can compute the pair (n_0, ϵ) ; if one of the kernels has a substantially larger ϵ , and smaller n_0 , then we might expect it to converge more quickly to the target of interest. Note that spectral gap techniques, see for example Diaconis & Saloff Coste (1993), can also be considered; see §5.3.

4.4. *Main result*

We now demonstrate that population Markov chain Monte Carlo approaches can be defined, using specially constructed component kernels, so that they are uniformly ergodic. For the following theorem, we denote a mutation kernel by K_M and an exchange kernel by K_E . A ‘mutation’ move is a Metropolis–Hastings kernel which attempts to change a single member of the population, dependent only on the current state of the chain for that member. An ‘exchange’ move is a Metropolis–Hastings kernel which proposes to swap the current states of two different members of the population; mathematically, it is defined as $K_E^{(il)} : E^2 \times \mathcal{E}^2 \rightarrow [0, 1]$, such that

$$K_E^{(il)}(x_{i,l}, dx'_{i,l}) = \min \left\{ 1, \frac{\pi_i(x_l)\pi_l(x_i)}{\pi_i(x_i)\pi_l(x_l)} \right\} \delta_{x_l}(dx'_i) \delta_{x_i}(dx'_l) \\ + \left[1 - \min \left\{ 1, \frac{\pi_i(x_l)\pi_l(x_i)}{\pi_i(x_i)\pi_l(x_l)} \right\} \right] \delta_{x_i}(dx'_i) \delta_{x_l}(dx'_l).$$

For simplicity, we assume that $E_i = E$, $i \in \mathbb{T}_N$. We now give our main theoretical result, the proof of which is in Appendix 2.

THEOREM 1. *Let $\tilde{K}$ be a Markov kernel defined on a measurable space $(E^N, \mathcal{E}^N)$ such that*

$$\tilde{K}(x_{1:N}, dx'_{1:N}) = \{\tau K_M + (1 - \tau)K_E\}(x_{1:N}, dx'_{1:N}), \\ K_M(x_{1:N}, dx'_{1:N}) = \prod_{i=1}^N K_i(x_i, dx'_i), \\ K_E(x_{1:N}, dx'_{1:N}) = \sum_{i=1}^{N-1} \sum_{l=i+1}^N \varepsilon_{il} K_E^{(il)}(x_{i,l}, dx'_{i,l}) \delta_{x_{-(i,l)}}(dx_{-(i,l)}),$$

where

$$\tau, \varepsilon_{il} \in (0, 1) \quad \sum_{i=1}^{N-1} \sum_{l=i+1}^N \varepsilon_{il} = 1$$

and, for each $i \in \mathbb{T}_N$, K_i is an aperiodic, λ -irreducible Markov kernel with invariant measure $\pi_i(x)\lambda(dx)$. Suppose that K_{j^*} is uniformly ergodic for one $j^* \in \mathbb{T}_N$ and for each $i \neq j^*$ there exists $\varrho_i \in (0, \infty)$ such that $\pi_i(x) \leq \varrho_i \pi_{j^*}(x)$ for all $x \in E$. Then $\tilde{K}$ is uniformly ergodic.

Remark 1. The assumption of uniform ergodicity for at least one K_{j^*} is not very restrictive. In many applications, such as Bayesian analyses with a proper prior and a bounded likelihood, a proposal under an independence kernel with proposal density q , where $\pi(x)/q(x) \leq \varrho$ for all $x \in E$ (Tierney, 1994; Mengersen & Tweedie, 1996), can be found, for example where q is the prior density, which ensures uniform ergodicity, at the cost of being a poor proposal for π . However, if applied to a related distribution in the population, this proposal may perform quite well. The assumption that $\pi_i(x) \leq \varrho_i \pi_{j^*}(x)$ for all $x \in E$ is quite reasonable and would apply in the framework described here, when π is bounded, for the case $j^* = N$. Crucially, for small N , we can establish that the fast rate of convergence for one of the kernels is propagated through the population; see §5.2. We note that the result holds for the case where K_M is a mixture of kernels, but have omitted the details for brevity.

Remark 2. Theorem 1 shows in simple cases where we can design uniformly ergodic chains for the target π which are likely to perform well in practice, how to compare population and single-chain approaches. In other words, we can construct a population sampler which has a faster rate of convergence to π^* , and hence π , which justifies the increased cost in computation. Furthermore, we can investigate the population kernels, such as $\tilde{K}$, which are likely to provide good mixing for more complex examples. However, because of the limitations of investigating the minorization conditions, it is difficult to use this result when N is large; we discuss this further in §5.3.

4.5. Impact of the exchange kernel

In the following result, proved in Appendix 2, we show that the exchange kernel, which is reducible, can indeed improve the rate of convergence relative to a parallel Markov chain Monte Carlo algorithm with no interaction between the chains. To study this phenomenon, we introduce the following mixing condition (e.g. Del Moral, 2004).

Condition 1. For the Markov kernel $K : E^N \times \mathcal{E}^N \rightarrow [0, 1]$, there exists $\epsilon > 0$ such that, for all $(x, y) \in E^{2N}$,

$$K(x, \cdot) \geq \epsilon K(y, \cdot). \quad (2)$$

Let $\mathbb{T}_N^c = \{(i, l) : i, l \in \mathbb{T}_N, i \neq l\}$.

PROPOSITION 1. Assume that K_M satisfies Condition 1. Then, for any initial distribution η and any $n \geq 1$,

$$\|\eta(K_M \circ K_E)^n - \pi^*\|_{\text{tv}} \leq \{2(1 - \alpha)(1 - \epsilon)\}^n \|\eta - \pi^*\|_{\text{tv}},$$

where $\alpha = \sum_{(i,l) \in \mathbb{T}_N^c} \varepsilon_{il} \{1 - \inf_{(x_i, x_l) \in E^2} \rho_{i,l}(x_i, x_l)\}$ and $\rho_{i,l}(x_i, x_l) = \min\{1, \pi_i(x_l)\pi_l(x_i)/\pi_i(x_i)\pi_l(x_l)\}$.

Remark 3. The result provides a sufficient condition for improving upon parallel Markov chain Monte Carlo, in terms of a tighter bound on the total variation distance. For the kernel K_M the bound is, under Condition 1, $(1 - \epsilon)^{3n/2} \|\eta - \pi^*\|_{\text{tv}}$; the factor $3/2$ is used to make a fair comparison, in terms of the CPU time. The condition is that, if the exchange probabilities are lower-bounded by $1 - \frac{1}{2}(1 - \epsilon)^{1/2}$, then the bound is tighter with an exchange step. Essentially, it is crucial to select a good sequence of densities and use population moves to improve convergence. As noted by a referee, this indicates that, if ϵ is small, that is K_M is poorly mixing, the exchange step can help to improve the convergence rate. In addition, if for all $(x, y) \in E^2$ and $(i, l) \in \mathbb{T}_N^c$ we have $q_L \leq \pi_i(x)/\pi_l(y) \leq q_U$, then the lower bound may be achieved if $q_L/q_U \geq 1 - \frac{1}{2}(1 - \epsilon)^{1/2}$, which suggests that we would require the densities to be similar, to improve convergence rates.

Remark 4. The result is only of use on compact spaces, as in the finite state space variable selection example in §5. In this scenario, it is possible to lower-bound acceptance probabilities and apply the mixing condition, Condition 1. However, even on compact spaces, it may be difficult to verify Condition 1 for Metropolis–Hastings kernels because of the rejection probability. This can be circumvented by iterating the kernel.

