ON MARKOV CHAIN MONTE CARLO ACCELERATION

A.E. Gelfand
S.K. Sahu

TECHNICAL REPORT No. 480
APRIL 4, 1994

Prepared Under Contract
N00014-92-J-1264 (NR-042-267)
FOR THE OFFICE OF NAVAL RESEARCH

Professor David Siegmund, Project Director

DEPARTMENT OF STATISTICS
STANFORD UNIVERSITY
STANFORD, CALIFORNIA 94305-4065

On Markov Chain Monte Carlo Acceleration

Alan E. Gelfand and Sujit K. Sahu

Abstract

Markov chain Monte Carlo (MCMC) methods are currently enjoying a surge of interest within the statistical community. The goal of this work is to formalize and support two distinct adaptive strategies which typically accelerate the convergence of a MCMC algorithm. One approach is through resampling; the other incorporates adaptive switching of the transition kernel. Support is both by analytic arguments and simulation study. Application is envisioned in low dimensional but non-trivial problems. Two pathological illustrations are presented. Connections with reparametrization are discussed as well as possible difficulties with infinitely often adaptation.

Key words: Adaptive chains; Gibbs sampler; L^1-convergence; Markov chain Monte Carlo; Metropolis-Hastings algorithm; Rejection method; Resampling.

1 Introduction

Markov chain Monte Carlo (henceforth MCMC) algorithms are currently experiencing a surge of interest within the statistical community. Though they first appeared in the literature forty years ago (Metropolis *et al.* 1953), the present enthusiasm arises from their much more recent recognition as powerful tools for implementing a wide range of statistical inference. For instance in Bayesian inference, MCMC simulation enables calculation of features of the posterior distribution of model unknowns given the observations (Gelfand and Smith 1990). For maximum likelihood estimation, where the likelihood is specified as a non-normalized joint density for the observations, MCMC techniques can be used for carrying out the maximization (Geyer and Thompson 1992; Gelfand and Carlin 1993).

In either case the object we wish to learn more about is a function, say $f(\mathbf{u})$, which we assume is strictly positive and integrable with respect to a measure μ over a set of interest denoted by $\mathcal{U}$. In practice μ is a product of Lebesgue and/ or counting measures. Utilizing the duality between population and sample we propose to learn about f by drawing samples from the normalized version of f, the density $h(\mathbf{u}) = f(\mathbf{u})/\int_{\mathcal{U}} f(\mathbf{u})d\mu(\mathbf{u})$. MCMC techniques permit handling of high dimensional u but there is also considerable interest in cases where dimension is not so large but asymptotic approximations are inappropriate and analytic methods are infeasible (see Gelfand, Smith and Lee 1992 for a broad range of examples). This is the situation we consider with the objective being to propose and justify modifications to MCMC techniques which tend to hasten convergence.

A few words regarding convergence of MCMC techniques are appropriate. There is by now a substantial theoretical literature providing conditions for and rates of convergence of MCMC algorithms. We mention a few: Geman and Geman (1984); Tierney (1994); Applegate *et al.* (1991); Roberts and Smith (1992); Schervish and Carlin (1992). In application, convergence conditions are readily checked but theoretical rates, usually in the form of bounds, seem to be of little practical value. As a result there has also arisen a substantial discussion of convergence diagnostics, using the output stream of the MCMC algorithm, e.g., Ritter and Tanner (1992), Raftery and Lewis (1992), Zellner and Min (1992), Gelman and Rubin (1992) and Geyer (1992). Since pathological choices of $f(\mathbf{u})$ can either deceive or render inapplicable such diagnostics, a cynical viewpoint suggests that we can only hope to demonstrate lack of convergence rather than convergence (Clifford 1993).

However, in practice, a broad range of starting values and a bit tuning of the MCMC algorithm usually enable the user to feel comfortable with regard to convergence.

In the present work we do not consider the problem of assessing convergence. Rather our goal is to formalize and support two distinct adaptive strategies for accelerating the convergence of a MCMC algorithm for low dimensional but non-trivial problems. Practically useful adaptive accelerators must be built solely from the iterations of the MCMC algorithm and must not be too costly to implement. This is the case with our proposals though we anticipate refinement. Support is provided both through theory and simulation. Related work of Gilks *et al.* (1992) describes a technique called adaptive direction sampling which, though not discussed here, falls under the umbrella of our second class of strategies. We confess at the outset that, in using adaptive strategies, we can not guarantee faster convergence. Occasionally we may be led in the "wrong direction". However, as we argue in the sequel, the proposed adaptation never compromises convergence itself.

It is well known that all numerical integration approaches perform better when correlation amongst the components of $\mathbf{u}$ is low. In particular, MCMC algorithms converge more rapidly. Hence acceleration might be attempted through a reparametrization to approximate orthogonality. An effective linear transformation to achieve roughly uncorrelated components of $\mathbf{u}$ might be identified through analytic investigation of $f(\mathbf{u})$ or adaptively from early output of the MCMC algorithm (see Müller 1994 in this regard). Such orthogonalization need not always be worthwhile. For instance, with a Gibbs sampler, conjugacies, which permit convenient draws from complete conditional distributions, may be sacrificed. In any event, such reparametrization is not in competition with our approaches, rather it may be employed in conjunction with them.

The format of the paper is as follows. Section two provides a brief review of aspects of MCMC simulation which we require, as well as a description of the two acceleration strategies. The first is based upon resampling and support is provided in Section 3. The second is based upon adaptive modification of the transition kernel and support is presented in Section 4. Section 5 provides two useful examples. The first fits a nonlinear model to a real data set resulting in a very poorly behaved likelihood. The second reveals that adaptation done infinitely often can modify the ergodic behavior of the MCMC algorithm.

2 MCMC Algorithms and Accelerators

MCMC algorithms proceed from the, possibly surprising, idea of creating a stationary Markov chain whose invariant distribution is H (here H is the probability measure associated with the density $h(\mathbf{u})$, i.e., $h = \frac{dH}{d\mu}$). We work with two such algorithms here, the Gibbs sampler (see, e.g., Geman and Geman 1984; Gelfand and Smith 1990) and the Metropolis-Hastings algorithm (see, e.g., Hastings 1970; Tierney 1994).

2.1 Notation and Review

We introduce a bit of notation. For a discrete time stationary Markov chain we denote its state at time t by $\mathbf{u}^{(t)}$ and its transition kernel by P, i.e., for a $\mu-$measurable set A, $P(\mathbf{u}^{(t-1)}; A) = P(\mathbf{u}^{(t)} \in A | \mathbf{u}^{(t-1)})$. The distribution H is an invariant distribution for P if for all $\mu-$measurable sets A, $H(A) = \int P(\mathbf{u}; A) dH(\mathbf{u})$. If $P^{(t)}(\mathbf{u}; A) = P(\mathbf{u}^{(t)} \in A | \mathbf{u}^{(0)} = \mathbf{u})$ then the invariant distribution H is an equilibrium distribution if

$$\lim_{t \to \infty} P^{(t)}(\mathbf{u}; A) = H(A) \tag{1}$$

for all $\mu-$measurable sets A. Expression (1) captures the basis of the MCMC simulation approach: after a sufficient number of transitions, $\mathbf{u}^{(t)}$ is approximately distributed according to H. All discussion of convergence diagnosis concerns itself with assessing when t is large enough so that the "distance" between $P^{(t)}$, which can rarely be computed analytically, and H is small.

