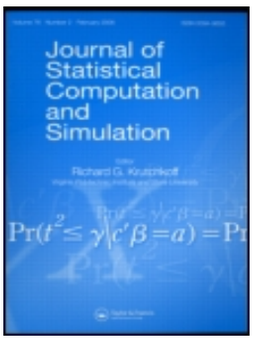

Journal of Statistical Computation and Simulation

ISSN: 0094-9655 (Print) 1563-5163 (Online) Journal homepage: http://www.tandfonline.com/loi/gscs20

A double Metropolis–Hastings sampler for spatial models with intractable normalizing constants

Faming Liang

To cite this article: Faming Liang (2010) A double Metropolis–Hastings sampler for spatial models with intractable normalizing constants, Journal of Statistical Computation and Simulation, 80:9, 1007-1022, DOI: 10.1080/00949650902882162

To link to this article: http://dx.doi.org/10.1080/00949650902882162

Published online: 23 Oct 2009.

Journal of Statistical Computation and Simulation
Vol. 80, No. 9, September 2010, 1007–1022

Taylor & Francis
Taylor & Francis Group

A double Metropolis–Hastings sampler for spatial models with intractable normalizing constants

Faming Liang*

Department of Statistics, Texas A&M University, College Station, TX 77843-3143, USA

(*Received 16 July 2008; final version received 10 March 2009*)

The problem of simulating from distributions with intractable normalizing constants has received much attention in recent literature. In this article, we propose an asymptotic algorithm, the so-called double Metropolis–Hastings (MH) sampler, for tackling this problem. Unlike other auxiliary variable algorithms, the double MH sampler removes the need for exact sampling, the auxiliary variables being generated using MH kernels, and thus can be applied to a wide range of problems for which exact sampling is not available. For the problems for which exact sampling is available, it can typically produce the same accurate results as the exchange algorithm, but using much less CPU time. The new method is illustrated by various spatial models.

Keywords: autologistic model; autonormal model; auxiliary variable MCMC algorithm; exchange algorithm; Metropolis–Hastings algorithm

1. Introduction

Spatial models, e.g. the autologistic model, Potts model, and autonormal model [1], have been used in the modelling of many scientific problems. Examples include image analysis [2], disease mapping [3], genetic analysis [4], among others. A major problem with the models is that the normalizing constant is intractable. The problem can be described as follows. Suppose we have a data set $\boldsymbol{X}$ generated from a statistical model with the likelihood function

$$f(\boldsymbol{x}|\theta) = \frac{1}{Z(\theta)} \exp\{-U(\boldsymbol{x}, \theta)\}, \quad \boldsymbol{x} \in \mathcal{X}, \quad \theta \in \Theta, \tag{1}$$

where θ is a parameter, and $Z(\theta)$, the normalizing constant that depends on θ and is not available in closed form. Let $\pi(\theta)$ denote the prior density of θ. The posterior distribution of θ given, $\boldsymbol{x}$ is then given by

$$\pi(\theta|\boldsymbol{x}) \propto \frac{1}{Z(\theta)} \exp\{-U(\boldsymbol{x}, \theta)\}\pi(\theta). \tag{2}$$

*Email: fliang@stat.tamu.edu

ISSN 0094-9655 print/ISSN 1563-5163 online

DOI: 10.1080/00949650902882162
http://www.informaworld.com

Since the closed form of $Z(\theta)$ is not available, inference from the model poses a great challenge on the current statistical methods.

The Metropolis–Hastings (MH) algorithm cannot be directly applied to simulate from $\pi(\theta|\boldsymbol{x})$, because the acceptance probability would involve the unknown ratio $Z(\theta)/Z(\theta')$, where θ' denotes the proposed value. To circumvent this difficulty, various approximation methods to the likelihood function or the normalizing constant function have been proposed for the models. The following are some examples. Besag [1] proposed the approximation of the likelihood function by a pseudo-likelihood function that is tractable. The method is easy to use, but it typically performs less well for the models for which neighbouring dependence is strong. This method was further discussed and generalized by Dryden *et al.* [5] and Huang and Ogata [6]. Geyer and Thompson [7] proposed an importance sampling-based approach to approximate $Z(\theta)$. This approach was refined by Liang *et al.* [8] by refining the choice of the trial density function using the stochastic approximation Monte Carlo algorithm. Liang [9] proposed an alternative Monte Carlo approach to approximate $Z(\theta)$, where $Z(\theta)$ is viewed as a marginal distribution of the unnormalized distribution $g(\boldsymbol{x}, \theta) = \exp\{-U(\boldsymbol{x}, \theta)\}$, and is estimated by an adaptive kernel density estimator using Monte Carlo draws.

Recently, Møller *et al.* [10] and Murray *et al.* [11] proposed auxiliary variable MCMC algorithms for simulating from the distribution (2). These algorithms require exact sampling [12] of $\boldsymbol{X}$. Unfortunately, exact sampling is very expensive or impossible for many statistical models whose normalizing constant is intractable.

In this article, we propose a new asymptotic algorithm, the so-called double MH sampler, for simulating from the distributions with intractable normalizing constants. The double MH sampler removes the need for exact sampling, the auxiliary variables being generated using MH kernels, and thus can be applied to many statistical models for which exact sampling is not available. While for the models for which exact sampling is available, e.g. the autologistic model, it can produce almost the same accurate results as the exchange algorithm, but using much less CPU time.

The remainder of this article is organized as follows: in Section 2, we give a brief review of the auxiliary variable MCMC algorithms. In Section 3, we describe the double MH sampler. In Section 4, we illustrate the double MH sampler with three spatial models, the autologistic model, autonormal model, and very-soft-core model. In Section 5, we conclude the article with a brief discussion.

2. Auxiliary variable MCMC algorithms

In this section, we give a brief review of the auxiliary variable MCMC algorithms proposed by Møller *et al.* [10] and Murray *et al.* [11].

2.1. *Møller et al.'s algorithm*

The key idea of Møller *et al.* [10] is to extend the distribution $\pi(\theta|\boldsymbol{x})$ to include an auxiliary variable, $\boldsymbol{y}$, which shares the same state space as $\boldsymbol{x}$:

$$f(\theta, \boldsymbol{y}|\boldsymbol{x}) = f(\boldsymbol{x}|\theta)\pi(\theta)f(\boldsymbol{y}|\theta, \boldsymbol{x}). \tag{3}$$

To simulate from Equation (3) using the MH algorithm, the authors suggested the following proposal distribution:

$$q(\theta', \boldsymbol{y}'|\theta, \boldsymbol{y}) = q(\theta'|\theta, \boldsymbol{y})q(\boldsymbol{y}'|\theta'), \tag{4}$$

which corresponds to the usual change in the parameter vector $\theta \to \theta'$, followed by exact sampling of $\boldsymbol{y}'$ from $q(\cdot|\theta')$. If $q(\boldsymbol{y}'|\theta')$ is set as $f(\boldsymbol{y}'|\theta)$, then the MH ratio can be written as

$$r(\theta, \boldsymbol{y}, \theta', \boldsymbol{y}'|\boldsymbol{x}) = \frac{f(\boldsymbol{x}|\theta')\pi(\theta')f(\boldsymbol{y}'|\theta', \boldsymbol{x})q(\theta|\theta', \boldsymbol{x})f(\boldsymbol{y}|\theta)}{f(\boldsymbol{x}|\theta)\pi(\theta)f(\boldsymbol{y}|\theta, \boldsymbol{x})q(\theta'|\theta, \boldsymbol{x})f(\boldsymbol{y}'|\theta')}, \tag{5}$$

where the unknown normalizing constant $Z(\theta)$ can be cancelled. To ease computation, the authors further suggested to set the auxiliary distributions

$$f(\boldsymbol{y}|\theta, \boldsymbol{x}) = f(\boldsymbol{y}|\hat{\theta}), \quad f(\boldsymbol{y}'|\theta', \boldsymbol{x}) = f(\boldsymbol{y}'|\hat{\theta}), \tag{6}$$

where $\hat{\theta}$ denotes an estimate of θ, for example, which can be obtained by maximizing a pseudo-likelihood function.