5. EXAMPLE: BAYESIAN VARIABLE SELECTION

5.1. *Model and data*

Consider the statistical model

$$y_i = \gamma_0 + \sum_{j=1}^{k_{\max}} \vartheta_j \gamma_j x_{ij} + \varpi_i,$$

with ϖ_i independently and identically distributed as $N(0, \sigma^2)$, $\vartheta_j \in \{0, 1\}$ and $\gamma_j \in \mathbb{R}$. If we consider the conjugate prior specification with $\gamma_{0:k} | \sigma, k \sim N_{k+1}(m, \sigma^2 V)$, $\sigma^2 \sim \text{IG}(a, b)$, where $\text{IG}(\cdot, \cdot)$ is the inverse Gamma distribution, and

$$p(\vartheta_{1:k_{\max}}, k) = \frac{1}{k_{\max} + 1} \binom{k_{\max}}{k}^{-1} \mathbb{I}_{S_k}(\vartheta_{1:k_{\max}}), \quad k = 0, \dots, k_{\max},$$

where $S_k = \{\vartheta_{1:k_{\max}} \in \{0, 1\}^{k_{\max}} : \sum_{j=0}^{k_{\max}} \vartheta_j = k\}$, then we can integrate out the parameters $(\sigma, \gamma_{0:k})$ and sample from a distribution on a finite state space.

We generated 100 data points from a linear model, with $k_{\max} = 8$. The posterior probability of a null model was 0.55 and that for the saturated model was 0.33. This is a typical situation for which a standard Markov chain Monte Carlo sampler would fail to move around the state space easily.

5.2. *Comparison of convergence rates*

To sample from the posterior distribution, we use a Markov chain Monte Carlo algorithm described in Denison et al. (2002, p. 53) modified for the variable selection case.

Since the state space is finite, it is clear that an ergodic Markov chain with appropriate stationary distribution is uniformly ergodic. To construct an appropriate ν in (1), that is, one that leads to a large ϵ , we used the approach discussed in Robert & Casella (2004, Ch. 6). Let K_{ij}^n ($i, j \in E$) denote the n -step transition probability and suppose that $\inf_i \{K_{ij}^n\} > 0$ for some j . Then for all j

$$K_{ij}^n \geq \inf_i \{K_{ij}^n\} = \epsilon \nu_j$$

say, where

$$\nu_j = \frac{\inf_i \{K_{ij}^n\}}{\sum_{l \in E} \inf_i \{K_{il}^n\}}, \quad \epsilon = \sum_{l \in E} \inf_i \{K_{il}^n\}.$$

For the algorithm discussed above, we found that the bound on the rate of convergence was reasonably constant in the range $n_0 \in \mathbb{T}_{1000:5000}$; we focus upon $(n_0, \epsilon) = (1000, 3.63 \times 10^{-3})$. To make the analysis computationally comparable to the population algorithm described below, we let 50 applications of this kernel be equivalent to a single step; this new kernel has $(20, 3.63 \times 10^{-3})$ as the (n_0, ϵ) pair.

For a population sampler, suppose we take a single auxiliary distribution

$$\pi_2(\vartheta_{1:k_{\max}}, k | x_{1:n}, y_{1:n}) \propto L(x_{1:n}, y_{1:n}; \vartheta_{1:k_{\max}})^\zeta p(\vartheta_{1:k_{\max}}, k),$$

with $\zeta = 0.01$, and let L be the likelihood function. The choice of 0.01 is used for illustration, and is adopted to demonstrate the impact of the use of a related, but easier to sample, distribution in the population. We concentrate upon a kernel which updates both chains via the Markov chain Monte Carlo algorithm mentioned above for 10 sweeps, followed by an exchange, and then

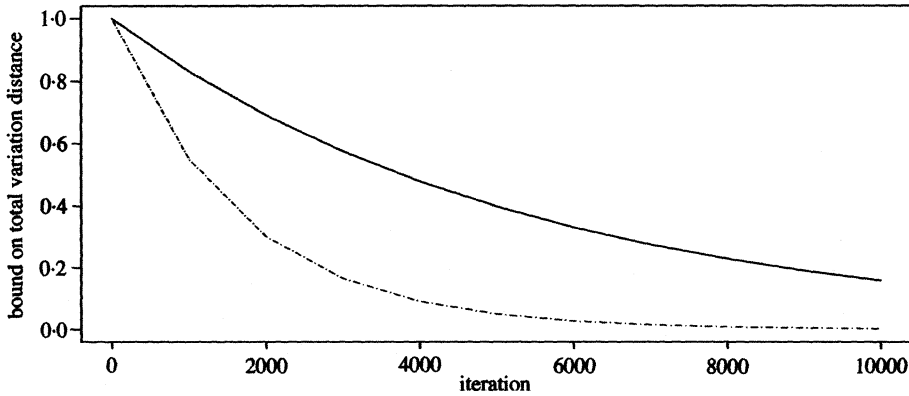


Fig. 2. Plot of the bound on the total variation distance for sampling from a Bayesian linear model, for a Markov chain Monte Carlo algorithm similar to that in Denison et al. (2002), solid, and a population algorithm, dashed.

another 10 sweeps. This corresponds to the kernel from which we sample; that is, it is a single time step.

It can be shown that the (n_0, ϵ) pair for the population sampler is $(1, 6.01 \times 10^{-4})$. This was computed by finding the ϵ and ν in the minorization condition for π_2 , and then

$$\phi = \sum_{(\vartheta_{1:k_{\max}}, k) \in E} \nu(\vartheta_{1:k_{\max}}, k) \min \left\{ 1, \frac{\pi_1(\vartheta_{1:k_{\max}}, k)}{\pi_2(\vartheta_{1:k_{\max}}, k) \varrho_1} \right\},$$

$$\varrho_1 = \sup_{(\vartheta_{1:k_{\max}}, k) \in E} \frac{\pi_1(\vartheta_{1:k_{\max}}, k)}{\pi_2(\vartheta_{1:k_{\max}}, k)}.$$

The constant in the minorization condition is then $\epsilon^2 \phi$. A graph comparing the bounds on total variation distance, R , can be seen in Fig. 2, suggesting a much faster rate of convergence for the population algorithm.

This example demonstrates that a simple extension of the original algorithm to include an auxiliary distribution that provides a good proposal, for the original target, allows an efficient movement around the state space; that is, it reiterates the point in Remark 1.

5.3. Discussion

A point of interest, raised by a referee, is that as N increased we found it difficult to find a Markov kernel that could improve upon the rate of convergence to stationarity when $N = 2$. This can be explained as follows. Suppose, for illustration, we run N parallel chains with the same marginal invariant measure $\pi(x)\lambda(dx)$ with kernel $K_M \circ K_E \circ K_M$ in the proof of Theorem 1. Assume that all the mutation kernels, except one, are strongly aperiodic ($n_0 = 1$) and mix almost perfectly, i.e. are uniformly ergodic with $\epsilon \simeq 1$. Then we can easily establish that the constant in the minorization condition is $\epsilon^N \rightarrow 0$ as $N \rightarrow \infty$, so that $R \simeq 0$. This establishes two points.

First, continually extending the state space will inevitably lead to slower convergence of the Markov chain, unless very efficient population moves can be constructed. That is, we should not naively extend the state space and expect our Markov chain to converge more quickly; unlike the convergence of particle algorithms (Del Moral, 2004), there is not necessarily an improved convergence property as $N \rightarrow \infty$. This is an open problem for future research.

Secondly, in cases in which R is not informative about the rate of convergence, it may be more beneficial to investigate other properties of the Markov kernel, such as the spectral gap. Such an

analysis is likely to be far more complicated than discussed here; see Madras & Zheng (2003) for an example.

6. POPULATION MOVES FOR THE MIXTURE EXAMPLE

6.1. Exchange moves

Our notation is such that $\theta_i = (\eta^i, w^i, \Psi^i, k_i)$, $i \in \mathbb{T}_N$ and (η^i, w^i) refers to all the component-specific parameters and weights for chain i . In our tempering approach, π^{ζ_i} , we will temper the likelihood terms only, rather than the full posterior, to avoid any integrability problems.