We can also work with the marginal distribution of $\mathbf{u}^{(t)}$, say $H^{(t)}$, i.e., $H^{(t)}(A) = \int P^{(t)}(\mathbf{u}; A)$ $dH^{(0)}(\mathbf{u})$ where $H^{(0)}$ is the distribution of starting states. If (1) holds then $\lim_{t \to \infty} H^{(t)}(A) = H(A)$ for all $\mu-$measurable sets A. Of course $H^{(t)}$ also will not be available explicitly but $H^{(t)}$ lends itself to Monte Carlo integration through the relationship

$$H^{(t)}(A) = \int P(\mathbf{u}; A) dH^{(t-1)}(\mathbf{u}). \tag{2}$$

The Gibbs sampler creates a transition from $\mathbf{u}^{(t-1)}$ to $\mathbf{u}^{(t)}$ as follows. If we partition u into k blocks, i.e., $\mathbf{u} = (\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_k)$, we obtain $\mathbf{u}_1^{(t)}$ as a draw from the conditional density, $h(\mathbf{u}_1 | \mathbf{u}_2^{(t-1)}, \mathbf{u}_3^{(t-1)}, \cdots, \mathbf{u}_k^{(t-1)})$. We then obtain $\mathbf{u}_2^{(t)}$ as a draw from $h(\mathbf{u}_2 | \mathbf{u}_1^{(t)}, \mathbf{u}_3^{(t-1)}, \cdots, \mathbf{u}_k^{(t-1)})$, etc. until finally $\mathbf{u}_k^{(t)}$ is drawn from $h(\mathbf{u}_k | \mathbf{u}_1^{(t)}, \mathbf{u}_2^{(t)}, \cdots, \mathbf{u}_{k-1}^{(t)})$ whence u has been fully updated.

Thus, the transition kernel of the Gibbs sampler has a density $p(\mathbf{u};\mathbf{v})$ taking the form

$$p(\mathbf{u};\mathbf{v}) = \prod_{i=1}^{k} h(\mathbf{v}_i|\mathbf{v}_1,\cdots,\mathbf{v}_{i-1},\mathbf{u}_{i+1},\cdots,\mathbf{u}_k). \tag{3}$$

Note that all of the conditional densities on the right hand side of (3) are proportional to f evaluated at the corresponding arguments. If any of these densities are not standard, normalization of f is needed in order that $p(\mathbf{u};\mathbf{v})$ can be computed. Therefore, for the Gibbs sampler, (2) can be differentiated yielding the marginal density of $\mathbf{u}^{(t)}$,

$$h^{(t)}(\mathbf{u}) = \int p(\mathbf{z};\mathbf{u})h^{(t-1)}(\mathbf{z})\, d\mu(\mathbf{z}). \tag{4}$$

The general Metropolis-Hastings algorithm updates $\mathbf{u}^{(t-1)}$ to $\mathbf{u}^{(t)}$ as follows. Suppose Q is a Markov transition kernel having strictly positive density $q(\mathbf{u};\mathbf{v})$ with respect to μ, i.e. $Q(\mathbf{u};A) = \int_A q(\mathbf{u};\mathbf{v})d\mu(\mathbf{v})$. Let $\alpha(\mathbf{u};\mathbf{v}) = \min\left(1, \frac{f(\mathbf{v})q(\mathbf{v};\mathbf{u})}{f(\mathbf{u})q(\mathbf{u};\mathbf{v})}\right)$ and define $r(\mathbf{u}) = 1 - \int_{\mathcal{U}} \alpha(\mathbf{u};\mathbf{v})q(\mathbf{u};\mathbf{v})d\mu(\mathbf{v})$. A transition from $\mathbf{u}^{(t-1)}$ to $\mathbf{u}^{(t)}$ is made by drawing a candidate transition state $\mathbf{v}$ from $q(\mathbf{u}^{(t-1)};\mathbf{v})$. Then, with probability $\alpha(\mathbf{u}^{(t-1)};\mathbf{v})$ we move from $\mathbf{u}^{(t-1)}$ to $\mathbf{v}$ and take $\mathbf{u}^{(t)} = \mathbf{v}$; with probability $1-\alpha(\mathbf{u}^{(t-1)};\mathbf{v})$ we stay at $\mathbf{u}^{(t-1)}$ and take $\mathbf{u}^{(t)} = \mathbf{u}^{(t-1)}$. Thus $r(\mathbf{u}^{(t-1)})$ is the chance of not moving and the transition kernel takes the form

$$P(\mathbf{u};A) = \int_A \alpha(\mathbf{u};\mathbf{v})q(\mathbf{u};\mathbf{v})d\mu(\mathbf{v}) + r(\mathbf{u})\delta_{\mathbf{u}}(A) \tag{5}$$

where $\delta_{\mathbf{u}}$ denotes the degenerate distribution at $\mathbf{u}$, i.e, $\delta_{\mathbf{u}}(A) = 1$ if $\mathbf{u} \in A$, 0 if $\mathbf{u} \notin A$. Thus $P(\mathbf{u};A)$ arises as a mixed measure. Returning to (2), in the Metropolis-Hastings case, though $P^{(t)}$ is atomic, $H^{(t)}$ is absolutely continuous with respect to μ if the starting distribution $H^{(0)}$ is. In fact, using (2) and (5), direct calculation of the Radon-Nikodym derivative yields

$$h^{(t)} \equiv \frac{dH^{(t)}}{d\mu} = s^{(t-1)} + rh^{(t-1)} \tag{6}$$

where $s^{(t-1)}(\mathbf{u}) = \int \alpha(\mathbf{z};\mathbf{u})q(\mathbf{z};\mathbf{u})h^{(t-1)}(\mathbf{z})d\mu(\mathbf{z})$.

2.2 Acceleration Approaches

The term "acceleration" is used in numerical analysis to indicate hastening of convergence. Analogously, for an MCMC algorithm, an acceleration approach is a technique to diminish the number of transitions, t, such that $\mathbf{u}^{(t)}$ is approximately distributed according to h. We consider two types

of accelerators. We present them in the setting of say m parallel chains whence at any iteration t we have m i.i.d observations. The approaches are more easily described in this case and the independence simplifies analytic arguments. However, several authors advocate the use of a few, perhaps even a single chain as most sensible and efficient (see, e.g., Geyer 1992). Intuitively, we would expect acceleration to ensue even if we used m iterates from a single chain possibly after an initial burn-in and possibly with spacing to reduce dependence.

