

Stochastic Approximation in Monte Carlo Computation

Faming LIANG, Chuanhai LIU, and Raymond J. CARROLL

The Wang–Landau (WL) algorithm is an adaptive Markov chain Monte Carlo algorithm used to calculate the spectral density for a physical system. A remarkable feature of the WL 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 reach only a limited statistical accuracy. In this article we propose the stochastic approximation Monte Carlo (SAMC) algorithm, which overcomes the shortcomings of the WL algorithm. We establish a theorem concerning its convergence. The estimates produced by SAMC can be improved continuously as the simulation proceeds. SAMC also extends applications of the WL 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.

KEY WORDS: Importance sampling; Markov chain Monte Carlo; Model selection; Spatial autologistic model; Stochastic approximation; Wang–Landau algorithm.

1. INTRODUCTION

Suppose that we are interested in sampling from a distribution that, for convenience, we write in the following form:

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

where c is a constant and \mathcal{X} is the sample space. As known by many researchers, the Metropolis–Hastings (MH) sampler (Metropolis, Rosenbluth, Rosenbluth, Teller, and Teller 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, $-\log\{p_0(\mathbf{x})\}$ is called the energy function of the distribution.] Over the last two decades, a various advanced Monte Carlo algorithms have been proposed to overcome this problem, based mainly 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), and others. 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 in which 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). These differ 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 that carries the information of past samples.

Among the algorithms described here, the WL algorithm has received much recent attention in physics. It can be described as follows. Suppose that the sample space \mathcal{X} is finite. Let $U(\mathbf{x}) = -\log\{p_0(\mathbf{x})\}$ denote the energy function, let $\{u_1, \dots, u_m\}$ be a set of real numbers containing all possible values of $U(\mathbf{x})$, and let $g(u) = \#\{\mathbf{x}: U(\mathbf{x}) = u\}$ 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 what follows. The WL algorithm is an adaptive Markov chain Monte Carlo (MCMC) algorithm designed to estimate $\mathbf{g} = (g_1, \dots, g_m)$. Let \hat{g}_i be the working estimate of g_i . A run of the WL algorithm consists of several stages. The first stage starts with the initial estimates $\hat{g}_1 = \dots = \hat{g}_m = 1$ and a sample drawn from \mathcal{X} at random, and iterates between the following steps:

The WL algorithm.

1. Simulate a sample \mathbf{x} by a single Metropolis update with the invariant distribution $\hat{p}(\mathbf{x}) \propto 1/\hat{g}(U(\mathbf{x}))$.
2. Set $\hat{g}_i \leftarrow \hat{g}_i \delta^{I(U(\mathbf{x})=u_i)}$ for $i = 1, \dots, m$, where δ is a gain factor > 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 $\hat{g}(u)$ 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, $\log(\delta) \leq 10^{-8}$. Wang and Landau (2001) considered a histogram flat if the sampling frequency for each energy value is not $< 80\%$ of the average sampling frequency.

Faming Liang is Associate Professor, Department of Statistics, Texas A&M University, College Station, TX 77843 (E-mail: ftiang@stat.tamu.edu). Chuanhai Liu is Professor, Department of Statistics, Purdue University, West Lafayette, IN 47907 (E-mail: chuanhai@stat.purdue.edu). Raymond J. Carroll is Distinguished Professor, Department of Statistics, Texas A&M University, College Station, TX 77843 (E-mail: carroll@stat.tamu.edu). Liang's research was supported by grants from the National Science Foundation (DMS-04-05748) and the National Cancer Institute (CA104620). 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 through a grant from the National Institute of Environmental Health Sciences (P30ES09106). The authors thank Walter W. Piegorsch, the associate editor, and three referees for their suggestions and comments, which led to significant improvement of this article.

Liang (2005) generalized the WL algorithm to continuum systems. This generalization is mainly in terms of 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 $U(\mathbf{x})$, into m disjoint subregions denoted by $E_1 = \{\mathbf{x}: U(\mathbf{x}) \leq u_1\}$, $E_2 = \{\mathbf{x}: u_1 < U(\mathbf{x}) \leq u_2\}$, \dots , $E_{m-1} = \{\mathbf{x}: u_{m-2} < U(\mathbf{x}) \leq u_{m-1}\}$, and $E_m = \{\mathbf{x}: U(\mathbf{x}) > u_{m-1}\}$, where u_1, \dots, u_{m-1} are $m-1$ specified real numbers. Let $\psi(\mathbf{x})$ be a nonnegative function defined on the sample space with $0 < \int_{\mathcal{X}} \psi(\mathbf{x}) d\mathbf{x} < \infty$. In practice, $\psi(\mathbf{x})$ is often set to $p_0(\mathbf{x})$ defined in (1). Let $g_i = \int_{E_i} \psi(\mathbf{x}) d\mathbf{x}$. One iteration of GWL consists of the following steps:

The GWL algorithm.

1. Simulate a sample \mathbf{x} by a number, κ , of MH steps of which the invariant distribution is defined as

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

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

The extension of g_i from the density of states to the integral $\int_{E_i} \psi(\mathbf{x}) d\mathbf{x}$ is of great interest to statisticians, because this leads to direct applications of the algorithm to model selection highest posterior density (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 de Pablo (2003) noted that estimates of g_i can only reach a limited statistical accuracy that will not be improved with further iterations, and the large number of configurations generated toward 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 let δ_s be the gain factor used in stage s . Let $n_1 = \dots = n_s = \dots = n$, where n is large enough such that a flat histogram can be reached in each stage. Let $\log \delta_s = \frac{1}{2} \log \delta_{s-1}$ decrease 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 that the tail sum represents the total correction to the current estimate in the iterations that follow. Hence the numerous configurations generated toward 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 in the total number of iterations required by the simulation.

In this article we propose a stochastic approximation Monte Carlo (SAMC) algorithm, which can be considered 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 \mathbf{g} can be improved continuously as the simulation proceeds. 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 must be sampled equally. It 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 difficult optimization problems. In addition, 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 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, and in Section 5 we discuss the use of SAMC in model selection. In Section 6 we conclude the article with a brief discussion.

2. STOCHASTIC APPROXIMATION MONTE CARLO

Consider the distribution defined in (1). For 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 $\{\mathbf{x}: p_0(\mathbf{x}) \geq p_{\min}\}$, where p_{\min} is sufficiently small such that the region $\{\mathbf{x}: p_0(\mathbf{x}) < p_{\min}\}$ is not of interest. As in GWL, we let E_1, \dots, E_m denote m disjoint regions that form a partition of \mathcal{X} . In practice, $\sup_{\mathbf{x} \in \mathcal{X}} p_0(\mathbf{x})$ 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 $g_i = \int_{E_i} \psi(\mathbf{x}) d\mathbf{x} = 0$. 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 $\theta_{ii} = \log(\widehat{g}_i^{(t)})$ and $\theta_t = (\theta_{t1}, \dots, \theta_{tm})$. The distribution (2) can then be rewritten as

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

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 $\Theta = \mathbb{R}^m$. Because $p_{\theta_t}(\mathbf{x})$ is invariant with respect to a location transformation of θ_t —that is, adding to or subtracting a constant vector from θ_t will not change $p_{\theta_t}(\mathbf{x})$ — θ_t can be kept in the compact set in simulations by adjusting with a constant vector. Because \mathcal{X} and Θ are both assumed to be compact, an additional assumption is that $p_{\theta_t}(\mathbf{x})$ is bounded away from 0 and ∞ on \mathcal{X} . Let the proposal distribution $q(\mathbf{x}, \mathbf{y})$ satisfy the following condition: For every $x \in \mathcal{X}$, there exist $\epsilon_1 > 0$ and $\epsilon_2 > 0$ such that

$$\|\mathbf{x} - \mathbf{y}\| \leq \epsilon_1 \implies q(\mathbf{x}, \mathbf{y}) \geq \epsilon_2. \quad (4)$$

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

Let $\pi = (\pi_1, \dots, \pi_m)$ 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, π is called the desired sampling distribution. Let $\{\gamma_t\}$ be a positive, nondecreasing sequence satisfying

$$(a) \quad \sum_{t=1}^{\infty} \gamma_t = \infty \quad \text{and} \quad (b) \quad \sum_{t=1}^{\infty} \gamma_t^\zeta < \infty \quad (5)$$

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

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

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

The SAMC algorithm.

1. Simulate a sample $\mathbf{x}^{(t+1)}$ by a single MH update, of which the proposal distribution is $q(\mathbf{x}^{(t)}, \cdot)$ and the invariant distribution is $p_{\theta_t}(\mathbf{x})$.
2. Set $\theta^* = \theta_t + \gamma_{t+1}(\mathbf{e}_{t+1} - \pi)$, where $\mathbf{e}_{t+1} = (e_{t+1,1}, \dots, e_{t+1,m})$ and $e_{t+1,i} = 1$ if $\mathbf{x}^{(t)} \in E_i$ and 0 otherwise. If $\theta^* \in \Theta$, then set $\theta_{t+1} = \theta^*$; otherwise, set $\theta_{t+1} = \theta^* + \mathbf{c}^*$, where $\mathbf{c}^* = (c^*, \dots, c^*)$ can be an arbitrary vector that satisfies the condition $\theta^* + \mathbf{c}^* \in \Theta$.

Remark. The explanation for 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) (assuming that adjustment of θ_t does not occur), $\sum_{t=1}^{\infty} |\theta_{t+1,i} - \theta_{t,i}| \leq \sum_{t=1}^{\infty} \gamma_t |e_{t,i} - \pi_i| \leq \sum_{t=1}^{\infty} \gamma_t < \infty$, where the second inequality follows from the fact $0 \leq e_{t,i}, \pi_i \leq 1$. Thus the value of $\theta_{t,i}$ does not reach $\log(g_i)$ if, for example, the initial point $\theta_{0,i}$ is sufficiently far away from $\log(g_i)$. On the other hand, γ_t cannot be too large; an overly 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 \mathbf{e}_t . When it holds, we have $\gamma_t |e_{t,i} - \pi_i| \leq \gamma_t \rightarrow 0$ as $t \rightarrow \infty$.