An exchange move is used to swap information between two different parallel tempered chains. In our strategy, at iteration t , we select two chains that are adjacent in terms of the temperature parameter, uniformly at random, and propose to swap their values. In order to achieve a reasonable interaction between the chains, the temperature ladder is set so that this move is accepted about half of the time (Liu, 2001); see §7 for further discussion.

One way of improving this move is to use the delayed rejection method, as suggested by Green & Mira (2001). At iteration t , we select any two chains i_1 and i_2 , accepting or rejecting with the usual Hastings ratio, that is, with probability

$$\rho_1(\theta_{1:N}, \theta'_{1:N}) = \min \left\{ 1, \frac{\pi_{i_1}(\theta_{i_2})\pi_{i_2}(\theta_{i_1})}{\pi_{i_1}(\theta_{i_1})\pi_{i_2}(\theta_{i_2})} \right\},$$

where the labelling of the chains is with respect to the current state of the chain and where $\theta'_{1:N}$ denotes the new configuration of chains. If this is rejected, we select two adjacent chains i_3 and i_4 to swap, denoting this configuration by $\theta''_{1:N}$. To ensure reversibility with respect to the target, as part of the delayed rejection method, we construct a pseudo move which consists of starting from the second-stage proposed state $\theta''_{1:N}$, proposing to move to $\theta^*_{1:N}$, which swaps θ''_{i_1} and θ''_{i_2} , and rejecting it with probability $\rho_1(\theta''_{1:N}, \theta^*_{1:N})$. The second-stage move is accepted with probability

$$\rho_2(\theta_{1:N}, \theta''_{1:N}) = \min \left\{ 1, \frac{\pi_{i_3}(\theta_{i_4})\pi_{i_4}(\theta_{i_3})\{1 - \rho_1(\theta'', \theta^*)\}}{\pi_{i_3}(\theta_{i_3})\pi_{i_4}(\theta_{i_4})\{1 - \rho_1(\theta, \theta')\}} \right\}.$$

This move allows for an increased interaction within the population. At the first stage, we allow any pair of chains to be swapped, so that, if a state of a chain at a high temperature is consistent with one of the distributions at a lower temperature, it is allowed to quickly jump down the ladder.

6.2. Crossover moves

Liang & Wong (2001) employ various crossover moves in an evolutionary Monte Carlo algorithm; the objective is to increase the interaction within the population. In our algorithm, we use two move types: variable-dimensional and fixed-dimensional crossover.

To construct a variable-dimension crossover move that is likely to have high acceptance probability, we proceed as follows. The move begins by reassigning mixture component labels in each chain so that they satisfy an ordering constraint on the weights, in order approximately to match the labels of components in different chains. When the chain is in state θ , we select a variable-dimension crossover with probability $v(\theta)$, where

$$v(\theta) = \begin{cases} 1 & \text{if } k_i \neq k_j \text{ for some } i \neq j \\ 0 & \text{otherwise.} \end{cases}$$

Note that the case $v(\theta) = 0$ corresponds to a ‘do nothing’ move. We select a pair of chains with different dimensions with probability inversely proportional to the squared difference between the

dimensions. We then propose the new states of the population members, by swapping k 's and the weights. We take the lowest-weighted component-specific parameters of the higher-dimensional chain to the lower-dimensional chain; that is, if $k_{i_1} > k_{i_2}$ for the selected chains i_1, i_2 , we propose

$$\begin{aligned}\eta'_{i_1} &= (\eta_{k_{i_1}-k_{i_2}}^{i_1}, \dots, \eta_{k_{i_1}}^{i_1}), \\ \eta'_{i_2} &= (\eta_1^{i_1}, \dots, \eta_{k_{i_1}-k_{i_2}-1}^{i_1}, \eta_1^{i_2}, \dots, \eta_{k_{i_2}}^{i_2}),\end{aligned}$$

where $\eta_j^{i_1}$ denotes an element of η^{i_1} . The acceptance probability is easily calculated and is thus omitted. After the move has been accepted or rejected, or the 'do nothing' move, we propose and accept a uniformly random permutation of the labels of the parameters of all the chains. This final permutation ensures invariance with respect to the target.

In the fixed-dimensional crossover, we begin by reassigning the mixture component labels in each chain by ordering on the first dimension of the means. When the chain is in state θ , we select a fixed-dimension crossover with probability 1, if it can be selected, i.e. there are at least two chains with the same dimensionality; otherwise, we select a 'do nothing' move. Select a pair of chains (i_1, i_2) with the same dimension, with probability

$$p(i_1, i_2 | \theta) \propto |\zeta_{i_1} - \zeta_{i_2}|^{-1} \mathbb{I}_{k_{i_1}=k_{i_2}}. \quad (3)$$

We select a position $j = 1, \dots, k_{i_1} - 1$ to cross over, this selection being made with probability proportional to $1/j$, and switch all component-specific parameters to the left of j inclusive, noting that if the identifiability constraint is not satisfied in the proposed state of the chain we immediately reject, and accept/reject on the basis of the Hastings ratio. After the accept/reject decision has been made, we again propose a random permutation of the labels of the parameters.

6.3. Snooker jumps

One of the most important ways in which we can use the information in the population is by targeting variable-dimension jumps by using another chain. This idea is linked to the snooker algorithm of Gilks et al. (1994) and is performed in the following way. When in state θ , we select a birth with probability $b(\theta)$, where

$$b(\theta) = \begin{cases} 1 & \text{if } k_i = 1 \text{ for all } i \\ 0 & \text{if } k_i = k_{\max} \text{ for all } i \\ 1/2 & \text{otherwise,} \end{cases}$$

then we select a chain at random, the current point θ_c , for which a birth is possible, letting $m_b(\theta)$ be the number of chains such that a birth can occur when in state θ , and we select an anchor point θ_a with the probability inversely proportional to the absolute value of the difference between the inverse temperatures. We then generate $w \sim \text{Be}(1, k_c)$, with $\text{Be}(\cdot, \cdot)$ the beta distribution, and draw a new (μ, Λ) pair from

$$q(\mu, \Lambda) = \sum_{j=1}^{k_a} \bar{h}(\eta_j) N_r(\mu_j^a, \sigma) \text{IW}(2r + 3, \Lambda_j^a),$$

where

$$\bar{h}(\eta_j) \propto \frac{1}{k_c} \sum_{l=1}^{k_c} h\{(\mu_j^a, \Lambda_j^a), (\mu_l^c, \Lambda_l^c)\}$$

and $h(\cdot, \cdot)$ is the Mahalanobis distance. We then perform the rest of the move as for the birth in Appendix 1. In the death move, we perform much the same as for Appendix 1, except that we

select a current point with probability $1/m_d(\theta)$, where $m_d(\theta)$ is the number of chains for which a death can occur when in state θ , and select an anchor point which is used in the reverse birth. The birth move is accepted with probability $\min\{1, A\}$, with

$$A = \frac{p(y_{1:n}|\eta'^c, w'_{1:k_c}, k_c + 1)^{\zeta_c} p(\mu)p(\Phi)p(k_c + 1)}{p(y_{1:n}|\eta^c, w_{1:k_c-1}^c, k_c)^{\zeta_c} p(k_c)} B(k_c\delta, \delta)^{-1} w^{\delta-1} (1-w)^{k_c(\delta-1)} \frac{(k_c + 1)!}{k_c!} \\ \times \frac{d(\theta')m_b(\theta)}{(k_c + 1)b(\theta)m_d(\theta')} \frac{(1-w)^{k_c-1}}{\text{Be}(w; 1, k_c)q(\mu, \Phi)},$$

where Φ is the Cholesky decomposition of Λ , $p(y_{1:n}|\eta^c, w_{1:k_c-1}^c, k_c)^{\zeta_c}$ is the tempered likelihood for the current point, $B(\cdot, \cdot)$ is the beta function and $\text{Be}(x; \cdot, \cdot)$ is the beta density evaluated at x . The objective of this move is to propose new component-specific parameters which are likely to be consistent with the data, but are markedly different from the current components. It also provides an adaptive element to the birth proposal, as it relies on current information, in the population, that is being continuously updated.

