Optimal Scaling for Various Metropolis–Hastings Algorithms

Gareth O. Roberts and Jeffrey S. Rosenthal

Abstract. We review and extend results related to optimal scaling of Metropolis–Hastings algorithms. We present various theoretical results for the high-dimensional limit. We also present simulation studies which confirm the theoretical results in finite-dimensional contexts.

Key words and phrases: Adaptive triangulations, AIC, density estimation, extended linear models, finite elements, free knot splines, GCV, linear splines, multivariate splines, regression.

1. INTRODUCTION

Metropolis–Hastings algorithms are an important class of Markov chain Monte Carlo (MCMC) algorithms [see, e.g., Smith and Roberts (1993), Tierney (1994) and Gilks, Richardson and Spiegelhalter (1996)]. Given essentially any probability distribution (the "target distribution"), these algorithms provide a way to generate a Markov chain $\mathbf{X}_0, \mathbf{X}_1, \ldots$ having the target distribution as a stationary distribution.

Specifically, suppose that the target distribution has density π (usually with respect to Lebesgue measure). Then, given $\mathbf{X}_n$, a "proposed value" $\mathbf{Y}_{n+1}$ is generated from some prespecified density $q(\mathbf{X}_n, \mathbf{y})$ and is then accepted with probability $\alpha(\mathbf{X}_n, \mathbf{Y}_{n+1})$, given by

$$
\begin{aligned}
&\alpha(\mathbf{x}, \mathbf{y}) \\
(1)\quad &= \begin{cases} \min\left\{\dfrac{\pi(\mathbf{y})}{\pi(\mathbf{x})}\dfrac{q(\mathbf{y}, \mathbf{x})}{q(\mathbf{x}, \mathbf{y})}, 1\right\}, & \pi(\mathbf{x})q(\mathbf{x}, \mathbf{y}) > 0, \\ 1, & \pi(\mathbf{x})q(\mathbf{x}, \mathbf{y}) = 0. \end{cases}
\end{aligned}
$$

If the proposed value is accepted, we set $\mathbf{X}_{n+1} = \mathbf{Y}_{n+1}$; otherwise, we set $\mathbf{X}_{n+1} = \mathbf{X}_n$. The function $\alpha(\mathbf{x}, \mathbf{y})$ above is chosen precisely to ensure that the Markov chain $\mathbf{X}_0, \mathbf{X}_1, \ldots$ is reversible with respect to the target density $\pi(\mathbf{y})$, so that the target density is stationary for the chain.

Gareth O. Roberts is Professor of Statistics, Department of Mathematics and Statistics, Fylde College, Lancaster University, Lancaster LA1 4YF, England (e-mail: g.o.robert@lancaster.ac.uk). Jeffrey S. Rosenthal is Professor, Department of Statistics, University of Toronto, Toronto, Ontario, Canada M5S 3G3. (e-mail: jeff@math.toronto.edu).

In applying Metropolis–Hastings algorithms, it is necessary to choose the proposal density $q(\mathbf{x}, \mathbf{y})$. Typically, q is chosen from some family of distributions, for example, normal distributions centered at $\mathbf{x}$. There is then a need to select the "scaling" of the proposal density (e.g., the variance of the normal distributions) in order to have some level of optimality in the performance of the algorithm.

An important special case of the Metropolis–Hastings method is the symmetric random-walk Metropolis (RWM) algorithm. In this case, we take the Markov chain described by q alone to be a simple symmetric random walk, so that $q(\mathbf{x}, \mathbf{y}) = q(\mathbf{y} - \mathbf{x})$ and the single-argument q density (sometimes called the *increment density*) is a symmetric function about $\mathbf{0}$. In this case, the acceptance probability in (1) simplifies to

$$
\alpha(\mathbf{x}, \mathbf{y}) = \min\left\{1, \frac{\pi(\mathbf{y})}{\pi(\mathbf{x})}\right\}.
$$

1.1 A First Example

A simple example illustrates the issues involved. Suppose the target $\pi(y)$ is the standard normal density. Suppose also that the proposal density $q(x, y)$ is taken to be the normal density $N(x, \sigma^2)$, where σ is to be chosen. That is, if $X_n = x$, then we choose $Y_{n+1} \sim N(x, \sigma^2)$ and set X_{n+1} to either Y_{n+1} [with probability $\alpha(x, y)$] or X_n [with probability $1 - \alpha(x, y)$], as above.