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

$$\theta_{t,i} \rightarrow \begin{cases} C + \log\left(\int_{E_i} \psi(\mathbf{x}) d\mathbf{x}\right) - \log(\pi_i + d) & \text{if } E_i \neq \emptyset \\ -\infty & \text{if } E_i = \emptyset, \end{cases} \quad (7)$$

as $t \rightarrow \infty$, where $d = \sum_{j \in \{i: E_i = \emptyset\}} \pi_j / (m - m_0)$, m_0 is the number of empty subregions, and C is an arbitrary constant. Because $p_{\theta_t}(\mathbf{x})$ is invariant with respect to a location transformation of θ_t , C cannot be determined by the samples drawn from $p_{\theta_t}(\mathbf{x})$. To

determine the value of C , extra information is needed; for example, $\sum_{i=1}^m e^{\theta_{t,i}}$ is equal to a known number. Let $\hat{\pi}_{t,i}$ denote the realized sampling frequency of the subregion E_i at iteration t . As $t \rightarrow \infty$, $\hat{\pi}_{t,i}$ converges to $\pi_i + d$ if $E_i \neq \emptyset$ and 0 otherwise. Note that for a nonempty subregion, the sampling frequency is independent of its probability, $\int_{E_i} p(\mathbf{x}) d\mathbf{x}$. 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 nonempty subregions (if each subregion is considered a “point”) with the sampling frequency of each nonempty subregion being proportional to $\pi_i + d$.

The subject area 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 has also served as a prototype for the development of recursive algorithms for on-line estimation and control of stochastic systems. (See Lai 2003 for an overview.) Recently, it has been used with MCMC to solve 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 lies in 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 HPD interval construction, normalizing constant estimation, and other problems, as discussed by 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. We will explore this issue further elsewhere. It is noteworthy that Geyer and Thompson (1995) and Geyer (1996) mentioned that stochastic approximation can be used to determine the “pseudopriors” for simulated tempering (i.e., determining the normalizing constants of a sequence of distributions scaled by temperature), although they provided no details. In Geyer's applications, the sample space is partitioned automatically according to the temperature variable.

For effective implementation of SAMC, several issues must be considered:

- *Sample space partition.* This can be done according to our 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 Sec. 4) or to minimize the energy function, then 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, to ensure 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 $\psi(\mathbf{x})$. If our goal is model selection, then the sample space can be partitioned according to the index of models, as illustrated in Section 5.

- *The desired sampling distribution.* If we aim to estimate \mathbf{g} , then we may set the desired distribution to be uniform, as is done in all examples in this article. However, if our goal is optimization, then we 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, we will report these results elsewhere.
- *Choice of t_0 and the number of iterations.* To estimate \mathbf{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 0 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, the larger the value of t_0 that should be chosen. 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 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 determined by checking the convergence of multiple runs (starting with different points) through examining for the variation of $\widehat{\mathbf{g}}$ or $\widehat{\pi}$, where $\widehat{\mathbf{g}}$ and $\widehat{\pi}$ denote the estimates of \mathbf{g} and π obtained at the end of a run. A rough examination for $\widehat{\mathbf{g}}$ is to check visually whether or not the $\widehat{\mathbf{g}}$ vectors produced in the multiple runs follow the same pattern. Existence of different patterns implies that the gain factor is still large at the end of the runs, or that some parts of the sample space are not visited in all runs. The examination for $\widehat{\mathbf{g}}$ can also be done by a statistical test under the assumption of multivariate normality. (See Jobson 1992, pp. 150–153, for the testing methods for multivariate outliers.)

To examine the variation of $\widehat{\pi}$, 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

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

for $i = 1, \dots, m$, 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 that are not visited. Note that \widehat{d} can be considered an estimate of d in (7). It is said that $\{\epsilon_f(E_i)\}$, output from all runs and for all subregions, matches well if the following two conditions are satisfied: (a) There does not exist such a subregion that is visited in some runs but not in others, and (b) $\max_{i=1}^m |\epsilon_f(E_i)|$ is less than a threshold value, say 10%, for all runs. A group of $\{\epsilon_f(E_i)\}$ that 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 by Gelman and Rubin (1992) and Geyer (1992).

In practice, to have a reliable diagnostic for the convergence, we may check both $\widehat{\mathbf{g}}$ and $\widehat{\pi}$. In the case where a failure of

multiple-run convergence is detected, SAMC should be rerun with more iterations or a larger value of t_0 . Determining t_0 and the number of iterations is a trial-and-error process.

3. TWO DEMONSTRATION EXAMPLES

Example 1. In this example we compare the convergence and efficiency of the 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 that 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 \mathbf{g} is then $\mathbf{g} = (1, 1, 2, 2, 4)$, 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 $Dir(1, \dots, 1)$. The desired sampling distribution is uniform, that is, $\pi_1 = \dots = \pi_5 = 1/5$. 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 \mathbf{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, $t = 5 \times 10^4, \dots, 5 \times 10^5$. 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 boxplots of the $\epsilon_f(E_i)$'s of the 100 runs. The deviations are $< 3\%$. 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 , including $t_0 = 20$ and 30 , were also tried, and the results were similar.

We applied the WL algorithm to this example with the same proposal distribution as that used in SAMC. In the runs, the gain factor was set as done by Wang and Landau (2001); it started with $\delta_0 = 2.718$ and then decreased in the scheme $\delta_{s+1} \rightarrow \sqrt{\delta_s}$. Let n_s denote the number of iterations performed in stage s . For simplicity, we set n_s to a constant that was large enough such that a flat histogram can be formed in each stage. The choices of n_s that we tried included $n_s = 1,000, 2,500, 5,000$, and $10,000$. The estimation error was also measured by $\epsilon_e(t)$ evaluated at $t = 5 \times 10^4, \dots, 5 \times 10^5$, 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 proceeds, whereas in WL the estimates can reach only a certain accuracy depending on the value of n_s .

Example 2. As pointed out by Liu (2001), umbrella sampling (Torrie and Valleau 1977) can be seen as a precursor of

Table 1. The Unnormalized Mass Function of the 10-State Distribution

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

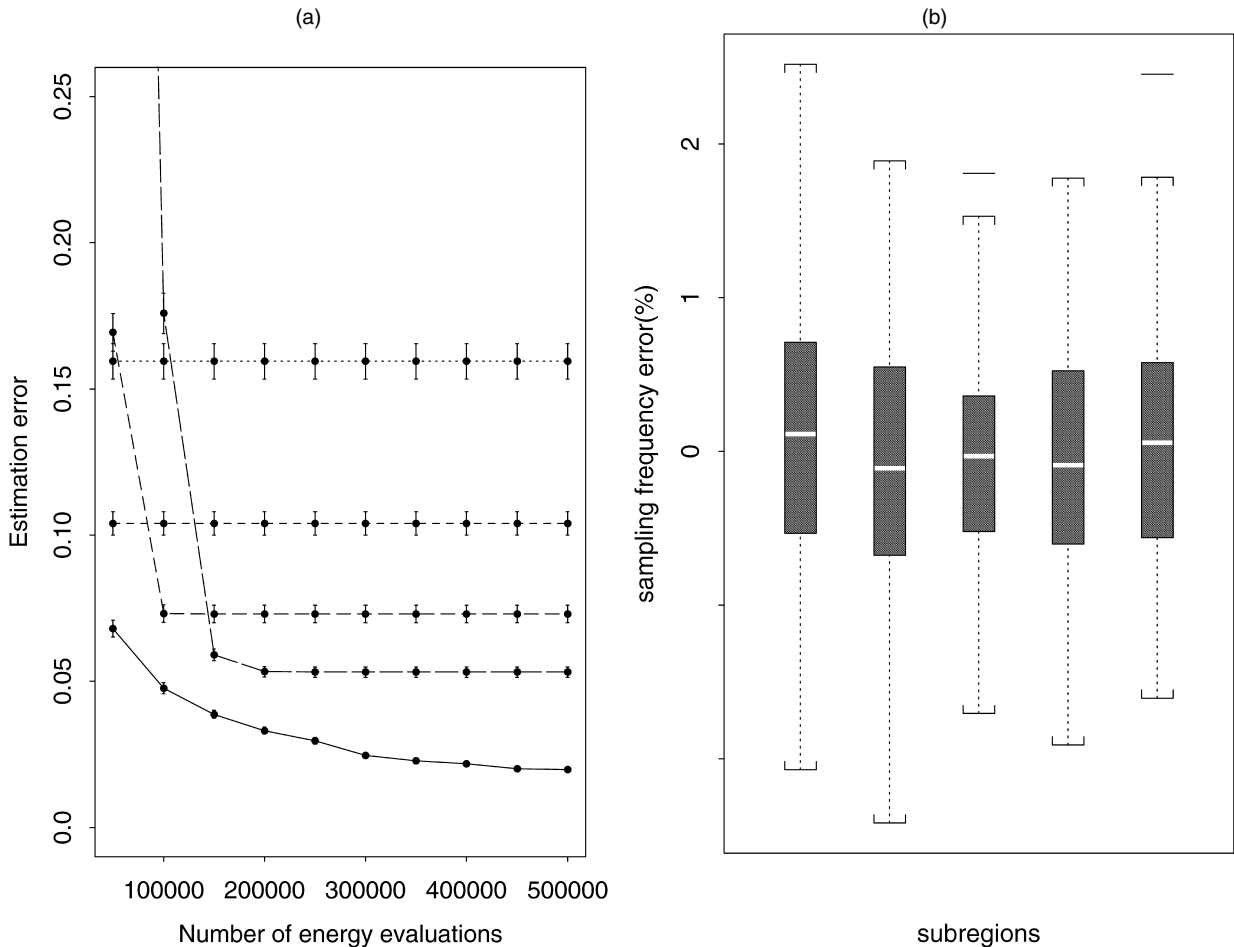


Figure 1. Comparison of the WL and SAMC Algorithms. (a) Average $\epsilon_e(t)$ curves obtained by SAMC and WL. The vertical bars show the ± 1 standard deviation of the average of the estimates (— SAMC; WL, $n = 1,000$; ---- WL, $n = 2,500$; - · - WL, $n = 5,000$; — · — WL, $n = 10,000$). (b) Boxplots of $\{\epsilon_i(E_i)\}$ obtained in 100 runs of SAMC.

