Stochastic Approximation in Monte Carlo Computation

Faming Liang, Chuanhai Liu and Raymond J. Carroll[1]

June 26, 2006

Abstract

The Wang-Landau algorithm is an adaptive Markov chain Monte Carlo algorithm to calculate the spectral density for a physical system. A remarkable feature of the algorithm is that it is not trapped by local energy minima, which is very important for systems with rugged energy landscapes. This feature has led to many successful applications of the algorithm in statistical physics and biophysics. However, there does not exist rigorous theory to support its convergence, and the estimates produced by the algorithm can only reach a limited statistical accuracy. In this paper, we propose the stochastic approximation Monte Carlo (SAMC) algorithm, which overcomes the shortcomings of the Wang-Landau algorithm. We establish a theorem concerning its convergence. The estimates produced by SAMC can be improved continuously as the simulation goes on. SAMC also extends applications of the Wang-Landau algorithm to continuum systems. The potential uses of SAMC in statistics are discussed through two classes of applications, importance sampling and model selection. The results show that SAMC can work as a general importance sampling algorithm and a model selection sampler when the model space is complex.

Keywords: Importance Sampling; Markov Chain Monte Carlo; Model Selection; Spatial Autologistic Model; Stochastic Approximation; Wang-Landau algorithm.

[1]Faming Liang is Associate Professor, Department of Statistics, Texas A&M University, College Station, TX 77843-3143 (E-mail: fliang@stat.tamu.edu). Liang's research was supported by grants from the National Science Foundation (DMS-0405748) and the National Cancer Institute (CA104620). Chuanhai Liu is Professor, Department of Statistics, Purdue University, 150 N. University Street, West Lafayette, IN 47907-2067 (Email: chuanhai@stat.purdue.edu). Raymond J. Carroll is Distinguished Professor, Department of Statistics, Texas A&M University, College Station, TX 77843-3143 (Email: carroll@stat.tamu.edu). Carroll's research was supported by a grant from the National Cancer Institute (CA57030), and by the Texas A&M Center for Environmental and Rural Health via a grant from the National Institute of Environmental Health Sciences (P30ES09106). The authors thank Professor Walter W. Piegorsch, the associate editor and three referees for their suggestions and comments which have led to a significant improvement of this paper,

1 Introduction

Suppose that we are interested in sampling from a distribution which, for convenience, is written in the following form,

$$p(\boldsymbol{x}) = cp_0(\boldsymbol{x}), \quad \boldsymbol{x} \in \mathcal{X}, \tag{1}$$

where c is a constant and $\mathcal{X}$ is the sample space. As known by many researchers, the Metropolis-Hastings (MH) sampler (Metropolis *et al*, 1953; Hastings, 1970) is prone to becoming trapped in local energy minima when the energy landscape of the distribution is rugged (in terms of physics, $-\log\{p_0(\boldsymbol{x})\}$ is called the energy function of the distribution). Over the last two decades a number of advanced Monte Carlo algorithms have been proposed to overcome this problem, mainly based on the following two ideas.

The first idea is the use of auxiliary variables. Included in this category are the Swendsen-Wang algorithm (Swendsen and Wang, 1987), simulated tempering (Marinari and Parisi, 1992; Geyer and Thompson, 1995), parallel tempering (Geyer, 1991, Hukushima and Nemoto, 1996), evolutionary Monte Carlo (Liang and Wong, 2001), etc. In these algorithms, the temperature is typically treated as an auxiliary variable. Simulations at high temperatures broaden sampling of the sample space, and thus are able to help the system escape from local energy minima.

The second idea is the use of past samples. The multicanonical algorithm (Berg and Neuhaus, 1991) is apparently the first work in this direction. This algorithm is essentially a dynamic importance sampling algorithm, where the trial distribution is learned dynamically from past samples. Related works include the $1/k$-ensemble algorithm (Hesselbo and Stinchcombe, 1995), the Wang-Landau (WL) algorithm (Wang and Landau, 2001), and the generalized Wang-Landau (GWL) algorithm (Liang, 2004, 2005). They are different from the multicanonical algorithm only in the specification and/or the learning scheme for the trial distribution. Other work included in this category is dynamic weighting (Wong and Liang, 1997; Liu, Liang and Wong, 2001, Liang, 2002), where the acceptance rate of the MH moves is adjusted dynamically with an importance weight which carries the information of past samples.

Among the algorithms described above, the WL algorithm has received much attention in physics recently. It can be described as follows. Suppose that the sample space $\mathcal{X}$ is finite. Let $U(\boldsymbol{x}) = -\log\{p_0(\boldsymbol{x})\}$ denote the energy function, $\{u_1, \ldots, u_m\}$ be a set of real numbers containing all possible values of $U(\boldsymbol{x})$, and $g(u) = \#\{\boldsymbol{x} : U(\boldsymbol{x}) = u\}$ be the number

of states with energy equal to u. In physics, $g(u)$ is called the spectral density or the density of states of the distribution. For simplicity, we also denote $g(u_i)$ by g_i in the following. The WL algorithm is an adaptive Markov chain Monte Carlo algorithm designed to estimate $\boldsymbol{g} = (g_1, \ldots, g_m)$. Let $\widehat{g}_i$ be the working estimate of g_i. A run of WL consists of several stages. The first stage starts with the initial estimates $\widehat{g}_1 = \cdots = \widehat{g}_m = 1$ and a sample drawn from $\mathcal{X}$ at random, and iterates between the following steps.

The WL algorithm:

(*a*) Simulate a sample $\boldsymbol{x}$ by a single Metropolis update with the invariant distribution $\widehat{p}(\boldsymbol{x}) \propto 1/\widehat{g}(U(\boldsymbol{x}))$.

(*b*) Set $\widehat{g}_i \leftarrow \widehat{g}_i \delta^{I(U(\boldsymbol{x})=u_i)}$ for $i = 1, \ldots, m$, where δ is a gain factor greater than 1 and $I(\cdot)$ is an indicator function.

The algorithm iterates until a flat histogram has been produced in the space of energy. Once the histogram is flat, the algorithm will restart by passing on $\widehat{g}(u)$ as the initial value of the new stage and reducing δ to a smaller value according to a pre-specified scheme, say, $\delta \leftarrow \sqrt{\delta}$. The process is repeated until δ is very close to 1, say, $\log(\delta) \leq 10^{-8}$. In Wang and Landau (2001), a histogram is regarded as flat if the sampling frequency for each energy value is not less than 80% of the average sampling frequency.

Liang (2005) generalized the WL algorithm to continuum systems. The generalization is mainly in three respects, the sample space, the working function and the estimate updating scheme. Suppose that the sample space $\mathcal{X}$ is continuous and has been partitioned according to a chosen parameterization, say, the energy function $U(\boldsymbol{x})$, into m disjoint subregions denoted by $E_1 = \{\boldsymbol{x} : U(\boldsymbol{x}) \leq u_1\}$, $E_2 = \{\boldsymbol{x} : u_1 < U(\boldsymbol{x}) \leq u_2\}$, ..., $E_{m-1} = \{\boldsymbol{x} : u_{m-2} < U(\boldsymbol{x}) \leq u_{m-1}\}$, and $E_m = \{\boldsymbol{x} : U(\boldsymbol{x}) > u_{m-1}\}$, where $u_1, \ldots, u_{m-1}$ are $m-1$ specified real numbers. Let $\psi(\boldsymbol{x})$ be a non-negative function defined on the sample space with $0 < \int_{\mathcal{X}} \psi(\boldsymbol{x}) d\boldsymbol{x} < \infty$. In practice, $\psi(\boldsymbol{x})$ is often set to $p_0(\boldsymbol{x})$ defined in (1). Let $g_i = \int_{E_i} \psi(\boldsymbol{x}) dx$. One iteration of GWL consists of the following steps.

The GWL algorithm:

(a) Simulate a sample $\boldsymbol{x}$ by a number, denoted by κ, of MH steps of which the invariant distribution is defined as follows,

$$\widehat{p}(\boldsymbol{x}) \propto \sum_{i=1}^{m} \frac{\psi(\boldsymbol{x})}{\widehat{g}_i} I(\boldsymbol{x} \in E_i). \tag{2}$$

(b) Set $\widehat{g}_{J(\boldsymbol{x})+k} \leftarrow \widehat{g}_{J(\boldsymbol{x})+k} + \delta\varrho^k\widehat{g}_{J(\boldsymbol{x})+k}$ for $k = 0, \ldots, m - J(\boldsymbol{x})$, where $J(\boldsymbol{x})$ is the index of the subregion $\boldsymbol{x}$ belongs to and $\varrho > 0$ is a parameter which controls the sampling frequency for each of the subregions.

The extension of g_i from the density of states to the integral $\int_{E_i} \psi(\boldsymbol{x})dx$ is of great interest to statisticians, as this leads to direct applications of the algorithm to model selection, HPD region construction, and many other Bayesian computational problems. Liang (2005) also studied the convergence of the GWL algorithm: as κ becomes large, $\widehat{g}_i$ is consistent for g_i. However, when κ is small, say, $\kappa = 1$, the choice adopted by the WL algorithm, there is no rigorous theory to support the convergence of $\widehat{g}_i$. In fact, some deficiencies of the WL algorithm have been observed in simulations. Yan and Pablo (2003) noticed that estimates of g_i can only reach a limited statistical accuracy which will not be improved with further iterations, and the large number of configurations generated towards the end of the simulation make only a small contribution to the estimates.

We find that this deficiency of the WL algorithm is caused by the choice of the gain factor δ. This can be explained as follows. Let n_s be the number of iterations performed in stage s and δ_s be the the gain factor used in stage s. Let $n_1 = \ldots = n_s = \ldots = n$, where n is large enough such that a flat histogram can be reached in each stage. Let $\log \delta_s = \frac{1}{2} \log \delta_{s-1}$ decreases geometrically as suggested by Wang and Landau (2001). Then the tail sum $n \sum_{s=S+1}^{\infty} \log \delta_s < \infty$ for any value of S. Note the tail sum represents the total correction to the current estimate in the following iterations. Hence, the large number of configurations generated towards the end of the simulation make only a small contribution to the estimates. To overcome this deficiency, Liang (2005) suggested that n_s should increase geometrically with the rate $\log \delta_{s+1} / \log \delta_s$. However, this leads to an explosion of the total number of iterations required by the simulation.

In this paper, we propose a stochastic approximation Monte Carlo (SAMC) algorithm. SAMC can be regarded as a stochastic approximation correction of the WL and GWL algorithms. In SAMC, the choice of the gain factor is guided by a condition given in the stochastic approximation algorithm (Andrieu, Moulines and Priouret, 2005), which ensures that the estimates of $\boldsymbol{g}$ can be improved continuously as the simulation goes on. It is shown that under mild conditions SAMC will converge. In addition, SAMC can bias sampling to some subregions of interest, say, the low energy region, according to a distribution defined on the subspace of the partition. This is different from WL, where each energy has to be sampled equally. This is also different from GWL, where the sampling frequencies of the subregions

follow a certain pattern determined by the parameter ϱ. Hesselbo and Stinchcombe (1995) and Liang (2005) showed numerically that biasing sampling to low energy regions often results in a simulation with improved ergodicity. This makes SAMC attractive for hard optimization problems. SAMC is user-friendly. It avoids the requirement of histogram checking during simulations. We discuss the potential use of SAMC in statistics through two classes of examples, importance sampling and model selection. It turns out that SAMC can work as a general importance sampling method and a model selection sampler when the model space is complex.

The remaining part of this article is organized as follows. In Section 2, we describe the SAMC algorithm and study its convergence theory. In Section 3, we compare WL and SAMC through a numerical example. In Section 4, we explore the use of SAMC in importance sampling. In Section 5, we discuss the use of SAMC in model selection. In Section 6, we conclude the paper with a brief discussion.

2 Stochastic Approximation Monte Carlo

Consider the distribution defined in (1). For reasons of mathematical convenience, we assume that $\mathcal{X}$ is either finite (for a discrete system) or compact (for a continuum system). For a continuum system, $\mathcal{X}$ can be restricted to the region $\{\boldsymbol{x} : p_0(\boldsymbol{x}) \geq p_{\min}\}$, where $p_{\min}$ is sufficiently small such that the region $\{\boldsymbol{x} : p_0(\boldsymbol{x}) < p_{\min}\}$ is not of interest. As in GWL, we let $E_1, \ldots, E_m$ denote m disjoint regions which form a partition of $\mathcal{X}$. In practice, $\sup_{\boldsymbol{x}\in\mathcal{X}} p_0(\boldsymbol{x})$ is often unknown. An inappropriate specification of u_i's may result in that some subregions are empty. A subregion E_i is empty if $g_i = \int_{E_i} \psi(\boldsymbol{x}) d\boldsymbol{x} = 0$. SAMC allows the existence of empty subregions in simulations. Let $\widehat{g}_i^{(t)}$ denote the estimate of g_i obtained at iteration t. For convenience, we let $\theta_{ti} = \log(\widehat{g}_i^{(t)})$ and $\theta_t = (\theta_{t1}, \ldots, \theta_{tm})$. The distribution (2) can then be rewritten as

$$p_{\theta_t}(\boldsymbol{x}) = \frac{1}{Z_t} \sum_{i=1}^{m} \frac{\psi(\boldsymbol{x})}{e^{\theta_{ti}}} I(\boldsymbol{x} \in E_i), \quad i = 1, \ldots, m. \tag{3}$$

For theoretical simplicity, we assume that $\theta_t \in \Theta$ for all t, where Θ is a compact set. In this article, we set $\Theta = [-10^{100}, 10^{100}]^m$ for all examples, although as a practical matter this is essentially equivalent to setting $\Theta = \mathbb{R}^m$. Since $p_{\theta_t}(\boldsymbol{x})$ is invariant with respect to a location transformation of θ_t; that is, adding to or subtracting from θ_t a constant vector will not change $p_{\theta_t}(\boldsymbol{x})$, θ_t can be kept in the compact set in simulations by adjusting with a constant

vector. Since $\mathcal{X}$ and Θ are both assumed to be compact, a follow-on assumption is that $p_{\theta_t}(\boldsymbol{x})$ is bounded away from 0 and ∞ on $\mathcal{X}$. Let the proposal distribution $q(\boldsymbol{x}, \boldsymbol{y})$ satisfy the following condition. For every $x \in \mathcal{X}$, there exist $\epsilon_1 > 0$ and $\epsilon_2 > 0$ such that

$$|\boldsymbol{x} - \boldsymbol{y}| \leq \epsilon_1 \Longrightarrow q(\boldsymbol{x}, \boldsymbol{y}) \geq \epsilon_2. \tag{4}$$

This is a natural condition in a study of MCMC theory (Roberts and Tweedie, 1996). In practice, this kind of proposal can be designed easily for both continuum and discrete systems. For a continuum system, $q(\boldsymbol{x}, \boldsymbol{y})$ can be set to the random walk Gaussian proposal $\boldsymbol{y} \sim N(\boldsymbol{x}, \sigma^2 I)$ with σ^2 being calibrated to have a desired acceptance rate. For a discrete system, $q(\boldsymbol{x}, \boldsymbol{y})$ can be set to a discrete distribution defined on a neighborhood of $\boldsymbol{x}$ by assuming that the states have been ordered in a certain way.