2.2. *The exchange algorithm*

Murray *et al.*'s algorithm is motivated by the parallel tempering algorithm [13], [14], and can be described as follows. Consider the augmented distribution

$$f(\boldsymbol{y}_1, \ldots, \boldsymbol{y}_m, \theta|\boldsymbol{x}) = \pi(\theta)f(\boldsymbol{x}|\theta)\prod_{j=1}^{m} f(\boldsymbol{y}_j|\theta_j), \tag{7}$$

where θ_is are instantiated and fixed, and $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_m$ are auxiliary variables with the same support as $\boldsymbol{x}$. Suppose that a change to θ is proposed with probability $q(\theta_i|\theta)$. To ensure that $\boldsymbol{y}_i = \boldsymbol{x}$, we swap the settings of $\boldsymbol{x}$ and $\boldsymbol{y}_i$. The resulting MH ratio for the change is

$$r(\theta, \theta_i, \boldsymbol{y}_i|\boldsymbol{x}) = \frac{\pi(\theta_i)f(\boldsymbol{x}|\theta_i)f(\boldsymbol{y}_i|\theta)\prod_{j\neq i} f(\boldsymbol{y}_j|\theta_j)q(\theta|\theta_i)}{\pi(\theta)f(\boldsymbol{x}|\theta)f(\boldsymbol{y}_i|\theta_i)\prod_{j\neq i} f(\boldsymbol{y}_j|\theta_j)q(\theta_i|\theta)} = \frac{\pi(\theta_i)f(\boldsymbol{x}|\theta_i)f(\boldsymbol{y}_i|\theta)q(\theta|\theta_i)}{\pi(\theta)f(\boldsymbol{x}|\theta)f(\boldsymbol{y}_i|\theta_i)q(\theta_i|\theta)}. \tag{8}$$

Based on the above arguments, the authors proposed the following algorithm:

The exchange algorithm

(a) Propose $\theta' \sim q(\theta'|\theta, \boldsymbol{x})$.
(b) Generate an auxiliary variable $\boldsymbol{y} \sim f(\boldsymbol{y}|\theta')$ using an exact sampler.
(c) Accept θ' with probability $\min\{1, r(\theta, \theta', \boldsymbol{y}|\boldsymbol{x})\}$, where

$$r(\theta, \theta', \boldsymbol{y}|\boldsymbol{x}) = \frac{\pi(\theta')f(\boldsymbol{x}|\theta')f(\boldsymbol{y}|\theta)q(\theta|\theta', \boldsymbol{x})}{\pi(\theta)f(\boldsymbol{x}|\theta)f(\boldsymbol{y}|\theta')q(\theta'|\theta, \boldsymbol{x})}. \tag{9}$$

Since a swapping change between $(\theta, \boldsymbol{x})$ and $(\theta', \boldsymbol{y})$ is involved, the algorithm is called the exchange algorithm by the authors. This algorithm represents an improvement over Møller *et al.*'s algorithm, as it removes the need to estimate the parameter before sampling begins. Murray *et al.* [11] reported that the exchange algorithm tends to have a higher acceptance probability for the exact samples than Møller *et al.*'s algorithm.

3. A double Metropolis–Hastings sampler

Suppose that we are interested in simulating a sample $\boldsymbol{y}$ from $f(\boldsymbol{y}|\theta')$. If the sample is generated through m MH updates starting with the current state $\boldsymbol{x}$, the transition probability, $P_{\theta'}^{(m)}(\boldsymbol{y}|\boldsymbol{x})$, from

$\boldsymbol{x}$ to $\boldsymbol{y}$ is then

$$P_{\theta'}^{(m)}(\boldsymbol{y}|\boldsymbol{x}) = K_{\theta'}(\boldsymbol{x} \longrightarrow \boldsymbol{x}_1) \cdots K_{\theta'}(\boldsymbol{x}_{m-1} \longrightarrow \boldsymbol{y}), \tag{10}$$

where $K(\cdot \to \cdot)$ is the MH transition kernel. Thus, we have

$$\begin{aligned} \frac{P_{\theta'}^{(m)}(\boldsymbol{x}|\boldsymbol{y})}{P_{\theta'}^{(m)}(\boldsymbol{y}|\boldsymbol{x})} &= \frac{K_{\theta'}(\boldsymbol{y} \to \boldsymbol{x}_{m-1}) \cdots K_{\theta'}(\boldsymbol{x}_1 \to \boldsymbol{x})}{K_{\theta'}(\boldsymbol{x} \to \boldsymbol{x}_1) \cdots K_{\theta'}(\boldsymbol{x}_{m-1} \to \boldsymbol{y})} \\ &= \frac{f(\boldsymbol{x}|\theta')}{f(\boldsymbol{y}|\theta')} \frac{f(\boldsymbol{y}|\theta')}{f(\boldsymbol{x}|\theta')} \frac{K_{\theta'}(\boldsymbol{y} \to \boldsymbol{x}_{m-1}) \cdots K_{\theta'}(\boldsymbol{x}_1 \to \boldsymbol{x})}{K_{\theta'}(\boldsymbol{x} \to \boldsymbol{x}_1) \cdots K_{\theta'}(\boldsymbol{x}_{m-1} \to \boldsymbol{y})} \\ &= \frac{f(\boldsymbol{x}|\theta')}{f(\boldsymbol{y}|\theta')}, \end{aligned} \tag{11}$$

where the last equality follows from the detailed balance equality $f(\boldsymbol{x}|\theta')K_{\theta'}(\boldsymbol{x} \to \boldsymbol{x}_1) \cdots K_{\theta'}(\boldsymbol{x}_{m-1} \to \boldsymbol{y}) = f(\boldsymbol{y}|\theta')K_{\theta'}(\boldsymbol{y} \to \boldsymbol{x}_{m-1}) \cdots K_{\theta'}(\boldsymbol{x}_1 \to \boldsymbol{x})$.

Now we return to the problem of simulating from the posterior distribution (2). By Equation (11), the MH ratio (9) can be re-expressed as

$$r(\theta, \theta', \boldsymbol{y}|\boldsymbol{x}) = \frac{\pi(\theta')q(\theta|\theta', \boldsymbol{x})}{\pi(\theta)q(\theta'|\theta, \boldsymbol{x})} \frac{f(\boldsymbol{y}|\theta)P_{\theta'}^{(m)}(\boldsymbol{x}|\boldsymbol{y})}{f(\boldsymbol{x}|\theta)P_{\theta'}^{(m)}(\boldsymbol{y}|\boldsymbol{x})}. \tag{12}$$

It is easy to see that if we choose $q(\theta'|\theta, \boldsymbol{x})$ as a MH transition kernel that satisfies the detailed balance condition, then we have $\pi(\theta')q(\theta|\theta', \boldsymbol{x}) = \pi(\theta)q(\theta'|\theta, \boldsymbol{x})$, and the exchange update is reduced to a simple MH update for which $f(\boldsymbol{x}|\theta)$ works as the target distribution and $P_{\theta'}^{(m)}(\boldsymbol{y}|\boldsymbol{x})$ works as the proposal distribution. In summary, we have the following sampling scheme as a replacement for the exchange algorithm. Let t denote the index of iterations, and let θ_t denote the current state of the Markov chain.

The double MH Sampler

(a) Simulate a new sample θ' from $\pi(\theta)$ using the MH algorithm starting with θ_t.
(b) Generate an auxiliary variable $\boldsymbol{y} \sim P_{\theta'}^{(m)}(\boldsymbol{y}|\boldsymbol{x})$, and accept it with probability $\min\{1, r(\theta_t, \theta', \boldsymbol{y}|\boldsymbol{x})\}$, where, by Equation (11),

$$r(\theta_t, \theta', \boldsymbol{y}|\boldsymbol{x}) = \frac{f(\boldsymbol{y}|\theta_t)P_{\theta'}^{(m)}(\boldsymbol{x}|\boldsymbol{y})}{f(\boldsymbol{x}|\theta_t)P_{\theta'}^{(m)}(\boldsymbol{y}|\boldsymbol{x})} = \frac{f(\boldsymbol{y}|\theta_t)f(\boldsymbol{x}|\theta')}{f(\boldsymbol{x}|\theta_t)f(\boldsymbol{y}|\theta')}. \tag{13}$$

(c) Set $\theta_{t+1} = \theta'$ if the auxiliary variable is accepted in step (b), and set $\theta_{t+1} = \theta_t$ otherwise.