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 also can be used as a general importance sampling method. Recall that the basic idea of umbrella sampling is to sample from an “umbrella distribution” (i.e., a trial distribution 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 $p_{st}^{(i)}(\mathbf{x}) \propto [p_0(\mathbf{x})]^{1/T_i}$ for $T_m > T_{m-1} > \dots > T_1 = 1$. This leads directly to the simulated tempering algorithm. The other one is to sample a weighted distribution $p_u(\mathbf{x}) \propto \omega\{U(\mathbf{x})\}p_0(\mathbf{x})$, 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, motivated by discretization of continuum systems, provides a new methodology for applying umbrella sampling to continuum systems.

Although SAMC and simulated tempering both fall in the class of umbrella sampling algorithms, they have quite different dynamics. This can be illustrated by an example. The distribution is defined as $p(\mathbf{x}) \propto e^{-U(\mathbf{x})}$, where $\mathbf{x} \in [-1.1, 1.1]^2$ and

$U(\mathbf{x}) = -\{x_1 \sin(20x_2) + x_2 \sin(20x_1)\}^2 \cosh\{\sin(10x_1)x_1\} - \{x_1 \cos(10x_2) - x_2 \sin(10x_1)\}^2 \cosh\{\cos(20x_2)x_2\}$. This example is modified from example 5.3 of Robert and Casella (2004). Figure 2(a) shows that $U(\mathbf{x})$ has a multitude of local energy minima separated by high-energy barriers. When applying SAMC to this example, we partitioned the sample space into 41 subregions with an equal energy bandwidth, $E_1 = \{\mathbf{x} : U(\mathbf{x}) \leq -8.0\}$, $E_2 = \{\mathbf{x} : -8.0 < U(\mathbf{x}) \leq -7.8\}$, ..., and $E_{41} = \{\mathbf{x} : -0.2 < U(\mathbf{x}) \leq 0\}$, and set other parameters as follows: $\psi(\mathbf{x}) = e^{-U(\mathbf{x})}$, $t_0 = 200$, $\pi_1 = \dots = \pi_{41} = 1/41$, and a random-walk proposal, $q(\mathbf{x}_t, \cdot) = N_2(\mathbf{x}_t, .25^2 I_2)$. SAMC was run for 20,000 iterations, and 2,000 samples were collected at equally spaced time points. Figure 2(b) shows the evolving path of the 2,000 samples. For comparison, MH was applied to simulate from the distribution $p_{st}(\mathbf{x}) \propto e^{-U(\mathbf{x})/5}$. MH was run for 20,000 iterations with the same proposal $N_2(\mathbf{x}_t, .25^2 I_2)$, and 2,000 samples were collected at equally spaced time points. Figure 2(c) shows the evolving path of the 2,000 samples, which characterizes the performance of simulated tempering at high temperatures.

The result is clear. In the foregoing setting, SAMC samples almost uniformly in the space of energy [i.e., the energy bandwidth of each subregion is small, and the sample distribution closely matches the contour plot of $U(\mathbf{x})$], whereas simulated

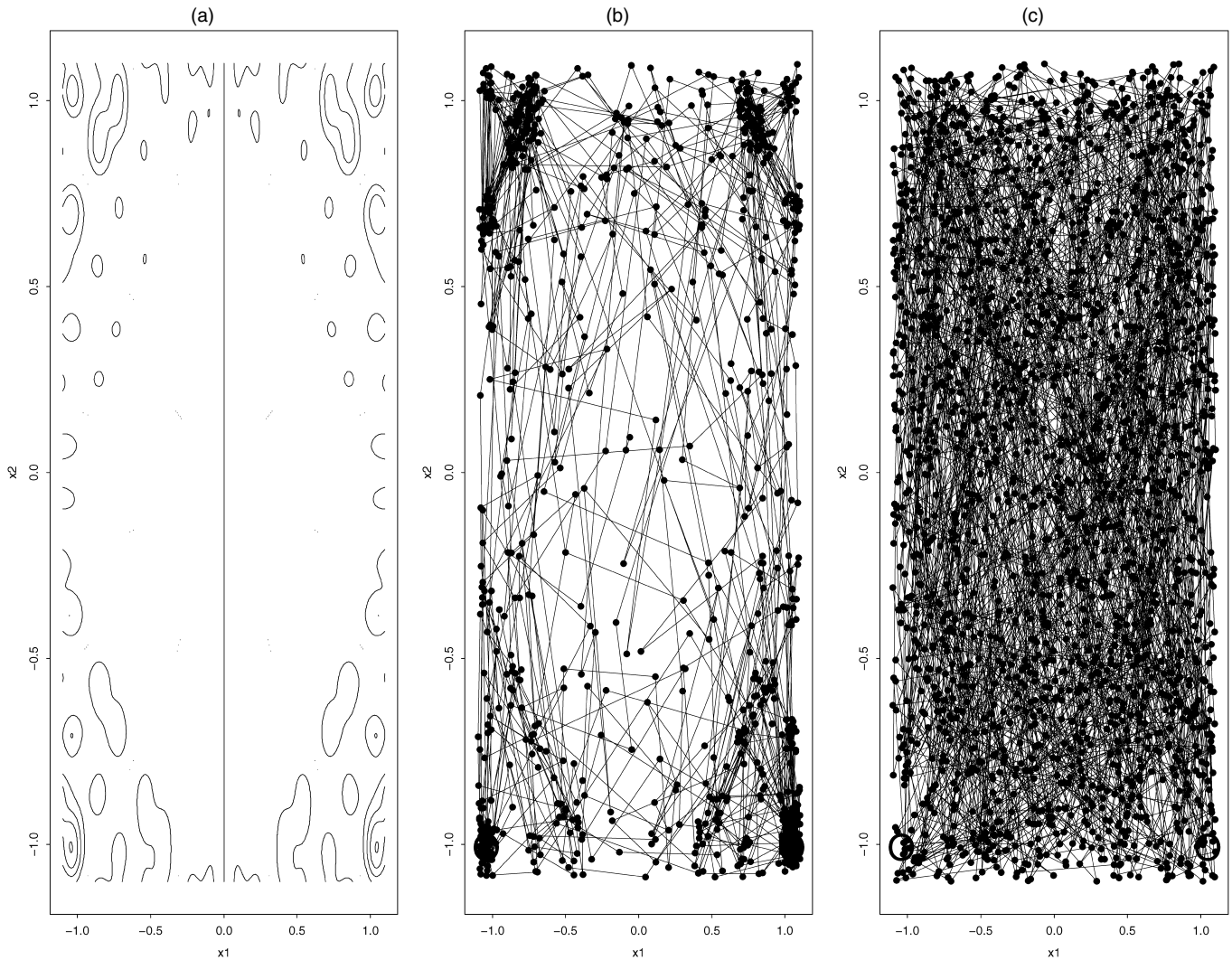


Figure 2. (a) Contour of $U(\mathbf{x})$, (b) Sample Path of SAMC, and (c) Sample Path of MH.

tempering tends to sample uniformly in the sample space \mathcal{X} when the temperature is high. Because we usually do not know where the high-energy and low-energy regions are and how much the ratio of their “volumes” is a priori, we cannot control the simulation time spent on low-energy 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-energy and high-energy regions in SAMC by choosing the desired sampling distribution π . SAMC can go to high-energy regions, but it spends only limited time there to help the system to escape from local energy minima, and also spends time 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, we do not explore this point in this article. But we note that Liang (2005) reported a neural network training example in which it was shown GWL is more efficient than simulated tempering in locating global energy minima.

4. USE OF STOCHASTIC APPROXIMATION MONTE CARLO IN IMPORTANCE SAMPLING

In this section we illustrate the use of SAMC as an importance sampling method. Suppose that because of its rugged en-

ergy landscape, the target distribution $p(\mathbf{x})$ is very difficult to simulate from with conventional Monte Carlo algorithms. Let $\mathbf{x}_1, \dots, \mathbf{x}_n$ denote the samples drawn from a trial density $p^*(\mathbf{x})$, and let w_1, \dots, w_n denote the associated importance weights, where $w_i = p(\mathbf{x}_i)/p^*(\mathbf{x}_i)$ for $i = 1, \dots, n$. The quantity $E_p h(\mathbf{x})$ then can be estimated by

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

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

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

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

- The importance weight is bounded; that is, there exists a number M such that $p(\mathbf{x})/p^*(\mathbf{x}) < M$ for all $\mathbf{x} \in \mathcal{X}$.
- The trial density $p^*(\mathbf{x})$ 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. Of course, how to specify an appropriate trial density for a general distribution has been a long-standing and difficult problem in statistics.

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

$$p^*(\mathbf{x}) = \lambda p(\mathbf{x}) + (1 - \lambda)\tilde{p}(\mathbf{x}), \quad (10)$$

where $0 < \lambda < 1$ and $\tilde{p}(\mathbf{x})$ is another density. However, in practice, $p^*(\mathbf{x})$ is rather poor. Although the resulting importance weights are bounded above by $1/\lambda$, it cannot be easily sampled from using conventional Monte Carlo algorithms. Because $p^*(\mathbf{x})$ contains $p(\mathbf{x})$ as a component, if we can sample from $p^*(\mathbf{x})$, then we can also sample from $p(\mathbf{x})$. In this case we do not need to use importance sampling! Stavropoulos and Titterton (2001), Warnes (2001), and Cappé, Giullin, Marin, and Robert (2004) suggested constructing the trial density based on previous Monte Carlo samples, but the trial densities resulting from their methods cannot 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 also may 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 that 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 foregoing 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

$$\hat{p}_\infty(\mathbf{x}) \propto \sum_{i=1}^m \frac{\psi(\mathbf{x})}{\hat{g}_i} I(\mathbf{x} \in E_i) \quad (11)$$

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

Besides satisfying the conditions (a) and (b), $\hat{p}_\infty(\mathbf{x})$ 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:

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

where the parameters λ_i , $i = 1, \dots, m$, control the sampling frequency of the subregions. Second, resampling can be made online if we are interested in generating equally weighted samples from $p(\mathbf{x})$. Let $\omega_i = \hat{g}_i / \max_{j=1}^m \hat{g}_j$ denote the resampling probability from the subregion E_i , and \mathbf{x}_t denote the sample drawn

from $\hat{p}_\infty(\mathbf{x})$ at iteration t . The resampling procedure consists of the following three steps:

The SAMC importance-resampling algorithm.