Let $\boldsymbol{\pi} = (\pi_1, \ldots, \pi_m)$ be an m-vector with $0 < \pi_i < 1$ and $\sum_{i=1}^m \pi_i = 1$, which defines the desired sampling frequency for each of the subregions. Henceforth, $\boldsymbol{\pi}$ will be called the desired sampling distribution. Let $\{\gamma_t\}$ be a positive, non-decreasing sequence satisfying

$$(i) \quad \sum_{t=1}^{\infty} \gamma_t = \infty, \qquad (ii) \quad \sum_{t=1}^{\infty} \gamma_t^{\zeta} < \infty, \tag{5}$$

for some $\zeta \in (1, 2)$. For example, in this paper we set

$$\gamma_t = \frac{t_0}{\max(t_0, t)}, \quad t = 1, 2, \ldots \tag{6}$$

for some specified value of $t_0 > 1$. With the above notation, one iteration of SAMC can be described as follows:

The SAMC algorithm:

(a) Simulate a sample $\boldsymbol{x}^{(t+1)}$ by a single MH update of which the proposal distribution is $q(\boldsymbol{x}^{(t)}, \cdot)$ and the invariant distribution is $p_{\theta_t}(\boldsymbol{x})$.

(b) Set $\theta^* = \theta_t + \gamma_{t+1}(\boldsymbol{e}_{t+1} - \boldsymbol{\pi})$, where $\boldsymbol{e}_{t+1} = (e_{t+1,1}, \ldots, e_{t+1,m})$ and $e_{t+1,i} = 1$ if $\boldsymbol{x}^{(t)} \in E_i$ and 0 otherwise. If $\theta^* \in \Theta$, set $\theta_{t+1} = \theta^*$; otherwise, set $\theta_{t+1} = \theta^* + \boldsymbol{c}^*$, where $\boldsymbol{c}^* = (c^*, \ldots, c^*)$ can be an arbitrary vector which satisfies the condition $\theta^* + \boldsymbol{c}^* \in \Theta$.

Remark: The explanation for the condition (5) can be found in advanced books on stochastic approximation, e.g., Nevel′son and Has′minskiĭ (1973). The first condition is necessary for the convergence of θ_t. If $\sum_{t=1}^{\infty} \gamma_t < \infty$, then as follows from step (b) below (assuming the adjustment of θ_t does not occur), $\sum_{t=1}^{\infty} |\theta_{t+1,i} - \theta_{ti}| \leq \sum_{t=1}^{\infty} \gamma_t |e_{ti} - \pi_i| \leq \sum_{t=1}^{\infty} \gamma_t < \infty$,

where the second inequality follows from the fact $0 \le e_{ti}, \pi_i \le 1$. Thus, the value of θ_{ti} does not reach $\log(g_i)$ if, for example, the initial point θ_{0i} is sufficiently far away from $\log(g_i)$. On the other hand, γ_t can not be too large. A large γ_t will prevent convergence. It turns out that the second condition in (5) asymptotically damps the effect of the random errors introduced by $\boldsymbol{e}_t$. When it holds, we have $\gamma_t|e_{ti} - \pi_i| \le \gamma_t \to 0$ as $t \to 0$.

SAMC falls into the category of stochastic approximation algorithms (Benveniste, Métivier and Priouret, 1990; Andrieu, Moulines and Priouret, 2005). Theoretical results on the convergence of SAMC are given in Appendix. The theory states that under mild conditions we have

$$\theta_{ti} \to \begin{cases} C + \log(\int_{E_i} \psi(\boldsymbol{x})d\boldsymbol{x}) - \log(\pi_i + d), & \text{if } E_i \neq \emptyset, \\ -\infty. & \text{if } E_i = \emptyset, \end{cases} \tag{7}$$

as $t \to \infty$, where $d = \sum_{j \in \{i:E_i=\emptyset\}} \pi_j/(m-m_0)$ and m_0 is the number of empty subregions, and C is an arbitrary constant. Since $p_{\theta_t}(\boldsymbol{x})$ is invariant with respect to a location transformation of θ_t, C can not be determined by the samples drawn from $p_{\theta_t}(\boldsymbol{x})$. To determine the value of C, extra information is needed, e.g., $\sum_{i=1}^m e^{\theta_{ti}}$ is equal to a known number. Let $\widehat{\pi}_{ti}$ denote the realized sampling frequency of the subregion E_i at iteration t. As $t \to \infty$, $\widehat{\pi}_{ti}$ converges to $\pi_i + d$ if $E_i \neq \emptyset$ and 0 otherwise. Note that for a non-empty subregion, its sampling frequency is independent of its probability $\int_{E_i} p(\boldsymbol{x})d\boldsymbol{x}$. This implies that SAMC is capable of exploring the whole sample space, even for the regions with tiny probabilities. Potentially, SAMC can be used to sample rare events from a large sample space. In practice, SAMC tends to lead to a "random walk" in the space of non-empty subregions (if each subregion is regarded as a "point") with the sampling frequency of each non-empty subregion being proportional to $\pi_i + d$.

The subject of stochastic approximation was founded by Robbins and Monro (1951). After five decades of continual development, it has developed into an important area in systems control and optimization, and it has also served as a prototype for the development of recursive algorithms for on-line estimation and control of stochastic systems. Refer to Lai (2003) for an overview on the subject. Recently, it has been used with Markov chain Monte Carlo for solving maximum likelihood estimation problems (Younes, 1988, 1999; Moyeed and Baddeley, 1991; Gu and Kong, 1998; Gelfand and Banerjee, 1998; Delyon, Lavielle and Moulines, 1999; Gu and Zhu, 2001). The critical difference between SAMC and other stochastic approximation MCMC algorithms is at sample space partitioning. With our use

of partitioning, many new applications can be established in Monte Carlo computation, for example, importance sampling and model selection as described in Sections 4 and 5. In the same spirit, SAMC can also be applied to highest posterior density (HPD) interval construction, normalizing constant estimation, and other problems as discussed in Liang (2005). In addition, sample space partitioning improves its performance in optimization. Control of the sampling frequency effectively prevents the system from getting trapped into local energy minima in simulations. This issue will be further explored elsewhere. It is notable that Geyer and Thompson (1995) and Geyer (1996) mentioned that stochastic approximation can be used to determine the "pseudo-priors" for simulated tempering (i.e., determining the normalizing constants of a sequence of distributions scaled by temperature), although no details were provided. In Geyer's applications, the sample space is partitioned automatically according to the temperature variable.

For an effective implementation of SAMC, several issues need to be considered.

• *Sample space partition.* This can be done according to one's goal and the complexity of the given problem. For example, if we aim to construct a trial density function for importance sampling (as illustrated in Section 4) or minimizing the energy function, the sample space can be partitioned according to the energy function. The maximum energy difference in each subregion should be bounded by a reasonable number, say, 2, which ensures that the local MH moves within the same subregion have a reasonable acceptance rate. Note that within the same subregion, sampling from the working density (3) is reduced to sampling from $\psi(\boldsymbol{x})$. If we aim at model selection, the sample space can be partitioned according to the index of models, as illustrated in Section 5.

• *The desired sampling distribution.* If one aims at estimating $\boldsymbol{g}$, one may set the desired distribution to be uniform, as is done in all examples of this paper. However, if one aims at optimization, one may set the desired distribution biased to low energy regions. As shown by Hesselbo and Stinchcombe (1995) and Liang (2005), biasing sampling to low energy regions often improves the ergodicity of the simulation. Our numerical results on BLN protein models (Honeycutt and Thirumalai, 1990) also strongly support this point. Due to space limitations, these results will be reported elsewhere.

• *The choice of t_0 and the number of iterations.* To estimate $\boldsymbol{g}$, γ_t should be very close to 0 at the end of simulations. Otherwise, the resulting estimates will have a large variation. The speed of γ_t going to zero can be controlled by t_0. In practice, t_0 can be chosen according to the complexity of the problem. The more complex the problem is, the larger one should

choose the value of t_0. A large t_0 will force the sampler to reach all subregions quickly, even in the presence of multiple local energy minima. In our experience, t_0 is often set to be a value between $2m$ and $100m$ with m being the number of subregions.

The appropriateness of the choice of t_0 and the number of iterations can be diagnosed by checking the convergence of multiple runs (starting with different points) through an examination for the variation of $\widehat{\boldsymbol{g}}$ or $\widehat{\boldsymbol{\pi}}$, where $\widehat{\boldsymbol{g}}$ and $\widehat{\boldsymbol{\pi}}$ denote, respectively, the estimates of $\boldsymbol{g}$ and $\boldsymbol{\pi}$ obtained at the end of a run. A rough examination for $\widehat{\boldsymbol{g}}$ is to see visually whether the $\widehat{\boldsymbol{g}}$ vectors produced in the multiple runs follow the same pattern or not. Existence of different patterns implies that the gain factor is still large at the end of the runs or some parts of the sample space are not visited in all runs. The examination for $\widehat{\boldsymbol{g}}$ can also be done by a statistical test under the assumption of multivariate normality. Refer to Jobson (1992) (page 150-153) for the testing methods for multivariate outliers.

To examine the variation of $\widehat{\boldsymbol{\pi}}$, we define the statistic $\epsilon_f(E_i)$, which measures the deviation of $\widehat{\pi}_i$, the realized sampling frequency of subregion E_i in a run, from its theoretical value. The statistic is defined as

$$\epsilon_f(E_i) = \begin{cases} \frac{\widehat{\pi}_i - (\pi_i + \widehat{d})}{\pi_i + \widehat{d}} \times 100\%, & \text{if } E_i \text{ is visited,} \\ 0, & \text{otherwise,} \end{cases} \tag{8}$$

for $i = 1, \ldots, m$, where $\widehat{d} = \sum_{j \in \{i: E_i \text{ is not visited}\}} \pi_j/(m - m_0')$ and m_0' is the number of subregions which are not visited. Note that $\widehat{d}$ can be regarded as an estimate of d in (7). It is said $\{\epsilon_f(E_i)\}$, output from all runs and for all subregions, matches well if the following two conditions are satisfied: (i) there does not exist such a subregion which is visited in some runs but not in others, and (ii) $\max_{i=1}^{m} |\epsilon_f(E_i)|$ is less than a threshold value, say, 10%, for all runs. A group of $\{\epsilon_f(E_i)\}$ which does not match well implies that some parts of the sample space are not visited in all runs, t_0 is too small (the self-adjusting ability is thus weak), or the number of iterations is too small. We note that the idea of monitoring convergence of MCMC simulations using multiple runs was discussed in Gelman and Rubin (1992) and Geyer (1992).

In practice, to have a reliable diagnostic for the convergence, we may check both $\widehat{\boldsymbol{g}}$ and $\widehat{\boldsymbol{\pi}}$. In the case that a failure of multiple-run convergence is detected, SAMC should be re-run with more iterations or a larger value of t_0. The process of determining t_0 and the number of iterations is a trial and error process.

x	1	2	3	4	5	6	7	8	9	10
$P(x)$	1	100	2	1	3	3	1	200	2	1

Table 1: The unnormalized mass function of the 10-state distribution.

3 Two Demonstration Examples

Example 1 In this example, we compared the convergence and efficiency of WL and SAMC. The distribution of the example consists of 10 states with the unnormalized mass function $P(x)$ as given in Table 1. It has two modes which are well separated by low mass states.

The sample space was partitioned according to the mass function into the following five subregions: $E_1 = \{8\}$, $E_2 = \{2\}$, $E_3 = \{5, 6\}$, $E_4 = \{3, 9\}$ and $E_5 = \{1, 4, 7, 10\}$. In simulations, we set $\psi(x) = 1$. The true value of $\boldsymbol{g}$ is then $\boldsymbol{g} = (1, 1, 2, 2, 4)$, which is the number of states in the respective subregions. The proposal used in the MH step is a stochastic matrix of which each row is generated independently from the Dirichlet distribution $Dir(1, \ldots, 1)$. The desired sampling distribution is uniform, i.e., $\pi_1 = \ldots = \pi_5 = 1/5$. The sequence $\{\gamma_t\}$ is as given in (6) with $t_0 = 10$. SAMC was run for 100 times independently. Each run consists of 5×10^5 iterations. The estimation error of $\boldsymbol{g}$ was measured by the function $\epsilon_e(t) = \sqrt{\sum_{E_i \neq \emptyset} (\widehat{g}_i^{(t)} - g_i)^2 / g_i}$ at 10 equally spaced time points $t = 5 \times 10^4, \ldots, 5 \times 10^5$. Figure 1(a) shows the curve of $\epsilon_e(t)$ obtained by averaging over the 100 runs. The statistic $\epsilon_f(E_i)$ was calculated at time $t = 10^5$ for each run. The results show that they match well. Figure 1(b) shows the box-plots of $\epsilon_f(E_i)$'s of the 100 runs. The deviations are all less than three percent. This indicates that SAMC has achieved the desired sampling distribution and the choice of t_0 and the number of iterations are appropriate. Other choices of t_0 were also tried, say, $t_0 = 20$ and 30. The results are similar.

The WL algorithm was applied to this example with the same proposal distribution as that used in SAMC. In the runs, the gain factor was set as in Wang and Landau (2001). It starts with $\delta_0 = 2.718$ and then decreases in the scheme $\delta_{s+1} \to \sqrt{\delta_s}$. Let n_s denote the number of iterations performed in stage s. For simplicity, we set n_s to a constant which has been large enough such that a flat histogram can be formed in each stage. The choices of n_s we tried include $n_s = 1000$, 2500, 5000 and 10000. The estimation error was also measured by $\epsilon_e(t)$ evaluated at $t = 5 \times 10^4, \ldots, 5 \times 10^5$, where t is the total number of iterations made

so far in the run. Figure 1(a) shows the curves of $\epsilon_e(t)$ for each choice of n_s, where each curve was obtained by averaging over 100 independent runs.