6.4. Constrained sampling

One aspect of population-based simulation that is apparent is the need to maintain diversity of the population, as in sequential Monte Carlo; see Del Moral et al. (2006). In many cases for which it is difficult to traverse the state space, the chains at lower temperatures can become trapped, i.e. stuck in local modes, as for single chain Markov chain Monte Carlo, and this may lead to inaccurate Monte Carlo estimates of quantities of interest. To avoid this problem, we propose to constrain some of the members of the population; that is, for some subset $\mathbb{T}_{l:N}$ ($l \geq 2$) and $i \in \mathbb{T}_{l:N}$, π_i is a density constrained to $E_i \subset E$. In the setting of transdimensional problems, the E_i may be selected with respect to the model dimension; for example, $E_i = \bigcup_{k \in \mathcal{K}_i} (\{k\} \times E^k)$ for $\mathcal{K}_i \subset \mathcal{K}$ and $i \in \mathbb{T}_{l:N}$. In general, choosing an appropriate E_i is challenging; see an unpublished University of Ottawa technical report by Y. F. Atchadé and J. S. Liu for some discussion. This technique will prove to be very important in §7.

6.5. The algorithm

To sample from the augmented distribution, we use the algorithm below; we use the genetic algorithm terminology of Liang & Wong (2001).

ALGORITHM

Step 0. Initialize the chain θ and then for $t = 1, \dots, M$ iterate the following steps.

Step 1. Mutation. Select a chain $i \in \{1, \dots, N\}$ with probability τ_i and then perform one sweep of the reversible jump algorithm in Appendix 1 for this chain, with appropriate modification for constrained targets.

Step 2. Make a random choice between performing Step 3 or Step 4.

Step 3. Crossover. Propose a variable-dimension crossover move with probability $1/2$, or otherwise propose a fixed-dimension crossover.

Step 4. Snooker jump. Propose a birth with probability $b(\theta)$, or otherwise propose a death.

Step 5. Exchange. Perform the delayed rejection exchange move.

Note that, for constrained chains, we only allow them to be involved in fixed-dimensional crossovers and a special exchange move that augments Step 5; we propose to exchange a constrained and non-constrained chain, selecting the move only if such a move may be performed;

that is, if $k_i \in \mathcal{K}_j$ and $k_j \in \mathcal{K}_i$ for $i \neq j$. All selections are made with uniform probability and no delayed rejection is used.

7. GENE EXPRESSION EXAMPLE REVISITED

7.1. *Specification of simulation parameters*

To run the population algorithm in §6.5 we used $N = 25$ with 5 chains constrained. We recommend a large population size in general, although the discussion in §5.3 should be considered, so that results are reasonably similar for separate runs of the algorithm; for more guidance, see Jasra et al. (2007).

For the main population, the unconstrained chains, the following inverse temperatures were selected:

$$\zeta_1 = 1, \zeta_i = \zeta_{i-1} - \varsigma\varphi^{i-1}, i \in \mathbb{T}_{2:20},$$

for constants $\varsigma > 0$, $\varphi > 1$. We selected $\varsigma = 10^{-6}$ and $\varphi = 1.85$, so that $\zeta_{20} = 0.74$. Our choice of cooling schedule and population size was based upon pilot tuning. We selected a slowly decreasing sequence of ζ 's since we observed a poor acceptance rate for the exchange move for distributions that were further away from each other. We found that the inverse temperature at which the reversible jump algorithm performed best, that is, achieving reasonable acceptance rates along with the regions of high support that were similar to the target, was $\zeta = 0.75$; thus, we attempted to include a distribution with this temperature. Note that we need to be careful when specifying temperatures, since for 'low' temperatures the distribution starts to favour dimensionalities that are small, although this may be alleviated by specifying priors for k which penalize small values. For this particular problem, we spent at least two to three hours in tuning the parameters; a more automatic procedure may be implemented as described below.

The term 'reasonable acceptance rate' deserves a quantitative definition. A useful criterion put forward by Iba (2001) is that the expected, with respect to the target, log Hastings ratio of the exchange probability is equal to 1. This in turn means that, approximately, the algorithm will accept the exchange about half the time. Such an approach may be used to provide an automatic temperature selection and will vastly reduce the time spent on tuning the algorithm; as noted by a referee, this aspect should be taken into account in the computational cost and comparison of the algorithm. We refer the reader to Iba (2001) for the details and to Goswami & Liu (2007) for an alternative approach.

To select the constrained sets, E_i , $i \in \mathbb{T}_{21:25}$, we used a pilot simulation. We ran the algorithm with $N = 25$ and the inverse temperature parameters discussed above but with only 9 chains in the main population and 16 constrained chains which were given the inverse temperature parameter 0.999. We selected the subspaces E_i with respect to the dimensionality; that is, we had 10 chains constrained to lie in $k \in \mathbb{T}_{1:2}, \dots, \mathbb{T}_{19:20}$ and then 6 other chains constrained to lie in $k \in \mathbb{T}_{3:6}, \mathbb{T}_{6:9}, \mathbb{T}_{9:12}, \mathbb{T}_{12:15}$ and $\mathbb{T}_{15:18}$. This was adopted in order to determine whether or not there was any support outside $\mathbb{T}_{3:11}$ found in §3.3. Based upon a short run, the five constrained chains were taken as $\mathbb{T}_{2:4}, \mathbb{T}_{4:6}, \mathbb{T}_{5:7}, \mathbb{T}_{7:9}$ and $\mathbb{T}_{9:11}$. The idea of the pilot tuning is to avoid wasting the CPU time on population members constrained to lie in areas of support that have low density with respect to the original target density. Furthermore, the constrained chains need to be able to interact with the main population and the preliminary tuning allows us to make this choice. The inverse temperature parameters for the constrained chains were 0.999, since we seek to maintain diversity with respect to the population at colder temperatures. For further discussion on the setting of partitions, especially in the context of reversible jump, see the technical report of Atchadé and Liu.

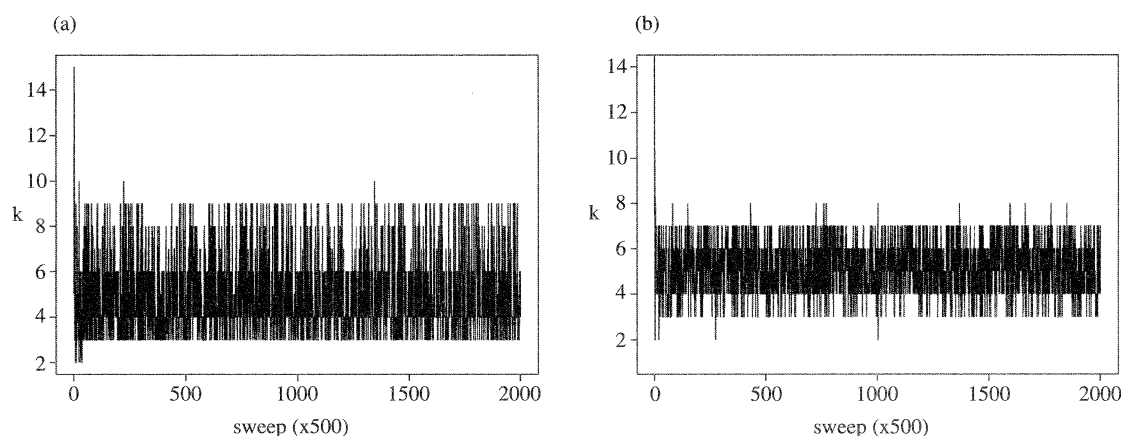


Fig. 3. Gene expression example. Sampled k (a) and sampled k without constrained targets (b) from the population-based reversible jump algorithm in §6.5. The algorithm was run for 1 million sweeps and every 500th sample was retained.