1. Draw a sample $\mathbf{x}_t \sim \hat{p}_\infty(\mathbf{x})$ using a conventional Monte Carlo algorithm, say, the MH algorithm.
2. Draw a random number $U \sim \text{uniform}(0, 1)$. If $U < \omega_k$, then save \mathbf{x}_t as a sample of $p(\mathbf{x})$, where k is the index of the subregion \mathbf{x}_t belongs to.
3. Set $t \leftarrow t + 1$ and go to step 1, until sufficient samples have been collected.

Consider the following distribution:

$$p(\mathbf{x}) = \frac{1}{3} \text{N} \left[\begin{pmatrix} -8 \\ -8 \end{pmatrix}, \begin{pmatrix} 1 & .9 \\ .9 & 1 \end{pmatrix} \right] \\ + \frac{1}{3} \text{N} \left[\begin{pmatrix} 6 \\ 6 \end{pmatrix}, \begin{pmatrix} 1 & -.9 \\ -.9 & 1 \end{pmatrix} \right] \\ + \frac{1}{3} \text{N} \left[\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \right],$$

which is identical to that given by 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 used to simulate from $p(\mathbf{x})$ with a random-walk proposal $\text{N}(\mathbf{x}, I_2)$, but it failed to mix the three components. However, an advanced MCMC sampler, such as simulated tempering, parallel tempering, or evolutionary Monte Carlo, should work well for this example. Our purpose in studying this example was 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 multimodal distribution.

We applied SAMC to this example with the same proposal as used in the MH algorithm. Let $\mathcal{X} = [-10^{100}, 10^{100}]^2$ be compact. It was partitioned with an equal energy bandwidth $\Delta u = 2$ into the following subregions: $E_1 = \{\mathbf{x} : -\log p(\mathbf{x}) < 0\}$, $E_2 = \{\mathbf{x} : 0 \leq -\log p(\mathbf{x}) < 2\}$, \dots , and $E_{12} = \{\mathbf{x} : -\log p(\mathbf{x}) > 20\}$. Set $\psi(\mathbf{x}) = p(\mathbf{x})$, $t_0 = 50$ 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 \hat{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 $p(\mathbf{x})$. Figure 3(c) shows the sample path of the first 500 samples generated by the algorithm. All three components had 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 multimodal distribution.

5. USE OF STOCHASTIC APPROXIMATION MONTE CARLO 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

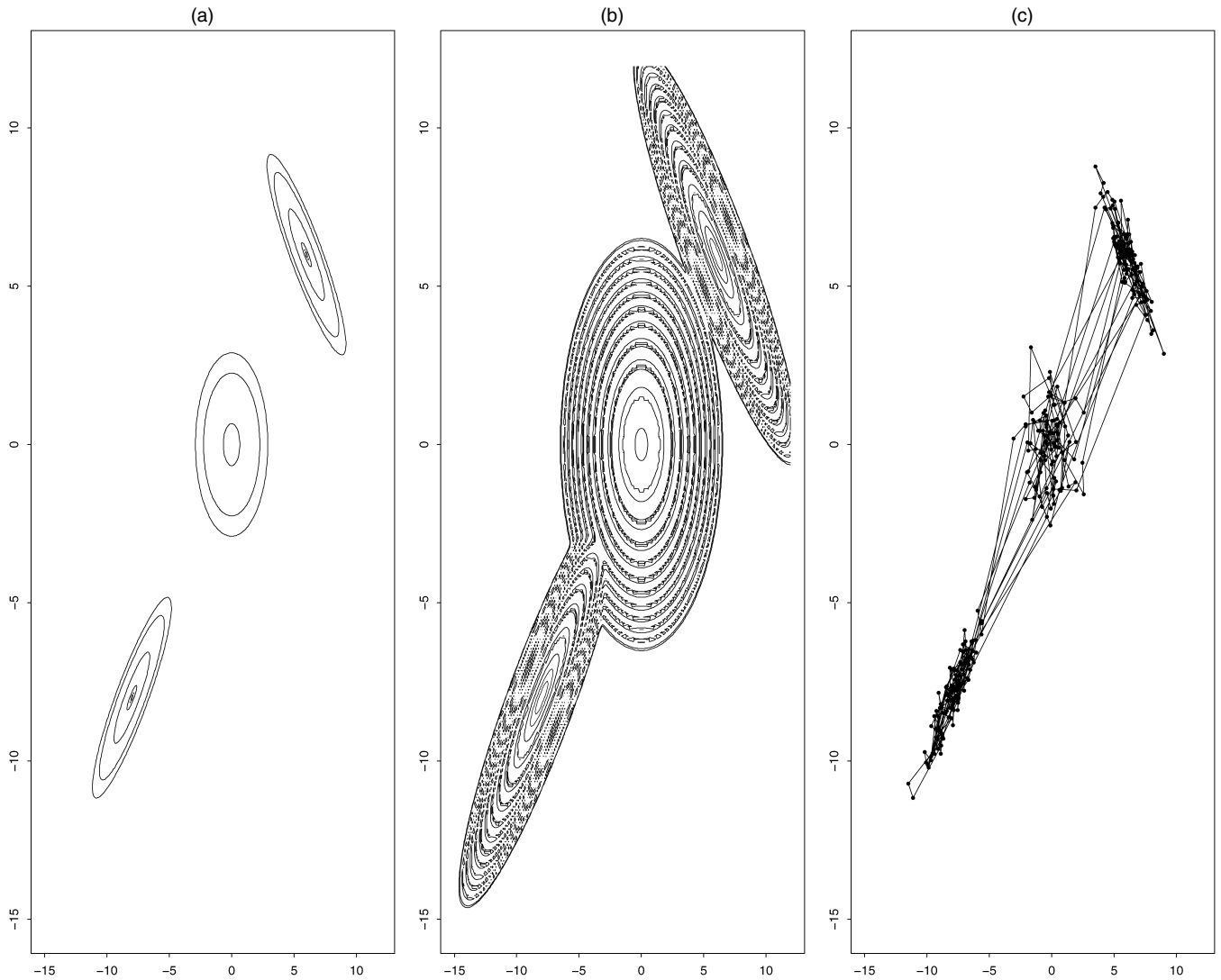


Figure 3. Computational Results for the Mixture Gaussian Example. (a) and (b) Contour plots of the true and trial densities. The contour lines correspond to 99%, 95%, 50%, 5%, and 1% of the total mass. (c) The path of the first 500 samples simulated from $p(\mathbf{x})$ by the SAMC importance-sampling algorithm.

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 that 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 $i = 1, \dots, m$, and $\psi(\cdot) \propto f(M, \vartheta_M|D)$, it follows from (7) that $\hat{g}_i^{(t)}/\hat{g}_j^{(t)} = e^{\theta_{ii}-\theta_{ij}}$ forms a consistent estimator for the Bayes factor of the models M_i and M_j , $1 \leq i, j \leq m$. We note that reversible-jump MCMC (RJMCMC) (Green 1995) can also estimate the Bayes factors of m models simultaneously. For comparison, in what follows we give explicitly the iterative procedures of the two methods for Bayesian model selection.

Let $Q(M_i \rightarrow M_j)$ denote the proposal probability for a transition from model M_i to model M_j , and $T(\vartheta_{M_i} \rightarrow \vartheta_{M_j})$ 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 . One iteration of SAMC comprises of the following steps:

The SAMC model-selection algorithm.

1. Generate model M^* according to the proposal matrix Q .
2. If $M^* = M^{(t)}$, then simulate a sample ϑ^* from $f(\vartheta_{M^{(t)}}|M^{(t)}, D)$ by a single MCMC iteration and set $(M^{(t+1)}, \vartheta^{(t+1)}) = (M^*, \vartheta^*)$.
3. If $M^* \neq M^{(t)}$, then generate ϑ^* according to the proposal distribution T and accept the sample (M^*, ϑ^*) with probability

$$\min \left\{ 1, \frac{e^{\theta_{t,M^{(t)}}} f(M^*, \vartheta^*|D)}{e^{\theta_{t,M^*}} f(M^{(t)}, \vartheta^{(t)}|D)} \times \frac{Q(M^* \rightarrow M^{(t)}) T(\vartheta^* \rightarrow \vartheta^{(t)})}{Q(M^{(t)} \rightarrow M^*) T(\vartheta^{(t)} \rightarrow \vartheta^*)} \right\}. \quad (13)$$

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

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

Let $\Xi_i^{(t)} = \#\{M^{(k)} = M_i : k = 1, 2, \dots, t\}$ 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.

1. Generate model M^* according to the proposal matrix Q .
2. If $M^* = M^{(t)}$, then simulate a sample ϑ^* from $f(\vartheta_{M^*} | M^{(t)}, D)$ by a single MCMC iteration and set $(M^{(t+1)}, \vartheta^{(t+1)}) = (M^*, \vartheta^*)$.
3. If $M^* \neq M^{(t)}$, then generate ϑ^* according to the proposal density T and accept the sample (M^*, ϑ^*) with probability

$$\min \left\{ 1, \frac{f(M^*, \vartheta^* | D)}{f(M^{(t)}, \vartheta^{(t)} | D)} \times \frac{Q(M^* \rightarrow M^{(t)}) T(\vartheta^* \rightarrow \vartheta^{(t)})}{Q(M^{(t)} \rightarrow M^*) T(\vartheta^{(t)} \rightarrow \vartheta^*)} \right\}. \quad (14)$$

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

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

The standard MCMC theory (Tierney 1994) implies that as $t \rightarrow \infty$, $\Xi_i^{(t)} / \Xi_j^{(t)}$ 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 just described is not the most general one, where the proposal distribution $T(\cdot \rightarrow \cdot)$ is assumed such that the Jacobian term in (14) is reduced to 1. This observation is also applicable to the SAMC model-selection algorithm. The MCMC algorithm used in step 2 of the foregoing 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 Sec. 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 different only at steps 3 and 4, that is, in the manner 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 convergence of the algorithm. In simulations, we can see that SAMC can overcome any difficulties in dimension-jumping moves and provide a full exploration for all models. Recall that the proposal distributions have been assumed to satisfy the condition (4). Because RJMCMC does not have 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 it possible for SAMC to work for a group of models with huge differences in probability. This is beyond the ability of RJMCMC, which can work only for a group of models with comparable probabilities.