The comparison shows that for this example SAMC produces more accurate estimates for g and converges much faster than WL. More importantly, in SAMC the estimates can be improved continuously as the simulation goes on, while in WL the estimates can only reach a certain accuracy depending on the value of n_s.

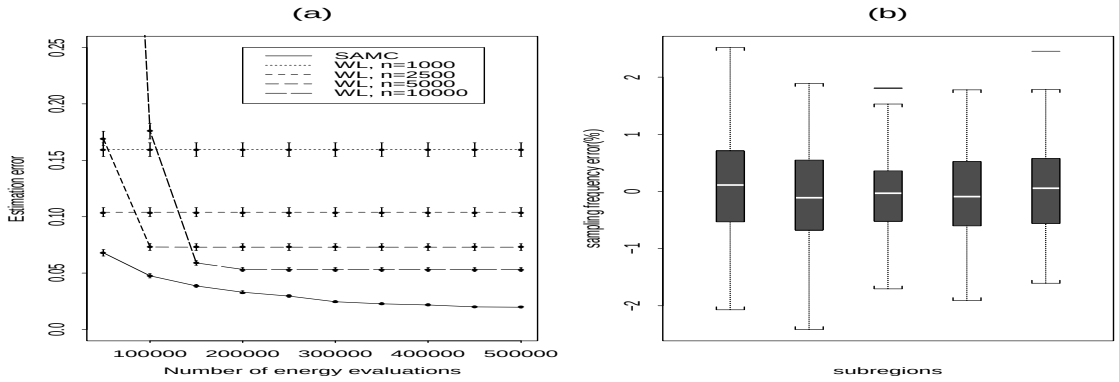

Figure 1: Comparison of the WL and SAMC algorithms. (a) Average $\epsilon_e(t)$ curves obtained by SAMC and WL. The vertical bars show the ±one-standard-deviation of the average of the estimates. (b) Box-plots of $\{\epsilon_f(E_i)\}$ obtained in 100 runs of SAMC.

Example 2 As pointed out by Liu (2001), umbrella sampling (Torrie and Valleau, 1977) can be seen as a precursor of many advanced Monte Carlo algorithms, including simulated tempering, multicanonical and thus WL, GWL and SAMC. Although umbrella sampling was proposed originally for estimating the ratio of two normalizing constants, it can also be used as a general importance sampling method. Recall that the basic idea of umbrella sampling is to sample from an "umbrella distribution" (trial distributions in terms of importance sampling) which covers the important regions of both target distributions. Torrie and Valleau (1977) proposed two possible schemes for construction of umbrella distributions. One is to sample intermediate systems of the temperature-scaling form $p_{st}^{(i)}(\boldsymbol{x}) \propto [p_0(\boldsymbol{x})]^{1/T_i}$ for $T_m > T_{m-1} > \cdots > T_1 = 1$. This leads to directly the simulated tempering algorithm. The other one is to sample a weighted distribution $p_u(\boldsymbol{x}) \propto \omega\{U(\boldsymbol{x})\}p_0(\boldsymbol{x})$, where the weight function $\omega(\cdot)$ is a function of the energy variable and can be determined by a pilot study. Thus, umbrella sampling can be seen as a precursor of multicanonical, WL, GWL

and SAMC. Sample space partitioning, which is motivated by discretization of continuum systems, provides a new methodology for applications of umbrella sampling to continuum systems.

Although SAMC and simulated tempering both fall into the class of umbrella sampling algorithms, they have quite different dynamics. This can be illustrated by the following example. The distribution is defined as $p(\boldsymbol{x}) \propto e^{-U(\boldsymbol{x})}$, where $\boldsymbol{x} \in [-1.1, 1.1]^2$ and $U(\boldsymbol{x}) = -\{x_1 \sin(20x_2) + x_2 \sin(20x_1)\}^2 \cosh\{\sin(10x_1)x_1\} - \{x_1 \cos(10x_2) - x_2 \sin(10x_1)\}^2 \cosh\{\cos(20x_2)x_2\}$. This example is modified from Example 5.3 of Robert and Casella (2004). Figure 2 (a) shows that $U(\boldsymbol{x})$ has a multitude of local energy minima separated by high-energy barriers. In applying SAMC to this example, we partitioned the sample space into 41 subregions with an equal energy bandwidth: $E_1 = \{\boldsymbol{x} : U(\boldsymbol{x}) \leq -8.0\}$, $E_2 = \{\boldsymbol{x} : -8.0 < U(\boldsymbol{x}) \leq -7.8\}$, ..., and $E_{41} = \{\boldsymbol{x} : -0.2 < U(\boldsymbol{x}) \leq 0\}$, and set other parameters as follows, $\psi(\boldsymbol{x}) = e^{-U(\boldsymbol{x})}$, $t_0 = 200$, $\pi_1 = \cdots = \pi_{41} = 1/41$, and a random walk proposal $q(\boldsymbol{x}_t, \cdot) = N_2(\boldsymbol{x}_t, 0.25^2 I_2)$. SAMC was run for 20000 iterations, and 2000 samples were collected at equally spaced time points. Figure 2 (b) shows the evolving path of the 2000 samples. For comparison, MH was applied to simulate from the distribution $p_{st}(\boldsymbol{x}) \propto e^{-U(\boldsymbol{x})/5}$. MH was run for 20000 iterations with the same proposal $N_2(\boldsymbol{x}_t, 0.25^2 I_2)$, and 2000 samples were collected at equally spaced time points. Figure 2 (c) shows the evolving path of the 2000 samples, which characterizes the performance of simulated tempering at high temperatures.

The result is clear: under the above setting, SAMC almost samples uniformly in the space of energy (the energy bandwidth of each subregion is small, and the sample distribution matches with the contour plot of $U(\boldsymbol{x})$ very well), while simulated tempering tends to sample uniformly in the sample space $\mathcal{X}$ when the temperature is high. As we usually do not know where the high and low energy regions are and how much the ratio of their "volumes" is *a priori*, we can not control the simulation time spent on low and high energy regions in simulated tempering. However, we can control almost exactly, up to the constant d in (7), the simulation time spent on low and high energy regions in SAMC by choosing the desired sampling distribution $\boldsymbol{\pi}$. SAMC can go to high energy regions, but it only spends limited time over there to help the system to escape from local energy minima and spends other time in exploring low energy regions. This smart simulation time distribution scheme makes SAMC potentially more efficient than simulated tempering in optimization. Due to the space limitations, this point is not explored in this paper. But we note that Liang (2005) reported

a neural network training example where it was shown GWL is more efficient than simulated tempering in locating global energy minima.

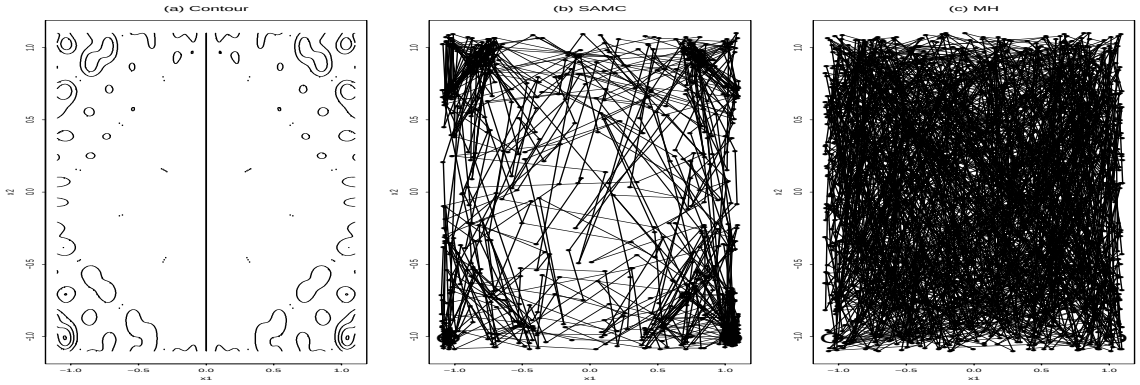

Figure 2: (a) Contour of $U(\boldsymbol{x})$. (b) Sample path of SAMC. (c) Sample path of MH.

4 Use of SAMC in Importance Sampling

In this section, we illustrate the use of SAMC as an importance sampling method. Suppose that due to its rugged energy landscape, the target distribution $p(\boldsymbol{x})$ is very difficult to simulate from with conventional Monte Carlo algorithms. Let $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ denote the samples drawn from a trial density $p^*(\boldsymbol{x})$, and let $w_1, \ldots, w_n$ denote the associated importance weights, where $w_i = p(\boldsymbol{x}_i)/p^*(\boldsymbol{x}_i)$ for $i = 1, \ldots, n$. The quantity $E_p h(\boldsymbol{x})$ can then be estimated by

$$\widehat{E_p h(\boldsymbol{x})} = \frac{\sum_{i=1}^n h(\boldsymbol{x}_i) w_i}{\sum_{i=1}^n w_i}. \tag{9}$$

Although this estimate converges almost surely to $E_p h(\boldsymbol{x})$, its variance is finite only if

$$E_{p^*} h^2(\boldsymbol{x}) \big(\frac{p(\boldsymbol{x})}{p^*(\boldsymbol{x})}\big)^2 d\boldsymbol{x} = \int_{\mathcal{X}} h^2(\boldsymbol{x}) \frac{p^2(\boldsymbol{x})}{p^*(\boldsymbol{x})} d\boldsymbol{x} < \infty.$$

If the ratio $p(\boldsymbol{x})/p^*(\boldsymbol{x})$ is unbounded, the weight $p(\boldsymbol{x}_i)/p^*(\boldsymbol{x}_i)$ will vary widely, and the resulting estimate will be unreliable. A good trial density should necessarily satisfy the following two conditions:

(a) The importance weight is bounded, that is, there exists a number M such that $p(\boldsymbol{x})/p^*(\boldsymbol{x}) < M$ for all $\boldsymbol{x} \in \mathcal{X}$.

(b) The trial density $p^*(\boldsymbol{x})$ can be easily simulated from using conventional Monte Carlo algorithms.

In addition, the trial density should be chosen to have a similar shape to the true density. This will minimize the variance of the resulting importance weights. How to specify an appropriate trial density for a general distribution has, of course, been a long standing and difficult problem in statistics.

The defensive mixture method (Hesterberg, 1995) suggests the following trial density

$$p^*(\boldsymbol{x}) = \lambda p(\boldsymbol{x}) + (1-\lambda)\widetilde{p}(\boldsymbol{x}), \tag{10}$$

where $0 < \lambda < 1$ and $\widetilde{p}(\boldsymbol{x})$ is another density. However, in practice, $p^*(\boldsymbol{x})$ is rather poor. Although the resulting importance weights are bounded above by $1/\lambda$, it can not be easily sampled from using conventional Monte Carlo algorithms. Since $p^*(\boldsymbol{x})$ contains $p(\boldsymbol{x})$ as a component, if we can sample from $p^*(\boldsymbol{x})$, we can also sample from $p(\boldsymbol{x})$. In this case, we do not need to use importance sampling! Stavropoulos and Titterington (2001), Warnes (2001), and Cappé et al (2004) suggested to construct the trial density based on previous Monte Carlo samples, but the trial densities resulting from their methods can not guarantee that the importance weights are bounded. We note that these methods are similar to SAMC in the spirit of learning from historical samples. Other trial densities based on simple mixtures of normals or t-distributions may also result in unbounded importance weights, although they can be sampled from easily.

Suppose that the sample space has been partitioned according to the energy function, and the maximum energy difference in each subregion has been bounded by a reasonable number such that the local MH move within the same subregion has a reasonable acceptance rate. It is then easy to see that the distribution defined in (2) or (3) satisfies the above two conditions and can work as a universal trial density even in the presence of multiple local minima on the energy landscape of the true density. Let

$$\widehat{p}_\infty(\boldsymbol{x}) \propto \sum_{i=1}^{m} \frac{\psi(\boldsymbol{x})}{\widehat{g}_i} I(\boldsymbol{x} \in E_i), \tag{11}$$

denote the trial density constructed by SAMC with $\psi(\boldsymbol{x}) = p_0(\boldsymbol{x})$, where $\widehat{g}_i = \lim_{t\to\infty} e^{\theta_{ti}}$. Assuming that $\widehat{g}_i$ has been normalized by an additional constraint, e.g., $\sum_{i=1}^m \widehat{g}_i$ is a known constant, the importance weights are then bounded above by $\max_{i=1}^m \widehat{g}_i < \int_{\mathcal{X}} \psi(\boldsymbol{x}) d\boldsymbol{x} < \infty$. As shown in Section 2, sampling from $\widehat{p}_\infty(\boldsymbol{x})$ will lead to a "random walk" in the space of non-empty subregions. Hence, the whole sample space can be well explored.

In addition to satisfying the conditions (a) and (b), $\widehat{p}_\infty(\boldsymbol{x})$ has two additional advantages over other trial densities. First, the similarity of the trial density to the target density can be controlled to some extent by the user. For example, instead of (11) we can sample from the following density,

$$\widehat{p}_\infty(\boldsymbol{x}) \propto \sum_{i=1}^{m} \frac{\psi(\boldsymbol{x})}{\lambda_i \widehat{g}_i} I(\boldsymbol{x} \in E_i), \tag{12}$$

where the parameters $\lambda_i, i = 1, \ldots, m$, control the sampling frequency of the subregions. Second, resampling can be made on-line if we are interested in generating equally weighted samples from $p(\boldsymbol{x})$. Let $\omega_i = \widehat{g}_i / \max_{j=1}^{m} \widehat{g}_j$ denote the resampling probability from the subregion E_i, and $\boldsymbol{x}_t$ denote the sample drawn from $\widehat{p}_\infty(\boldsymbol{x})$ at iteration t. The resampling procedure consists of the following three steps.

The SAMC-importance-resampling algorithm

(a) Draw a sample $\boldsymbol{x}_t \sim \widehat{p}_\infty(\boldsymbol{x})$ using a conventional Monte Carlo algorithm, say, the MH algorithm.

(b) Draw a random number $U \sim \text{Uniform}(0, 1)$. If $U < \omega_k$, save $\boldsymbol{x}_t$ as a sample of $p(\boldsymbol{x})$, where k is the index of the subregion $\boldsymbol{x}_t$ belongs to.