2.3 Acceleration through Resampling

Suppose, then, that we have $\mathbf{u}_j^{(t)}, j = 1, 2, \ldots, m$, i.i.d $h^{(t)}$ for each t. The first type of accelerator attempts to convert the sample from $h^{(t)}$ to a sample approximately from h. The idea is that if our current marginal distribution is moved "closer" to h, within fewer transitions our draws will be essentially from h. Indeed, if our current $\mathbf{u}_j^{(t)}$ were drawn exactly from h, then all subsequent draws would be as well. Such conversion can be accomplished using ideas in Smith and Gelfand (1992) who suggest two resampling methods, the rejection method (see also Devroye 1986 and Ripley 1987) and the weighted bootstrap (see also Rubin 1988).

Applied to our setting the rejection method proceeds as follows. Compute $M = \sup_{\mathbf{u}} \frac{f(\mathbf{u})}{h^{(t)}(\mathbf{u})}$. For each $\mathbf{u}_j^{(t)}$ draw $z_j \sim \mathrm{U}(0,1)$. If $z_j \leq \frac{f(\mathbf{u}_j^{(t)})}{Mh^{(t)}(\mathbf{u}_j^{(t)})}$ retain $\mathbf{u}_j^{(t)}$ as a draw from h. The weighted bootstrap computes, for each $\mathbf{u}_j^{(t)}$, the weights $w_j = \frac{f(\mathbf{u}_j^{(t)})}{h^{(t)}(\mathbf{u}_j^{(t)})}$ and $q_j = \frac{w_j}{\sum_{j=1}^{m} w_j}$. The $\mathbf{u}_j^{(t)}$ are then resampled according to the probabilities q_j. A resampled $\mathbf{u}^*$ is approximately distributed according to h. The rejection method is desirable in that retained observations have exactly the distribution h. However, computation of M is usually difficult and only a portion (often small) of the m $\mathbf{u}_j^{(t)}$ are retained. Hence, in practice and in our simulation investigation we use the weighted bootstrap. It is easy to implement and permits as many resampled observations as desired. However, their distribution is only approximately h.

Unfortunately, except in the simplest cases, $h^{(t)}$ will not be known, so, in implementing a resampling method, we must replace $h^{(t)}$ by an estimate $\hat{h}^{(t)}$. This extra level of approximation should not be viewed as a deterrent. Our goal is only to draw from a distribution which we expect to be "closer" to h than $h^{(t)}$. In fact, denoting a resampled draw by $\mathbf{u}^{*(t)}$, we would make the next transition via P to obtain $\mathbf{u}^{(t+1)}$. Indeed, if $\mathbf{u}^* \sim h$, then so would all subsequent draws under this

chain.

How shall we estimate $h^{(t)}$? One possibility is a kernel density estimate based upon the $\mathbf{u}_j^{(t)}$ (see, e.g., Silverman 1986). An alternative arises using Monte Carlo integration. In particular, corresponding to (4), we obtain

$$\hat{h}^{(t)}(\mathbf{u}) = m^{-1} \sum_{j=1}^{m} p(\mathbf{u}_j^{(t-1)}; \mathbf{u}) \tag{7}$$

where $p(.;.)$ is defined in (3). Monte Carlo estimation of h is more computationally demanding than a kernel density estimate. However, the Rao-Blackwellization encompassed in (7) suggests, under a wide range of loss functions, a better estimate of h will result (see Gelfand and Smith 1990 in this regard). The form (7) was used in the simulation investigation of the next section. Corresponding to (6), if we assume $h^{(t)} \approx h^{(t-1)}$ then $h^{(t)} \approx \frac{s^{(t-1)}}{(1-r)}$. A Monte Carlo estimate of $s^{(t-1)}$ is straightforward again using the $\mathbf{u}_j^{(t-1)}$. For a given $\mathbf{u}$, we can obtain a Monte Carlo estimate of $r(\mathbf{u})$ by making draws $\mathbf{v}_j$ given $\mathbf{u}$ from $q(\mathbf{u}; \mathbf{v})$.

Note that the proposed adaptive modification of the MCMC algorithm changes the distribution at the t^{th} iteration from $h^{(t)}$ to a *random distribution* $\hat{h}^{(t)}$ and results in a non-Markovian transition. Were we to do this at each iteration there is no assurance that the resulting MCMC algorithm converges or that it converges to h (see Section 5.2). However, we envision such adaptive modification for only a few iterations, thereafter running a stationary chain whose invariant distribution is h so convergence is assured.

2.4 Acceleration by Changing P

The second type of acceleration replaces the current transition kernel say P_1 by a new choice P_2 which is "better than" P_1 . What do we mean by "better than" ? It is easier to work with a finite state space (and, of course, so do computing machines) whence transition kernels become stochastic matrices. Two such matrices having the same unique invariant distribution may be compared using their eigenvalues. In particular if P is $r \times r$ having eigenvalues $1 = \beta_1 > \beta_2 > \cdots > \beta_r$, let $\beta^{(*)} = \max(\beta_2, |\beta_r|)$. Then $\beta^{(*)}$ can be used to bound the distance between $P^{(t)}$ and h. This is usually referred to as Perron-Frobenius theory and a weak result (see, e.g., Iosifescu 1980) is the following. Suppose the invariant distribution h is written as an $r \times 1$ vector, $\mathbf{h}$, and we define the

$r \times r$ matrix, say $H = (\mathrm{h}, \mathrm{h}, \ldots, \mathrm{h})$. Noting that $P^{(t)}$ is in fact $(P)^t$, we have

$$(P)^t - H^T = O\left(t^{\gamma-1}(\beta^{(*)})^t\right) \tag{8}$$

where γ is the multiplicity of $\beta^{(*)}$. Expression (8) is not as useful as we would like since it does not provide an explicit bound on a cell difference.

A stronger result is available if P is reversible, i.e., if $h_r P_{rs} = h_s P_{sr}$ for all r and s. Diaconis and Stroock (1991) obtain a bound on the variation distance to equilibrium in terms of $\beta^{(*)}$. Defining this distance for probability distributions h and g to be $\| h - g \| = \sum_l |h_l - g_l|/2$ we have

Theorem 1(Diaconis and Stroock): *If P is a reversible Markov chain with unique invariant distribution h and P is irreducible then for all j and t*

$$\left(\sum_{l=1}^{r} |(P^t)_{jl} - h_l|\right)^2 \leq \frac{1 - h_j}{h_j}(\beta^{(*)})^{2t}. \tag{9}$$

Thus for $P_i, i = 1, 2$, with associated $\beta_i^{(*)}$ we will say that P_2 is better than P_1 if $\beta_2^{(*)} < \beta_1^{(*)}$.