Finally, we point out that for a problem that contains only several models with comparable probabilities, SAMC may not be better than RJMCMC, because in this case its self-adjusting ability is no longer crucial for mixing of the models. SAMC is essentially an importance sampling method (i.e., the samples are not equally weighted); hence its efficiency should be lower than that of RJMCMC for a problem in which RJMCMC succeeds. In summary, we suggest using SAMC when the model space is complex, for example, when the distribution $f(M|D)$ has well-separated multiple modes or when there are probability models that are tiny 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; Augustin, Muggleston, and Buckland 1996). Let $\mathbf{s} = \{s_i : i \in D\}$ denote a configuration of the model in which 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 let $N(i)$ denote a set of the neighbors of spin i . The probability mass function of the model is

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

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

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

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, because it requires summary over all $2^{|D|}$ possible realizations of \mathbf{s} . Because $\varphi(\alpha, \beta)$ is unknown, importance sampling is perhaps the most convenient technique if we aim at calculating the expectation $E_{\alpha, \beta} h(\mathbf{s})$ over the parameter space. This problem is a little different than 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

$$p_{\text{mix}}^*(\mathbf{s}) = \frac{1}{m^*} \sum_{j=1}^{m^*} p(\mathbf{s} | \alpha_j, \beta_j), \quad (16)$$

where the values of the parameters $(\alpha_1, \beta_1), \dots, (\alpha_{m^*}, \beta_{m^*})$ are prespecified. We note that this idea has been suggested

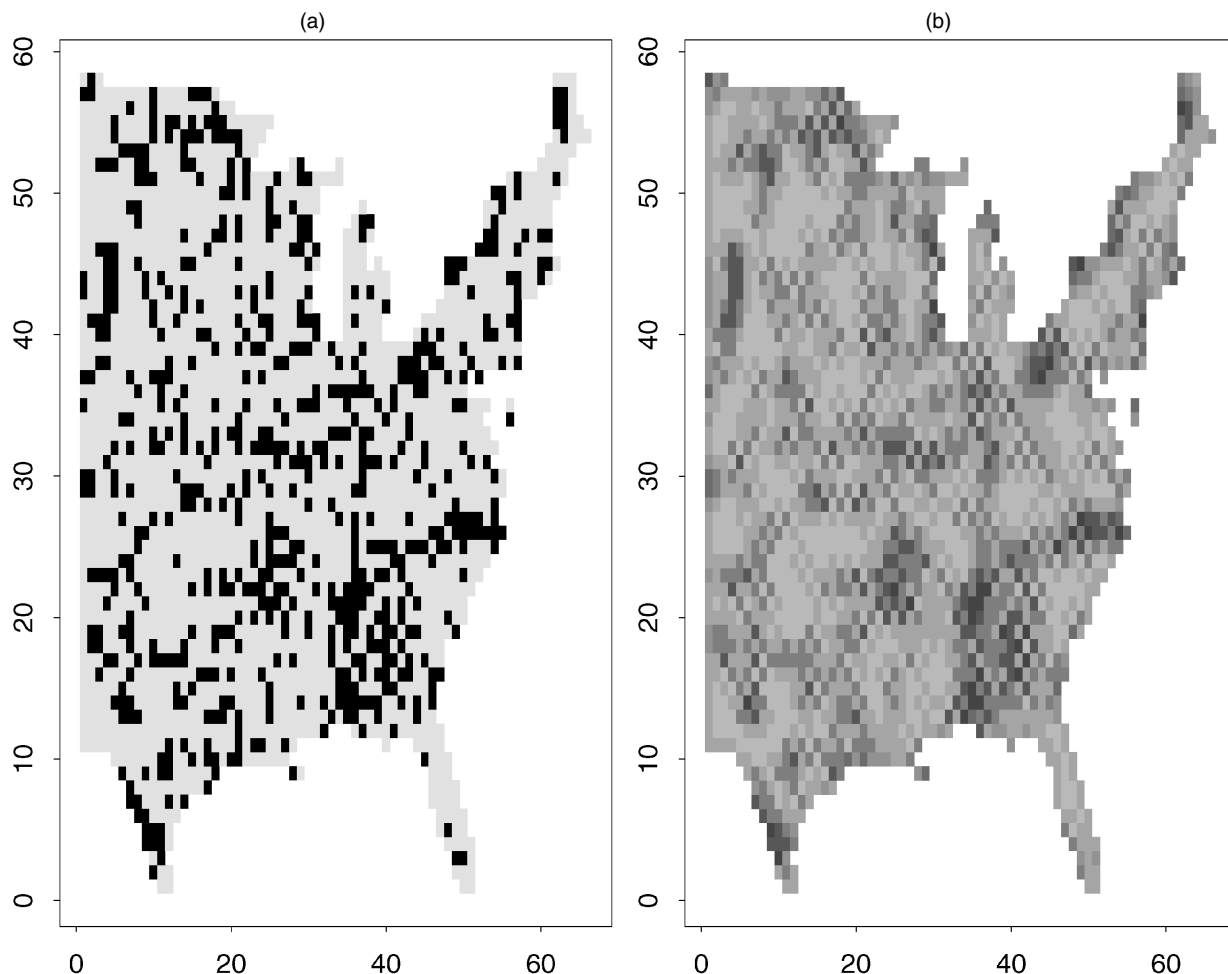


Figure 4. U.S. Cancer Mortality Rate Data. (a) The mortality map of liver and gallbladder cancer (including bile ducts) for white males during the decade 1950–1959. The black squares denote the counties of high cancer mortality rate, and the white squares denote the counties of low cancer mortality rate.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

by Geyer (1996). To complete this idea, the key is to estimate $\varphi(\alpha_j, \beta_j), \dots, \varphi(\alpha_{m^*}, \beta_{m^*})$. The estimation can be up to a common multiplicative constant that will be canceled out in calculation of $E_{\alpha, \beta} h(\mathbf{s})$ in (9). Geyer (1996) also suggested stochastic approximation as a feasible method for simultaneously estimating $\varphi(\alpha_1, \beta_1), \dots, \varphi(\alpha_{m^*}, \beta_{m^*})$, 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 maximum likelihood estimators (MLEs) of the parameters. A similar idea was also been proposed by Green and Richardson (2002) for analyzing a disease-mapping example.

In this article we explore the idea of Geyer (1996) and give details about how SAMC can be used to simultaneously estimate $\varphi(\alpha_1, \beta_1), \dots, \varphi(\alpha_{m^*}, \beta_{m^*})$ 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 $|D| = 2,293$. Suppose that the parameter points used in (16) form a 21×11 lattice ($m^* = 231$) with α equally spaced between $-.5$ and $.5$ and β between 0 and $.5$. Because $\varphi(\alpha, \beta)$ is a symmetric function about α , we only need to estimate it on a sublattice with α between 0 and $.5$. The sublattice consists of $m = 121$ points. Estimating the quantities $\varphi(\alpha_1, \beta_1), \dots, \varphi(\alpha_m, \beta_m)$ can be treated as a Bayesian model selection problem, although no observed data are involved. This is because $\varphi(\alpha_1, \beta_1), \dots, \varphi(\alpha_m, \beta_m)$ correspond to the normalizing constants of different distributions. In what follows, 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 $p(\mathbf{s}|\alpha_j, \beta_j)\varphi(\alpha_j, \beta_j)$ as the posterior distribution used in (13) and (14).

SAMC was first applied to this problem. The proposal matrix Q , the proposal distribution T , and the MCMC sampler used in step 2 are specified as follows. Let the m models be coded as a matrix (M_{ij}) with $i = 0, \dots, 10$ and $j = 0, \dots, 10$. The proposal matrix Q is then defined as

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

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

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

$$P(s_i = -1|N(i)) = 1 - P(s_i = +1|N(i)),$$

for all $i \in D$ in a prespecified order.

SAMC was run five times independently. Each run consisted of two stages. The first stage estimated 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 drew importance samples from the trial distribution,

$$\hat{p}_{\text{mix}}^*(\mathbf{s}) \propto \frac{1}{m^*} \sum_{k=1}^{m^*} \frac{1}{\hat{\varphi}(\alpha_k, \beta_k)} \times \exp \left\{ \alpha_k \sum_{i \in D} s_i + \frac{\beta_k}{2} \sum_{i \in D} s_i \left(\sum_{j \in N(i)} s_j \right) \right\}, \quad (18)$$

which represents an approximation to (16), with $\varphi(\alpha_j, \beta_j)$ 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 115 minutes of CPU time in a 2.8-GHz 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 earlier, 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 $h(\mathbf{s}) = \sum_{i \in D} (s_i +$

$1)/(2|D|)$. 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.