Since two types of MH updates are performed in step (b), one for drawing the auxiliary variable $\boldsymbol{y}$ and one for acceptance of θ', we call the algorithm the double MH sampler. Note that the MH update performed in step (a) is not essential, which can be incorporated into step (b) by changing Equation (13) to Equation (12). We also note that Equation (13) holds regardless of the value of m. A remarkable feature of the algorithm is that it removes the need of exact sampling with a delicate use of the detailed balance condition. Therefore, the algorithm can be applied to a wide range of problems for which exact sampling is impossible or very expensive.

It is obvious that the samples will converge to the correct posterior distribution for a large value of m. In practice, to get good samples from the posterior distribution, m is not necessarily large. This can be justified as follows: suppose that the current state θ_t is a sample from $\pi(\theta|\boldsymbol{x})$. If θ' is a

good proposal, i.e. $\theta' \sim \pi(\theta|\boldsymbol{x})$, then we have $\boldsymbol{x} \sim f(\boldsymbol{x}|\theta')$. This further implies that $\boldsymbol{y} \sim f(\boldsymbol{y}|\theta')$ for any value of m, because $\boldsymbol{y}$ is generated through a sequence of MH updates, which starts with $\boldsymbol{x}$ and admits $f(\cdot|\theta')$ as the invariant distribution. Hence, in this case, the transition $\theta_t \to \theta_{t+1}$ leaves the posterior distribution $\pi(\theta|\boldsymbol{x})$ invariant regardless the value of m. If θ' is a bad proposal, i.e. θ' is unlikely a sample from $\pi(\theta|\boldsymbol{x})$, then the ratio $f(\boldsymbol{x}|\theta')/f(\boldsymbol{y}|\theta')$ should be small, as $\boldsymbol{y}$ has moved some steps towards the equilibrium of $f(\cdot|\theta')$ than $\boldsymbol{x}$; and the ratio $f(\boldsymbol{y}|\theta_t)/f(\boldsymbol{x}|\theta_t)$ should also be small, as $\boldsymbol{y}$ has moved away from the equilibrium of $f(\cdot|\theta_t)$ than $\boldsymbol{x}$. This results in a very small MH ratio $r(\theta_t, \theta', \boldsymbol{y}|\boldsymbol{x})$. As a very likely consequence, the proposal is rejected, and we set $\theta_{t+1} = \theta_t$. In all examples of this article, the auxiliary variable $\boldsymbol{y}$ is generated through a single cycle of Gibbs iterations, and this translates to a value of m being equal to the dimension of $\boldsymbol{y}$. Note that the value of m used here is rather small, only one Gibbs update per component of $\boldsymbol{y}$. The key to the efficiency of the MH kernel $P^{(m)}_{\theta'}(\boldsymbol{y}|\boldsymbol{x})$ is of starting with $\boldsymbol{x}$, which also directly leads to the validity of Equation (13) by the detailed balance condition.

Suppose that a sequence of samples $\theta_1, \ldots, \theta_n$ has been collected from a run of the double MH sampler. An approximate Bayesian estimator of θ can then be obtained by averaging over the samples,

$$\bar{\theta} = \frac{1}{n}\sum_{i=1}^{n} \theta_i.$$

This estimator can also be named as an ensemble averaging estimator, as $\theta_1 \ldots, \theta_n$ are only approximately distributed as $\pi(\theta|\boldsymbol{x})$. As discussed by Haykin [15, p. 355], the ensemble averaging estimator has the same bias as, but a much smaller variance than the single sample estimator. This estimator can potentially be robustified by downweighting the samples for which the corresponding MH ratio $r(\theta_t, \theta', \boldsymbol{y}|\boldsymbol{x})$ was small or the corresponding proposal was rejected, with the reasons as explained above. Please refer to Haykin [15] again for discussions on weight setting for ensemble averages. In this article, we simply assign an equal weight to each sample, and call the resulting estimator $\bar{\theta}$ the approximate Bayesian estimator.

For an effective implementation of the double MH sampler, we need to consider two more issues. The first issue is on the choice of proposal distributions, namely, the proposal distribution used in step (a) for generating a new sample of θ', and the proposal distribution used in step (b) for generating an auxiliary variable of $\boldsymbol{y}$. As for conventional MCMC algorithms, these proposals should be adjusted such that they have a reasonable acceptance rate, e.g. a rate between 0.2 and 0.4 as suggested by Gelman *et al.* [16]. Since the double MH sampler undergoes two acceptance/rejection steps, a sample θ' is counted as acceptance only when acceptance is made in both steps (a) and (b). In the article, we call the acceptance rate of θ' the acceptance rate of double MH moves. Note that when a Gibbs sampler is used for generating the auxiliary variables, the choice of the proposal is automatic.

The second issue is on diagnostic for the convergence of simulations. Since the double MH sampler belongs to the class of MCMC algorithms, its convergence can be monitored using the tools existing in the literature. In this article, we adopted the the multiple run-based diagnostic method developed by Gelman and Rubin [17].

4. Approximate Bayesian analysis for various spatial models

4.1. *Spatial autologistic models*

The autologistic model [1] has been widely used for spatial data analysis, [18–20]. Let $\boldsymbol{x} = \{x_i : i \in D\}$ denote the observed binary data, where x_i is called a spin and D is the set of indices of

the spins. Let $|D|$ denote the total number of spins in D, and let $n(i)$ denote the set of neighbours of spin i. The likelihood function of the model is

$$f(\boldsymbol{x}|\alpha, \beta) = \frac{1}{Z(\alpha, \beta)} \exp\left\{\alpha \sum_{i \in D} x_i + \frac{\beta}{2} \sum_{i \in D} x_i \left(\sum_{j \in n(i)} x_j\right)\right\}, \quad (\alpha, \beta) \in \Theta, \tag{14}$$

where the parameter α determines the overall proportion of $x_i = +1$, the parameter β determines the intensity of interaction between x_i and its neighbours, and $Z(\alpha, \beta)$ is the intractable normalizing constant defined by

$$Z(\alpha, \beta) = \sum_{\text{for all possible } \boldsymbol{x}} \exp\left\{\alpha \sum_{j \in D} x_j + \frac{\beta}{2} \sum_{i \in D} x_i \left(\sum_{j \in n(i)} x_j\right)\right\}.$$

An exact evaluation of $Z(\alpha, \beta)$ is impossible even for a moderate system.

To conduct a Bayesian analysis for the model, we assume a uniform prior on

$$(\alpha, \beta) \in \Theta = [-1, 1] \times [0, 1]$$

for all examples studied in Section 4.1. Then the double MH sampler can be applied to simulate from the posterior distribution $\pi(\alpha, \beta|\boldsymbol{x})$. In step (a), (α_t, β_t), the current state of the Markov chain, is updated by a single MH step with a random walk proposal $N_2((\alpha_t, \beta_t)', s^2 I_2)$, where s is the step size, and I_2 is the 2×2 identity matrix. In step (b), the auxiliary variable $\boldsymbol{y}$ is generated by a single cycle of Gibbs updates, and this translates to a value of $m = 2293$. Two or more cycles, which means a larger value of m, have also been tried for the examples, the results are similar. The acceptance rate of the double MH moves can be controlled by the choice of s. In this subsection, we set $s = 0.03$ for all examples.

4.1.1. *US cancer mortality data*

United States cancer mortality maps have been compiled by Riggan *et al.* [21] for investigating possible association of cancer with unusual demographic, environmental, industrial characteristics, or employment patterns. Figure 1(a) shows the mortality map of liver and gallbladder (including bile ducts) cancers for white males during the decade 1950–1959, which indicates some apparent geographic clustering. Refer to Sherman *et al.* [20] for more descriptions of the data. Following Sherman *et al.* [20], we modelled the data by a spatial autologistic model. The total number of spins is $|D| = 2293$. A free boundary condition is assumed for the model, under which the boundary points have less neighbouring points than the interior points. The assumption is natural to this example, as the lattice has an irregular shape.