Again considering m parallel chains each taken to the t^{th} iteration, the proposal is to utilize the $\mathrm{u}_j^{(t)}$ to adaptively revise P_1 to P_2 which we expect to be better than P_1. For a Metropolis-Hastings algorithm this means adaptive modification of the proposal transition kernel $q(u; \mathrm{v})$. Such change results in a chain which is no longer stationary and hence need not converge to h (see Section 5.2). As at the end of Section 2.3, we propose only a few adaptive transitions before settling into a stationary chain, again to insure convergence. Analytic justification and simulation support is provided in Section 4. The idea of switching between different transition mechanisms as the simulation proceeds is discussed in Besag and Green (1993). There, the issue is one of conflicting demands upon an MCMC algorithm, namely speed of convergence vs. small variability in estimation using ergodic averages. For the former we want $\beta^{(*)}$ small. For the latter they show that small values of $\frac{1+\beta_l}{1-\beta_l}$ help, i.e., negative eigenvalues help. Our proposed switching is only concerned with the first demand.

2.5 Additional Remarks

We have ignored the cost in computing time to implement the implementation. This cost is clearly problem specific and might offset the benefit of acceleration. In our simulation investigation this

was not the case. We might expect resampling to be more effective than changing P, since the explicit objective of the former is to sample approximately from h rather than to improve upon the current P. However, so many factors affect the convergence of an adaptive procedure that such a conclusion is not generally supportable. Both approaches are rather sensitive to the choice of m. If m is too small a poor estimate of $h^{(t)}$ or a poor revision of P may arise yielding a potentially adverse effect on convergence.

3 Acceleration through Resampling

Here we further investigate the resampling approach of Section 2.3. Again our goal is to resample from the set of $\mathbf{u}_j^{(t)}, j = 1, 2, \ldots, m$, i.i.d $h^{(t)}$, a new sample $\mathbf{u}_j^{*(t)}, j = 1, 2, \ldots, m^*$ whose distribution $\hat{\hat{h}}^{(t)}$ is closer to h than $h^{(t)}$ is. Recall that, since $h^{(t)}$ is not available, an estimate, $\hat{h}^{(t)}$ must be used to implement the resampling. Hence $\hat{\hat{h}}^{(t)}$ is random.

In Section 3.1 we argue that such resampling works in the sense that, under mild conditions, the variation distance, $J_m^{(t)} \equiv \| \hat{\hat{h}}^{(t)} - h \| \to 0$ *a.s.* as $m \to \infty$. The practical implication is that for a suitably large m, resampling may be expected to produce draws from a distribution "closer" to h. In Section 3.2 we present some very encouraging simulation results regarding the distribution of $J_m^{(t)}$, in comparison with the constant $\| h^{(t)} - h \|$.

3.1 Theoretical Results

Recalling the rejection method described in Section 2.3, suppose $M = \sup_{\mathbf{u}} \frac{f(\mathbf{u})}{g_2(\mathbf{u})}$, $z \sim \mathrm{U}(0,1)$ but $\mathbf{u} \sim g_1(\mathbf{u})$. Suppose further that we keep $\mathbf{u}$ if $z \leq \frac{f(\mathbf{u})}{M g_2(\mathbf{u})}$. Then the density of $\mathbf{u}$ is

$$\frac{f(\mathbf{u}) g_1(\mathbf{u})}{g_2(\mathbf{u})} \Big/ \int \frac{f(\mathbf{u}) g_1(\mathbf{u})}{g_2(\mathbf{u})} d\mathbf{u}. \tag{10}$$

The proof is the same as that of the usual rejection method (see, e.g., Ripley 1987). Similarly, for the weighted bootstrap of Section 2.3, if we sample $\mathbf{u}_j$ i.i.d $g_1(\mathbf{u})$, $j = 1, 2, \ldots, m$ but we resample using weights $w_j = f(\mathbf{u}_j)/g_2(\mathbf{u}_j)$ it is straight forward to show that the distribution we are approximately sampling is again (10). In our setting g_1 is $h^{(t)}$, g_2 is $\hat{h}^{(t)}$ whence

$$\hat{\hat{h}}^{(t)} = \frac{f(\mathbf{u}) h^{(t)}(\mathbf{u})}{\hat{h}^{(t)}(\mathbf{u})} \Big/ \int \frac{f(\mathbf{u}) h^{(t)}(\mathbf{u})}{\hat{h}^{(t)}(\mathbf{u})} d\mathbf{u}. \tag{11}$$

We set $t = 1$, without loss of generality. We assume that $\hat{h}^{(1)}$ converges *a.s.* to $h^{(1)}$. This is the case with usual kernel density estimates (see Devroye and Gyorfi 1985) as well as estimator (7). In either case, $\hat{h}^{(1)}$ is of the form $m^{-1}\sum_{j=1}^{m} g_j(\mathrm{u})$ where g_j is a (random) density. Hence the numerator of (11) converges *a.s.* to $f(\mathrm{u})$ as $m \to \infty$. If, as $m \to \infty$, the denominator converges to $\int f(\mathrm{u})$, the density $\hat{\hat{h}}^{(1)}$ converges *a.s.* to the density h. Then by a standard theorem (see, e.g., Glick 1974) $\int |\hat{\hat{h}}^{(1)} - h| \to 0$ *a.s.* as $m \to \infty$.

Thus we investigate the limiting behavior of $\int f h^{(1)}/\hat{h}^{(1)}$. We write $\hat{h}^{(1)}$ as $\hat{h}^{(1)}_m$ to indicate that it is an average over m terms. In fact the i.i.d sequence $\{\mathrm{u}_j^{(0)}, j = 1, 2, \ldots\}$ drawn from $h^{(0)}$ determines $\{\hat{h}^{(1)}_m, m = 1, 2, \ldots\}$ under (7). The i.i.d. sequence $\{\mathrm{u}_j^{(1)}, j = 1, 2, \ldots\}$ drawn from $h^{(1)}$ determines $\{\hat{h}^{(1)}_m, m = 1, 2, \ldots\}$ under a kernel density estimate. In either case, it is clear that for certain sequences the random functions of u, $h^{(1)}(\mathrm{u})/\hat{h}^{(1)}_m(\mathrm{u})$, can be badly behaved in the tails and that, with $f_m = f h^{(1)}/\hat{h}^{(1)}_m$, $\int f_m$ need not exist.

Fortunately in practice, the situation is more encouraging. For the Gibbs sampler, suppose f is continuous and strictly positive, as it will be in most statistical applications where u is a parameter vector. Since the complete conditional distributions associated with f are all proportional to f, from (3), the $p(\mathrm{u}_j^{(t-1)}; \mathrm{u})$ are continuous and strictly positive. Hence, from (7), $\hat{h}^{(t)}$ and thus f_m are as well. Using a kernel density estimate for $h^{(1)}$ based upon a kernel function which is continuous and strictly positive where f is, again, f_m will be as well.

Fix an underlying sequence such that $f_m(\mathrm{u})$ converges to $f(\mathrm{u})$. By Egoroff's theorem $f_m(\mathrm{u})$ converges uniformly to $f(\mathrm{u})$ on a compact subset of $\mathcal{U}$, say Ω, having arbitrarily large probability under f. But then, since f_m is continuous on Ω we have $\int_\Omega f_m \to \int_\Omega f$. Since $f_m \to f$ *a.s.* we can claim such convergence of integrals for almost every underlying sequence. In the case of a Metropolis-Hastings algorithm, to obtain similar results we require that, for almost every v, $q(\mathrm{v}; \mathrm{u})$ to be strictly positive and continuous where f is.