(c) Set $t \leftarrow t + 1$ and go to step (a), until enough samples have been collected.

Consider the following distribution,

$$p(\boldsymbol{x}) = \frac{1}{3} N \left[\begin{pmatrix} -8 \\ -8 \end{pmatrix}, \begin{pmatrix} 1 & 0.9 \\ 0.9 & 1 \end{pmatrix} \right] + \frac{1}{3} N \left[\begin{pmatrix} 6 \\ 6 \end{pmatrix}, \begin{pmatrix} 1 & -0.9 \\ -0.9 & 1 \end{pmatrix} \right] + \frac{1}{3} N \left[\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \right],$$

which is identical to the one given in Gilks, Roberts and Sahu (1998), except that the mean vectors are separated by a larger distance in each dimension. Figure 3 (a) shows its contour plot, which contains three well separated components. The MH algorithm has been tried to simulate from $p(\boldsymbol{x})$ with a random walk proposal $N(\boldsymbol{x}, I_2)$, but it failed to mix the three components. However, an advanced MCMC sampler, such as simulated tempering, parallel tempering and evolutionary Monte Carlo, should work well for this example. The purpose we study this example is just to illustrate how SAMC can be used in importance sampling as a universal trial distribution constructor and how SAMC can be used as an advanced sampler to sample from a multi-modal distribution.

SAMC was applied to this example with the same proposal as that used in the MH algorithm. Let $\mathcal{X} = [-10^{100}, 10^{100}]^2$ be compact. It was partitioned with an equal energy

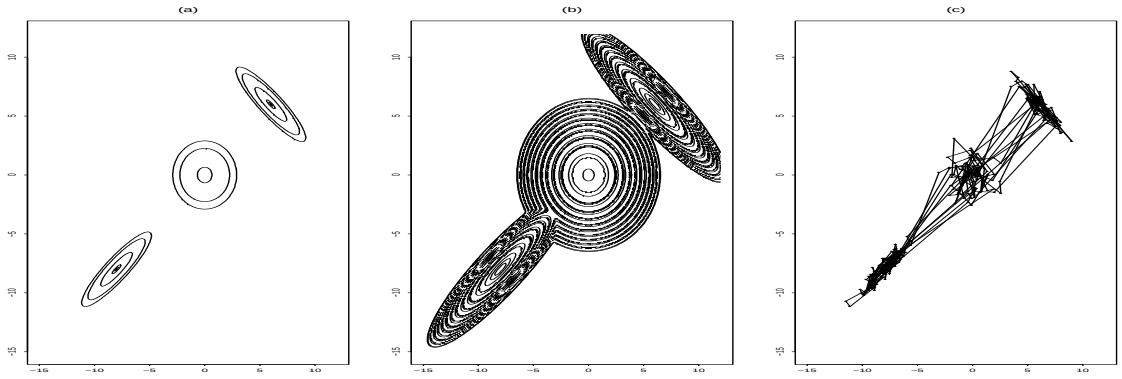

Figure 3: Computational results for the mixture Gaussian example. Plots (a) and (b) are the contour plots of the true and trial densities, respectively. The contour lines correspond to 99%, 95%, 50%, 5% and 1% of the total mass. Plot (c) shows the path of the first 500 samples simulated from $p(\boldsymbol{x})$ by the SAMC-importance-sampling algorithm.

bandwidth $\Delta u = 2$ into the following subregions, $E_1 = \{\boldsymbol{x} : -\log p(\boldsymbol{x}) < 0\}$, $E_2 = \{\boldsymbol{x} : 0 \le -\log p(\boldsymbol{x}) < 2\}$, ..., and $E_{12} = \{\boldsymbol{x} : -\log p(\boldsymbol{x}) > 20\}$. Set $\psi(\boldsymbol{x}) = p(\boldsymbol{x})$, $t_0 = 50$ and the desired sampling distribution to be uniform. In a run of 500,000 iterations, SAMC produced a trial density with the contour plot as shown in Figure 3(b). On the plot there are many contour circles formed due to the density adjustment by $\widehat{g}_i$'s. The adjustment means that many points of the sample space have the same density value. The SAMC-importance-sampling algorithm was then applied to simulate samples from $p(\boldsymbol{x})$. Figure 3(c) shows the sample path of the first 500 samples generated by the algorithm. All three components have been well mixed. Later, the run was lengthened, and the mean and variance of the distribution were estimated accurately using the simulated samples. The results indicate that SAMC can indeed work as a general trial distribution constructor for importance sampling and an advanced sampler for simulation from a multi-modal distribution.

5 Use of SAMC in Model Selection Problems

5.1 Algorithms

Suppose that we have a posterior distribution denoted by $f(M, \vartheta_M|D)$, where D denotes the data, M is the index of models, and ϑ_M is the vector of parameters associated with

model M. Without loss of generality, we assume that only a finite number, m, of models are under consideration, and the models are subject to a uniform prior. The sample space of $f(M, \vartheta_M|D)$ can be written as $\bigcup_{i=1}^{m} \mathcal{X}_{M_i}$, where $\mathcal{X}_{M_i}$ denotes the sample space of $f(\vartheta_{M_i}|M_i, D)$. If we let $E_i = \mathcal{X}_{M_i}$ for $i = 1, \ldots, m$, and $\psi(\cdot) \propto f(M, \vartheta_M|D)$, it follows from (7) that $\widehat{g}_i^{(t)}/\widehat{g}_j^{(t)} = e^{\theta_{ti}-\theta_{tj}}$ forms a consistent estimator for the Bayes factor of the models M_i and M_j, $1 \le i, j \le m$. We note that reversible jump MCMC (RJMCMC) (Green, 1995) can also estimate the Bayes factors of m models simultaneously. For comparison, in the following we give explicitly the iterative procedures of the two methods for Bayesian model selection.

Let $Q(M_i \to M_j)$ denote the proposal probability for a transition from model M_i to model M_j, and $T(\vartheta_{M_i} \to \vartheta_{M_j})$ denote the proposal distribution of generating ϑ_{M_j} conditional on ϑ_{M_i}. Assume that both Q and T satisfy the condition (4). Let $M^{(t)}$ and $\vartheta^{(t)}$ denote the model and the model parameters sampled at iteration t, respectively. One iteration of SAMC consists of the following steps.

The SAMC-model-selection algorithm:

(a) Generate model M^* according to the proposal matrix Q.

(b) If $M^* = M^{(t)}$, simulate a sample ϑ^* from $f(\vartheta_{M^{(t)}}|M^{(t)}, D)$ by a single MCMC iteration and set $(M^{(t+1)}, \vartheta^{(t+1)}) = (M^*, \vartheta^*)$.

(c) If $M^* \neq M^{(t)}$, generate ϑ^* according to the proposal distribution T and accept the sample (M^*, ϑ^*) with probability

$$\min\{1, \frac{e^{\theta_{t,M^{(t)}}}}{e^{\theta_{t,M^*}}} \frac{f(M^*, \vartheta^*|D)}{f(M^{(t)}, \vartheta^{(t)}|D)} \frac{Q(M^* \to M^{(t)})}{Q(M^{(t)} \to M^*)} \frac{T(\vartheta^* \to \vartheta^{(t)})}{T(\vartheta^{(t)} \to \vartheta^*)}\}. \tag{13}$$

If it is accepted, set $(M^{(t+1)}, \vartheta^{(t+1)}) = (M^*, \vartheta^*)$; otherwise, set $(M^{(t+1)}, \vartheta^{(t+1)}) = (M^{(t)}, \vartheta^{(t)})$.

(d) Set $\theta^* = \theta_t + \gamma_{t+1}(\boldsymbol{e}_{t+1} - \boldsymbol{\pi})$, where $\boldsymbol{e}_{t+1} = (e_{t+1,1}, \ldots, e_{t+1,m})$, and $e_{t+1,i} = 1$ if $M^{(t+1)} = M_i$ and 0 otherwise. If $\theta^* \in \Theta$, set $\theta_{t+1} = \theta^*$; otherwise, set $\theta_{t+1} = \theta^* + \boldsymbol{c}^*$, where $\boldsymbol{c}^*$ is chosen such that $\theta^* + \boldsymbol{c}^* \in \Theta$.

Let $\Xi_i^{(t)} = \#\{M^{(k)} = M_i : k = 1, 2, \ldots t\}$ be the sampling frequency of model M_i during the first t iterations in a run of RJMCMC. With the same proposal matrix Q, the same proposal distribution T and the same MH step (or Gibbs cycle) as those used by SAMC, one iteration of RJMCMC can be described as follows.

The RJMCMC algorithm:

(a) Generate model M^* according to the proposal matrix Q.

(b) If $M^* = M^{(t)}$, simulate a sample ϑ^* from $f(\vartheta_{M^{(t)}}|M^{(t)}, D)$ by a single MCMC iteration and set $(M^{(t+1)}, \vartheta^{(t+1)}) = (M^*, \vartheta^*)$.

(c) If $M^* \neq M^{(t)}$, generating ϑ^* according to the proposal density T and accept the sample (M^*, ϑ^*) with probability

$$\min\{1, \frac{f(M^*, \vartheta^*|D)}{f(M^{(t)}, \vartheta^{(t)}|D)} \frac{Q(M^* \to M^{(t)})}{Q(M^{(t)} \to M^*)} \frac{T(\vartheta^* \to \vartheta^{(t)})}{T(\vartheta^{(t)} \to \vartheta^*)}\}. \tag{14}$$

If it is accepted, set $(M^{(t+1)}, \vartheta^{(t+1)}) = (M^*, \vartheta^*)$; otherwise, set $(M^{(t+1)}, \vartheta^{(t+1)}) = (M^{(t)}, \vartheta^{(t)})$.

(d) Set $\Xi_i^{(t+1)} = \Xi_i^{(t)} + I(M^{(t+1)} = M_i)$ for $i = 1, \ldots, m$.

The standard MCMC theory (Tierney, 1994) implies that as $t \to \infty$, $\Xi_i^{(t)}/\Xi_j^{(t)}$ forms a consistent estimator for the Bayes factor of model M_i and model M_j.

We note that the form of the RJMCMC algorithm described above is not the most general one, where the proposal distribution $T(\cdot \to \cdot)$ is assumed such that the Jacobian term in (14) is reduced to 1. This note is also applicable to the SAMC-model-selection algorithm. The MCMC algorithm employed in step (b) of the above two algorithms can be the MH algorithm, the Gibbs sampler (Geman and Geman, 1984) or any other advanced MCMC algorithms, such as simulated tempering, parallel tempering, evolutionary Monte Carlo and SAMC-importance-resampling (discussed in Section 3). When the distribution $f(M, \vartheta_M|D)$ is complex, an advanced MCMC algorithm may be chosen and multiple iterations may be used in this step.

SAMC and RJMCMC are only different at steps (c) and (d), i.e., the ways of acceptance for a new sample and estimation for the model probabilities. In SAMC, a new sample is accepted with an adjusted probability. The adjustment always works in the reverse direction of the estimation error of the model probability or, equivalently, the frequency discrepancy between the realized sampling frequency and the desired one. Thus, it guarantees the convergence of the algorithm. In simulations, we can see that SAMC can overcome any difficulties in dimension-jumping moves and have a full exploration for all models. Recall that the proposal distributions have been assumed to satisfy the condition (4). Since RJMCMC does not possess the self-adjusting ability, it samples each model in a frequency proportional to its probability. In simulations, we can see that RJMCMC often stays on a model for a long

time if that model has a significantly higher probability than its neighboring models. In SAMC, the estimates of the model probabilities are updated in the logarithmic scale; this makes SAMC potentially work for a group of models with huge differences in probability. This is beyond the ability of RJMCMC, which can only work for a group of models with comparable probabilities.

At last, we point out that for a problem which contains only several models with comparable probabilities, SAMC may not be better than RJMCMC, as in this case its self-adjusting ability is no longer crucial for mixing of the models. SAMC is essentially an importance sampling method (the samples are not equally weighted), hence, its efficiency should be lower than RJMCMC for a problem that RJMCMC succeeds. In summary, we suggest to use SAMC when the model space is complex, for example, when the distribution $f(M|D)$ has well separated multiple modes, or when there are tiny probability models, but, of interest to us.

5.2 Numerical Results

The autologistic model (Besag, 1974) has been widely used for spatial data analysis, see, e.g., Preisler (1993) or Augustin *et al* (1996). Let $\boldsymbol{s} = \{s_i : i \in D\}$ denote a configuration of the model, where the binary response $s_i \in \{-1, +1\}$ 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 $N(i)$ denote a set of the neighbors of spin i. The probability mass function of the model is

$$p(\boldsymbol{s}|\alpha, \beta) = \frac{1}{\varphi(\alpha, \beta)} \exp\{\alpha \sum_{i \in D} s_i + \frac{\beta}{2} \sum_{i \in D} s_i (\sum_{j \in N(i)} s_j)\}, \quad (\alpha, \beta) \in \Omega, \tag{15}$$

where Ω is the parameter space, and $\varphi(\alpha, \beta)$ is the normalizing constant defined by

$$\varphi(\alpha, \beta) = \sum_{\text{for all possible } \boldsymbol{s}} \exp\{\alpha \sum_{i \in D} s_i + \frac{\beta}{2} \sum_{i \in D} s_i (\sum_{j \in N(i)} s_j)\}.$$

The parameter α determines the overall proportion of s_i with a value of $+1$, and the parameter β determines the intensity of the interaction between s_i and its neighbors.