Using the importance samples collected earlier, 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 $p^*(\mathbf{s}) = c^* p_0^*(\mathbf{s})$ denote an arbitrary trial distribution for (15), where $p_0^*(\mathbf{s})$ is completely specified and c^* is an unknown constant. Let $\psi(\alpha, \beta, \mathbf{s}) = \varphi(\alpha, \beta) p(\mathbf{s}|\alpha, \beta)$ and let $L(\alpha, \beta|\mathbf{s})$ denote the log-likelihood function of an observation \mathbf{s} . Thus

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

approaches $L(\alpha, \beta|\mathbf{s})$ as $n \rightarrow \infty$, where $\mathbf{s}^{(1)}, \dots, \mathbf{s}^{(n)}$ are MCMC samples simulated from $p^*(\mathbf{s})$. The estimate $(\hat{\alpha}, \hat{\beta}) = \arg \max_{\alpha, \beta} L_n(\alpha, \beta|\mathbf{s})$ is called the Monte Carlo MLE (MCMLE) of (α, β) . The maximization can be done using a conventional optimization procedure, say the conjugate gradient method. Setting $p^*(\mathbf{s}) = \hat{p}_{\text{mix}}^*(\mathbf{s})$, the five runs of SAMC resulted in five estimates of (α, β) . The mean and standard deviation vectors of these estimates were $(-.2994, .1237)$ and $(.00063, .00027)$. 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 $(-.2999, .1234)$.

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

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

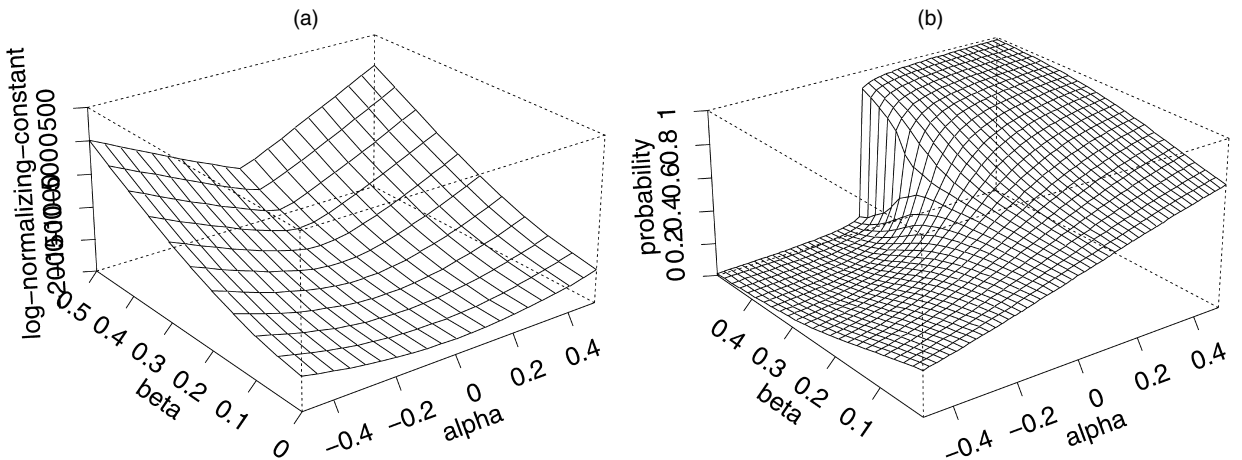


Figure 5. Computational Results of SAMC. (a) Estimate of $\log \varphi(\alpha, \beta)$ on a 21×11 lattice with $\alpha \in \{-.5, -.45, \dots, .5\}$ and $\beta \in \{0, .05, \dots, .5\}$. (b) Estimate of $P(s_i = +1|\alpha, \beta)$ on a 50×25 lattice with $\alpha \in \{-.49, -.47, \dots, .49\}$ and $\beta \in \{.01, .03, \dots, .49\}$.

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 pseudolikelihood estimate (Besag 1975) of (α, β) , which is the MLE of the pseudolikelihood function

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

We repeated the procedure of Sherman et al. for the cancer data five times with $n = 10^5$ and the MCMC samples 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 $(-.3073, .1262)$ and $(.00837, .00946)$. 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 Tibshirani 1993). Let $\mathbf{T}_1 = \sum_{i \in D} s_i$ and $\mathbf{T}_2 = \frac{1}{2} \sum_{i \in D} s_i (\sum_{j \in N(i)} s_j)$. It is easy to see that $\mathbf{T} = (\mathbf{T}_1, \mathbf{T}_2)$ forms a sufficient statistic of (α, β) . Given an estimate $(\hat{\alpha}, \hat{\beta})$, we can reversely estimate

Table 2. Comparison of the Accuracy of the Mix-MCMLEs and Single-MCMLEs for the U.S. Cancer Data

Estimate	Single-MCMLE	Mix-MCMLE
RMSE($\mathbf{t}_1^{\text{sim}}$)	59.51	2.90
RMSE($\mathbf{t}_2^{\text{sim}}$)	114.91	4.61

NOTE: RMSE($\mathbf{t}_i^{\text{sim}}$) is calculated as $\sqrt{\sum_{k=1}^S (\mathbf{t}_i^{\text{sim},k} - \mathbf{t}_i^{\text{obs}})^2 / 5}$, where $i = 1, 2$, and $\mathbf{t}_i^{\text{sim},k}$ denotes the value of $\mathbf{t}_i^{\text{sim}}$ calculated based on the k th estimate of (α, β) .

the quantities \mathbf{T}_1 and \mathbf{T}_2 by drawing samples from the distribution $f(\mathbf{s} | \hat{\alpha}, \hat{\beta})$. If $(\hat{\alpha}, \hat{\beta})$ is accurate, then we should have $\mathbf{t}^{\text{obs}} \approx \mathbf{t}^{\text{sim}}$, where \mathbf{t}^{obs} and \mathbf{t}^{sim} denote the values of \mathbf{T} calculated from the true observation and from the simulated samples. To calculate \mathbf{t}^{sim} , we generated 1,000 independent configurations conditional on each estimate, with each configuration generated by a short run of the Gibbs sampler. The Gibbs sampler started with a random configuration and was iterated for 1,000 cycles. A convergence diagnostic shows that 1,000 iteration cycles were long enough for the Gibbs sampler to reach equilibrium for simulation of $f(\mathbf{s} | \hat{\alpha}, \hat{\beta})$. Table 2 compares the root mean squared errors (RMSEs) of \mathbf{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.

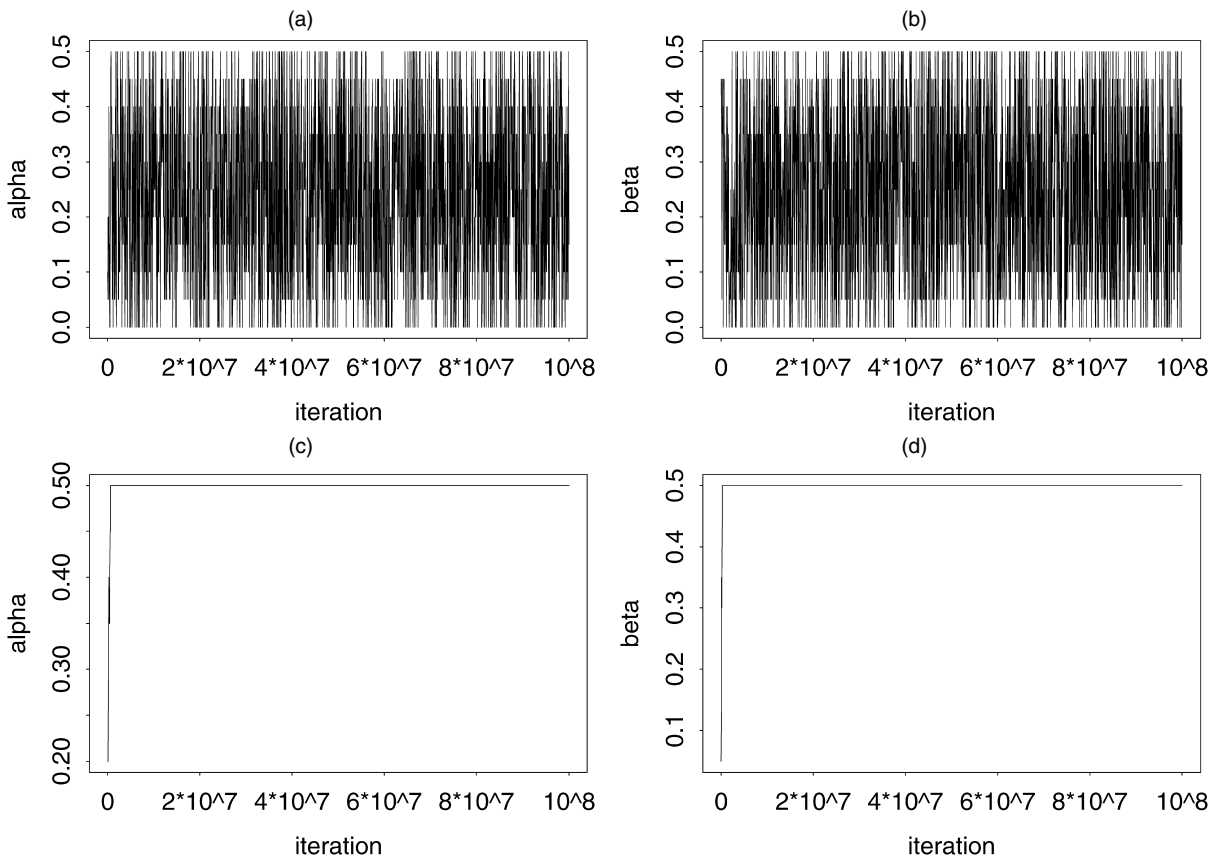


Figure 6. Comparison of SAMC and RJMCMC. (a) and (b) The sample paths of α and β in a run of SAMC. (c) and (d) The sample paths of α and β in a run of RJMCMC.

This is shown in Figures 6(c) and 6(d), where the parameter vector $(\alpha, \beta) = (0, 0)$ corresponds to model M_0 and $(.5, .5)$ corresponds to model $M_{10,10}$. RJMCMC failed to estimate $\varphi(\alpha_1, \beta_1), \dots, \varphi(\alpha_m, \beta_m)$ simultaneously. This phenomenon can be easily understood from Figure 5(a), which shows that model $M_{10,10}$ has a dominated probability over other models. Fives runs of SAMC produced an estimate of the log-odds ratio $\log P(M_{10,10})/P(M_{0,0})$. The estimate is 1,775.7 with standard deviation .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 difficult for SAMC due to its ability to sample rare events from a large sample space. For comparison, Figures 6(a) and 6(b) plot 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.