3.2 Simulation Results

The simulation study investigates the distribution of $J_m^{(1)}/2 = \int |\hat{\hat{h}}^{(1)} - h|$ with $\hat{\hat{h}}^{(1)}$ as in (11), as well as the exact value of $J/2 = \int |h^{(1)} - h|$. We confine ourselves to the Gibbs sampler and to illustrations which are, of necessity, simple in order to permit exact calculations and thousands of

replications within a reasonable amount of computing time. In fact we take $f(\mathrm{u})$ to be bivariate normal or a bimodal mixture of two bivariate normals with high correlation between the components, a situation where convergence is known to be slow. In either case we ran the sampler drawing from $h(u_1|u_2)$ first taking $h^{(0)}(u_2)$ to be $N(\mu_0, \sigma_0^2)$. We chose μ_0 away from the mean of $f(u_2)$ and considered σ_0^2 both smaller and larger than the variance of $f(u_2)$ in order to examine under and overdispersed initial distributions relative to $f(u_2)$. In the mixture case $h^{(1)}(u_1)$ is not available explicitly. Straightforward calculation yields

$$J_m^{(1)}/2 = \int f(u_1)|\frac{h^{(1)}(u_1)}{\hat{h}^{(1)}(u_1)\int f(u_1)h^{(1)}(u_1)/\hat{h}^{(1)}(u_1)} - 1|du_1. \tag{12}$$

Hence, given $h^{(1)}(u_1)$ and $\hat{h}^{(1)}(u_1)$, (12) can be calculated using Monte Carlo integration by taking draws from $f(u_1)$; 2000 draws were used. The exact value of $\int |h^{(1)} - h|$ can also be calculated in this fashion. If $h^{(1)}(u_1)$ is not available explicitly it was obtained by Monte Carlo integration using draws from $h^{(0)}(u_2)$. 500 draws were made to evaluate the integral in the denominator of right hand side of equation (12).

For each f and $h^{(0)}$, the simulation involves an outer loop over the number of replicates (we choose 1000 here). Each replicate yields a random value of $J_m^{(1)}/2$. For each replicate we ran $m = 500$ parallel strings of the Gibbs sampler. Of course since we only took one iteration this amounts to making 500 draws from $h^{(0)}$ after which $\hat{h}^{(1)}(u_1)$ is determined and (12) can be calculated.

For the single bivariate normal case we took $f = h = BVN(0,0,1,1,0.8)$ and $h^{(0)}(u_2) = N(2,\sigma_0^2)$, $\sigma_0^2 = 9, 0.09$. For the mixture case our $h = 0.4 \times BVN(2.0,-2.0,1,1,0.8) + 0.6 \times BVN(0,0,1,1,0.8)$ with $h^{(0)}(u_2) = N(0,1)$. The exact value of J and some features of the distribution of $J_m^{(1)}$ are summarized in Table 1. Note that the starting distribution does not affect $P(J_m^{(1)}) < J)$ which equals 1 in all the cases indicating the effectiveness of the the adaptive strategy. The distribution of the $J_m^{(1)}$ is fairly symmetric and lies well below the J in each case. Fixing m, in general, performance will depend upon how often and at which iterates we resample as well as the dimension of u.

{ Insert Table 1 here }

4 Acceleration by Changing P

Here we further investigate the approach of adaptive switching of transition kernels introduced in Section 2.4. The goal is to replace the current kernel P_1 by a new one, P_2 which we expect to accelerate convergence. We envision application of this approach to the Metropolis-Hastings class of algorithms by changing $q(\mathrm{u};\mathrm{v})$ the proposed kernel. As noted by Tierney (1994) various types of q's are useful. For example if v is generated by adding a random increment to u, drawn according to a density g then $q(\mathrm{u};\mathrm{v}) = g(\mathrm{v}-\mathrm{u})$. If g is elliptically symmetric, i.e., $g(\mathrm{z}) = g(\mathrm{z}^T\Delta\mathrm{z})$ then switching might mean changing Δ. If v is chosen independently of the current u then $q(\mathrm{u};\mathrm{v}) = g(\mathrm{v})$. Here g functions similarly to an importance sampling density. Transition depends upon the relative sizes of the weights $w(\mathrm{u}) = \frac{f(\mathrm{u})}{g(\mathrm{u})}$ and $w(\mathrm{v}) = \frac{f(\mathrm{v})}{g(\mathrm{v})}$. If $g \propto f$ then the chain produces i.i.d draws from h. Under suitable rescaling of u, g is often taken to be a multivariate normal or t density so that switching would change the mean and /or covariance of g to make g more resemble f.

We remind the reader that, for the Metropolis-Hastings algorithm, any such g, obtained adaptively or otherwise, yields a chain whose invariant distribution is h. Hence initial adaptive switching for a finite number of transitions does not affect desired ergodic behavior. In Section 4.1 we argue that, in finite state spaces, for given choices of P_1 and P_2, if P_2 is better than P_1 in the sense of Section 2.4, then for a fixed total number of transitions, a smaller bound on variation distance results by running some under P_2 than all under P_1. Here we extend Theorem 1 of Section 2.4. Of course, it would have been preferable to run all transitions under P_2 but a potentially better P_2 is only identified through (parallel) transitions under P_1. The fact that P_2 is thus random does not violate the conclusions of Theorem 2 below. However, simulation investigation is required to demonstrate that such adaptive development of P_2 does, in fact, tend to yield better P's. This is taken up in Section 4.2 with illustrations using a random increment choice for g. We find that adaptive change of Δ results in a distribution of variation distance from h which tends to yield smaller values than the fixed variation distance without switching. The work of Gilks *et al.* (1992) incorporates a different sort of adaptive strategy to identify a potentially better P_2.

4.1 Theoretical Results

We consider the rate of convergence of a finite state space Markov chain arising under a finite deterministic sequence of transition matrices P_i, each having h as an invariant distribution. It is routine that such a chain has h as its stationary distribution. We illustrate in the case of just two P_i where P_1 is run for the first s iterations and then P_2 for the next $t-s$ iterations. We obtain a bound on the variation distance at the t^{th} iteration as in Theorem 1 which may be compared with (9). The reader will readily see extension to more than two P_i's. Recalling the notation surrounding Theorem 1, we have for all j and $1 \leq s \leq t$,

Theorem 2:

$$\left(\sum_{l=1}^{r} |(P_1^s P_2^{t-s})_{jl} - h_l|\right)^2 \leq \frac{1-h_j}{h_j}(\beta_1^{(*)})^{2s}(\beta_2^{(*)})^{2(t-s)} \tag{13}$$

where $(P_1^s P_2^{t-s})_{jl} = \sum_{k=1}^{r}(P_1^s)_{jk}(P_2^{t-s})_{kl}$.