A major difficulty with this model is that the function $\varphi(\alpha, \beta)$ is generally unknown analytically. Evaluating $\varphi(\alpha, \beta)$ exactly is prohibitive even for a moderate system, since it requires summary over all $2^{|D|}$ possible realizations of $\boldsymbol{s}$. Since $\varphi(\alpha, \beta)$ is unknown, importance sampling is perhaps the most convenient technique if we aim at calculating the expectation $E_{\alpha,\beta} h(\boldsymbol{s})$ over the parameter space. This problem is a little bit different from

conventional importance sampling problems discussed in Section 4, where we have only one target distribution, whereas here we have multiple target distributions indexed by their parameter values. A natural choice for the trial distribution is a mixture distribution of the form

$$p^*_{mix}(\boldsymbol{s}) = \frac{1}{m^*}\sum_{j=1}^{m^*} p(\boldsymbol{s}|\alpha_j, \beta_j), \tag{16}$$

where the values of the parameters $(\alpha_1, \beta_1), \ldots, (\alpha_{m^*}, \beta_{m^*})$ are pre-specified. We note that this idea has been suggested by Geyer (1996). To complete this idea, the key is to estimate $\varphi(\alpha_j, \beta_j), \ldots, \varphi(\alpha_{m^*}, \beta_{m^*})$. The estimation can be up to a common multiplicative constant, which will be canceled out in calculation of $E_{\alpha,\beta}h(\boldsymbol{s})$ in (9). Geyer (1996) also suggested stochastic approximation as a feasible method for simultaneously estimating $\varphi(\alpha_1, \beta_1), \ldots, \varphi(\alpha_{m^*}, \beta_{m^*})$, but gave no details. Several authors have based their inferences for a distribution like (15) on the estimates of the normalizing constant function at a finite number of points. For example, Diggle and Gratton (1984) proposed estimating the normalizing constant function on a grid, smoothing the estimates using a kernel method, and then substituting the smooth estimates into (15) as known for finding MLEs of the parameters. A similar idea has also been proposed by Green and Richardson (2002) in analyzing a disease mapping example.

In this paper, we explore the idea of Geyer (1996) and give details about how SAMC can be used to simultaneously estimate $\varphi(\alpha_1, \beta_1), \ldots, \varphi(\alpha_{m^*}, \beta_{m^*})$ and how the estimates can be further used in estimation of the model parameters. The dataset considered is the U.S. cancer mortality rate as shown in Figure 4(a). Following Sherman, Apamasovich and Carroll (2006), we modeled the data by a spatial autologistic model. The total number of spins is $|D| = 2293$. Suppose that the parameter points used in (16) form a 21×11 lattice ($m^* = 231$) with α equally spaced between -0.5 and 0.5 and β between 0 and 0.5. Since $\varphi(\alpha, \beta)$ is a symmetric function about α, we only need to estimate it on a sublattice with α between 0 and 0.5. The sublattice consists of $m = 121$ points. Estimating the quantities $\varphi(\alpha_1, \beta_1), \ldots, \varphi(\alpha_m, \beta_m)$ can be treated as a Bayesian model selection problem, although no observed data are involved. This is because $\varphi(\alpha_1, \beta_1), \ldots, \varphi(\alpha_m, \beta_m)$ correspond to the normalizing constants of different distributions. In the following, the SAMC-model-selection algorithm and the RJMCMC algorithm were applied to this problem by treating each grid point (α_j, β_j) as a different model and $p(\boldsymbol{s}|\alpha_j, \beta_j)\varphi(\alpha_j, \beta_j)$ as the posterior distribution used in (13) and (14).

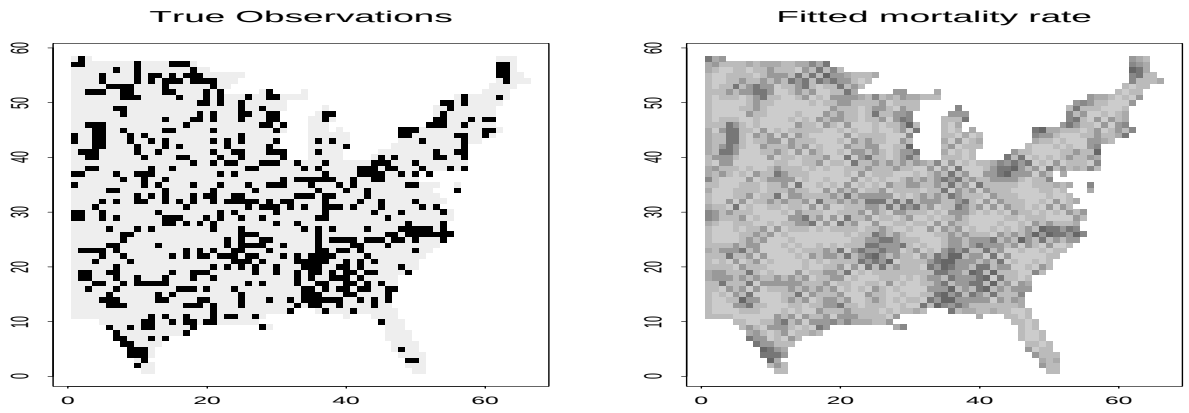

Figure 4: The U.S. cancer mortality rate data. (a) The mortality map of liver and gallbladder cancer (including bile ducts) for white males during the decade 1950-1959. The black squares denote the counties of high cancer mortality rate, and the white squares denote the counties of low cancer mortality rate. (b) Fitted cancer mortality rates. The cancer mortality rate of each county is represented by the gray level of the corresponding square.

SAMC was first applied to this problem. The proposal matrix Q, the proposal distribution T and the MCMC sampler used in step (b) are specified as follows. Let the m models be coded as a matrix (M_{ij}) with $i = 0, \ldots, 10$ and $j = 0, \ldots, 10$. The proposal matrix Q is then defined as follows,

$$Q(M_{ij} \to M_{i'j'}) = q_{ii'}^{(\alpha)} q_{jj'}^{(\beta)},$$

where $q_{i,i-1}^{(\alpha)} = q_{i,i}^{(\alpha)} = q_{i,i+1}^{(\alpha)} = 1/3$ for $i = 1, \ldots, 9$, $q_{0,0}^{(\alpha)} = q_{10,10}^{(\alpha)} = 2/3$, and $q_{0,1}^{(\alpha)} = q_{10,9}^{(\alpha)} = 1/3$; and $q_{i,i-1}^{(\beta)} = q_{i,i}^{(\beta)} = q_{i,i+1}^{(\beta)} = 1/3$ for $i = 1, \ldots, 9$, $q_{0,0}^{(\beta)} = q_{10,10}^{(\beta)} = 2/3$, and $q_{0,1}^{(\beta)} = q_{10,9}^{(\beta)} = 1/3$. For this example, ϑ corresponds to the configuration $\boldsymbol{s}$ of the model. The proposal distribution $T(\vartheta^{(t)} \to \vartheta^*)$ is an identical mapping, i.e., keeping the current configuration unchanged when a model is proposed to be changed to another one. Thus, we have $T(\vartheta^{(t)} \to \vartheta^*) = T(\vartheta^* \to \vartheta^{(t)}) = 1$. The MCMC sampler used in step (b) is the Gibbs sampler (Geman and Geman, 1984): sampling spin i from the conditional distribution

$$P(s_i = +1|N(i)) = \frac{1}{1 + e^{-2(\alpha + \beta \sum_{j \in N(i)} s_j)}}, \quad P(s_i = -1|N(i)) = 1 - P(s_i = +1|N(i)), \quad (17)$$

for all $i \in D$ in a pre-specified order.

SAMC was run 5 times independently. Each run consisted of two stages. The first stage was to estimate the function $\varphi(\alpha, \beta)$ on the sublattice. In this stage, SAMC was run with

$t_0 = 10^4$ and for 10^8 iterations. The second stage was to draw importance samples from the trial distribution,

$$\widehat{p}^*_{mix}(\boldsymbol{s}) \propto \frac{1}{m^*} \sum_{k=1}^{m^*} \frac{1}{\widehat{\varphi}(\alpha_k, \beta_k)} \exp\{\alpha_k \sum_{i \in D} s_i + \frac{\beta_k}{2} \sum_{i \in D} s_i (\sum_{j \in N(i)} s_j)\}, \tag{18}$$

which represents an approximation to (16) with $\varphi(\alpha_j, \beta_j)$ being replaced by its estimate obtained in the first stage. In this stage, SAMC was run with $\delta_t \equiv 0$ and for 10^7 iterations, and a total of 10^5 samples were harvested at equally spaced time points. Each run cost about 115m CPU time in a 2.8GHZ computer. In the second stage, SAMC is reduced to RJMCMC by setting $\delta_t = 0$. Figure 5(a) shows one estimate of $\varphi(\alpha, \beta)$ obtained in a run of SAMC.

Using the importance samples collected above, we estimated the probability $P(s_i = +1|\alpha, \beta)$, which is a function of (α, β). The estimation can be done in (9) by setting $h(\boldsymbol{s}) = \sum_{i \in D}(s_i + 1)/(2|D|)$. By averaging over the five runs, we obtained one estimate of the function as shown in Figure 5(b). To assess the variation of the estimate, we calculated the standard deviation of the estimate at each grid point of the lattice. The average of the standard deviations is 3×10^{-4}. The estimate is fairly stable.

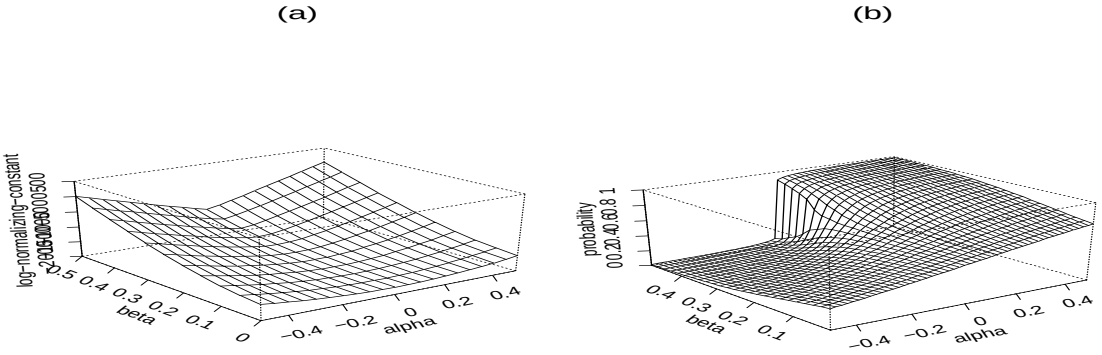

Figure 5: Computational results of SAMC. (a) Estimate of $\log \varphi(\alpha, \beta)$ on a 21×11 lattice with $\alpha\{-0.5, -0.45, \ldots, 0.5\}$ and $\beta \in \{0, 0.05, \ldots, 0.5\}$. (b) Estimate of $P(s_i = +1|\alpha, \beta)$ on a 50×25 lattice with $\alpha \in \{-0.49, -0.47, \ldots, 0.49\}$ and $\beta \in \{0.01, 0.03, \ldots, 0.49\}$.

Using the importance samples collected above, we also estimated the parameters (α, β) for the cancer data shown in Figure 4(a). The estimation can be done using the Monte

Carlo maximum likelihood method (Geyer and Thompson, 1992; Geyer, 1994) as follows. Let $p^*(\boldsymbol{s}) = c^* p_0^*(\boldsymbol{s})$ denote an arbitrary trial distribution for (15), where $p_0^*(\boldsymbol{s})$ is completely specified and c^* is an unknown constant. Let $\psi(\alpha, \beta, \boldsymbol{s}) = \varphi(\alpha, \beta) p(\boldsymbol{s}|\alpha, \beta)$, and $L(\alpha, \beta|\boldsymbol{s})$ denote the log-likelihood function of an observation $\boldsymbol{s}$. Thus,

$$L_n(\alpha, \beta|\boldsymbol{s}) = \alpha \sum_{i \in D} s_i + \frac{\beta}{2} \sum_{i \in D} s_i \Big(\sum_{j \in N(i)} s_j \Big) + \log c^* - \log \Big[\frac{1}{n} \sum_{k=1}^{n} \frac{\psi(\alpha, \beta, \boldsymbol{s}^{(k)})}{p_0^*(\boldsymbol{s}^{(k)})} \Big], \tag{19}$$

approaches $L(\alpha, \beta|\boldsymbol{s})$ as $n \to \infty$, where $\boldsymbol{s}^{(1)}, \ldots, \boldsymbol{s}^{(n)}$ are MCMC samples simulated from $p^*(\boldsymbol{s})$. The estimate $(\widehat{\alpha}, \widehat{\beta}) = \arg\max_{\alpha,\beta} L_n(\alpha, \beta|\boldsymbol{s})$ is called the Monte Carlo maximum likelihood estimator (MCMLE) of (α, β). The maximization can be done using a conventional optimization procedure, say, the conjugate gradient method. Setting $p^*(\boldsymbol{s}) = \widehat{p}_{mix}^*(\boldsymbol{s})$, the five runs of SAMC resulted in five estimates of (α, β). The mean and standard deviation vectors of these estimates are (-0.2994,0.1237) and (0.00063,0.00027), respectively. Henceforth, these estimates are called mix-MCMLEs, because they are obtained based on a mixture trial distribution. Figure 4(b) shows the fitted mortality map based on one mix-MCMLE $(-0.2999, 0.1234)$.

In the literature, $p^*(\boldsymbol{s})$ is often constructed based on a single parameter point; that is, setting

$$p^*(\boldsymbol{s}) \propto \exp\{\alpha^* \sum_{i \in D} s_i + \frac{\beta^*}{2} \sum_{i \in D} s_i (\sum_{j \in N(i)} s_j)\}, \tag{20}$$

where (α^*, β^*) denotes the parameter point. The point (α^*, β^*) should be chosen to be close to the true parameter point, otherwise, a large value of n would be required for the convergence of (19). Sherman *et al* (2006) set (α^*, β^*) to be the maximum pseudo-likelihood estimate (Besag, 1975) of (α, β), which is the MLE of the pseudo-likelihood function

$$PL(\alpha, \beta|\boldsymbol{s}) = \prod_{i \in D} \frac{\exp\{s_i(\alpha + \beta \sum_{j \in N(i)} s_j)\}}{\exp\{\alpha + \beta \sum_{j \in N(i)} s_j\} + \exp\{-\alpha - \beta \sum_{j \in N(i)} s_j\}}. \tag{21}$$

We repeated Sherman *et al*'s procedure for the cancer data five times with $n = 10^5$ and the MCMC samples being collected at equally spaced time points in a run of the Gibbs sampler of 10^7 iteration cycles. The mean and standard deviation vectors of the resulting estimates are (-0.3073,0.1262) and (0.00837, 0.00946), respectively. These estimates have a significantly higher variation than the mix-MCMLEs. Henceforth, these estimates are called single-MCMLEs, because they are obtained based on a single-point trial distribution.