The algorithm in §6.5 was run for 1 million sweeps, which took approximately $9\frac{1}{2}$ hours. A minor modification to the algorithm was made: if a crossover move was selected, we also proposed an exchange for the chain of interest.

7.2. Comparison with the standard sampler

Figure 3(a) shows sampled values of k for the original target density. The improvement over the standard sampler is substantial: on average, the chain of interest took 6.75 sweeps to jump between a mixture model with fewer than four components to a mixture with more than six.

The observed inability of the standard reversible jump algorithm to move around the state space, i.e. from $k \in \mathbb{T}_{3;5}$ to $k \in \mathbb{T}_{8;11}$, is not present for the population sampler, since it may represent both parts of the space simultaneously. Note that, because of the complexity of the target, we cannot claim that the sampler has converged; there may be regions of high posterior density that are still unexplored.

Figure 3(b) displays the sampled k for a population-based algorithm with no constrained target. It is apparent that, despite the substantially improved performance when compared to the standard sampler, the sampler has missed the region $k \in \{8, 9, 10\}$. This occurs for the reasons discussed earlier, and demonstrates that constrained chains can help guard against such problems.

The effective sample size, ESS, is a standard measure of the relative efficiency of a Markov chain Monte Carlo sampler; see Liu (2001) for a definition. For our population algorithm, the effective sample size for k was 59 745 corresponding to 60 000 samples, using a lag of 10 in the autocorrelation calculation, compared with 2998 and 3591 for both the standard algorithms. Define

$$F := 2 \frac{\text{ESS}_{\text{pop}}}{M_{\text{pop}} T_{\text{pop}}} \bigg/ \left(\frac{\text{ESS}_{\text{stan}_1}}{M_{\text{stan}_1} T_{\text{stan}_1}} + \frac{\text{ESS}_{\text{stan}_2}}{M_{\text{stan}_2} T_{\text{stan}_2}} \right),$$

where the subscripts refer to the population, standard algorithm runs 1 and 2, i.e. the chains run in §3.3, respectively, M is the sample size and T is the CPU time. We obtain $F = 2.07$; similar conclusions are drawn when the tuning time is taken into account. Thus, the population approach is far superior; note that all coincidental simulation parameters are the same between algorithms.

7.3. *The efficiency of sampler moves*

The exchange move was accepted 44% of the time at the first stage and 75% at the second stage. This indicates that delayed rejection helps us to ensure that the algorithm is constantly swapping information between the chains; for 86% of the sweeps there is at least one exchange.

The snooker and variable-dimension crossover moves have acceptance rate less than 1%. That this occurs is to be expected. Liang & Wong (2001) report fairly small acceptance rates for their crossover moves; our rates are smaller as our algorithm operates upon a more complex space. Our experience with the snooker and variable-dimension crossover moves in more simple examples is that they are generally not worth the extra coding effort given their performance. However, we were satisfied with the fixed-dimensional crossover move which was accepted 2.9% of the time. The snooker birth is accepted more often than the standard birth, but the reverse snooker death move is rarely accepted, as with a birth move with a proposal that has low variance. Hence the move is less successful overall than the standard birth. The variable-dimension move acceptance rates, averaged over all chains, were still below 1%, with the split/combine move being less effective.

Overall, our recommendation is that the exchange, with delayed rejection, and fixed-dimensional crossover be implemented, in conjunction with constrained targets.

7.4. *Comparison with simulated tempering*

A more appropriate single-chain competitor is a simulated tempering algorithm; see Hodgson (1999) for an example of another variable-dimension simulated tempering algorithm. Here the target distribution is

$$\pi(\theta, \zeta, k | y_{1:n}) \propto p(y_{1:n} | \theta, k)^\zeta p(\theta, k) p(\zeta); \quad (4)$$

that is, $\zeta \in Z$ is stochastic with Z finite and pseudo prior $p(\zeta)$. In this example, we set the prior as opposed to constructing one adaptively; see Geyer & Thompson (1995). Note that our formulation does not require any normalization constants to be known, although such an algorithm will allow us to find a 'good' pseudo prior; see Geyer & Thompson (1995).

To sample from (4) we use the reversible jump algorithm in Appendix 1, conditional upon ζ , and to update ζ a delayed rejection move was adopted: propose a temperature uniformly at random from Z and accept or reject with the Hastings ratio. If this is rejected, select an adjacent temperature and perform a pseudo move that selects a move from the proposed temperature at the second stage to the proposed temperature at the first stage.

Using some trial simulations we were unable to find a pseudo prior such that, for a reasonable number of temperatures, e.g. 25, the algorithm could jump between the target and the inverse temperature 0.75. As a result, for any sensible number of distributions, the performance of this approach improves only slightly over the standard algorithm. An example of a run of the simulated tempering algorithm was setting $p(\zeta = \zeta_i) \propto 1/i$, with $\zeta_1 = 1$ and having a difference of 1×10^{-4} between each temperature. We found, with $|Z| = 25$, that the algorithm only visited the distribution of interest 10% of the time in a run of 250 000 sweeps.

In this section, it was seen that simulated tempering has been difficult to set up in such a way that it can be operated efficiently; see the technical report of Atachd  & Liu for a more automatic procedure. In addition, see Zheng (2003) for a theoretical comparison of population Markov chain Monte Carlo and simulated tempering.

8. DISCUSSION

One of the problems of our approach is the limited amount of success of our crossover moves. Whilst this was observed for simpler problems in Liang & Wong (2001), we would still hope that the population can provide more information when proposing moves. The search for general guidelines for constructing constrained subspaces of E and finding efficient ways to make them interact with the population is an important area for further investigation. There are various methodological extensions that may be considered.

The first is to combine our approach with adaptive Markov chain Monte Carlo methods; see Andrieu & Moulines (2006) and Chauveau & Vandekerckhove (2002) in the population context. This is likely to be superior to the standard adaptive algorithms, since there is more information with which to update proposals.

Secondly, we may consider combining some of our ideas with many recent stochastic simulation algorithms. As an example, we might use the constrained chains in a similar context to the equi-energy sampler of Kou et al. (2006). In the transdimensional case, the energy rings could be replaced with dimension rings. This is likely to produce a highly diverse sample with respect to the dimensionality.

ACKNOWLEDGEMENT

The first author was supported by a studentship from the U.K. Engineering and Physical Sciences Research Council. We thank two referees, an associate editor and the editor for comments which led to a substantial improvement in the content of the paper. The first author acknowledges many useful conversations with Nick Heard, Matthew Gander and Zhicheng Zhang. We also thank Adam Johansen and Tso-Jung Yen for some comments on earlier versions of the paper. The last author would like to thank the Medical Research Council, U.K., for their support.

APPENDIX 1

The reversible jump sampler of §3.3

One of the drawbacks of the model we have selected is the need for Λ to be positive definite. To deal with this problem, we consider the Cholesky decomposition Φ . That is, we write $\Lambda = \Phi\Phi'$, where Φ is lower-triangular with positive diagonal elements; recall that the Jacobian is $2^r \prod_{l=1}^r \phi_{ll}^{r-l+1}$. Our reversible jump Markov chain Monte Carlo algorithm is as follows, all moves being Metropolis–Hastings steps unless otherwise stated.