The double MH sampler was first applied to this example. The sampler started with the initial value $(\alpha_0, \beta_0) = (0, 0)$ and was run five times independently. Each run consisted of 10,500 iterations. The CPU time cost by each run was 4.2 s on a 2.8 GHz computer (all computations reported in this article were done in the same computer). The overall acceptance rate of the double MH moves was about 0.23. Figure 2 provides two diagnostic plots for the convergence of the runs, where the statistic Gelman–Rubin $\hat{R}$ [17] was plotted against iterations. The simulations are usually considered as converged, when the statistic Gelman–Rubin $\hat{R}$ falls below the horizontal line 1.1. Figure 2 indicates that for this example, the simulations converged very fast, usually within two hundreds of iterations. Based on this diagnostic, we discarded the first 500 iterations of each run for the burn-in process, and collected 2000 samples from the remaining

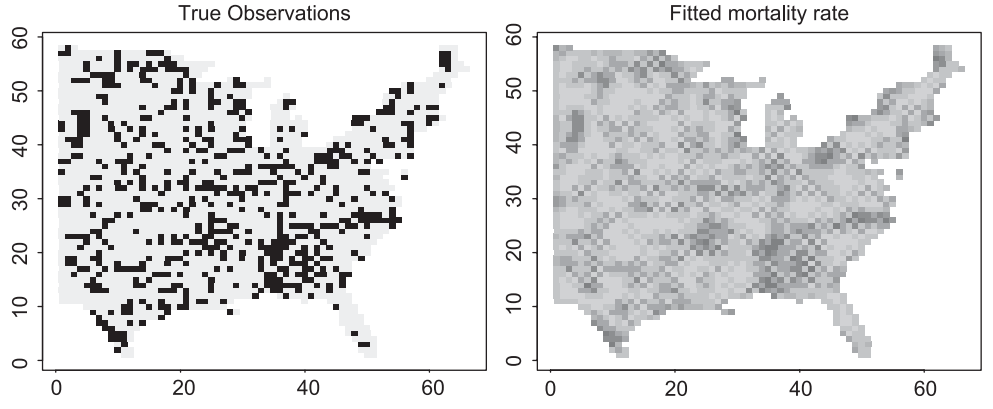

Figure 1. US cancer mortality data. (a) The mortality map of liver and gallbladder cancers (including bile ducts) for white males during the decade 1950–1959. Black squares denote counties of high cancer mortality rate, and white squares denote counties of low cancer mortality rate. (b) Fitted cancer mortality rates by the autologistic model with the parameters being replaced by its approximate Bayesian estimates. The cancer mortality rate of each county is represented by the gray level of the corresponding square.

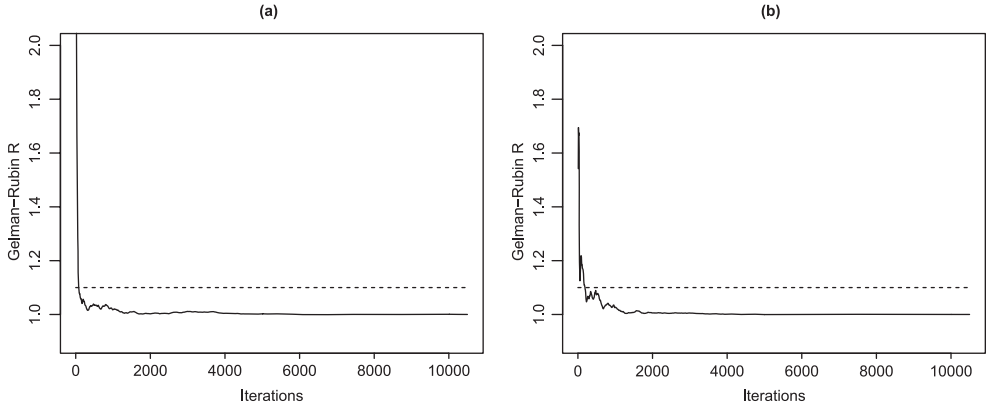

Figure 2. Convergence diagnostic of the double MH sampler for the US cancer mortality example: (a) Gelman–Rubin diagnostic based on the samples of α generated in five runs; (b) Gelman–Rubin diagnostic based on the samples of β generated in five runs.

iterations at equally spaced time points. Averaging over the estimates obtained from respective runs, we got the following estimate: $(\hat{\alpha}, \hat{\beta}) = (-0.3028, 0.1228)$ with the standard error $(8.2 \times 10^{-4}, 2.7 \times 10^{-4})$.

For comparison, the exchange algorithm was also applied to this example. It was run as the double MH sampler except that the auxiliary variable $\boldsymbol{y}$ was generated using an exact sampler. Following Murray *et al.* [11], we adopted the summary state algorithm [22] as our exact sampler, which is suitable for high-dimensional binary spaces. The algorithm was also run five times, and each run consisted of 10,500 iterations. The CPU time cost by each run was 111.5 s, about 27 times longer than that cost by the double MH sampler. The overall acceptance rate of the exact auxiliary variables was 0.2. Averaging over the estimates obtained from respective runs, we got the estimate $(\hat{\alpha}, \hat{\beta}) = (-0.3030, 0.1219)$ with the standard error $(1.1 \times 10^{-3}, 6.0 \times 10^{-4})$.

It is easy to see that the double MH sampler and the exchange algorithm produced almost identical estimates for this example. These estimates are also very close to the estimate $(-0.3008, 0.1231)$ obtained by Liang [9] using contour Monte Carlo, and the estimate $(-0.2999,$

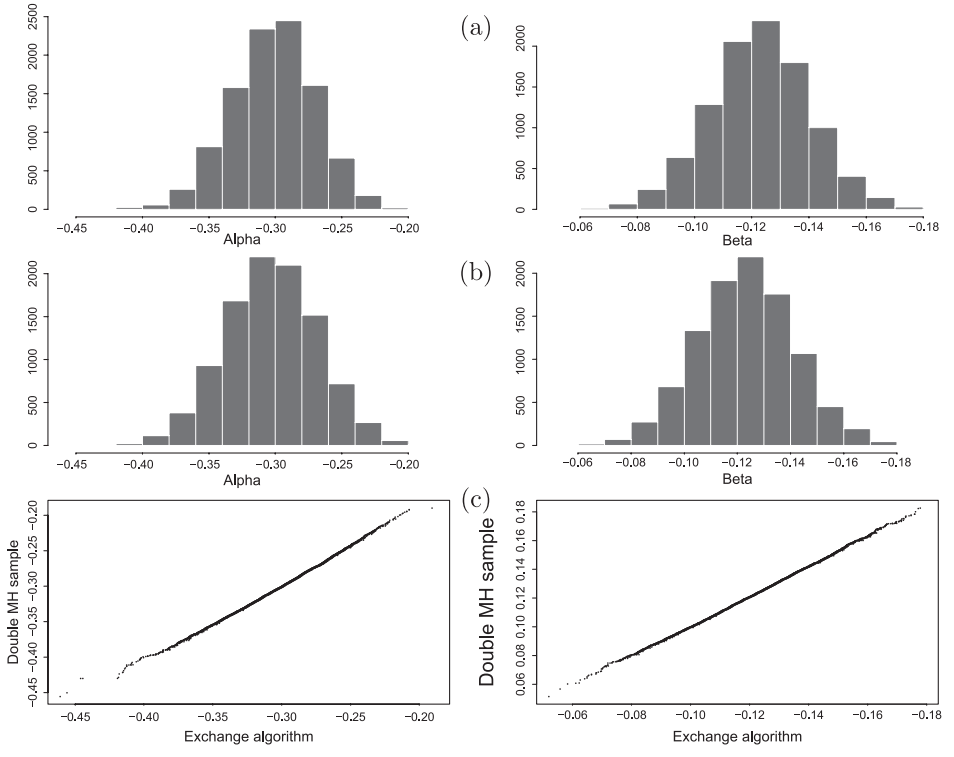

Figure 3. Panel (a) shows the histograms of the samples generated by the exchange algorithm. Panel (b) shows the histograms of the samples generated by the double MH sampler. Panel (c) shows the Q–Q plots for the samples generated by these two algorithms, the left plot is for the α samples, and the right plot is for the β samples.