It is intuitively clear that we can make the algorithm arbitrarily poor by making σ either very small or very large. For extremely small σ, the algorithm will propose small jumps. These will almost all be accepted (since α will be approximately 1, because of the continuity of π and q). However, the size of the jumps is too small for the
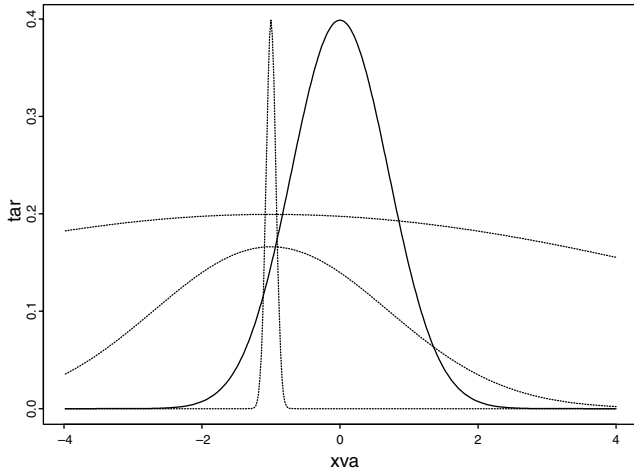
FIG. 1. *Standard normal target density with proposal distributions for a normal proposal random walk Metropolis algorithm with three alternative proposal scalings. The alternative proposals are all centered about the current point $x = -1$ and are shown using dotted lines.*

algorithm to explore the space rapidly, and the algorithm will therefore take a long time to converge to its stationary distribution. On the other hand, if σ is taken to be extremely large, the algorithm will nearly always propose large jumps to regions where π is extremely small. It will therefore reject most of its proposed moves and hence stay fixed for large numbers of iterations. It seems reasonable that there exist "good" values for σ, between these two extremes, where the algorithm performs optimally. This is illustrated by simulation in Figure 1.

Figure 2 shows trace plots giving examples of all three types of behavior: a situation where the proposal variance is too high so the chain gets stuck in different regions of the space, a situation where the proposal variance is too low so that the chain crawls to stationarity and a case where the proposal variance is tuned appropriately (using Theorem 1 below) and the chain converges at a reasonable rate.

1.2 Efficiency of Markov Chains and Related Concepts

To compare different implementations of MCMC, we require some notion of efficiency of Markov chains to guide us. For an arbitrary square-integrable function g, we define its integrated autocorrelation time by

$$\tau_g = 1 + 2 \sum_{i=1}^{\infty} \mathrm{Corr}(g(X_0), g(X_i)),$$

where X_0 is assumed to be distributed according to π. If a central limit theorem for X and g exists, then the variance of the estimator $\sum_{i=1}^{n} g(X_i)/n$ for

estimating $\mathbf{E}(g(X))$ is approximately $\mathrm{Var}_{\pi}(g(X)) \times \tau_g/n$. This suggests that the efficiency of Markov chains can be compared by comparing the reciprocal of their integrated autocorrelation times, that is,

$$
\begin{aligned}
(2) \quad e_g(\sigma) &= \left(\mathrm{Var}_{\pi}(g(X)) \tau_g \right)^{-1} \\
&= \left(\lim_{n \to \infty} n \, \mathrm{Var}\left(\frac{\sum_{i=1}^{n} g(X_i)}{n} \right) \right)^{-1}.
\end{aligned}
$$

However, this measure of efficiency is highly dependent on the function g chosen. Thus, for two different Markov chains, different functions g could order their efficiency differently. Where specific interest is in a particular function g, $e_g(\sigma)$ is a sensible criterion to be using, but where interest is in a whole collection of functionals of the target distribution (perhaps the cdf of a component of interest for example), its use is more problematic.

We shall see later in Section 2.2 that, in the high-dimensional limit, at least for algorithms which "behave like diffusion processes," all efficiency measures e_g are virtually equivalent. In such cases, we can produce a unique limiting efficiency criterion. We shall use this criterion to define a function-independent notion of efficiency.

It will turn out that a quantity related to efficiency is the algorithm's *acceptance rate*, that is, the probability in stationarity that the chain's proposed move is accepted. Alternatively, by the ergodic theorem, the acceptance rate equals the long-term proportion of proposed moves accepted by the chain:

$$
\begin{aligned}
(3) \quad a &= \int \alpha(\mathbf{x}, \mathbf{y}) \pi(\mathbf{x}) q(\mathbf{x}, \mathbf{y}) \, d\mathbf{x} \, d\mathbf{y} \\
&= \lim_{n \to \infty} n^{-1} \#\{\text{accepted moves}\},
\end{aligned}
$$

the latter identity indicating a sensible way to estimate a without extra computational effort.