First, the fixed-dimensional moves are described. The component-specific means μ_j and component-specific lower-triangular part of Φ_j are both updated via an additive Cauchy random walk, independent in each dimension. The component-specific diagonals of Φ_j are updated via a multiplicative lognormal random walk, independent in each dimension. The weights are proposed using an additive normal random walk on the logit scale. Finally, Ψ is generated using a Gibbs kernel; the full conditional is $W_r\{2(g + k\alpha'), (2h + 2 \sum_{j=1}^k \Lambda_j^{-1})^{-1}\}$.

Secondly, the birth/death of a component follows Richardson & Green (1997). We draw a new μ and Φ from the prior and $w \sim \text{Be}(1, k)$, setting the new weights as $(w_1(1 - w), \dots, w_k(1 - w), w)$ and selecting the move with probability b_k , when in state k . The death, selected with probability d_k , is performed by selecting a component to die at random and inverting the jump function.

Finally, we describe the split/combine of a component. We select a split with probability s_k and choose a component j^* uniformly at random to split into components labelled as (j_1, j_2) . The split requires the following actions:

- (i) split the weight by drawing $u_1 \sim \text{Be}(\gamma, \gamma)$ and set $w_{j_1} = u_1 w_{j^*}$, $w_{j_2} = (1 - u_1) w_{j^*}$;
- (ii) split the mean vector by drawing $u_{1(2)}, \dots, u_{r(2)} \sim \mathcal{N}(0, \sigma_\mu^2)$ and take $\mu_{l(j_1)} = \mu_{l(j^*)} + u_{l(2)}$, $\mu_{l(j_2)} = \mu_{l(j^*)} - u_{l(2)}$;
- (iii) split the off-diagonals of Φ by drawing $u_{21(3)}, \dots, u_{r(r-1)(3)} \sim \mathcal{N}(0, \sigma_\phi^2)$ and take $\phi_{lm(j_1)} = \phi_{lm(j^*)} + u_{lm(3)}$, $\phi_{lm(j_2)} = \phi_{lm(j^*)} - u_{lm(3)}$, where $l = 2, \dots, r$ and $m = 1, \dots, l - 1$;
- (iv) split the diagonals of Φ by drawing $u_{11(3)}, \dots, u_{rr(3)}$ from the lognormal distribution with parameters 0 and σ_ϕ^2 and take $\phi_{ll(j_1)} = \frac{\phi_{ll(j^*)}}{u_{ll(3)}}$, $\phi_{ll(j_2)} = \phi_{ll(j^*)} u_{ll(3)}$.

In order to combine, we select the move with probability c_k and invert the jump function above. As a result of the symmetry constraint imposed on the jump function, it does not matter which way we combine the components; see Cappé et al. (2003) for details. We choose two components to combine, when in state k , with probability inversely proportional to the Mahalanobis distance between them; that is,

$$p_k(j_1, j_2) \propto \{(\mu_{j_1} - \mu_{j_2})' \Lambda_{j_1}^{-1} (\mu_{j_1} - \mu_{j_2}) + (\mu_{j_2} - \mu_{j_1})' \Lambda_{j_2}^{-1} (\mu_{j_2} - \mu_{j_1})\}^{-1}.$$

The split in state k is accepted with probability $\min\{1, A\}$, where

$$A = (\text{likelihood ratio}) \frac{p(\Phi_{j_1}) p(\Phi_{j_2})}{p(\Phi_{j^*})} \frac{p(\mu_{j_1}) p(\mu_{j_2})}{p(\mu_{j^*})} B(k\delta, \delta)^{-1} \{w_{j^*} u_1 (1 - u_1)\}^{\delta-1} \\ \times \frac{p(k+1)}{p(k)} \frac{(k+1)!}{k!} \frac{kc_{k+1} p_{k+1}(j_1, j_2)}{s_k} \frac{|J|}{2q_1(u_1)q_2(u_2)q_3(u_3)},$$

where $|J|$ is the Jacobian,

$$|J| = 2^{\frac{r(r+3)}{2}} w_{j^*} \prod_{l=1}^r \frac{\phi_{ll(j^*)}}{u_{ll(3)}},$$

with obvious notation for the prior and proposal densities.

The algorithm is performed in a deterministic sweep over all fixed-dimension moves followed by a random choice of birth/death or split/merge. The particular transdimensional move is selected with uniform probability, assuming that we allow a move, i.e. no birth or split when $k = k_{\max}$ or no death or combine when $k = 1$.

APPENDIX 2

Proofs

Proof of Theorem 1. Our strategy is to show that $\hat{K}^{N-1}(x_{1:N}, A) = (K_M \circ K_E \circ K_M)^{N-1}(x_{1:N}, A)$ ($A \in \mathcal{E}^N$) is a uniformly ergodic Markov kernel. Then, to prove that $\tilde{K}^{3(N-1)}$ is uniformly ergodic, we may use the fact that $\tilde{K}^{3(N-1)}(x_{1:N}, A) \geq \tau^{2(N-1)}(1 - \tau)^{N-1} \hat{K}^{N-1}(x_{1:N}, A)$; this is proved below. We begin by proving the case $N = 2$ and then use an induction on N . The strategy of the proof is to use the uniform ergodicity of K_{j^*} , which we take to be K_N , and the acceptance of an exchange move. We assume that $n_0 = 1$ as the proof can be extended to the case $n_0 > 1$ with only notational complications. We denote by $x_i^{(l)}$ the value of x_i after l steps.

We first establish that $\tilde{K}^{3(N-1)}(x_{1:N}, A) \geq \tau^{2(N-1)}(1 - \tau)^{N-1} \hat{K}^{N-1}(x_{1:N}, A)$, $A \in \mathcal{E}^N$. Consider $\tilde{K}^3$. Then

$$\tilde{K}^3(x_{1:N}, A) = \int_{E^N} \tilde{K}(x_{1:N}, dx_{1:N}^{(1)}) \int_{E^N} \tilde{K}(x_{1:N}^{(1)}, dx_{1:N}^{(2)}) \tilde{K}(x_{1:N}^{(2)}, A) \\ \geq \int_{E^N} \tilde{K}(x_{1:N}, dx_{1:N}^{(1)}) \int_{E^N} \tilde{K}(x_{1:N}^{(1)}, dx_{1:N}^{(2)}) \tau K_M(x_{1:N}^{(2)}, A),$$

where we have applied Chapman–Kolmogorov and used the fact that $\tilde{K}(x_{1:2}, \cdot) \geq \tau K_M(x_{1:N}, \cdot)$ for all $x_{1:N} \in E^N$. We then have

$$\tilde{K}^3(x_{1:N}, A) \geq \tau^2(1 - \tau) \int_{E^N} K_M(x_{1:N}, dx_{1:N}^{(1)}) \int_{E^N} K_E(x_{1:N}^{(1)}, dx_{1:N}^{(2)}) K_M(x_{1:N}^{(2)}, A), \quad (\text{A1})$$

which corresponds to selecting a mutation followed by an exchange and another mutation. This clearly follows for $\tilde{K}^{3(N-1)}(x_{1:N}, A)$, by the argument above:

$$\begin{aligned} \tilde{K}^{3(N-1)}(x_{1:N}, A) &\geq \tau^2(1 - \tau) \int_{E^N} \tilde{K}^{3(N-2)}(x_{1:N}, dx_{1:N}^{3(N-2)}) \hat{K}(x_{1:N}^{3(N-2)}, A) \\ &\geq \tau^4(1 - \tau)^2 \int_{E^N} \tilde{K}^{3(N-3)}(x_{1:N}, dx_{1:N}^{3(N-3)}) \hat{K}^2(x_{1:N}^{3(N-3)}, A). \end{aligned}$$

We may thus apply the argument above, recursively, to demonstrate the result. We now drop $\tau^2(1 - \tau)$ and prove uniform ergodicity of $\hat{K}$.