0.1234) obtained by Liang *et al.* [8] using stochastic approximation Monte Carlo. We note that both the contour Monte Carlo and stochastic approximation Monte Carlo algorithms try to first approximate the unknown normalizing constant function, and then estimate the parameters based on the approximated normalizing constant function. As reported by the authors, both the algorithms take hours of CPU time to approximate the normalizing constant function. This data has also been analysed by Sherman *et al.* [20] using the Monte Carlo maximum likelihood algorithm [7], resulting in a similar estimate of $(-0.304, 0.117)$.

Later, both the double MH sampler and the exchange algorithm were re-run with the same parameter setting as specified above except for the number of iterations being lengthened to 100,500. In each run, 10,000 samples were collected from the last 100,000 iterations at equally spaced time points. The empirical distributions of the samples were studied in Figure 3. The plots indicate that the samples generated by the double MH sampler are almost identically distributed as those generated by the exchange algorithm.

In summary, for this example the double MH sampler produced almost identical results with the exchange algorithm while using much less CPU time. This advantage can be seen clearly in Section 4.1.2, where for some cases the exact sampler is impossible while the double MH sampler still works well.

4.1.2. *Simulation studies*

To assess the general accuracy of the estimates produced by the double MH sampler, we simulated 50 independent samples for the US cancer mortality data under each setting of (α, β)

Table 1. Computational results for the simulated US cancer mortality data.

	Double MH sampler			Exchange algorithm			MPLE	
(α, β)	$\hat{\alpha}$	$\hat{\beta}$	CPU[a](s)	$\hat{\alpha}$	$\hat{\beta}$	CPU[b](s)	$\hat{\alpha}$	$\hat{\beta}$
(0,0.1)[c]	−.0038	.1010	4.2	−.0038	.1002	103	−.0035	.1016
	(.0024)	(.0018)		(.0024)	(.0018)		(.0024)	(.0019)
(0,0.2)[c]	−.0026	.2018	4.2	−.0025	.2007	251	−.0024	.2025
	(.0021)	(.0019)		(.0020)	(.0019)		(.0022)	(.0022)
(0,0.3)[c]	−.0018	.2994	4.2	−.0014	.2971	821	−.0019	.2981
	(.0014)	(.0018)		(.0014)	(.0018)		(.0016)	(.0022)
(0,0.4)[c]	.0013	.4023	4.2	−.0007	.3980	7938	.0020	.4013
	(.0009)	(.0015)		(.0004)	(.0012)		(.0012)	(.0020)
(0.1,0.1)[c]	.1025	.0993	4.2	.1030	.0986	110	.1023	.0999
	(.0025)	(.0022)		(.0025)	(.0022)		(.0025)	(.0023)
(0.3,0.3)[c]	.2944	.3032	4.2	.3012	.3008	321	.2904	.3041
	(.0098)	(.0043)		(.0098)	(.0043)		(.0102)	(.0046)
(0.5,0.5)[d]	.5040	.5060	4.2	—	—	—	.5610	.4847
	(.0227)	(.0085)		—	—		(.0393)	(.0123)

Notes: The numbers in the parentheses denote the standard error of the estimates. —, not available.
[a]The CPU time cost by a single run of the double MH sampler.
[b]The CPU time cost by a single run of the exchange algorithm.
[c]The data were simulated using the exact sampler.
[d]The data were simulated using the Gibbs sampler, starting with a random configuration and then iterating for 100,000 Gibbs cycles.

given in Table 1. Since the lattice is irregular, the free boundary condition was again assumed in the simulations. We then re-estimated the parameters using the double MH sampler and the exchange algorithm. Both algorithms were run as in Section 4.1.1. The computational results were summarized in Table 1. For a thorough comparison, we also included in Table 1 the maximum pseudo-likelihood estimators (MPLE) of the parameters [1] , which were obtained by maximizing the pseudo-likelihood function

$$\tilde{L}(\alpha, \beta|\boldsymbol{x}) = \prod_{i \in D} \left[\frac{\mathrm{e}^{\alpha \mathrm{x_i} + \beta \sum_{\mathrm{j} \in \mathrm{n(i)}} \mathrm{x_i x_j}}}{\mathrm{e}^{\alpha + \beta \sum_{j \in n(i)} x_j} + \mathrm{e}^{-\alpha - \beta \sum_{j \in n(i)} \sum x_j}} \right]$$

using the downhill simplex method [23]. The advantage of this method is that it does not require the gradient information of the objective function, and can thus be easily applied to the constraint optimization problems.

Table 1 indicates that the double MH sampler can produce almost the same accurate results as does the exchange algorithm, and more accurate results than does the MPLE especially when α and β are large. It is remarkable that the CPU time cost by the double MH sampler is independent of the values of (α, β). While as β increases, the CPU time cost by the exchange algorithm increases exponentially. Childs *et al.* [22] studied the behaviour of the exact sampler for the Ising model, a simplified autologistic model. For the Ising model, they fitted an exponential law for the convergence time, and reported that the exact sampler may diverge at a value of β lower than the critical value ($\approx$0.44). Childs *et al.*'s finding is consistent with our results. It takes an extremely long CPU time for the exact sampler to generate a sample under the settings $(0, 0.4)$ and $(0.5, 0.5)$. We note that due to the effect of α, it usually takes a longer CPU time for the exact sampler to generate a sampler under the setting $(0, \beta)$ than under the setting (α, β); and that when α and β are large, the accuracy of the estimates tends to be reduced by their correlation.

4.2. *Autonormal models*

Consider a second-order zero-mean Gaussian Markov random field $\boldsymbol{X} = (X_{ij})$ defined on an $M \times N$ lattice, whose conditional density function is given by

$$f(x_{ij}|\boldsymbol{\beta}, \sigma^2, x_{uv}; (u, v) \neq (i, j)) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left\{-\frac{1}{2\sigma^2}(x_{ij} - \beta_h \sum_{(u,v)\in n_h(i,j)} x_{uv} - \beta_v \sum_{(u,v)\in n_v(i,j)} x_{uv} - \beta_d \sum_{(u,v)\in n_d(i,j)} x_{uv})^2\right\}, \tag{15}$$

where $\boldsymbol{\beta} = (\beta_h, \beta_v, \beta_d)$ and σ^2 are parameters, $n_h(i, j) = \{(i, j-1), (i, j+1)\}$, $n_v(i, j) = \{(i-1, j), (i+1, j)\}$ and $n_d(i, j) = \{(i-1, j-1), (i-1, j+1), (i+1, j-1), (i+1, j+1)\}$ are neighbours of (i, j). This model is stationary when $|\beta_h| + |\beta_v| + 2|\beta_d| < 0.5$ [24]. The joint likelihood function of this model can be written as

$$f(\boldsymbol{x}|\beta, \sigma^2) = (2\pi\sigma^2)^{-MN/2}|B|^{1/2} \exp\left\{-\frac{1}{2\sigma^2}\boldsymbol{x}'B\boldsymbol{x}\right\},$$

where B is an $(MN \times MN)$-dimensional matrix, e.g. a 2500×2500 matrix corresponding to a small lattice of size 50×50, and $|B|$ is intractable except for some special cases [25].