To compare the accuracy of the mix-MCMLEs and single-MCMLEs, we conducted the following experiment based on the principle of the parametric bootstrap method (Efron and

Estimate	single-MCMLE	mix-MCMLE
RMSE($\boldsymbol{t}_1^{sim}$)	59.51	2.90
RMSE($\boldsymbol{t}_2^{sim}$)	114.91	4.61

Table 2: Comparison of the accuracy of the mix-MCMLEs and single-MCMLEs for the US cancer data. RMSE($\boldsymbol{t}_i^{sim}$) is calculated as $\sqrt{\sum_{k=1}^{5}(\boldsymbol{t}_i^{sim,k} - \boldsymbol{t}_i^{obs})^2/5}$, where $i = 1, 2$, and $\boldsymbol{t}_i^{sim,k}$ denotes the value of $\boldsymbol{t}_i^{sim}$ calculated based on the k^{th} estimate of (α, β).

Tibshirani, 1993). Let $\boldsymbol{T}_1 = \sum_{i\in D} s_i$ and $\boldsymbol{T}_2 = \frac{1}{2}\sum_{i\in D} s_i \left(\sum_{j\in N(i)} s_j\right)$. It is easy to see that $\boldsymbol{T} = (\boldsymbol{T}_1, \boldsymbol{T}_2)$ forms a sufficient statistic of (α, β). Given an estimate $(\widehat{\alpha}, \widehat{\beta})$, we can reversely estimate the quantities $\boldsymbol{T}_1$ and $\boldsymbol{T}_2$ by drawing samples from the distribution $f(\boldsymbol{s}|\widehat{\alpha}, \widehat{\beta})$. If $(\widehat{\alpha}, \widehat{\beta})$ is accurate, one should have $\boldsymbol{t}^{obs} \approx \boldsymbol{t}^{sim}$, where $\boldsymbol{t}^{obs}$ and $\boldsymbol{t}^{sim}$ denote the values of $\boldsymbol{T}$ calculated from the true observation and from the simulated samples, respectively. To calculate $\boldsymbol{t}^{sim}$, 1000 independent configurations were generated conditional on each estimate with each configuration being generated by a short run of the Gibbs sampler. The Gibbs sampler started with a random configuration and was iterated for 1000 cycles. A convergence diagnostic shows that 1000 iteration cycles have been long enough for the Gibbs sampler to reach equilibrium for simulation of $f(\boldsymbol{s}|\widehat{\alpha}, \widehat{\beta})$. Table 2 compares the root mean squared errors (RMSEs) of $\boldsymbol{t}^{sim}$'s calculated from the mix-MCMLEs and single-MCMLEs. The comparison shows that the mix-MCMLEs are much more accurate than the single-MCMLEs for this example.

For comparison, RJMCMC was also run for this example for 10^8 iterations. The simulation started with model $M_{0,0}$, moved to model $M_{10,10}$ very fast and then got stuck there. This is shown in Figures 6 (c)&(d), where the parameter vector $(\alpha, \beta) = (0, 0)$ corresponds to model M_0 and $(0.5, 0.5)$ corresponds to model $M_{10,10}$. RJMCMC failed to estimate $\varphi(\alpha_1, \beta_1), \ldots, \varphi(\alpha_m, \beta_m)$ simultaneously. This phenomenon can be easily understood from Figure 5(a), which indicates that model $M_{10,10}$ has a dominated probability over other models. Fives runs of SAMC produced an estimate of the log-odd ratio $\log P(M_{10,10})/P(M_{0,0})$. The estimate is 1775.7 with standard deviation 0.6. Making transitions between models with such a huge difference in probability is beyond the ability of RJMCMC. It is also beyond the ability of other advanced MCMC samplers, such as simulated tempering, parallel tempering and evolutionary Monte Carlo, because the strength of these advanced MCMC samplers is

at making transitions between different modes of the distribution instead of sampling from low probability models. However, it is not hard for SAMC due to its capability of sampling rare events from a large sample space. For comparison, we plot in Figures 6 (a)&(b) the sample paths of α and β obtained in a run of SAMC. This figure indicates that even though the models have very large differences in probabilities, SAMC can still mix them well and sample each model equally. Note that the desired sampling distribution has been set to the uniform distribution for this example and other examples of this section.

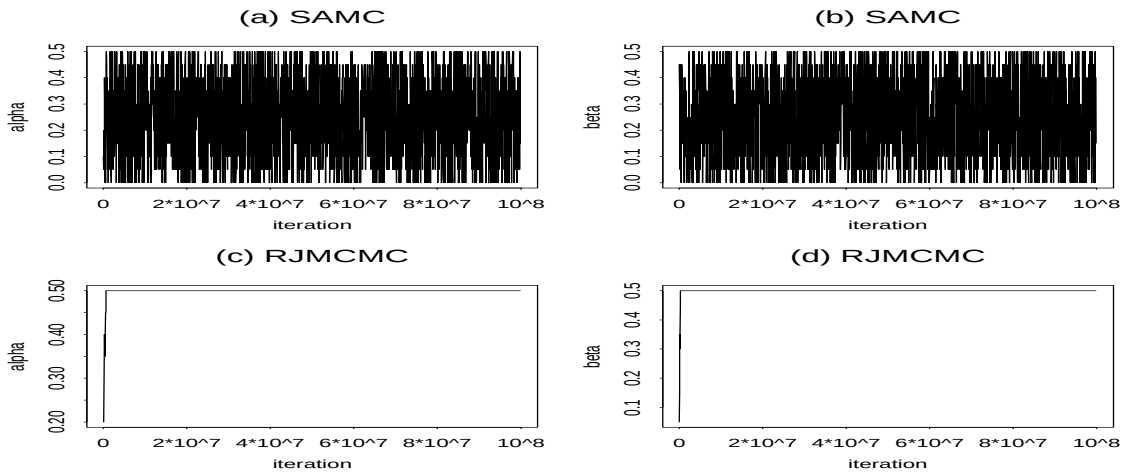

Figure 6: Comparison of SAMC and RJMCMC. Plots (a) and (b) show, respectively, the sample paths of α and β in a run of SAMC. Plots (c) and (d) show, respectively, the sample paths of α and β in a run of RJMCMC.

6 Discussion

In this paper, we have introduced the SAMC algorithm and studied its convergence. SAMC overcomes the shortcomings of the WL algorithm. It can improve the estimates continuously as the simulation goes on. Two classes of applications, importance sampling and model selection, are discussed. SAMC can work as a general importance sampling method and a model selection sampler when the model space is complex.

As with many other Monte Carlo algorithms, such as slice sampling (Neal, 2003), SAMC also suffers from the curse of dimensionality. For example, consider the modified witch's hat

distribution studied in Geyer and Thompson (1995),

$$p(\boldsymbol{x}) = \begin{cases} 1+\beta, & \boldsymbol{x} \in [0,\alpha]^k, \\ 1, & \boldsymbol{x} \in [0,1]^k \setminus [0,\alpha]^k, \end{cases} \tag{22}$$

where $k = 30$ is the dimension of $\boldsymbol{x}$, $\alpha = 1/3$, and $\beta \approx 10^{14}$ which is chosen such that the probability of the peak is $1/3$ exactly. For this distribution, the small hypercube is called the peak and the rest the brim. It is easy to see that SAMC is not better than MH for sampling from this distribution, if the sample space is partitioned according to the energy function. The peak is like an atom, so SAMC will make a random walk in the brim just like MH. The chance for SAMC to jump into the peak from the brim is decreasing geometrically as the dimension increases. One way to overcome this difficulty is to include an auxiliary variable in (22) and to work on the joint distribution,

$$p(\boldsymbol{x}_I, I) = \begin{cases} 1+\beta_I, & \boldsymbol{x}_I \in [0,\alpha]^I, \\ 1, & \boldsymbol{x}_I \in [0,1]^I \setminus [0,\alpha]^I, \end{cases} \tag{23}$$

where I is the dimension of $\boldsymbol{x}_I$ with $I \in \{1, \ldots, 30\}$, and β_I is chosen such that the peak probability is $1/3$ exactly. To sample from (23), we can make a joint partition on I and energy. Let $E_{11}, E_{12}, \ldots, E_{k1}, E_{k2}$ denote the partition, where E_{i1} and E_{i2} denote, respectively, the peak and brim sets of $p(\boldsymbol{x}_i, i)$. SAMC can then work on the distribution with this partition and appropriate proposal distributions (dimension jumping will be involved). As shown by Liang (2003), working on such a sequence of trial distributions indexed by dimension can help the sampler to reduce the curse of dimensionality. We note that the auxiliary variable used in construction of the joint distribution is not necessarily the dimension variable; the temperature variable can be used as in simulated tempering for some problems for which the dimension change is not sensible.

In our theoretical results on convergence, we assume that the sample space $\mathcal{X}$ and the parameter space Θ are both compact. At least in principle, these restrictions can be removed as in Andrieu, Moulines and Priouret (2005). If the restrictions are removed, we may need to put some other constraints on the tails of the target distribution $p(\boldsymbol{x})$ and the proposal distribution $q(\boldsymbol{x}, \boldsymbol{y})$ to ensure the minorisation condition holds, see Roberts and Tweedie (1996), Rosenthal (1995) and Roberts and Rosenthal (2004) for more discussions on the issue. Our numerical experience indicates that SAMC should have some type of convergence

even when the minorisation condition does not hold, in a manner similar to the Metropolis-Hastings algorithm (Mengersen and Tweedie, 1996). A further study in this direction is of some interest.

Appendix: Theoretical Results on SAMC

The appendix is organized as follows. In part I, we describe a theorem for the convergence of the SAMC algorithm. In part II, we briefly review the published results on the convergence of a general stochastic approximation algorithm. In part III , we give a proof for the theorem described in Section 1.

I. A Convergence Theorem for SAMC Without loss of generality, we only show the convergence presented in Equation (7) of the paper for the case that all subregions are non-empty or, equivalently, $d = 0$. Extension to the case $d \neq 0$ is trivial, because changing step (ii) of the SAMC algorithm to (ii)$'$ (given below) will not change the process of simulation.

(ii)$'$ Set $\theta' = \theta_t + \gamma_t(\boldsymbol{e}_t - \boldsymbol{\pi} - \boldsymbol{d})$, where $\boldsymbol{d}$ is an m-vector of d.

Theorem 6.1 *Let $E_1, \ldots, E_m$ be a partition of a compact sample space $\mathcal{X}$ and $\psi(\boldsymbol{x})$ be a non-negative function defined on $\mathcal{X}$ with $0 < \int_{E_i} \psi(\boldsymbol{x}) d\boldsymbol{x} < \infty$ for all E_i's. Let $\boldsymbol{\pi} = (\pi_1, \ldots, \pi_m)$ be an m-vector with $0 < \pi_i < 1$ and $\sum_{i=1}^m \pi_i = 1$. Let Θ be a compact set of m dimensions, and there exists a constant C such that $\breve{\theta} \in \Theta$, where $\breve{\theta} = (\breve{\theta}_1, \ldots, \breve{\theta}_m)$ and $\breve{\theta}_i = C + \log(\int_{E_i} \psi(\boldsymbol{x}) d\boldsymbol{x}) - \log(\pi_i)$. Let $\theta_0 \in \Theta$ be an initial estimate of $\breve{\theta}$, and $\theta_t \in \Theta$ be the estimate of $\breve{\theta}$ at iteration t. Let $\{\gamma_t\}$ be an non-increasing, positive sequence as specified in (6). Suppose that $p_{\theta_t}(\boldsymbol{x})$ is bounded away from 0 and ∞ on $\mathcal{X}$, and the proposal distribution satisfies the condition (4). As $t \to \infty$, we have*

$$P\{\lim_{t\to\infty} \theta_{ti} = C + \log(\int_{E_i} \psi(\boldsymbol{x}) d\boldsymbol{x}) - \log(\pi_i)\} = 1, \quad i = 1, \ldots, m, \tag{24}$$

where C is an arbitrary constant.

II. Existing Results on the Convergence of a General Stochastic Approximation Algorithm Suppose that our target is to solve the following integration equation for the parameter vector θ,

$$h(\theta) = \int_{\mathcal{X}} H(\theta, \boldsymbol{x}) p(d\boldsymbol{x}) = 0, \quad \theta \in \Theta.$$

The stochastic approximation algorithm with MCMC innovations (noise) works iteratively as follows. Let $K(\boldsymbol{x}_t, \cdot)$ be a MCMC transition kernel. For example, it can be the MH kernel of the form,

$$K(\boldsymbol{x}_t, d\boldsymbol{y}) = s(\boldsymbol{x}_t, d\boldsymbol{y}) + I(\boldsymbol{x}_t \in d\boldsymbol{y})[1 - \int_{\mathcal{X}} s(\boldsymbol{x}_t, \boldsymbol{z})d\boldsymbol{z}],$$

where $s(\boldsymbol{x}_t, d\boldsymbol{y}) = q(\boldsymbol{x}_t, d\boldsymbol{y}) \min\{1, [p(\boldsymbol{y})q(\boldsymbol{y}, \boldsymbol{x}_t)]/[p(\boldsymbol{x})q(\boldsymbol{x}, \boldsymbol{y})]\}$, and $q(\cdot, \cdot)$ is the proposal distribution and $p(\cdot)$ is the invariant distribution. Let $\Theta \subset \tilde{\Theta}$ be a compact subset of $\tilde{\Theta}$. Let $\{\gamma_t\}_{t=0}^{\infty}$ be a monotone, non-increasing sequence governing the step size. Also define a function $\Phi : \mathcal{X} \times \tilde{\Theta} \to \mathcal{X} \times \Theta$, which reinitializes the non-homogeneous Markov chain $\{(\boldsymbol{x}_t, \theta_t)\}$. The function Φ can for instance generate a random or fixed point, or project $(\boldsymbol{x}_{t+1}, \theta_{t+1})$ onto $\mathcal{X} \times \Theta$. An iteration of the algorithm is as follows.

(i) Generate $\boldsymbol{y} \sim K_{\theta_t}(\boldsymbol{x}_t, \cdot)$.

(ii) Set $\theta^* = \theta_t + \gamma_{t+1} H(\theta_t, \boldsymbol{y})$.

(iii) If $\theta^* \in \Theta$, then set $(\boldsymbol{x}_{t+1}, \theta_{t+1}) = (\boldsymbol{y}, \theta^*)$; otherwise, set $(\boldsymbol{x}_{t+1}, \theta_{t+1}) = \Phi(\boldsymbol{y}, \theta^*)$.

This algorithm is actually a simplified version of the algorithm presented in Andrieu *et al* (2005). Let $\mathbb{P}_{\boldsymbol{x}_0,\theta_0}$ denote the probability measure of the Markov chain $\{(\boldsymbol{x}_t, \theta_t)\}$, started in $(\boldsymbol{x}_0, \theta_0)$, and implicitly defined by the sequences $\{\gamma_t\}$. Define $D(\boldsymbol{x}, A) = \inf_{\boldsymbol{y} \in A} |\boldsymbol{x} - \boldsymbol{y}|$.