6. DISCUSSION

In this article 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 proceeds. 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 by Geyer and Thompson (1995),

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

where $k = 30$ is the dimension of \mathbf{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 are called 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 likelihood of SAMC jumping 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,

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

where I is the dimension of \mathbf{x}_I with $I \in \{1, \dots, 30\}$ 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 $E_{11}, E_{12}, \dots, E_{k1}, E_{k2}$ denote the partition, where E_{i1} and

E_{i2} denote the peak and brim sets of $p(\mathbf{x}_i, i)$. 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 reduce the curse of dimensionality. We note that the auxiliary variable used in constructing 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 was done by Andrieu et al. (2005). If the restrictions are removed, then we may need to put some other constraints on the tails of the target distribution $p(\mathbf{x})$ and the proposal distribution $q(\mathbf{x}, \mathbf{y})$ to ensure the minorization condition holds (see Roberts and Tweedie 1996; Rosenthal 1995; 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 minorization condition does not hold, in a manner similar to the MH algorithm (Mengersen and Tweedie 1996). A further study in this direction is of some interest.

APPENDIX: THEORETICAL RESULTS ON STOCHASTIC APPROXIMATION MONTE CARLO

The appendix is organized as follows. In Section A.1 we describe a theorem for the convergence of the SAMC algorithm. In Section A.2 we briefly review the published results on the convergence of a general stochastic approximation algorithm. In Section A.3 we give a proof for the theorem described in Section 1.

A.1 A Convergence Theorem for SAMC

Without loss of generality, we show only the convergence presented in (7) for the case where all subregions are nonempty or, equivalently, $d = 0$. Extension to the case $d \neq 0$ is trivial, because changing step (2) of the SAMC algorithm to (2)' will not change the process of simulation:

$$(2)' \text{ Set } \theta^t = \theta_t + \gamma_t(\mathbf{e}_t - \pi - \mathbf{d}), \text{ where } \mathbf{d} \text{ is an } m\text{-vector of } d.$$

Theorem A.1. Let E_1, \dots, E_m be a partition of a compact sample space \mathcal{X} and $\psi(\mathbf{x})$ be a nonnegative function defined on \mathcal{X} with $0 < \int_{E_i} \psi(\mathbf{x}) d\mathbf{x} < \infty$ for all E_i 's. Let $\pi = (\pi_1, \dots, \pi_m)$ 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 let there exist a constant C such that $\check{\theta} \in \Theta$, where $\check{\theta} = (\check{\theta}_1, \dots, \check{\theta}_m)$ and $\check{\theta}_i = C + \log(\int_{E_i} \psi(\mathbf{x}) d\mathbf{x}) - \log(\pi_i)$. Let $\theta_0 \in \Theta$ be an initial estimate of $\check{\theta}$ and let $\theta_t \in \Theta$ be the estimate of $\check{\theta}$ at iteration t . Let $\{\gamma_t\}$ be a nonincreasing, positive sequence as specified in (6). Suppose that $p_{\theta_t}(\mathbf{x})$ is bounded away from 0 and ∞ on \mathcal{X} , and the proposal distribution satisfies the condition (4). As $t \rightarrow \infty$, we have

$$P \left\{ \lim_{t \rightarrow \infty} \theta_{ti} = C + \log \left(\int_{E_i} \psi(\mathbf{x}) d\mathbf{x} \right) - \log(\pi_i) \right\} = 1, \quad i = 1, \dots, m, \quad (\text{A.1})$$

where C is an arbitrary constant.

A.2 Existing Results on the Convergence of a General Stochastic Approximation Algorithm

Suppose that our goal is to solve the following integration equation for the parameter vector θ :

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

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

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

where $s(\mathbf{x}_t, d\mathbf{y}) = q(\mathbf{x}_t, d\mathbf{y}) \min\{1, [p(\mathbf{y})q(\mathbf{y}, \mathbf{x}_t)]/[p(\mathbf{x})q(\mathbf{x}, \mathbf{y})]\}$, $q(\cdot, \cdot)$ 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 nonincreasing sequence governing the step size. In addition, define a function $\Phi: \mathcal{X} \times \tilde{\Theta} \rightarrow \mathcal{X} \times \Theta$, which reinitializes the nonhomogeneous Markov chain $\{(\mathbf{x}_t, \theta_t)\}$. For instance, the function Φ can generate a random or fixed point, or project $(\mathbf{x}_{t+1}, \theta_{t+1})$ onto $\mathcal{X} \times \Theta$. An iteration of the algorithm is as follows:

1. Generate $\mathbf{y} \sim K_{\theta_t}(\mathbf{x}_t, \cdot)$.
2. Set $\theta^* = \theta_t + \gamma_{t+1}H(\theta_t, \mathbf{y})$.
3. If $\theta^* \in \Theta$, then set $(\mathbf{x}_{t+1}, \theta_{t+1}) = (\mathbf{y}, \theta^*)$; otherwise, set $(\mathbf{x}_{t+1}, \theta_{t+1}) = \Phi(\mathbf{y}, \theta^*)$.

This algorithm is actually a simplified version of the algorithm given by Andrieu et al. (2005). Let $\mathbb{P}_{\mathbf{x}_0, \theta_0}$ denote the probability measure of the Markov chain $\{(\mathbf{x}_t, \theta_t)\}$, started in (\mathbf{x}_0, θ_0) , and implicitly defined by the sequences $\{\gamma_t\}$. Define $D(\mathbf{x}, A) = \inf_{\mathbf{y} \in A} \|\mathbf{x} - \mathbf{y}\|$.

Theorem A.2 (Thm. 5.5 and prop. 6.1 of Andrieu et al. 2005). Assume that the conditions (A₁) and (A₄) hold, and there exists a drift function $V(\mathbf{x})$ such that $\sup_{\mathbf{x} \in \mathcal{X}} V(\mathbf{x}) < \infty$ and the drift condition holds (see Andrieu et al. 2005 for descriptions of the conditions). Let the sequence $\{\theta_n\}$ be defined as in the stochastic approximation algorithm. Then for all $(\mathbf{x}_0, \theta_0) \in \mathcal{X} \times \Theta$,

$$\lim_{t \rightarrow \infty} D(\theta_t, \mathcal{L}) = 0, \quad \mathbb{P}_{\mathbf{x}_0, \theta_0}\text{-a.e.}$$

A.3 Proof of Theorem A.2

To prove Theorem A.2, it suffices to verify that (A₁), (A₄), and the drift condition hold for the SAMC algorithm. To simplify notation, in the proof we drop the subscript t by denoting \mathbf{x}_t by \mathbf{x} and $\theta_t = (\theta_1, \dots, \theta_m)$ by $\theta = (\theta_1, \dots, \theta_m)$. Because the invariant distribution of the MH kernel is $p_\theta(\mathbf{x})$, for any fixed θ , we have

$$\begin{aligned} E(e_{\mathbf{x}}^{(i)} - \pi_i) &= \int_{\mathcal{X}} (e_{\mathbf{x}}^{(i)} - \pi_i) p_\theta(\mathbf{x}) d\mathbf{x} \\ &= \frac{\int_{E_i} \psi(\mathbf{x}) d\mathbf{x} / e^{\theta_i}}{\sum_{k=1}^m [\int_{E_k} \psi(\mathbf{x}) d\mathbf{x} / e^{\theta_k}]} - \pi_i \\ &= \frac{S_i}{S} - \pi_i, \quad i = 1, \dots, m, \end{aligned} \tag{A.2}$$

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

$$h(\theta) = \int_{\mathcal{X}} H(\theta, \mathbf{x}) p(d\mathbf{x}) = \left(\frac{S_1}{S} - \pi_1, \dots, \frac{S_m}{S} - \pi_m \right)'$$

Condition A₁. It follows from (A.2) 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 later, $w(\theta)$ has continuous partial derivatives of the first order. Because $0 \leq w(\theta) \leq \frac{1}{2} [\sum_{k=1}^m (\frac{S_k}{S})^2 + \pi_k^2] \leq 1$ for all $\theta \in \Theta$, and Θ itself is compact, the level set $\mathcal{W}_M = \{\theta \in \Theta, w(\theta) \leq M\}$ is compact for any positive integer M . Condition (A₁-b) is satisfied.

Solving the system of equations formed by (A.2), we have

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

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

To verify the conditions (A₁-a), (A₁-c), and (A₁-d), we have the following calculations:

$$\begin{aligned} \frac{\partial S}{\partial \theta_i} &= \frac{\partial S_i}{\partial \theta_i} = -S_i, \\ \frac{\partial S_i}{\partial \theta_j} &= \frac{\partial S_j}{\partial \theta_j} = 0, \\ \frac{\partial (S_i/S)}{\partial \theta_i} &= -\frac{S_i}{S} \left(1 - \frac{S_i}{S} \right), \quad \text{and} \\ \frac{\partial (S_i/S)}{\partial \theta_j} &= \frac{\partial (S_j/S)}{\partial \theta_i} = \frac{S_i S_j}{S^2} \end{aligned} \tag{A.3}$$

for $i, j = 1, \dots, m$ and $i \neq j$, and

$$\begin{aligned} \frac{\partial w(\theta)}{\partial \theta_i} &= \frac{1}{2} \sum_{k=1}^m \frac{\partial (S_k/S - \pi_k)^2}{\partial \theta_i} \\ &= \sum_{j \neq i} \left(\frac{S_j}{S} - \pi_j \right) \frac{S_i S_j}{S^2} - \left(\frac{S_i}{S} - \pi_i \right) \frac{S_i}{S} \left(1 - \frac{S_i}{S} \right) \\ &= \sum_{j=1}^m \left(\frac{S_j}{S} - \pi_j \right) \frac{S_i S_j}{S^2} - \left(\frac{S_i}{S} - \pi_i \right) \frac{S_i}{S} \\ &= \mu_\eta \frac{S_i}{S} - \left(\frac{S_i}{S} - \pi_i \right) \frac{S_i}{S} \end{aligned} \tag{A.4}$$

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

$$\begin{aligned} \langle \nabla w(\theta), h(\theta) \rangle &= \mu_\eta \sum_{i=1}^m \left(\frac{S_i}{S} - \pi_i \right) \frac{S_i}{S} - \sum_{i=1}^m \left(\frac{S_i}{S} - \pi_i \right)^2 \frac{S_i}{S} \\ &= - \left\{ \sum_{i=1}^m \left(\frac{S_i}{S} - \pi_i \right)^2 \frac{S_i}{S} - \mu_\eta^2 \right\} \\ &= -\sigma_\eta^2 \leq 0, \end{aligned} \tag{A.5}$$

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

State (η)	$\frac{S_1}{S} - \pi_1$	\dots	$\frac{S_m}{S} - \pi_m$
Probability	$\frac{S_1}{S}$	\dots	$\frac{S_m}{S}$

If $\theta \in \mathcal{L}$, then $\langle \nabla w(\theta), h(\theta) \rangle = 0$; 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 (A₁-a) is satisfied.