To conduct a Bayesian analysis for the model, we assume the following priors:

$$\pi(\beta) \propto I(|\beta_h| + |\beta_v| + 2|\beta_d| < 0.5), \quad \pi(\sigma^2) \propto \frac{1}{\sigma^2}, \tag{16}$$

which $I(\cdot)$ is the indicator function. Under the free boundary condition for which the boundary pixels have fewer neighbours, we have the following posterior distribution

$$\pi(\beta, \sigma^2|\boldsymbol{x}) \propto (\sigma^2)^{-(MN/2)-1}|B|^{1/2} \exp\left\{-\frac{MN}{2\sigma^2}(S_x - 2\beta_h X_h - 2\beta_v X_v - 2\beta_d X_d)\right\} \times I(|\beta_h| + |\beta_v| + 2|\beta_d| < 0.5), \tag{17}$$

where

$$S_x = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N} x_{ij}^2, \quad X_h = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N-1} x_{ij}x_{i,j+1},$$

$$X_v = \frac{1}{MN}\sum_{i=1}^{M-1}\sum_{j=1}^{N} x_{ij}x_{i+1,j}, \quad X_d = \frac{1}{MN}\left(\sum_{i=1}^{M-1}\sum_{j=1}^{N-1} x_{ij}x_{i+1,j+1} + \sum_{i=1}^{M-1}\sum_{j=2}^{N} x_{ij}x_{i+1,j-1}\right).$$

Although σ^2 can be integrated out from the posterior, we do not suggest to do so. Working on the joint posterior will ease the generation of auxiliary variables for the double MH sampler. To ease implementation of sampling from the prior distribution, we reparametrize σ^2 by $\tau = \log(\sigma^2)$, and then we have

$$\pi(\beta, \tau) \propto I(|\beta_h| + |\beta_v| + 2|\beta_d| < 0.5).$$

The double MH sampler can be applied to simulate from the posterior distribution of β and τ. In step (a), (β_t, τ_t) is updated by a single MH step with a random walk proposal $N((\beta_t, \tau_t)', s^2 I_4)$.

In this subsection, we set $s = 0.02$ unless otherwise stated. In step (b), the auxiliary variable $\boldsymbol{y}$ is generated by a single cycle of Gibbs updates:

$$y_{ij}|\boldsymbol{y}_{(u,v)\in n(i,j)} \sim N\left(\beta_h \sum_{(u,v)\in n_h(i,j)} y_{uv} + \beta_v \sum_{(u,v)\in n_v(i,j)} y_{uv} + \beta_d \sum_{(u,v)\in n_d(i,j)} y_{uv}, \mathrm{e}^{\tau_t}\right),$$

for $i = 1, \ldots, M$ and $j = 1, \ldots, N$, starting with $\boldsymbol{y} = \boldsymbol{x}$.

The exchange algorithm is not applicable to the autonormal model, as exact sampling is not available for it. However, under the free boundary condition, the log-likelihood function of the model admits the following analytic form [24]:

$$l(\boldsymbol{X}|\beta, \sigma^2) = \text{Constant} - \frac{MN}{2}\log(\sigma^2) - \frac{MN}{2\sigma^2}(S_x - 2\beta_h X_h - 2\beta_v X_v - 2\beta_d X_d) + \frac{1}{2}\sum_{i=1}^{M}\sum_{j=1}^{N} \times \log\left(1 - 2\beta_v \cos\frac{i\pi}{M+1} - 2\beta_h \cos\frac{j\pi}{N+1} - 4\beta_d \cos\frac{i\pi}{M+1}\cos\frac{j\pi}{N+1}\right), \tag{18}$$

where S_x, X_h, X_v, and X_d are as defined in Equation (17). The Bayesian inference for the model is then standard, with the priors as specified in Equation (16). In this article, we call the Bayesian analysis based on this analytic likelihood function, the true Bayesian analysis, and call the resulting estimator, the true Bayesian estimator.

For a thorough comparison, we also considered MPLEs for this model, which are to find the parameter values that maximize the pseudo-likelihood function:

$$\widetilde{L}(\beta, \sigma^2|\boldsymbol{x}) = \prod_{i=1}^{M}\prod_{j=1}^{N} f(x_{ij}|\beta, \sigma^2, x_{uv}; (u, v) \neq (i, j)).$$

The maximization can be accomplished using the downhill simplex method [23].

4.2.1. *Wheat yield data*

We first work on the wheat yield data collected on a 20×25 rectangular lattice [26, Table 6.1]. The data was shown in Figure 4(a), which indicates positive correlation between neighbouring observations. This data has been analysed by a number of authors, [1], Huang and Ogata [27] and Gu and Zhu [28]. Following the previous authors, we subtracted the mean from the data and then fitted them by the autonormal model. In our analysis, the free boundary condition is assumed. This is natural, as for the real data the lattice is often irregular.

The double MH sampler is applied to this example. The sampler was run five times independently. Each run started with the point (0,0,0,0) and consisted of 50,500 iterations, for which the first 500 iterations were discarded for the burn-in process and 10,000 samples were collected from the remaining iterations at equally spaced time points. The CPU time cost by each run was 5.8 s. The overall acceptance rate of the double MH moves was about 0.23. The results were summarized in Table 2. To assess the quality of the estimators, we also included in the table the mean squared error of the fitted values (FMSE), which is defined as

$$\text{FMSE} = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}(x_{ij} - \hat{x}_{ij}),$$

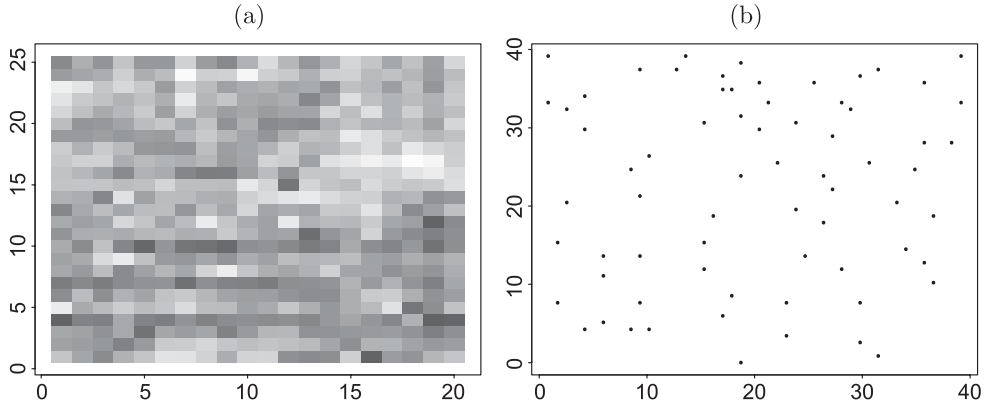

Figure 4. (a) Image of the wheat yield data: black squares denote high yield ares, and white squares denote low yield areas. (b) Locations of 69 Spanish town in an area of 40×40 square miles.

Table 2. Computational results for the wheat yield data.

Algorithm	β_h	β_v	β_d	σ^2	FMSE
True Bayes	0.102(4e−4)	0.355(3e−4)	0.006(2e−4)	0.123(2e−4)	0.123(0.0)
DMH	0.099(6e−4)	0.351(5e−4)	0.006(3e−4)	0.126(3e−4)	0.123(0.0)
MPLE	0.140	0.340	−0.010	0.122	0.122

Note: The numbers in the parentheses denote the standard error of the estimates.

and $\hat{x}_{ij}$ denotes the fitted value of x_{ij}. FMSE provides a square measure for the difference between the fitted and true observations. For the Bayesian method, $\hat{x}_{ij}$ can be calculated by

$$\hat{x}_{ij} = \frac{1}{n}\sum_{t=1}^{n}\left(\beta_h^{(t)}\sum_{(u,v)\in n_h(i,j)} x_{uv} + \beta_v^{(t)}\sum_{(u,v)\in n_v(i,j)} x_{uv} + \beta_d^{(t)}\sum_{(u,v)\in n_d(i,j)} x_{uv}\right),$$

given the MCMC samples $(\boldsymbol{\beta}^{(t)}, \tau^{(t)})$, $t = 1, \ldots, n$.

For comparison, the true Bayesian and MPLE methods were also applied to this example. In the true Bayesian analysis, the posterior was simulated using the MH algorithm five times. Each run also consisted of 50,500 iterations, with the first 500 iterations being discarded for the burn-in process and 10,000 samples being collected from the remaining iterations at equally spaced time points. The proposal adopted here was a random walk proposal with the variance–covariance matrix $0.02^2 I_4$. The overall acceptance rate of the proposals was about 0.22. The numerical results were also summarized in Table 2. The comparison indicates that the estimates produced by DMH are much closer to the true Bayesian estimates than are MPLEs, while the three methods all produced about the same FMSEs.