1.3 Goal of This Paper

The goal of this paper is to review and further develop guidelines for choosing good values of σ in situations such as those in Section 1.1, especially when the dimension of the space is large. The guidelines we describe are in terms of the asymptotic overall acceptance rate of the chain, which makes them easily implementable. Much, but not all, of what follows is a synthesis of Roberts, Gelman and Gilks (1997), Roberts (1998), Roberts and Rosenthal (1998) and Breyer and Roberts (2000). Extensions to consider heterogeneity in scale are also given.

Our results provide theoretical justification for a commonly used strategy for implementing the multivariate random-walk Metropolis algorithm, which dates back at least as far as Tierney (1994).
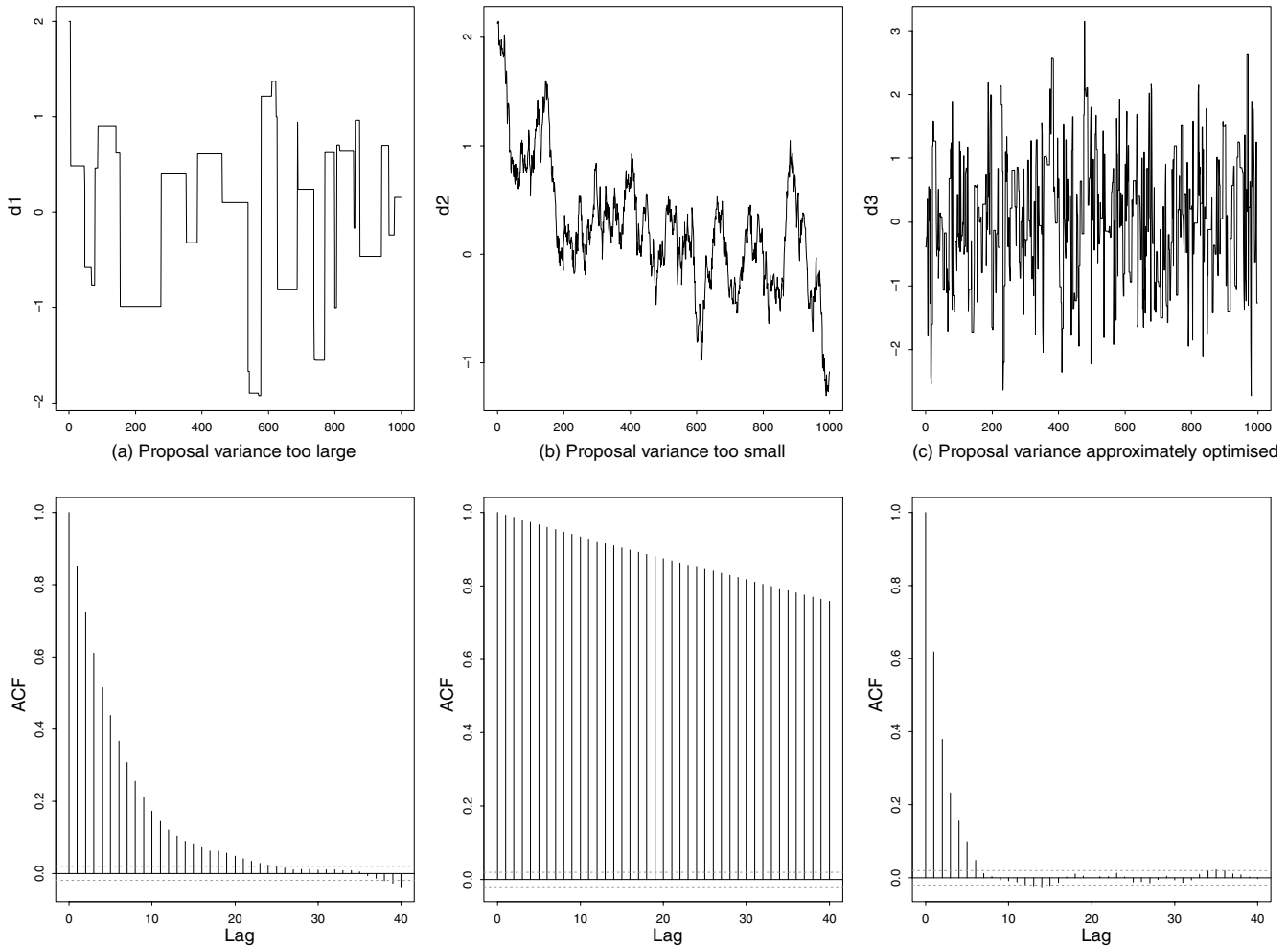
FIG. 2.   *Simple Metropolis algorithm with* (a) *too-large variance* (*left plots*), (b) *too-small variance* (*middle*) *and* (c) *appropriate variance* (*right*). *Trace plots* (*top*) *and autocorrelation plots* (*below*) *are shown for each case.*

The strategy involves estimating the correlation structure of the target distribution, either empirically based on a pilot sample of MCMC output or perhaps numerically from curvature calculations on the target density itself, and using a proposal distribution for the random walk algorithm to be a scalar multiple of the estimated correlation matrix. In the Gaussian case, if the correlation structure is accurately estimated, then scaling the proposal in each direction proportional to the target scaling can be shown to optimize the algorithm's efficiency.