It follows from (A.5) 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 , because it contains only one free parameter c . Therefore, the interior set of $w(\mathcal{L})$ is empty. Conditions (A₁-c) and (A₁-d) are satisfied.

Condition A₄. Let p be arbitrarily large, $\beta = 1$, $\alpha = 1$, and $\zeta \in (\frac{1}{\tau}, 2)$. Thus conditions $\sum_{t=1}^\infty \gamma_t = \infty$ and $\sum_{t=1}^\infty \gamma_t^\zeta < \infty$ hold. Because $|H(\theta, \mathbf{x})|$ is bounded above by c_1 , as shown in (A.8), $|\gamma_t H(\theta_{t-1}, \mathbf{x}_t)| < c_1 \gamma_t < c_1 \gamma_t^\eta$ holds. Condition (A₄) 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 $\theta \in \Theta$. Because \mathcal{X} is compact, \mathcal{X} is a small set, and thus the minorization condition is satisfied; that is there exists an integer l such that

$$\inf_{\theta \in \Theta} K_\theta^l(\mathbf{x}, A) \geq \delta \nu(A), \quad \forall \mathbf{x} \in \mathcal{X}, \forall A \in \mathcal{B}. \quad (\text{A.6})$$

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

$$\begin{aligned} \sup_{\theta \in \Theta_0} K_\theta^l V^p(\mathbf{x}) &\leq \lambda V^p(\mathbf{x}) + bI(\mathbf{x} \in C), & \forall \mathbf{x} \in \mathcal{X}; \\ \sup_{\theta \in \Theta_0} K_\theta V^p(\mathbf{x}) &\leq \kappa V^p(\mathbf{x}), & \forall \mathbf{x} \in \mathcal{X}, \end{aligned} \quad (\text{A.7})$$

by choosing the drift function $V(\mathbf{x}) = 1$, $\Theta_0 = \Theta$, $0 < \lambda < 1$, $b = 1 - \lambda$, $\kappa > 1$, $p \in [2, \infty)$ and any integer l . Equations (A.6) and (A.7) imply that (DRI1) is satisfied.

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

$$\sup_{\theta \in \Theta} |H(\theta, \mathbf{x})| \leq c_1. \quad (\text{A.8})$$

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

$$\sup_{(\theta, \theta') \in \Theta \times \Theta} |H(\theta, \mathbf{x}) - H(\theta', \mathbf{x})| \leq c_1 |\theta - \theta'|. \quad (\text{A.9})$$

Equations (A.8) and (A.9) imply that (DRI2) is satisfied by choosing $\beta = 1$ and $V(\mathbf{x}) = 1$.

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

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

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

$$|s_\theta(\mathbf{x}, \mathbf{y}) - s_{\theta'}(\mathbf{x}, \mathbf{y})| \leq q(\mathbf{x}, \mathbf{y}) c_2 |\theta - \theta'|, \quad (\text{A.10})$$

which implies that

$$\begin{aligned} \sup_{\mathbf{x}} \|s_\theta(\mathbf{x}, \cdot) - s_{\theta'}(\mathbf{x}, \cdot)\|_1 &= \sup_{\mathbf{x}} \int_{\mathcal{X}} |s_\theta(\mathbf{x}, \mathbf{y}) - s_{\theta'}(\mathbf{x}, \mathbf{y})| d\mathbf{y} \\ &\leq c_2 |\theta - \theta'|. \end{aligned} \quad (\text{A.11})$$

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

$$\begin{aligned} &|K_\theta(\mathbf{x}, A) - K_{\theta'}(\mathbf{x}, A)| \\ &= \left| \int_A [s_\theta(\mathbf{x}, \mathbf{y}) - s_{\theta'}(\mathbf{x}, \mathbf{y})] d\mathbf{y} \right. \\ &\quad \left. + I(\mathbf{x} \in A) \int_{\mathcal{X}} [s_{\theta'}(\mathbf{x}, \mathbf{z}) - s_\theta(\mathbf{x}, \mathbf{z})] d\mathbf{z} \right| \\ &\leq \int_{\mathcal{X}} |s_\theta(\mathbf{x}, \mathbf{y}) - s_{\theta'}(\mathbf{x}, \mathbf{y})| d\mathbf{y} \end{aligned}$$

$$\begin{aligned} &+ I(\mathbf{x} \in A) \int_{\mathcal{X}} |s_{\theta'}(\mathbf{x}, \mathbf{z}) - s_\theta(\mathbf{x}, \mathbf{z})| d\mathbf{z} \\ &\leq 2 \int_{\mathcal{X}} |s_\theta(\mathbf{x}, \mathbf{y}) - s_{\theta'}(\mathbf{x}, \mathbf{y})| d\mathbf{y} \leq 2c_2 |\theta - \theta'|. \end{aligned} \quad (\text{A.12})$$

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

$$\begin{aligned} &\|K_\theta g - K_{\theta'} g\|_V \\ &= \left\| \int (K_\theta(\mathbf{x}, d\mathbf{y}) - K_{\theta'}(\mathbf{x}, d\mathbf{y})) g(\mathbf{y}) \right\|_V \\ &= \left\| \int_{\mathcal{X}^+} (K_\theta(\mathbf{x}, d\mathbf{y}) - K_{\theta'}(\mathbf{x}, d\mathbf{y})) g(\mathbf{y}) \right. \\ &\quad \left. + \int_{\mathcal{X}^-} (K_\theta(\mathbf{x}, d\mathbf{y}) - K_{\theta'}(\mathbf{x}, d\mathbf{y})) g(\mathbf{y}) \right\|_V \\ &\leq \left\| \max \left\{ \int_{\mathcal{X}^+} (K_\theta(\mathbf{x}, d\mathbf{y}) - K_{\theta'}(\mathbf{x}, d\mathbf{y})) g(\mathbf{y}), \right. \right. \\ &\quad \left. \left. - \int_{\mathcal{X}^-} (K_\theta(\mathbf{x}, d\mathbf{y}) - K_{\theta'}(\mathbf{x}, d\mathbf{y})) g(\mathbf{y}) \right\} \right\|_V \\ &\leq \|g\|_V \max \{ |K_\theta(\mathbf{x}, \mathcal{X}^+) - K_{\theta'}(\mathbf{x}, \mathcal{X}^+)|, \\ &\quad |K_\theta(\mathbf{x}, \mathcal{X}^-) - K_{\theta'}(\mathbf{x}, \mathcal{X}^-)| \} \\ &\leq 2c_2 \|g\|_V |\theta - \theta'| \quad [\text{following from (A.12)}], \end{aligned}$$

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

[Received June 2005. Revised July 2006.]

REFERENCES

- Andrieu, C., Moulines, É., and Priouret, P. (2005), "Stability of Stochastic Approximation Under Verifiable Conditions," *SIAM Journal of Control and Optimization*, 44, 283–312.
- Augustin, N., Muggleston, 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, Ser. B*, 36, 192–236.
- (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," *The 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, Ser. 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, Dept. 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*, ed. E. M. Keramigas, Fairfax, VA: Interface Foundation, pp. 156–163.

- (1992), "Practical Monte Carlo Markov Chain" (with discussion), *Statistical Science*, 7, 473–511.
- (1994), "On the Convergence of Monte Carlo Maximum Likelihood Calculations," *Journal of the Royal Statistical Society*, Ser. B, 56, 261–274.
- (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*, Ser. B, 54, 657–699.
- (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*, Ser. 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 Stinchcombe, 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," *Journal of the Physics Society of Japan*, 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.
- (2003), "Use of Sequential Structure in Simulation From High-Dimensional Systems," *Physical Review E*, 67, 56101–56107.
- (2004), "Annealing Contour Monte Carlo for Structure Optimization in an Off-Lattice Protein Model," *Journal of Chemical Physics*, 120, 6756–6763.
- (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-Verlag.
- 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," *Scandinavian Journal of Statistics*, 18, 39–50.
- Neal, R. M. (2003), "Slice Sampling" (with discussion), *The Annals of Statistics*, 31, 705–767.
- Nevel'son, M. B., and Has'minskiĭ, R. Z. (1973), *Stochastic Approximation and Recursive Estimation*, Providence, RI: American Mathematical 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," *The Annals of Mathematical Statistics*, 22, 400–407.
- Robert, C. P., and Casella, G. (2004), *Monte Carlo Statistical Methods* (2nd ed.), New York: Springer-Verlag.
- 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 Titterton, D. M. (2001), "Improved Particle Filters and Smoothing," in *Sequential MCMC in Practice*, eds. A. Doucet, N. de Freitas, 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), *The 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.
- 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, Dept. 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," *Physics Review Letters*, 90, 035701.
- Younes, L. (1988), "Estimation and Annealing for Gibbsian Fields," *Annales de l'Institut Henri Poincaré*, 24, 269–294.
- (1999), "On the Convergence of Markovian Stochastic Algorithms With Rapidly Decreasing Ergodicity Rates," *Stochastics and Stochastics Reports*, 65, 177–228.