4.2.2. *Simulation studies*

To assess the general accuracy of the double MH estimator for the autonormal models, we simulated 50 independent samples of size 100×100 under each setting of (β, σ^2) given in Table 3. Without loss of generality, we set $\sigma^2 = 1.0$. The simulations were done using the Gibbs sampler, starting with a random configuration with each entry being drawn independently from $N(0, 1)$, and then iterating for 50,000 Gibbs cycles. The free boundary condition was again assumed in the simulations. The parameters were re-estimated using the double MH sampler, true Bayesian

Table 3. Computational results for the simulated autonormal data. All entries of the table have been scaled by a factor of 1000. The true parameter values $(\beta_h, \beta_v, \beta_d, \sigma^2)$ are (0.1,0.1,0.0,1), (0.15,0.1,0.1,1.0), and (0.2,0.15,0.05,1.0) for the settings I, II, and III, respectively. Let θ be the true value of a parameter, and let $\hat{\theta}^{(i)}$ denote its estimate obtained from data set i, then 'bias'$= \sum_{i=1}^{50}(\hat{\theta}^{(i)} - \theta)/50$, 'se' is the standard error of $\sum_{i=1}^{50}\hat{\theta}^{(i)}/50$, and 'rmse'$= \sqrt{\sum_{i=1}^{50}(\hat{\theta}^{(i)} - \theta)^2/50}$.

Algorithm		True Bayesian method			DMH			MPLE		
Setting		I	II	III	I	II	III	I	II	III
β_h	bias	0.34	−0.75	3.04	−0.29	−1.42	1.60	0.22	−1.96	2.68
	se	1.55	1.22	1.12	1.52	1.22	1.14	1.63	1.46	1.25
	rmse	10.83	8.56	8.44	10.66	8.67	8.14	11.39	10.40	9.18
β_v	bias	−1.73	0.92	0.35	−2.28	0.73	−0.32	−2.65	1.77	−1.39
	se	1.25	1.26	1.41	1.24	1.25	1.41	1.31	1.85	1.67
	rmse	8.95	8.90	9.91	8.98	8.80	9.89	9.52	13.06	11.75
β_d	bias	−0.44	0.24	−1.15	−0.23	0.57	0.16	−0.39	0.36	−0.63
	se	1.08	0.88	1.04	1.08	0.84	2.30	1.11	0.92	1.17
	rmse	7.60	6.15	7.36	7.57	5.94	7.57	7.79	6.48	8.23
σ^2	bias	−4.08	−13.02	−14.01	0.36	−0.48	0.16	0.05	−1.68	2.18
	se	2.02	2.46	2.27	2.05	2.43	2.30	2.05	2.93	2.90
	rmse	14.69	21.57	21.21	14.36	17.02	16.07	14.36	20.55	20.43
Score	mean	1418.7	1418.7	1419.1	1418.7	1418.7	1419.1	1418.8	1418.9	1419.2
	se	0.96	0.99	1.08	0.96	0.97	1.07	0.96	0.96	1.07

method, and MPLE method. These methods were run as in Section 4.2.1 except for the choice of s. Here we set $s = 0.005$ and this resulted in an acceptance rate of 0.35 for the double MH sampler and an acceptance rate of 0.37 for the true Bayesian method. The numerical results were summarized in Table 3. Instead of FMSEs, we considered for this example the predictive log-score [29], which is defined as

$$-\frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}\log\left\{\frac{1}{n}\sum_{t=1}^{n}P(z_{ij}|\beta^{(t)}, \tau^{(t)}, z_{uv}; (u, v) \neq (i, j))\right\},$$

given the MCMC samples $(\beta^{(t)}, \tau^{(t)})$, $t = 1, \ldots, n$. Here $\boldsymbol{z}$ denotes a sample simulated independently of $\boldsymbol{x}$ but under the same parameter setting. As argued by Hoeting *et al.* [29], the predictive score is a combined measure of the predictive bias and the lack of calibration. The smaller the score, the better the predictive performance.

Table 3 indicates that the double MH sampler outperforms the MPLE method for this example; the double MH estimates consistently have smaller biases, standard errors, RMSEs, and predictive log-scores than the MPLEs. It is remarkable that for the parameters β_h, β_v, and β_d, the DMH estimates are quite comparable with the true Bayesian estimates; and for σ^2, the DMH estimate is even better than the true Bayesian estimate.

4.3. *Pairwise interacted spatial point process*

The spatial point process is described by the coordinates of points $\boldsymbol{x} = \{x_i \in A : i = 1, \ldots, n\}$ in a planar region A. If the points are pairwise interacted, the joint density can be written as

$$f(\boldsymbol{x}|\theta) = \frac{1}{Z(\theta)}\exp\left\{-\sum_{i=1}^{n}\sum_{j>i}\phi(\|x_i - x_j\|, \theta)\right\}, \quad \theta > 0, \tag{19}$$

where $\phi(\cdot)$ is called the pairwise potential function. If we define

$$\phi(t, \theta) = -\log\{1 - \exp(-\rho t^2/\theta)\},$$

then the model (19) is called the very-soft-core model, where $\rho = n/|A|$ and $|A|$ is the area of region A. The normalizing constant of the model is

$$Z(\theta) = \int_A \cdots \int_A \exp\left\{-\sum_{i=1}^{n}\sum_{j>i}\phi(\|x_i - x_j\|, \theta)\right\} \mathrm{d}x_1 \cdots \mathrm{d}x_n,$$

which is intractable. To estimate the parameter θ, various approximations to the function $Z(\theta)$ have been proposed in the literature, see Huang and Ogata [27] for an overview.

To conduct a Bayesian analysis for the model, we let θ be subject to the following prior:

$$\pi(\theta) \propto \frac{1}{\theta},$$

and then re-parametrize it by $\tau = \log(\theta)$. The double MH sampler can then be applied to simulate from the posterior density of τ. In step (a), τ_t, the current state of the Markov chain, is updated by a MH step with a random walk proposal $N(\tau_t, s^2)$, where we set $s = 1.5$ for the Spanish town example studied below. In step (b), the auxiliary variable $\boldsymbol{y}$ is simulated through a single cycle of Metropolis-within-Gibbs moves [30] :

$$y_i \sim f(y_i|\boldsymbol{y}_{[-i]}) \propto \exp\left\{-\sum_{j\neq i}\phi(\|y_j - y_i\|, \mathrm{e}^{\tau})\right\}, \quad i = 1, \ldots, n, \tag{20}$$

where $\boldsymbol{y}$ is initialized at $\boldsymbol{x}$, and $\boldsymbol{y}_{[-i]}$ denotes a subset of $\boldsymbol{y}$ with y_i being deleted.

We fitted the very-soft-core model to the Spanish town data shown in Figure 4(b). The data consists of $n = 69$ points in an area of 40×40 square miles. This data set has been analysed by Ripley [31], Ogata and Tanemura [32] and Gu and Zhu [28] using the same model. The double MH sampler was applied to this example. In the Metropolis-within-Gibbs cycle, each y_i is updated through 10 consecutive MH steps with a random walk proposal $N(y_i, 5^2 I_2)$. The sampler was run five times independently. Each run consisted of 20,500 iterations, where the first 500 iterations were discarded for the burn-in process, and 4000 samples were collected from the remaining iterations at equally spaced time points. The overall acceptance rate of the double MH moves was 0.27, and the CPU time cost by each run was about 266 s. Averaging over the estimates obtained from the five runs, we got an estimate $\hat{\theta} = 0.176$ with the standard error 0.001. This estimate is consistent with the Monte Carlo MLE 0.167 obtained by Gu and Zhu [28], but the latter has a large standard error of 0.078.

5. Discussion

We have proposed the double MH sampler for conducting an approximate Bayesian analysis for the models with intractable normalizing constants. The double MH sampler removes the need for exact sampling, the auxiliary variables being generated using MH kernels, and thus can be applied to a wide range of problems for which exact sampling is not available or very expensive. Besides the spatial models studied in the article, the double MH sampler can be directly applied to many other scientific models or problems, such as image segmentation [2], social network modelling [33], and genetic analysis [4].