Proof: Following Diaconis and Stroock we have

$$4\left(\sum_{l=1}^{r} |(P_1^s P_2^{t-s})_{jl} - h_l|\right)^2 \leq \sum_k \frac{1}{h_k}\left(P_1^s P_2^{t-s}\right)_{jk}^2 - 1.$$

But $h_j(P_i^k)_{jl} = h_l(P_i^k)_{lj}, i = 1,2$ and for any integer $k \geq 1$. Hence it is easy to see that $h_j(P_1^s P_2^{t-s})_{jl} = h_l(P_2^{t-s} P_1^s)_{lj}$, so that

$$\sum_k \frac{\left(P_1^s P_2^{t-s}\right)_{jk}^2}{h_k} = \frac{(P_1^s P_2^{t-s} P_1^s)_{jj}}{h_j}.$$

Let D be a diagonal matrix with j^{th} diagonal entry $\sqrt{h_j}$. Let $DP_iD^{-1} = \Gamma_i B_i \Gamma_i^T$ where B_i is diagonal with eigenvalues of P_i and $\Gamma_i\Gamma_i^T = I$. Therefore,

$$\left(P_1^s P_2^{2(t-s)} P_1^s\right)_{jj} = \sum_l \sum_k a_{jl} b_{lk} a_{jk} \tag{14}$$

where $a_{jl} = (\Gamma_1 B_1 \Gamma_1^T)_{jl}$ and $b_{lk} = (\Gamma_2 B_2 \Gamma_2^T)_{lk}$.

Now, if $(B_1)_{zz} = 1$, $a_{jl} = \sum_w \Gamma_{1,jw} B_{1,ww}^s \Gamma_{1,lw} \leq \Gamma_{1,jz}\Gamma_{1,lz} + (\beta_1^{(*)})^s \sum_{w \neq z} \Gamma_{1,jw}\Gamma_{1,lw}$. Hence if $l \neq j$ $a_{jl} \leq a'_{jl} \equiv (1-(\beta_1^{(*)})^s)\Gamma_{1,jz}\Gamma_{1,lz} = (1-(\beta_1^{(*)})^s)\sqrt{h_j h_l}$ while if $l = j$ $a_{jj} \leq a'_{jj} \equiv \Gamma_{1,jz}^2 + (\beta_1^{(*)})^s(1-\Gamma_{1,jz}^2) = (1-(\beta_1^{(*)})^s)h_j + (\beta_1^{(*)})^s$. Similarly if $k \neq l$ $b_{lk} \leq b'_{lk} \equiv (1-(\beta_2^{(*)})^{2(t-s)})\sqrt{h_l h_k}$ while if $k = l$ $b_{ll} \leq b'_{ll} \equiv (1-(\beta_2^{(*)})^{2(t-s)})h_l + (\beta_2^{(*)})^{2(t-s)}$. Hence (14) is bounded by $\sum_l \sum_k a'_{jl} b'_{lk} a'_{jk}$ which after a bit of algebraic manipulation becomes $\left(1-(\beta_1^{(*)})^{2s}(\beta_2^{(*)})^{2(t-s)}\right)h_j + (\beta_1^{(*)})^{2s}(\beta_2^{(*)})^{2(t-s)}$. Thus the bound in (13) follows.

Comparing theorems 1 and 2, if $\beta_2^{(*)} \leq \beta_1^{(*)}$, a tighter bound on the variation distance arises by switching. In this sense we argue that switching accelerates convergence.

4.2 Simulation Results

The simulation study is patterned after that of Section 3.2 taking u to be three dimensional, high enough to allow interesting structure while again keeping computations manageable. We assume that f (= h, here) is either trivariate normal or a bimodal mixture of two trivariate normals. We utilize a random increment trivariate normal proposal transition kernel with covariance matrix identity. We examine the benefit of adaptive switching in the following illustrative way. We first run m parallel strings each for t Metropolis steps obtaining $\mathbf{u}_j^{(t)}, j = 1, 2, \ldots, m$. Secondly, using the output of the first s Metropolis steps $\mathbf{u}_j^{(s)}, j = 1, 2, \ldots, m$, we compute $\Sigma_{\mathbf{u}}$ the sample covariance matrix of the $\mathbf{u}_j^{(s)}$. We then switch the covariance matrix of the proposal transition kernel to this $\Sigma_{\mathbf{u}}$ and run an additional $t - s$ Metropolis steps to obtain $\mathbf{u}_j^{*(t)}, j = 1, 2, \ldots, m$. We compare the L^1 distance between f and the density $h^{(t)}$ of the $\mathbf{u}_j^{(t)}$ with that between f and $h^{*(t)}$, the density of $\mathbf{u}_j^{*(t)}$, i.e., $J^{(t)} = \int |h^{(t)} - f|$ and $J^{(t)}(\Sigma_{\mathbf{u}}) = \int |h^{*(t)} - f|$. Of course both $h^{(t)}$ and $h^{*(t)}$ are unknown and $h^{*(t)}$ is random. $J^{(t)}$ can be calculated arbitrarily accurately by making m very large, obtaining a kernel density estimate $\hat{h}^{(t)}$ of $h^{(t)}$ and then using Monte Carlo integration with draws from f to calculate $\int |\hat{h}^{(t)} - f| = \int f |\frac{\hat{h}^{(t)}}{f} - 1|$. On the other hand since $\Sigma_{\mathbf{u}}$ varies with the sample, $\mathbf{u}_j^{(s)}$, we treat $J^{(t)}(\Sigma_{\mathbf{u}})$ as random. That is, here (as in practice) we do not take m so large that $\Sigma_{\mathbf{u}}$ is essentially the covariance of $\mathbf{u}_j^{(s)}$, whence the new transition kernel is essentially fixed. Instead our simulation replicates adaptive switching to obtain the distribution of $J^{(t)}(\Sigma_{\mathbf{u}})$ and to compare with $J^{(t)}$. The $J^{(t)}(\Sigma_{\mathbf{u}})$ are still computed by Monte Carlo integration.

Our two illustrative cases are as follows, Case I: $f(\mathbf{u}) = h(\mathbf{u}) = N_3(\mu, \mathbf{I} + 911^{\mathbf{T}})$ where $\mu = (-5, 5, 15)^T$ and Case II: $f(\mathbf{u}) = h(\mathbf{u}) = 0.4 \times N_3(\mathbf{0}, \mathbf{I} + 411^{\mathbf{T}}) + 0.6 \times N_3(151, \mathbf{I} + 911^{\mathbf{T}})$, where $\mathbf{I}$ is the identity matrix and $\mathbf{1}$ is a column vector of ones. In either case our starting proposal density is a unit three dimensional normal distribution. As before we ran (m =) 500 parallel chains. We have taken 1000 draws to calculate the distances. We have replicated 1000 $J^{(t)}(\Sigma_{\mathbf{u}})$ to evaluate the probability that the adaptive distance is less than the actual distance. For Case I, we take $s = 10, t = 30$ while for Case II we take $s = 50, t = 110$. Table 2 summarizes the encouraging findings. For a fixed m, in general, performance is sensitive to s, t and the dimension of u.