Theorem 6.2 *(Theorem 5.5 and Proposition 6.1, Andrieu et al, 2005) Assume the conditions* (A_1) *and* (A_4) *hold, and there exists a drift function* $V(\boldsymbol{x})$ *such that* $\sup_{\boldsymbol{x} \in \mathcal{X}} V(\boldsymbol{x}) < \infty$ *and the drift condition holds (refer to Andrieu et al (2005) for the description of the conditions). Let the sequence* $\{\theta_n\}$ *be defined as in the stochastic approximation algorithm. Then for all* $(\boldsymbol{x}_0, \theta_0) \in \mathcal{X} \times \Theta$*,*

$$\lim_{t \to \infty} D(\theta_t, \mathcal{L}) = 0, \qquad \mathbb{P}_{\boldsymbol{x}_0,\theta_0} - a.e.$$

III. Proof of Theorem 6.1

PROOF: To prove Theorem 6.1, it suffices to verify that (A_1), (A_4) and the drift condition hold for the SAMC algorithm. To simplify notation, in the proof we will drop the subscript t by denoting $\boldsymbol{x}_t$ by $\boldsymbol{x}$ and $\theta_t = (\theta_{t1}, \ldots, \theta_{tm})$ by $\theta = (\theta_1, \ldots, \theta_m)$. Since the invariant distribution of the MH kernel is $p_\theta(\boldsymbol{x})$, for any fixed θ, we have

$$E(e_{\boldsymbol{x}}^{(i)} - \pi_i) = \int_{\mathcal{X}} (e_{\boldsymbol{x}}^{(i)} - \pi_i) p_\theta(\boldsymbol{x}) d\boldsymbol{x} = \frac{\int_{E_i} \psi(\boldsymbol{x}) d\boldsymbol{x}/e^{\theta_i}}{\sum_{k=1}^{m} [\int_{E_k} \psi(\boldsymbol{x}) d\boldsymbol{x}/e^{\theta_k}]} - \pi_i = \frac{S_i}{S} - \pi_i, \quad i = 1, \ldots, m, \tag{25}$$

where $S_i = \int_{E_i} \psi(\boldsymbol{x}) d\boldsymbol{x}/e^{\theta_i}$ and $S = \sum_{k=1}^m S_k$. Thus,

$$h(\theta) = \int_{\mathcal{X}} H(\theta, \boldsymbol{x}) p(d\boldsymbol{x}) = (\frac{S_1}{S} - \pi_1, \ldots, \frac{S_m}{S} - \pi_m)'.$$

• **Condition A_1:** It follows from (25) that $h(\theta)$ is a continuous function of θ. Let $w(\theta) = \frac{1}{2}\sum_{k=1}^m (\frac{S_k}{S} - \pi_k)^2$. As shown below, $w(\theta)$ has continuous partial derivatives of the first order. Since $0 \le w(\theta) \le \frac{1}{2}[\sum_{k=1}^m (\frac{S_k}{S})^2 + \pi_k^2)] \le 1$ for all $\theta \in \Theta$, and Θ itself is compact, the level set $\mathcal{W}_M = \{\theta \in \Theta, w(\theta) \le M\}$ is compact for any positive integer M. Condition (A_1-ii) is satisfied.

Solving the system of equations formed by (25), we have

$$\mathcal{L} = \{(\theta_1, \ldots, \theta_m) : \theta_i = c + \log(\int_{E_i} \psi(\boldsymbol{x}) d\boldsymbol{x}) - \log(\pi_i), i = 1, \ldots, m; \theta \in \Theta\},$$

where $c = \log(S)$ can be determined by imposing a constraint on S. For example, setting $S = 1$ leads to that $c = 0$. It is obvious that $\mathcal{L}$ is nonempty and $w(\theta) = 0$ for every $\theta \in \mathcal{L}$.

To verify the conditions (A_1-i), (A_1-iii) and (A_1-iv), we have the following calculations.

$$\begin{aligned} &\frac{\partial S}{\partial \theta_i} = \frac{\partial S_i}{\partial \theta_i} = -S_i, \qquad \frac{\partial S_i}{\partial \theta_j} = \frac{\partial S_j}{\partial \theta_i} = 0, \\ &\frac{\partial (\frac{S_i}{S})}{\partial \theta_i} = -\frac{S_i}{S}(1 - \frac{S_i}{S}), \qquad \frac{\partial (\frac{S_i}{S})}{\partial \theta_j} = \frac{\partial (\frac{S_j}{S})}{\partial \theta_i} = \frac{S_i S_j}{S^2}, \end{aligned} \tag{26}$$

for $i, j = 1, \ldots, m$ and $i \ne j$.

$$\begin{aligned} \frac{\partial w(\theta)}{\partial \theta_i} &= \frac{1}{2}\sum_{k=1}^m \frac{\partial (\frac{S_k}{S} - \pi_k)^2}{\partial \theta_i} = \sum_{j \ne i} (\frac{S_j}{S} - \pi_j)\frac{S_i S_j}{S^2} - (\frac{S_i}{S} - \pi_i)\frac{S_i}{S}(1 - \frac{S_i}{S}) \\ &= \sum_{j=1}^m (\frac{S_j}{S} - \pi_j)\frac{S_i S_j}{S^2} - (\frac{S_i}{S} - \pi_i)\frac{S_i}{S} = \mu_\eta \frac{S_i}{S} - (\frac{S_i}{S} - \pi_i)\frac{S_i}{S}, \end{aligned} \tag{27}$$

for $i = 1, \ldots, m$, where it is defined $\mu_\eta = \sum_{j=1}^m (\frac{S_j}{S} - \pi_j)\frac{S_j}{S}$. Thus,

$$\langle \nabla w(\theta), h(\theta) \rangle = \mu_\eta \sum_{i=1}^m (\frac{S_i}{S} - \pi_i)\frac{S_i}{S} - \sum_{i=1}^m (\frac{S_i}{S} - \pi_i)^2 \frac{S_i}{S} = -\{\sum_{i=1}^m (\frac{S_i}{S} - \pi_i)^2 \frac{S_i}{S} - \mu_\eta^2\} = -\sigma_\eta^2 \le 0, \tag{28}$$

where σ_η^2 denotes the variance of the discrete distribution defined by the following table,

State (η)	$\frac{S_1}{S} - \pi_1$	$\cdots$	$\frac{S_m}{S} - \pi_m$
Prob.	$\frac{S_1}{S}$	$\cdots$	$\frac{S_m}{S}$

If $\theta \in \mathcal{L}$, $\langle \nabla w(\theta), h(\theta) \rangle = 0$; otherwise, $\langle \nabla w(\theta), h(\theta) \rangle < 0$. For any $M_0 \in (0, 1]$, it is true that $\mathcal{L} \subset \{\theta \in \Theta, w(\theta) < M_0\}$. Hence, condition ($A_1$-i) is satisfied.

It follows from (28) that $\langle \nabla w(\theta), h(\theta) \rangle \leq 0$ for all $\theta \in \Theta$. The $w(\mathcal{L})$ forms a line in space Θ, as it contains only one free parameter c. Therefore, the interior set of $w(\mathcal{L})$ is empty. Conditions (A_1-iii) and (A_1-iv) are satisfied.

• **Condition** A_4**:** Let p be arbitrarily large, $\beta = 1$, $\alpha = 1$, and $\zeta \in (\frac{1}{\tau}, 2)$. Thus, the conditions $\sum_{t=1}^{\infty} \gamma_t = \infty$ and $\sum_{t=1}^{\infty} \gamma_t^{\zeta} < \infty$. Since $|H(\theta, \boldsymbol{x})|$ is bounded above by c_1 as shown in (31), $|\gamma_t H(\theta_{t-1}, \boldsymbol{x}_t)| < c_1 \gamma_t < c_1 \gamma_t^{\eta}$ holds. Condition (A_4) is satisfied by choosing $C = c_1$ and $\eta \in [(\zeta - 1)/\alpha, (p - \zeta)/p] = [\zeta - 1, 1)$.

•**(Drift condition)** Theorem 2.2 of Roberts and Tweedie (1996) shows that if the target distribution is bounded away from 0 and ∞ on every compact set of its support $\mathcal{X}$, then the MH chain with a proposal distribution satisfying the condition (4) is irreducible and aperiodic, and every nonempty compact set is small. Hence, K_θ, the MH kernel used in each iteration of SAMC, is irreducible and aperiodic for any $\theta \in \Theta$. Since $\mathcal{X}$ is compact, $\mathcal{X}$ is a small set and thus the minorisation condition is satisfied, i.e., there exists an integer l such that

$$\inf_{\theta \in \Theta} K_\theta^l(\boldsymbol{x}, A) \geq \delta \nu(A), \quad \forall \boldsymbol{x} \in \mathcal{X}, \ \forall A \in \mathcal{B}. \tag{29}$$

Define $K_\theta V(\boldsymbol{x}) = \int_{\mathcal{X}} K_\theta(\boldsymbol{x}, \boldsymbol{y}) V(\boldsymbol{y}) d\boldsymbol{y}$. Since $C = \mathcal{X}$ is small, the following conditions hold

$$\begin{aligned} &\sup_{\theta \in \Theta_0} K_\theta^l V^p(\boldsymbol{x}) \leq \lambda V^p(\boldsymbol{x}) + b I(\boldsymbol{x} \in C), \quad \forall \boldsymbol{x} \in \mathcal{X}, \\ &\sup_{\theta \in \Theta_0} K_\theta V^p(\boldsymbol{x}) \leq \kappa V^p(\boldsymbol{x}), \quad \forall \boldsymbol{x} \in \mathcal{X}, \end{aligned} \tag{30}$$

by choosing the drift function $V(\boldsymbol{x}) = 1$, $\Theta_0 = \Theta$, $0 < \lambda < 1$, $b = 1 - \lambda$, $\kappa > 1$, $p \in [2, \infty)$ and any integer l. Equations (29) and (30) imply that (DRI1) is satisfied.

Let $H^{(i)}(\theta, \boldsymbol{x})$ be the i^{th} component of the vector $H(\theta, \boldsymbol{x}) = (\boldsymbol{e}_{\boldsymbol{x}} - \boldsymbol{\pi})$. By construction, $|H^{(i)}(\theta, \boldsymbol{x})| = |e_{\boldsymbol{x}}^{(i)} - \pi_i| < 1$ for all $\boldsymbol{x} \in \mathcal{X}$ and $i = 1, \ldots, m$. Therefore, there exists a constant $c_1 = \sqrt{m}$ such that, for all $\boldsymbol{x} \in \mathcal{X}$,

$$\sup_{\theta \in \Theta} |H(\theta, \boldsymbol{x})| \leq c_1. \tag{31}$$

Also, $H(\theta, \boldsymbol{x})$ does not depend on θ for a given sample $\boldsymbol{x}$. Hence, $H(\theta, \boldsymbol{x}) - H(\theta', \boldsymbol{x}) = 0$ for all $(\theta, \theta') \in \Theta \times \Theta$, and the following condition holds for the SAMC algorithm,

$$\sup_{(\theta, \theta') \in \Theta \times \Theta} |H(\theta, \boldsymbol{x}) - H(\theta', \boldsymbol{x})| \leq c_1 |\theta - \theta'|. \tag{32}$$

Equations (31) and (32) imply that (DRI2) is satisfied by choosing $\beta = 1$ and $V(\boldsymbol{x}) = 1$.

Let $s_\theta(\boldsymbol{x}, \boldsymbol{y}) = q(\boldsymbol{x}, \boldsymbol{y}) \min\{1, r(\theta, \boldsymbol{x}, \boldsymbol{y})\}$, where $r(\theta, \boldsymbol{x}, \boldsymbol{y}) = \frac{p_\theta(\boldsymbol{y})q(\boldsymbol{y},\boldsymbol{x})}{p_\theta(\boldsymbol{x})q(\boldsymbol{x},\boldsymbol{y})}$. Thus, we have

$$\begin{aligned}\Big|\frac{\partial s_\theta(\boldsymbol{x}, \boldsymbol{y})}{\partial \theta_i}\Big| &= \Big| - q(\boldsymbol{x}, \boldsymbol{y}) I(r(\theta, \boldsymbol{x}, \boldsymbol{y}) < 1) I(J(\boldsymbol{x}) = i \text{ or } J(\boldsymbol{y}) = i) I(J(\boldsymbol{x}) \neq J(\boldsymbol{y})) r(\theta, \boldsymbol{x}, \boldsymbol{y})\Big| \\ &\leq q(\boldsymbol{x}, \boldsymbol{y}),\end{aligned}$$

where $I(\cdot)$ is the indicator function, and $J(\boldsymbol{x})$ denotes the index of the subregion where $\boldsymbol{x}$ belongs to. The mean-value theorem implies that there exists a constant c_2 such that

$$|s_\theta(\boldsymbol{x}, \boldsymbol{y}) - s_{\theta'}(\boldsymbol{x}, \boldsymbol{y})| \leq q(\boldsymbol{x}, \boldsymbol{y}) c_2 |\theta - \theta'|, \tag{33}$$

which implies that

$$\sup_{\boldsymbol{x}} \|s_\theta(\boldsymbol{x}, \cdot) - s_{\theta'}(\boldsymbol{x}, \cdot)\|_1 = \sup_{\boldsymbol{x}} \int_{\mathcal{X}} |s_\theta(\boldsymbol{x}, \boldsymbol{y}) - s_{\theta'}(\boldsymbol{x}, \boldsymbol{y})| d\boldsymbol{y} \leq c_2 |\theta - \theta'|. \tag{34}$$