Let $N = 2$ and $A = A_1 \times A_2 \in \mathcal{E} \otimes \mathcal{E}$. If we use the fact that $K_M(x_{1:2}, \cdot) = K_1(x_1, \cdot)K_2(x_2, \cdot)$ and apply the minorization condition for $K_2(x_2^{(2)}, A_2)$, the modified equation (A1) becomes

$$\begin{aligned} \hat{K}(x_{1:2}, A) &\geq \int_{E^2} K_1(x_1, dx_1^{(1)}) K_2(x_2, dx_2^{(1)}) \\ &\quad \times \int_{E^2} \delta_{x_1^{(1)}}(dx_2^{(2)}) \delta_{x_2^{(1)}}(dx_1^{(2)}) \min \left\{ 1, \frac{\pi_1(x_2^{(1)})\pi_2(x_1^{(1)})}{\pi_1(x_1^{(1)})\pi_2(x_2^{(1)})} \right\} \\ &\quad \times K_1(x_1^{(2)}, A_1) \nu(A_2), \end{aligned} \quad (\text{A2})$$

where we have ignored the rejection of an exchange move. Since, for all $(x_1, x_2) \in E^2$,

$$\min \left\{ 1, \frac{\pi_1(x_2)\pi_2(x_1)}{\pi_1(x_1)\pi_2(x_2)} \right\} \geq \min \left\{ 1, \frac{\pi_1(x_2)}{\pi_2(x_2)q_1} \right\}, \quad (\text{A3})$$

and using the measurability of the function, we can split the integrals in equation (A2) into $I_1 \times I_2$, where

$$\begin{aligned} I_1 &= \int_E K_1(x_1, dx_1^{(1)}) \int_E \delta_{x_1^{(1)}}(dx_2^{(2)}), \\ I_2 &= \int_{E^2} K_2(x_2, dx_2^{(1)}) \delta_{x_2^{(1)}}(dx_1^{(2)}) \min \left\{ 1, \frac{\pi_1(x_2^{(1)})}{\pi_2(x_2^{(1)})q_1} \right\} K_1(x_1^{(2)}, A_1). \end{aligned}$$

Clearly $I_1 = 1$. For I_2 , integrating with respect to Dirac measure and then applying the minorization condition, we obtain

$$I_2 \geq \epsilon \int_E \nu(dx_2^{(1)}) \min \left\{ 1, \frac{\pi_1(x_2^{(1)})}{\pi_2(x_2^{(1)})q_1} \right\} K_1(x_2^{(1)}, A_1).$$

Therefore, equation (A2) becomes

$$\hat{K}(x_{1:2}, A) \geq \theta \nu^*(A), \quad (\text{A4})$$

where

$$\begin{aligned}\theta &= \epsilon^2 \phi, \\ \nu^*(A) &= K^*(A_1) \nu(A_2), \\ K^*(A_1) &= \frac{1}{\phi} \int_E \nu(dx_2^{(1)}) \min \left\{ 1, \frac{\pi_1(x_2^{(1)})}{\pi_2(x_2^{(1)})_{Q_1}} \right\} K_1(x_2^{(1)}, A_1), \\ \phi &= \int_E \nu(dx_2) \min \left\{ 1, \frac{\pi_1(x_2)}{\pi_2(x_2)_{Q_1}} \right\}.\end{aligned}$$

Since (A4) holds for all $x_{1:2} \in E^2$ and $A \in \mathcal{E} \otimes \mathcal{E}$ and the product measure is nontrivial, we have that E^2 is $(1, \theta, \nu^*)$ small, and similarly $\tilde{K}^3$ is uniformly ergodic.

Now suppose that $\hat{K}(x_{2:N}, dx_{2:N})$ is uniformly ergodic, for arbitrary mixture parameters, under the specified conditions with $E^{N-1} \{(N-2), \epsilon, \nu\}$ small. Let $\theta_1 = \sum_{l=2}^N \epsilon_{1l}$. For $A = A_1 \times \cdots \times A_N = A_{1:N} \in \mathcal{E}^N$, we have that

$$\begin{aligned}\hat{K}^{(N-1)}(x_{1:N}, A) &\geq (1 - \theta_1)^{N-2} \int_{E^N} \hat{K}(x_{1:N}, dx_{1:N}^{(1)}) \hat{K}_{\theta_1}^{(N-2)}(x_{2:N}^{(1)}, A_{2:N}) \\ &\quad \times K_1^{2N-3}(x_1^{(1)}, A_1),\end{aligned}\tag{A5}$$

where $\hat{K}_{\theta_1}$ denotes that the mixture parameters in the exchange kernel have been modified by θ_1 . Equation (A5) refers to the fact that the probability of moving to A under $\hat{K}$ for any $x \in E^N$ is greater than considering the kernel that updates x_1 independently of $x_{2:N}$; that is, we can apply much the same arguments as for demonstrating $\tilde{K}^{3(N-1)}(x_{1:N}, A) \geq \tau^{2(N-1)}(1 - \tau)^{N-1} \hat{K}^{N-1}(x_{1:N}, A)$ and therefore

$$\hat{K}^{(N-1)}(x_{1:N}, A) \geq \epsilon(1 - \theta_1)^{N-2} \int_{E^N} \hat{K}(x_{1:N}, dx_{1:N}^{(1)}) \nu(A_{2:N}) K_1^{2N-3}(x_1^{(1)}, A_1).$$

At this point, we may apply the above arguments to yield that E^N is $(N-1, \theta, \nu^*)$ small for appropriate $\theta > 0$ and probability measure ν^* . Thus, the result follows by induction and the fact that $\tilde{K}^{3(N-1)}(x_{1:N}, A) \geq \tau^{2(N-1)}(1 - \tau)^{N-1} \hat{K}^{N-1}(x_{1:N}, A)$. $\square$

Proof of Proposition 1. Our approach is to focus upon the Dobrushin coefficient of $K_M \circ K_E$. Consider $N = 2$ and $f : E^2 \rightarrow [0, 1]$. Then

$$\begin{aligned}&|K_M \circ K_E(f)(u_{1:2}) - K_M \circ K_E(f)(v_{1:2})| \\ &\leq \left| \int_{E^4} \rho_{1,2}(x_1, x_2) f(x'_{1:2}) \delta_{x_1}(dx'_1) \delta_{x_2}(dx'_2) K_M(u_{1:2}, dx_{1:2}) \right. \\ &\quad \left. - \int_{E^4} \rho_{1,2}(x_1, x_2) f(y'_{1:2}) \delta_{x_1}(dy'_1) \delta_{x_2}(dy'_2) K_M(v_{1:2}, dx_{1:2}) \right| \\ &\quad + \left| \int_{E^4} [1 - \rho_{1,2}(x_1, x_2)] f(x'_{1:2}) \delta_{x_1}(dx'_1) \delta_{x_2}(dx'_2) K_M(u_{1:2}, dx_{1:2}) \right. \\ &\quad \left. - \int_{E^4} [1 - \rho_{1,2}(x_1, x_2)] f(y'_{1:2}) \delta_{x_1}(dy'_1) \delta_{x_2}(dy'_2) K_M(v_{1:2}, dx_{1:2}) \right| \\ &\leq [\text{osc}(f\rho_{1,2}) + \text{osc}\{f(1 - \rho_{1,2})\}] \beta(K_M) \\ &\leq 2\text{osc}(\rho_{1,2}) \beta(K_M),\end{aligned}$$

where we have used the fact that $\text{osc}(fg) \leq \sup(f)\text{osc}(g)$; this can be used since $\inf(f) = 0$ and $\sup(f) = 1$. Note that

$$\text{osc}(\rho_{1,2}) = 1 - \inf_{(x_1, x_2) \in E^2} \rho_{1,2}(x_1, x_2).$$

Since we have the mixing condition on K_M we have that

$$\beta(K_M \circ K_E) \leq 2\{1 - \inf_{(x_1, x_2) \in E^2} \rho_{1,2}(x_1, x_2)\}(1 - \epsilon).$$