With knowledge of such optimal scaling properties, the applied user of Metropolis–Hastings algorithms can tune "their" proposal variances by running various versions of the algorithm, with variance at the appropriate level [e.g., $O(d^{-1})$ or $O(d^{-1/3})$, say], and making finer adjustments to get the approximate acceptance rate close to optimal.

Thus, optimal scaling results are of importance in the practical implementation of Metropolis–Hastings algorithms.

1.4 Outline of This Paper

In Section 2, we shall review the results of Roberts, Gelman and Gilks (1997) for random-walk Metropolis (RWM) algorithms with target distribution consisting of approximately i.i.d. components. They showed that optimality is achieved if the variance of the proposal distribution is $O(d^{-1})$ and the overall acceptance rate (i.e., the long-run expected proportion of accepted moves) is close to 0.234. In Section 3, we review the results of Roberts (1998), who showed that the same acceptance rate is approximately optimal for a quite different class of discrete RWM algorithms.

In Section 4, we consider the Metropolis-adjusted Langevin (MALA) algorithms, which

are Metropolis–Hastings algorithms whose proposal distributions make clever use of the gradient of the target density. We review results of Roberts and Rosenthal (1998), who proved that, for target distributions of the form (5), optimality is achieved if the variance of the proposal distribution is $O(d^{-1/3})$ and we have an overall acceptance rate close to 0.574. That is, MALA algorithms have a larger optimal proposal variance and a larger optimal acceptance rate. This demonstrates that MALA algorithms asymptotically mix considerably faster than do RWM algorithms.

In Section 5, we shall review the results of Breyer and Roberts (2000), who showed that the optimal acceptance rate for the random-walk Metropolis algorithm is *again* optimal for a class of Markov random-field models, but only when the local correlations are small enough to avoid phase-transition behavior.

In Section 6, we extend some of the above results to cases other than target distributions of the form (5). In particular, we consider the case where the target distribution consists of components which each have different scaling C_i. Now, if the C_i values are known, then it is best to scale the proposal distribution components also proportional to C_i; that way, the resulting chain corresponds precisely to the previous case where all components are identical. However, if the quantities C_i are unknown and the proposal scaling is taken to be the same in each component, then we prove the following. Once again, the optimal acceptance rate is close to 0.234. However, in this case, the algorithm's asymptotic efficiency is multiplied by a factor $E(C_i)^2/E(C_i^2)$ compared to what it could have been with different proposal distribution scaling in each component. Since we always have $E(C_i)^2/E(C_i^2) < 1$ for nonconstant $\{C_i\}$, this shows that the RWM algorithm becomes *less* efficient when the components of the target distribution have significantly different scalings. For MALA algorithms, we show that this effect is even stronger, depending instead on the *sixth* moments of the target component scalings $\{C_i\}$. Simulations are presented which confirm our theoretical results.

In Section 7, we consider various examples related to our theorems. In some cases, the examples fit in well with our theorems, and we provide simulations to illustrate this. In other cases, the examples fall outside our theoretical results and exhibit different behavior, as we describe.

2. PROPOSAL VARIANCE OF CONTINUOUS i.i.d. RWM

Suppose we are given a density π with respect to Lebesgue measure on $\mathbf{R}^d$ and a class of symmetric proposal increment densities for the RWM algorithm given by

$$(4) \qquad q_\sigma(\mathbf{x}) = \sigma^{-d} q(\mathbf{x}/\sigma),$$

where q is some fixed density and where $\sigma > 0$ denotes some measure of dispersion (typically the standard deviation) of the distribution with density q_σ. Thus, from a random variable Z with density q, we can produce one with density q_σ as $Z_\sigma = \sigma Z$. As already noted, we can make the Metropolis–Hastings algorithm inefficient by taking σ either very small or very large, and our goal is to determine optimal choices of σ.