As a practical hint, we would like to point out that the MCMC sampler used for generating auxiliary variables is the key to the efficiency of the double MH sampler. In this article, we used the Gibbs sampler for the first and second examples, and used the Metropolis-within-Gibbs sampler

for the third example. Theoretically, any MCMC samplers can be used here, and some may be even more efficient than the ones we used. For example, the Swendsen–Wang algorithm [34] or the Wolff algorithm [35] can be used for the autologistic models, and they are expected to be more efficient than the plain Gibbs sampler we used. Similarly, the block Gibbs sampler [36] can be used for the spatial point process with the points being grouped appropriately.

When the dimension of the problem is high, the curse of dimensionality may be a serious difficulty for the MH sampler. To maintain a given level of quality of auxiliary variables, its number of iterations need to increase exponentially with dimension. In this case, the sequential parallel tempering algorithm [37] may be used. As demonstrated in Liang [37], the sequential parallel tempering algorithm can significantly reduce the curse of dimensionality suffered by the MH sampler.

References

[1] J.E. Besag, *Spatial interaction and the statistical analysis of lattice systems (with discussion)*, J. R. Statist. Soc. B 36 (1974), pp. 192–236.
[2] M. Hurn, O. Husby, and H. Rue, *A tutorial on image analysis*, Lecture Notes Statist. 173 (2003), pp. 87–141.
[3] P.J. Green and S. Richardson, *Hidden Markov models and disease mapping*, J. Amer. Statist. Assoc. 97 (2002), pp. 1055–1070.
[4] O. Francois, S. Ancelet, and G. Guillot, *Bayesian clustering using hidden Markov random fields in spatial population genetics*, Genetics 174 (2006), pp. 805–816.
[5] I. Dryden, L. Ippoliti, and L. Romagnoli, *Adjusted maximum likelihood and pseudo-likelihood estimation for noisy Gaussian Markov random fields*, J. Comput. Graph. Statist. 11 (2002), pp. 370–388.
[6] F. Huang and Y. Ogata, *Generalized pseudo-likelihood estimates for Markov random fields on lattice*, Ann. Inst. Statist. Math. 54 (2002), pp. 1–18.
[7] C.J. Geyer and E.A. Thompson, *Constrained Monte Carlo maximum likelihood for dependent data*, J. R. Statist. Soc. B 54 (1992), pp. 657–699.
[8] F. Liang, C. Liu, and R.J. Carroll, *Stochastic approximation in Monte Carlo computation*, J. Amer. Statist. Assoc. 102 (2007), pp. 305–320.
[9] F. Liang, *Continuous contour Monte Carlo for marginal density estimation with an application to a spatial statistical model*, J. Comput. Graph. Statist. 16 (2007), pp. 608–632.
[10] J. Møller, A.N. Pettitt, R. Reeves, and K.K. Berthelsen, *An efficient Markov chain Monte Carlo method for distributions with intractable normalizing constants*, Biometrika 93 (2006), pp. 451–458.
[11] I. Murray, Z. Ghahramani, and D.J.C. MacKay, *MCMC for doubly-intractable distributions*, Proceedings of the 22nd Annual Conference on Uncertainty in Artificial Intelligence (UAI), 2006.
[12] G. Propp and D.B. Wilson, *Exact sampling with coupled Markov chains and applications to statistical mechanics*, Random Struc. Algor. 9 (1996), pp. 223–252.
[13] C.J. Geyer, *Markov chain Monte Carlo maximum likelihood*, in *Computing Science and Statistics: Proceedings of the 23rd Symposium on the Interface*, E.M. Keramigas, ed., Interface Foundation, Fairfax, 1991, pp. 156–163.
[14] K. Hukushima and K. Nemoto, *Exchange Monte Carlo method and application to spin glass simulations*, J. Phys. Soc. Jpn. 65 (1996), pp. 1604–1608.
[15] S. Haykin, *Neural Networks: A Comprehensive Foundation*, 2nd ed., Prentice Hall International, Inc., New Jersey, 1999.
[16] A. Gelman, R.O. Roberts, and W.R. Gilks, *Efficient Metropolis jumping rules*, in *Bayesian Statistics 5*, J.M. Bernardo, J.O. Berger, A.P. Dawid, and A.F.M. Smith, eds., Oxford University Press, New York, 1996.
[17] A. Gelman and D.B. Rubin, *Inference from iterative simulation using multiple sequences (with discussion)*, Statist. Sci. 7 (1992), pp. 457–472.
[18] H.K. Preisler, *Modeling spatial patterns of trees attacked by bark-beetles*, Appl. Statist. 42 (1993), pp. 501–514.
[19] H. Wu and F.W. Huffer, *Modeling the distribution of plant species using the autologistic regression model*, Ecol. Statist. 4 (1997), pp. 49–64.
[20] M. Sherman, T.V. Apanasovich, and R.J. Carroll, *On estimation in binary autologistic spatial models*, J. Statist. Comput. Simul. 76 (2006), pp. 167–179.
[21] W.B. Riggan, J.P. Creason, W.C. Nelson, K.G. Manton, M.A. Woodbury, E. Stallard, A.C. Pellom, and J. Beaubier, *U.S. Cancer Mortality Rates and Trends 1950–1979* Vol. IV: Maps, U.S. Environmental Protection Agency, U.S. Government Printing Office, Washington DC, 1987.
[22] A.M. Childs, R.B. Patterson, and D.J.C. MacKay, *Exact sampling from nonattractive distributions using summary states*, Phys. Rev. E. 63 (2001), p. 036113.
[23] W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery, *Numerical Recipes in C*, 2nd ed., Cambridge University Press, Cambridge, 1992.
[24] N. Balram and J.M.F. Moura, *Noncausal Gauss Markov random fields: parameter structure and estimation*, IEEE Trans. Inform. Theory 39 (1993), pp. 1333–1355.

[25] J.E. Besag and P.A.P. Moran, *On the estimation and testing of spatial interaction in Gaussian lattice processes*, Biometrika 62 (1975), pp. 555–562.
[26] D.F. Andrews and A.M. Herzberg, *Data*, Springer, New York, 1985.
[27] F. Huang and Y. Ogata, *Improvements of the maximum pseudo-likelihood estimators in various spatial statistical models*, J. Comput. Graph. Statist. 8 (1999), pp. 510–530.
[28] M.G. Gu and H. Zhu, *Maximum likelihood estimation for spatial models by Markov chain Monte Carlo stochastic approximation*, J. R. Statist. Soc. B 63 (2001), pp. 339–355.
[29] J.A. Hoeting, D. Madigan, A.E. Raftery, and C.T. Volinsky, *Bayesian model averaging: A tutorial*, Statist. Sci. 14 (1999), pp. 382–417.
[30] P. Müller, *Alternatives to the Gibbs sampling scheme*, Tech. Rep., Institute pf Statistics and Decision Sciences, Duke University, 1993.
[31] B.D. Ripley, *Modeling spatial patterns (with discussion)*, J. R. Statist. Soc. B 39 (1977), pp. 172–212.
[32] Y. Ogata and M. Tanemura, *Likelihood analysis of spatial point patterns*, J. R. Statist. Soc. B 46 (1984), pp. 496–518.
[33] S. Wasserman and K. Faust, *Social Network Analysis: Methods and Applications*, Cambridge University Press Cambridge, 1994.
[34] R.H. Swendsen and J.S. Wang, *Nonuniversal critical dynamics in Monte Carlo simulations*, Phys. Rev. Lett. 58 (1987), pp. 86–88.
[35] U. Wolff, *Collective Monte Carlo updating for spin systems*, Phys. Rev. Lett. 62 (1989), pp. 361–364.
[36] J.S. Liu, W.H. Wong, and A. Kong, *Covariance structure and convergence rate of the Gibbs samples with various scans*, J. R. Statist. Soc. B 57 (1995), pp. 157–169.
[37] F. Liang, *Use of sequential structure in simulation from high dimensional systems*, Phys. Rev. E 67 (2003), pp. 56101–56107.