Since

$$\|\eta(K_M \circ K_E)^n - \pi^*\|_{\text{tv}} \leq \beta\{(K_M \circ K_E)^n\} \|\eta - \pi^*\|_{\text{tv}}$$

(Del Moral, 2004, Ch. 4), we easily obtain the desired result, for $N = 2$. The result for $N \geq 3$ can be proved using the above arguments, with only notational complications. $\square$

REFERENCES

- ANDRIEU, C. & MOULINES, E. (2006). On the ergodicity properties of some adaptive Markov chain Monte Carlo algorithms. *Ann. Appl. Prob.* **16**, 1462–505.
- BOZDECH, Z., LLINAS, M., PULLIUM, B. L., WONG, E. D., ZHU, J. & DERISI, J. L. (2003). The transcriptome of the intraerythrocytic developmental cycle of *Plasmodium falciparum*. *PLoS Biol.* **1**, 1–16.
- BROOKS, S. P., GIUDICI, P. & ROBERTS, G. O. (2003). Efficient construction of reversible jump Markov chain Monte Carlo proposal distributions (with Discussion). *J. R. Statist. Soc. B* **65**, 1–37.
- CAPPÉ, O., ROBERT, C. P. & RYDÉN, T. (2003). Reversible jump, birth-and-death and more general continuous time Markov chain Monte Carlo samplers. *J. R. Statist. Soc. B* **65**, 679–700.
- CARLIN, B. & CHIB, S. (1995). Bayesian model choice via Markov chain Monte Carlo methods. *J. R. Statist. Soc. B* **57**, 473–84.
- CHAUVEAU, D. & VANDEKERKHOVE, P. (2002). Improving convergence of the Hastings and Metropolis algorithm with an adaptive proposal. *Scand. J. Statist.* **29**, 13–29.
- DEL MORAL, P. (2004). *Feynman–Kac Formulae: Genealogical and Interacting Particle Systems with Applications*. New York: Springer.
- DEL MORAL, P., DOUCET, A. & JASRA, A. (2006). Sequential Monte Carlo samplers. *J. R. Statist. Soc. B* **68**, 411–36.
- DELLAPORTAS, P. & PAPAGEORGIOU, I. (2006). Multivariate mixtures of normals with an unknown number of components. *Statist. Comp.* **16**, 57–68.
- DENISON, D. G. T., HOLMES, C. C., MALLICK, B. K. & SMITH, A. F. M. (2002). *Bayesian Methods for Nonlinear Classification and Regression*. Chichester: Wiley.
- DIACONIS, P. & SALOFF-COSTE, L. (1993). Comparison theorems for reversible Markov chains. *Ann. Appl. Prob.* **3**, 696–750.
- DOBRUSHIN, R. L. (1956). Central limit theorem for non-stationary Markov chains I. *Theory Prob. Appl.* **1**, 65–80.
- GEYER, C. (1991). Markov chain maximum likelihood. In *Computing Science and Statistics: The 23rd Symposium on the Interface*, Ed. E. Keramigas, pp. 156–63. Fairfax: Interface Foundation.
- GEYER, C. & THOMPSON, E. A. (1995). Annealing Markov chain Monte Carlo with applications to ancestral inference. *J. Am. Statist. Assoc.* **90**, 909–20.
- GILKS, W. R., ROBERTS, G. O. & GEORGE, E. I. (1994). Adaptive direction sampling. *Statistician* **43**, 179–89.
- GODSILL, S. (2001). On the relationship between Markov chain Monte Carlo methods for uncertainty. *J. Comp. Graph. Statist.* **10**, 230–48.
- GOSWAMI, G. & LIU, J. S. (2007). On learning strategies for evolutionary Monte Carlo. *Statist. Comp.* **17**, 23–38.
- GREEN, P. J. (1995). Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika* **82**, 711–32.
- GREEN, P. J. (2003a). Discussion of a paper by Brooks et al. *J. R. Statist. Soc. B* **65**, 48–9.
- GREEN, P. J. (2003b). Trans-dimensional Markov chain Monte Carlo. In *Highly Structured Stochastic Systems*, Ed. P. J. Green, N. L. Hjort and S. Richardson, pp. 179–96. Oxford: Oxford University Press.
- GREEN, P. J. & MIRA, A. (2001). Delayed rejection in reversible jump Metropolis–Hastings. *Biometrika* **88**, 1035–53.
- HEARD, N. A., HOLMES, C. C. & STEPHENS, D. A. (2006). A quantitative study of gene regulation involved in the immune response of anopheline mosquitoes: an application of Bayesian hierarchical clustering of curves. *J. Am. Statist. Assoc.* **101**, 18–29.
- HODGSON, M. E. A. (1999). A Bayesian restoration of an ion channel signal. *J. R. Statist. Soc. B* **61**, 95–114.
- HUKUSHIMA, K. & NEMOTO, K. (1996). Exchange Monte Carlo method and application to spin glass simulations. *J. Phys. Soc. Jap.* **65**, 1604–8.
- IBA, Y. (2001). Extended ensemble Monte Carlo. *Int. J. Mod. Phys.* **12**, 653–6.
- JASRA, A., HOLMES, C. C. & STEPHENS, D. A. (2005). Markov chain Monte Carlo methods and the label switching problem in Bayesian mixture modelling. *Statist. Sci.* **20**, 50–67.
- JASRA, A., STEPHENS, D. A. & HOLMES, C. C. (2007). On population-based simulation, *Statist. Comp.* **17**, 263–79.

- JENNISON, C., HURN, M. A. & AL-AWADHI, F. (2003). Discussion of a paper by Brooks et al. *J. R. Statist. Soc. B* **65**, 44–5.
- KOU, S. C., ZHOU, Q. & WONG, W. H. (2006). Equi-energy sampler with applications in statistical inference and statistical mechanics. *Ann. Statist.* **34**, 1581–619.
- LIANG, F. & WONG, W. H. (2001). Real parameter evolutionary Monte Carlo with applications to Bayesian mixture models. *J. Am. Statist. Assoc.* **96**, 653–66.
- LIU, J. S. (2001). *Monte Carlo Strategies in Scientific Computing*. New York: Springer.
- MADRAS, N. & ZHENG, Z. (2003). On the swapping algorithm. *Random Struct. Algor.* **22**, 66–97.
- MCLACHLAN, G. J. & PEEL, D. (2000). *Finite Mixture Models*. Chichester: Wiley.
- MENGERSEN, K. L. & TWEEDIE, R. L. (1996). Rates of convergence of the Hastings and Metropolis algorithms. *Ann. Statist.* **24**, 101–21.
- RICHARDSON, S. & GREEN, P. J. (1997). On Bayesian analysis of mixture models with an unknown number of components (with Discussion). *J. R. Statist. Soc. B* **59**, 731–92.
- ROBERT, C. P. & CASELLA G. (2004). *Monte Carlo Statistical Methods*, 2nd ed. New York: Springer.
- ROBERTS, G. O. & ROSENTHAL, J. S. (2004). General state space Markov chains and Markov chain Monte Carlo algorithms. *Prob. Surveys* **1**, 20–71.
- STEPHENS, M. (2000). Bayesian analysis of mixture models with an unknown number of components—an alternative to reversible jump methods. *Ann. Statist.* **28**, 40–74.
- TIERNEY, L. (1994). Markov chains for exploring posterior distributions (with Discussion). *Ann. Statist.* **22**, 1701–62.
- ZHENG, Z. (2003). On swapping and simulated tempering algorithms. *Stoch. Proces. Appl.* **104**, 131–53.

[Received February 2005. Revised April 2007]