{ Insert Table 2 here }

5 Two Examples

Here we consider two examples employing the proposed accelerators. The first ···es resampling in fitting a non linear model with a very badly behaved likelihood; the second ···strates the collapse of ergodic behavior under infinitely often stochastic adaptation of the transition kernel.

5.1 A Nonlinear Model

Bates and Watts (1988 page 110-121) discuss the analysis of a data set on the utilization of nitrite in bush beans as a function of light intensity on each of two days. Assuming zero utilization at zero light intensity and an asymptote as light intensity increases, they investigate several nonlinear normal models with homogeneous variance using maximum likelihood analysis. A Michaelis-Menten form, which provided adequate fit, models the mean utilization as $(\theta_1 + \phi x_2)x_1/(\theta_2 + x_1 + \theta_3 x_1^2)$ where x_1 is the light intensity and $x_2 = 0$ for day 1, $= 1$ for day 2. The likelihood analysis yielded the summary given in Table 3. The mean square error yields the variance estimate $\hat{\sigma}^2 = 647723.0$. The correlation matrix indicates a very poorly behaved likelihood surface; in fact as we shall see, the surface is more pathological than the authors realized.

{ Insert Table 3 here }

We perform a Bayesian analysis with extremely vague priors on the different parameters, so that, up to standardization, the posterior density is essentially the likelihood surface. We show the benefit of resampling using the Gibbs sampler. The complete conditionals for θ_1, ϕ, σ^2 are standard. For θ_2 and θ_3 we used Metropolis subchains for 50 steps with Gaussian proposals having standard deviations essentially the asymptotic standard errors. We eschewed reparametrization since this would sacrifice easy sampling for θ_1, ϕ and σ^2. Initially we started ten chains in the vicinity of the MLE. Autocorrelation in any individual chain is extremely high, requiring roughly 1500 iterations to die down to insignificance. Because the joint posterior for θ_1, θ_2 and θ_3 is so nearly singular, even after several hundred thousand iterations of the ten chains (requiring more than 24 hours

of run time), we cannot capture tail behavior adequately or conclude convergence. However, as an approximate benchmark we obtained kernel density estimates for the θ_i and ϕ based upon a sample of size 500, 50 from each chain using every 1500 th iteration after 100,000. These estimates appear as the solid curves in figure 1. We then ran 500 parallel chains out to 100 iterations doing no resampling as well as 500 parallel chains out to 100 iterations resampling once at iteration 5. The kernel density estimates for the θ_i and ϕ without resampling appears as the dashed curves in figure 1, those with resampling as the dotted curves. Clearly, even a single resampling is beneficial. Lastly note that, the standard errors of the MLE's supplied by Bates and Watts are not useful, e.g., $\hat{\theta}_1 \pm 3\mathrm{SE}_{\hat{\theta}_1}$ includes approximately 50 % of the marginal posterior mass for θ_1.

5.2 A Pathological Example

We remarked in Sections 2.3 and 2.4 that, when stochastic adaptation is employed at each transition, even if $h(\mathrm{u})$ is the unique stationary distribution for each adaptive transition kernel, this does not assure that the resultant MCMC algorithm converges or, if it does, that it converges to h. Analytic examination of non-stationary Markov chains with stochastic choice of transition kernel is generally infeasible. However, we can give a simple example due to G. O. Roberts (personal communication) where the additional randomness introduced in the selection of the transition kernel changes the overall ergodic behavior of the chain. Assume u is two dimensional with $f(\mathrm{u}) = h(\mathrm{u}) = BVN(0,0,1,1,\rho)$, $\rho > 0$. Suppose P_1 is the transition kernel associated with the Gibbs sampler, i.e., we draw $u_2^{(t)}|u_1^{(t-1)} \sim N(\rho u_1^{(t-1)}, 1-\rho^2)$ and $u_1^{(t)}|u_2^{(t)} \sim N(\rho u_2^{(t)}, 1-\rho^2)$. Then f is the invariant distribution associated with P_1. Suppose P_2 is the transition kernel corresponding to exact sampling of u through the principal components transformation $v_1 = u_1 + u_2, \quad v_2 = u_1 - u_2$. That is v_1 and v_2 are independent with $v_1 \sim N(0, 2(1+\rho)), v_2 \sim N(0, 2(1-\rho))$. Inverting $(\mathrm{v})^T = (v_1, v_2)^T$ to solve for $(\mathrm{u})^T = (u_1, u_2)^T$ yields a draw from f.

Now suppose at iteration t we take P_1 if $u_1^{(t-1)} > 0$, P_2 otherwise. A trajectory from this chain will tend to show more positive $u_1^{(t)}$ than negative. This is clear since if $u_1^{(t-1)} < 0$ we have a 0.5 chance that $u_1^{(t)} > 0$ while if $u_1^{(t-1)} > 0$ simple calculation shows that $u_1^{(t)}|u_1^{(t-1)} \sim N(\rho^2 u_1^{(t-1)}, 1-\rho^4)$ whence $u_1^{(t)}$ has a chance greater than 0.5 of being positive. Hence the ergodic average of the $u_1^{(t)}$ will be positive and the ergodic estimate of $P(u_1 > 0) > \frac{1}{2}$. Thus u_1 can not have $N(\mathrm{C}, 1)$ as its stationary distribution. In fact, by simulation investigation using 1000 parallel strings each

starting from $u_1^{(0)} \sim N(0,1)$, the supposed stationary distribution, we can observe the marginal distribution of $u_1^{(t)}$ at various t's as well as the behavior of ergodic estimates under this scheme. Table 4 illustrates such calculations for $\rho = 0.9$.

{ Insert Table 4 here }

References

[1] Applegate, D., Kannan, R., and Polson, N. (1990). Random polynomial time algorithms for sampling from joint distributions, Technical Report, Department of Statistics, Carnegie Mellon University.

[2] Bates, D. M., and Watts, D. G. (1988). *Nonlinear Regression Analysis & Its Application.* New York: John Wiley and Sons.

[3] Besag, J. and Green, P. (1993). Spatial statistics and Bayesian computation. *Journal of the Royal Statistical Society B* **56** 25-38.

[4] Clifford, P. (1993). Discussion to the meeting on the Gibbs Sampler and other Markov chain Monte Carlo Methods. *Journal of the Royal Statistical Society B* **55** 39-40.

[5] Devroye L. (1986). *Non-uniform Random Variate Generation.* New York: Springer.

[6] Devroye, L. and Gyorfi, L. (1985). *Nonparametric Density Estimation: The L_1 view*, New York: John Wiley and Sons.

[7] Diaconis, P. and Stroock, D. (1991). Geometric bounds for eigenvalues of Markov chains. *The Annals of Applied Probability* **1** 36-61.

[8] Gelfand, A. E. and Carlin, B. P. (1993). Maximum likelihood estimation for constrained or missing data models. *Canadian Journal of Statistics* **21** 303-311.

[9] Gelfand, A. E. and Smith, A. F. M. (1990). Sampling based approaches to calculating marginal densities. *Journal of the American Statistical Association* **85** 398-409.