In addition, for any measurable set $A \subset \mathcal{X}$ we have

$$\begin{aligned}&|K_\theta(\boldsymbol{x}, A) - K_{\theta'}(\boldsymbol{x}, A)| = \Big| \int_A [s_\theta(\boldsymbol{x}, \boldsymbol{y}) - s_{\theta'}(\boldsymbol{x}, \boldsymbol{y})] d\boldsymbol{y} + I(\boldsymbol{x} \in A) \int_{\mathcal{X}} [s_{\theta'}(\boldsymbol{x}, \boldsymbol{z}) - s_\theta(\boldsymbol{x}, \boldsymbol{z})] d\boldsymbol{z}\Big| \\ &\leq \int_{\mathcal{X}} |s_\theta(\boldsymbol{x}, \boldsymbol{y}) - s_{\theta'}(\boldsymbol{x}, \boldsymbol{y})| d\boldsymbol{y} + I(\boldsymbol{x} \in A) \int_{\mathcal{X}} |s_{\theta'}(\boldsymbol{x}, \boldsymbol{z}) - s_\theta(\boldsymbol{x}, \boldsymbol{z})| d\boldsymbol{z} \\ &\leq 2 \int_{\mathcal{X}} |s_\theta(\boldsymbol{x}, \boldsymbol{y}) - s_{\theta'}(\boldsymbol{x}, \boldsymbol{y})| d\boldsymbol{y} \leq 2c_2 |\theta - \theta'|.\end{aligned} \tag{35}$$

For $g : \mathcal{X} \to \mathbb{R}^d$, define the norm $\|g\|_V = \sup_{\boldsymbol{x} \in \mathcal{X}} \frac{|g(\boldsymbol{x})|}{V(\boldsymbol{x})}$. Then, for any function $g \in \mathcal{L}_V = \{g : \mathcal{X} \to \mathbb{R}^d, \|g\|_V < \infty\}$, we have

$$\begin{aligned}&\|K_\theta g - K_{\theta'} g\|_V = \| \int (K_\theta(\boldsymbol{x}, d\boldsymbol{y}) - K_{\theta'}(\boldsymbol{x}, d\boldsymbol{y})) g(\boldsymbol{y})\|_V \\ &= \| \int_{\mathcal{X}^+} (K_\theta(\boldsymbol{x}, d\boldsymbol{y}) - K_{\theta'}(\boldsymbol{x}, d\boldsymbol{y})) g(\boldsymbol{y}) + \int_{\mathcal{X}^-} (K_\theta(\boldsymbol{x}, d\boldsymbol{y}) - K_{\theta'}(\boldsymbol{x}, d\boldsymbol{y})) g(\boldsymbol{y})\|_V \\ &\leq \| \max\{ \int_{\mathcal{X}^+} (K_\theta(\boldsymbol{x}, d\boldsymbol{y}) - K_{\theta'}(\boldsymbol{x}, d\boldsymbol{y})) g(\boldsymbol{y}), - \int_{\mathcal{X}^-} (K_\theta(\boldsymbol{x}, d\boldsymbol{y}) - K_{\theta'}(\boldsymbol{x}, d\boldsymbol{y})) g(\boldsymbol{y})\}\|_V \\ &\leq \|g\|_V \max\{|K_\theta(\boldsymbol{x}, \mathcal{X}^+) - K_{\theta'}(\boldsymbol{x}, \mathcal{X}^+)|, |K_\theta(\boldsymbol{x}, \mathcal{X}^-) - K_{\theta'}(\boldsymbol{x}, \mathcal{X}^-)|\} \\ &\leq 2c_2 \|g\|_V |\theta - \theta'|, \qquad \text{(following from (35))}\end{aligned}$$

where $\mathcal{X}^+ = \{\boldsymbol{y} : \boldsymbol{y} \in \mathcal{X}, (K_\theta(\boldsymbol{x}, d\boldsymbol{y}) - K_{\theta'}(\boldsymbol{x}, d\boldsymbol{y})) g(\boldsymbol{y}) > 0\}$ and $\mathcal{X}^- = \mathcal{X} \setminus \mathcal{X}^+$. This implies that condition (DRI3) is satisfied by choosing $V(\boldsymbol{x}) = 1$ and $\beta = 1$. The proof is completed. □

References

Andrieu, C., Moulines, É., and Priouret, P. (2005), "Stability of Stochastic Approximation Under Verifiable Conditions," *SIAM J. Control and Optimization*, 44, 283-312.

Augustin, N., Mugglestone, M. and Buckland, S. (1996), "An Autologistic Model for Spatial Distribution of Wildlife," *Journal of Applied Ecology*, 33, 339-347.

Besag, J.E. (1974), "Spatial Interaction and the Statistical Analysis of Lattice Systems (with Discussion)," *Journal of the Royal Statistical Society, Series B*, 36, 192-236.

Besag, J.E. (1975)," Statistical Analysis of Non-Lattice Data," *The Statistician*, 24, 179-195.

Benveniste, A., Métivier, M., and Priouret, P. (1990), *Adaptive Algorithms and Stochastic Approximations*, New York: Springer-Verlag.

Berg, B.A. and Neuhaus, T. (1991), "Multicanonical Algorithms for 1st Order Phase-Transitions," *Physics Letters B*, 267, 249-253.

Cappé, O., Guillin, A., Marin, J.M. and Robert, C.P. (2004), "Population Monte Carlo," *Journal of Computational and Graphical Statistics*, 13, 907-929.

Delyon, B., Lavielle, M. and Moulines, E. (1999), "Convergence of a Stochastic Approximation Version of the EM Algorithm," *Annals of Statistics*, 27, 94-128.

Diggle, P.J. and Gratton, R.J. (1984), "Monte Carlo Methods of Inference for Implicit Statistical Models (with Discussion)," *Journal of the Royal Statistical Society, Series B*, 46, 193-227.

Efron, B. and Tibshirani, R.J. (1993), *An Introduction to the Bootstrap*, London: Chapman & Hall.

Gelfand, A.E. and Banerjee, S. (1998), "Computing Marginal Posterior Modes Using Stochastic Approximation," Technical report, University of Connecticut, Department of Statistics.

Gelman, A. and Rubin, D.B. (1992), "Inference from Iterative Simulation Using Multiple Sequences (with Discussion)," *Statistical Science*, 7, 457-472.

Geman, S. and Geman, D. (1984), "Stochastic Relaxation, Gibbs Distributions and the Bayesian Restoration of Images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6, 721-741.

Geyer, C.J. (1991), "Markov Chain Monte Carlo Maximum Likelihood," in *Computing Science and Statistics: Proceedings of the 23rd Symposium on the Interface*, eds. E.M. Keramigas, Fairfax: Interface Foundation, pp.156-163.

Geyer, C.J. (1992), "Practical Monte Carlo Markov Chain (with Discussion)," *Statistical Science*, 7, 473-511.

Geyer, C.J. (1994), "On the Convergence of Monte Carlo Maximum Likelihood Calculations," *Journal of the Royal Statistical Society, Series B*, 56, 261-274.

Geyer, C.J. (1996), "Estimation and Optimization of Functions," in *Markov Chain Monte Carlo in Practice*, eds. W.R. Gilks, S. Richardson and D.J. Spiegelhalter, London: Chapman & Hall, pp.241-258.

Geyer, C.J. and Thompson, E.A. (1992), "Constrained Monte Carlo Maximum Likelihood for Dependent Data," *Journal of the Royal Statistical Society, Series B*, 54, 657-699.

Geyer, C.J. and Thompson, E.A. (1995), "Annealing Markov Chain Monte Carlo with Applications to Ancestral Inference," *Journal of the American Statistical Association*, 90, 909-920.

Gilks, W.R., Roberts, R.O., and Sahu, S.K. (1998), "Adaptive Markov chain Monte Carlo through regeneration," *Journal of the American Statistical Association*, 93, 1045-1054.

Green, P.J. (1995), "Reversible Jump Markov Chain Monte Carlo Computation and Bayesian Model Determination," *Biometrika*, 82, 711-732.

Green, P.J. and Richardson, S. (2002), "Hidden Markov Models and Disease Mapping," *Journal of the American Statistical Association*, 97, 1055-1070.

Gu, M.G. and Kong, F.H. (1998), "A Stochastic Approximation Algorithm with Markov Chain Monte Carlo Method for Incomplete Data Estimation Problems," *Proceedings of the National Academy of Sciences USA*, 95, 7270-7274.

Gu, M.G. and Zhu, H.T. (2001), "Maximum Likelihood Estimation for Spatial Models by Markov Chain Monte Carlo Stochastic Approximation," *Journal of the Royal Statistical Society, Series B*, 63, 339-355.

Hastings, W.K. (1970), "Monte Carlo Sampling Methods Using Markov Chains and Their Applications," *Biometrika*, 57, 97-109.

Hesselbo, B. and Stinchcomble, R.B. (1995), "Monte Carlo Simulation and Global Optimization without Parameters," *Physical Review Letters*, 74, 2151-2155.

Hesterberg, T. (1995), "Weighted Average Importance Sampling and Defensive Mixture Distributions," *Technometrics*, 37, 185-194.

Honeycutt, J.D. and Thirumalai, D. (1990), "Metastability of the Folded States of Globular Proteins," *Proceedings of the National Academy of Sciences USA*, 87, 3526-3529.

Hukushima K. and Nemoto, K. (1996), "Exchange Monte Carlo Method and Application to Spin Glass Simulations," *J. Phys. Soc. Jpn.*, 65, 1604-1608.

Jobson, J.D. (1992), *Applied Multivariate Data Analysis, Vol. II: Categorical and Multivariate Methods*, New York: Springer-Verlag.

Lai, T.L. (2003), "Stochastic Approximation," *The Annals of Statistics*, 31, 391-406.

Liang, F. (2002), "Dynamically Weighted Importance Sampling in Monte Carlo Computation," *Journal of the American Statistical Association*, 97, 807-821.

Liang, F. (2003), "Use of Sequential Structure in Simulation from High Dimensional Systems," *Physical Review E*, 67, 56101-56107.

Liang, F. (2004), "Annealing Contour Monte Carlo for Structure Optimization in an Off-Lattice Protein Model," *Journal of Chemical Physics*, 120, 6756-6763.

Liang, F. (2005), "Generalized Wang-Landau Algorithm for Monte Carlo Computation," *Journal of the American Statistical Association*, 100, 1311-1327.

Liang, F. and Wong, W.H. (2001), "Real Parameter Evolutionary Monte Carlo with Applications in Bayesian Mixture Models," *Journal of the American Statistical Association*, 96, 653-666.

Liu, J.S. (2001), *Monte Carlo Strategies in Scientific Computing*, New York: Springer.

Liu, J.S., Liang, F., and Wong. W.H. (2001), "A theory for Dynamic Weighting in Monte Carlo," *Journal of the American Statistical Association*, 96, 561-573.

Marinari, E., and Parisi, G. (1992), "Simulated Tempering: A New Monte Carlo Scheme," *Europhysics Letters*, 19, 451-458.

Mengersen, K.L. and Tweedie, R.L. (1996), "Rates of Convergence of the Hastings and Metropolis Algorithms," *The Annals of Statistics*, 24, 101-121.

Metropolis, N., Rosenbluth, A.W., Rosenbluth, M.N., Teller, A.H., and Teller, E. (1953), "Equation of State Calculations by Fast Computing Machines," *Journal of Chemical Physics*, 21, 1087-1091.

Moyeed, R.A. and Baddeley, A.J. (1991), "Stochastic Approximation of the MLE for a Spatial Point Pattern," *Scand. J. Statist.*, 18, 39-50.

Neal, R.M. (2003), "Slice sampling (with Discussion)," *Annals of Statistics*, 31, 705-767.

Nevel′son, M.B. and Has′minskiĭ, R.Z. (1973), *Stochastic Approximation and Recursive Estimation*, Rhode Island: the American Mathmatical Society.

Preisler, H.K. (1993), "Modeling Spatial Patterns of Trees Attacked by Bark-Beetles," *Applied Statistics*, 42, 501-514.

Robbins, H. and Monro, S. (1951), "A Stochastic Approximation Method," *Annals of Mathematical Statistics*, 22, 400-407.

Robert, C.P. and Casella, G. (2004), *Monte Carlo Statistical Methods* (2nd edition), Springer.

Roberts, G.O. and Rosenthal, J.S. (2004), "General State Space Markov Chains and MCMC Algorithms," *Probability Surveys*, 1, 20-71.

Roberts, G.O. and Tweedie, R.L. (1996), "Geometric Convergence and Central Limit Theorems for Multidimensional Hastings and Metropolis Algorithms," *Biometrika*, **83**, 95-110.

Rosenthal, J.S. (1995), "Minorization Conditions and Convergence Rate for Markov Chain Monte Carlo," *Journal of the American Statistical Association*, 90, 558-566.

Sherman, M., Apanasovich, T.V. and Carroll, R.J. (2006), "On Estimation in Binary Autologistic Spatial Models," *Journal of Statistical Computation and Simulation*, 76, 167-179.

Stavropoulos, P. and Titterington, D.M. (2001), "Improved Particle Filters and Smoothing," in *Sequential MCMC in Practice*, eds. A. Doucet, N. deFreitas and N. Gordon, New York: Springer-Verlag, pp.295-318.

Swendsen, R.H. and Wang, J.S. (1987), "Nonuniversal Critical Dynamics in Monte Carlo Simulations," *Physical Review Letters*, 58, 86-88.

Tierney, L. (1994), "Markov Chains for Exploring Posterior Distributions (with Discussion)," *Annals of Statistics*, 22, 1701-1762.

Torrie, G.M. and Valleau, J.P. (1977), "Non-Physical Sampling Distributions in Monte Carlo Free Energy Estimation: Umbrella Sampling," *Journal of Computational Physics*, 23, 187-199.

Valleau, J.P. (1999), "Thermodynamic Scaling Methods in Monte Carlo and Their Application to Phase Equilibria," *Advances in Chemical Physics*, 105, 369-404.

Wang, F. and Landau, D.P. (2001), "Efficient, Multiple-Range Random Walk Algorithm to Calculate the Density of States," *Physical Review Letters*, 86, 2050-2053.

Warnes, G.R. (2001), "The Normal Kernel Coupler: An Adaptive Markov Chain Monte Carlo Method for Efficiently Sampling from Multi-Modal Distributions," Technical Report 395, University of Washington, Department of Statistics.

Wong, W.H. and Liang, F. (1997), "Dynamic Weighting in Monte Carlo and Optimization," *Proceedings of the National Academy of Sciences USA*, 94, 14220-14224.

Yan, Q. and de Pablo, J.J. (2003), "Fast Calculation of the Density of States of a Fluid by Monte Carlo Simulations," *Phys. Rev. Lett.*, **90**, 035701.

Younes, L. (1988), "Estimation and Annealing for Gibbsian fields," *Ann. Inst. Henri Poincaré*, 24, 269-294.

Younes, L. (1999), "On the Convergence of Markovian Stochastic Algorithms with Rapidly Decreasing Ergodicity Rates," *Stochastics and Stochastics Reports*, 65, 177-228.