[10] Gelfand, A. E., Smith, A. F. M. and Lee, T (1992). Bayesian analysis of constrained parameter and truncated data models. *Journal of the American Statistical Association* **87** 523-532.

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

[12] 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.

[13] Geyer, C. J. (1993). Practical Markov chain Monte Carlo. *Statistical Science* 7 473-483.

[14] Geyer, C. J. and Thompson, E. A. (1992). Constrained Monte Carlo maximum likelihood estimation for dependent data (with discussion). *Journal of the Royal Statistical Society B* 54 657-700.

[15] Gilks, W. R., Roberts, G. O., and George, E. (1992). Adaptive direction sampling, Research Report 92-31, Statistical Laboratory, University of Cambridge.

[16] Glick, F. P. (1974). Consistency conditions for probability estimators and integrals of density estimators. *Utilitas Mathematica* 6 61-74.

[17] Hastings W. K. (1970). Monte Carlo sampling methods using Markov chains and their applications. *Biometrika* 57 97-109.

[18] Iosifescu, M. (1980). *Finite Markov Processes and Their Applications.* Romania: John Wiley and Sons.

[19] Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., and Teller, E. (1953). Equations of state calculations by fast computing machines. *Journal of Chemical Physics* 21 1087-1091.

[20] Müller, P. (1994). A generic approach to posterior integration and Gibbs sampling. *Journal of the American Statistical Association*, to appear.

[21] Raftery, A. and Lewis, S. (1992). How many iterations in the Gibbs sampler? *Bayesian Statistics 4* (eds J. M. Bernado, J. O. Berger, A. P. Dawid and A. F. M. Smith), Oxford: Oxford University Press, 765-776.

[22] Ripley, B. D. (1987). *Stochastic Simulation.* New York: John Wiley and Sons.

[23] Ritter, C. and Tanner, M. A. (1992). Facilitating the Gibbs sampler; the Gibbs stopper and the griddy-Gibbs sampler. *Journal of the American Statistical Association* **87** 861-868.

[24] Roberts, G. O. and Smith A. F. M. (1992). Simple conditions for the convergence of the Gibbs sampler and Metropolis-Hastings algorithms. Technical Report 92-16, Department of Mathematics, Statistics Section, Imperial College, London.

[25] Rubin D. B. (1988). Using the SIR algorithm to simulate posterior distribution. *Bayesian Statistics 3* (eds, J. M. Bernado, M.H. DeGroot, D. V. Lindley and A. F. M. Smith), Oxford: Oxford University Press, 395-402, (with discussion).

[26] Schervish, M. J. and Carlin, B. P. (1992). On the convergence of successive substitution sampling. *Journal of Computational and Graphical Statistics* **1** 111-127.

[27] Silverman, B. W. (1986). *Density Estimation for Statistics and Data Analysis.* London: Chapman and Hall.

[28] Smith, A. F. M. and Gelfand, A. E. (1992). Bayesian Statistics without tears : a sampling resampling perspective. *The American Statistician* **46** 84-88.

[29] Tierney, L. (1994). Markov Chains for Exploring Posterior Distributions. *The Annals of Statistics*, to appear.

[30] Zellner, A. and Min C. (1992). Gibbs sampler convergence criteria (GSC^2). Technical Report, Graduate School of Business, University of Chicago.

Table 1. Simulation for Resampling

h	Normal	Normal	Mixture
$h^{(0)}(u_2)$	$N(2,9)$	$N(2,0.09)$	$N(0,1)$
J	0.7626	1.3010	0.2485
$P(J_m^{(1)}) < J)$	1.0	1.0	1.0
$E(J_m^{(1)})$	0.11	1.0658	0.1542
$\mathrm{Var}(J_m^{(1)})$	0.0012	0.00036	0.00056
$\mathrm{Med}(J_m^{(1)})$	0.1075	1.0661	0.1543

Table 2. Simulation for switching P

h	Normal	Mixture
$J^{(t)}$	1.6335	1.3260
$P(J^{(t)}(\Sigma_u) < J^{(t)})$	1.0	1.0
$E(J^{(2)}(\Sigma_u))$	1.0251	1.1014
$\mathrm{Var}(J^{(t)}(\Sigma_u))$	0.0190	0.0016
$\mathrm{Med}(J^{(t)}(\Sigma_u))$	1.0056	1.0976

Table 3. Parameter Summary for the Michaelis-Menten model

Parameter	Estimate	Standard Error	t Ratio	Correlation Matrix			
θ_1	70096	16443	4.3	1.00			
θ_2	139.4	39.3	3.6	1.00	1.00		
θ_3	0.01144	0.00404	2.8	0.99	0.99	1.00	
ϕ	-5381	1915	-2.8	-0.69	-0.66	-0.66	1.00

Table 4. Simulation for the pathological example $\rho = 0.9$

t(iteration)	$E(u_1^{(t)})$	$P(u_1^{(t)} > 0)$	$\sum_t u_1^{(t)}/t$	$\sum_t 1_{(0,\infty)}(u_1^{(t)})/t$
5	0.5957	0.767	0.4840	0.7232
10	0.5953	0.777	0.5268	0.7390
50	0.6018	0.750	0.5710	0.7516
100	0.6047	0.784	0.5796	0.7534
500	0.5553	0.744	0.5895	0.7572
1000	0.6093	0.777	0.5938	0.7589

Figure 1: Estimated Posteriors for the Nonlinear Model of Section 5.1.
— = "true", -- = no resampling, ... = one resampling.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) On Markov Chain Monte Carlo Acceleration		5. TYPE OF REPORT & PERIOD COVERED Technical 6. PERFORMING ORG. REPORT NUMBER 480
7. AUTHOR(s) Alan E. Gelfand and Sujit K. Sahu		8. CONTRACT OR GRANT NUMBER(s) N00014-92-J-1264
9. PERFORMING ORGANIZATION NAME AND ADDRESS Department of Statistics Stanford University Stanford, CA 94305-4065		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS NR-042-267
11. CONTROLLING OFFICE NAME AND ADDRESS Office of Naval Research Statistics & Probability Program Code 1111		12. REPORT DATE 4 April 1994 13. NUMBER OF PAGES 21
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

Also issued as Technical Report No. 93-11, Department of Statistics, University of Connecticut.

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Adaptive chains; Gibbs sampler; L^1-convergence; Markov chain Monte Carlo; Metropolis-Hastings algorithm; Rejection method; Resampling

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

see reverse side

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

20. ABSTRACT

Markov chain Monte Carlo (MCMC) methods are currently enjoying a surge of interest within the statistical community. The goal of this work is to formalize and support two distinct adaptive strategies which typically accelerate the convergence of a MCMC algorithm. One approach is through resampling; the other incorporates adaptive switching of the transition kernel. Support is both by analytic arguments and simulation study. Application is envisioned in low dimensional but non-trivial problems. Two pathological illustrations are presented. Connections with reparametrization are discussed as well as possible difficulties with infinitely often adaptation.