For ease of analysis, we here let π have the simple product form

$$(5) \qquad \pi(\mathbf{x}) = \prod_{i=1}^{d} f(x_i)$$

and suppose that the proposal is of the form

$$(6) \qquad q_\sigma(\mathbf{x})\,d\mathbf{x} \sim N(0, I_d \sigma_d^2),$$

a normal distribution with mean 0 and variance σ_d^2 times the identity matrix. Our goal is to characterize the optimal values of σ_d^2 in a practically useful way.

We shall assume that there exists a positive constant ℓ such that

$$(7) \qquad \sigma_d^2 = \ell^2/d.$$

Indeed, if the variance is larger than $O(d^{-1})$, then the acceptance rate of the algorithm converges to 0 too rapidly, whereas for smaller order scalings, the jumps of the algorithm (which are almost all accepted) are too small. That is, it can be shown that taking σ_d^2 to be $O(d^{-1})$ is optimal. Our goal is then to optimize the choice of ℓ in (7) above.

For fixed d, the algorithm involves d components, $X^{(1)}, \ldots, X^{(d)}$, say, which are constantly interacting with each other. Thus, there is no reason to think that any proper subset of the d components should itself form a Markov chain. However, for target densities of the form (5), each component acts like an independent Markov chain as $d \to \infty$. Therefore, by considering any one component (say $X^{(1)}$), by independence this gives us information about the behavior of any finite collection of the components of interest. For simplicity, therefore, we shall write all our results in terms of convergence of single components.

Note that, when we consider dependent densities in Section 5, asymptotic independence of the constituent components is not achieved; therefore, to consider the limiting behavior, we need to look at a genuinely infinite-dimensional limit process. As a result of this, we refrain from a formal statement of that result.

2.1 RWM Algorithm as $d \to \infty$

For each component, since we are making steps of decreasing size according to (7) as $d \to \infty$, any individual component's movement will grind to a halt unless we somehow speed up time to allow steps of the algorithm to happen more quickly. We do this by stipulating that the algorithm is updated every d^{-1} time units. As $d \to \infty$, therefore, the algorithm makes smaller jumps more and more frequently, and its limit needs to be described as a continuous-time stochastic process. Since its jumps get smaller and smaller, the limiting trajectory will resemble a continuous sample path, and, as we shall see, it will be a diffusion process.

Let the RWM chain on $\mathbf{R}^d$ be denoted $\{X_n^{(1)}, \ldots, X_n^{(d)}\}$ and consider the related one-dimensional process given by

$$(8) \qquad Z_t^d = X_{[td]}^{(1)},$$

where $[\cdot]$ denotes the integer part function. That is, Z^d is a speeded-up continuous-time version of the first coordinate of the original algorithm, parameterized to make jumps [of size $O(\sigma_d) = O(d^{-1/2})$, by (7)] every d^{-1} time units. The process Z^d is not itself Markovian, since it only considers the first component of a d-dimensional chain. However, in the limit as d goes to ∞, the process Z^d converges to a Markovian limit, as we now describe.

We require a number of smoothness conditions on the density f in (5). The details of sufficient conditions appear in Roberts, Gelman and Gilks (1997), although these conditions can be weakened also. The essential requirements are that $\log f$ is continuously differentiable and that

$$(9) \qquad I \equiv \mathsf{E}_f\left[\left(\frac{f'(X)}{f(X)}\right)^2\right] < \infty.$$

We denote weak convergence by $\Rightarrow$, let B_t be standard Brownian motion and write $\Phi(z) = (1/\sqrt{2\pi})\int_{-\infty}^z e^{-s^2/2}\,ds$ for the cumulative distribution function of a standard normal distribution. The following result is taken from Roberts, Gelman and Gilks (1997).

THEOREM 1. *Consider the RWM* $\{X_n^{(1)}, \ldots, X_n^{(d)}\}$ *on* $\mathbf{R}^d$. *Define the process* Z_t^d *by* (8). *Suppose that* π *and* q *are given by* (5) *and* (6), *respectively, with* σ^2 *given by* (7). *Under the above regularity conditions on* f,

$$(10) \qquad Z^d \Rightarrow Z,$$

where Z *is a diffusion process which satisfies the stochastic differential equation*

$$(11) \quad dZ_t = h(\ell)^{1/2}dB_t + \frac{h(\ell)\nabla \log f(Z_t)}{2}\,dt,$$

with

$$(12) \qquad h(\ell) = \ell^2 \times 2\Phi\left(-\frac{\sqrt{I}\ell}{2}\right) := \ell^2 \times A_I(\ell)$$

and I *given by* (9). *Here the acceptance rate* $a = A_I(\ell) = 2\Phi(-\sqrt{I}\ell/2)$. *The speed of the limiting diffusion, as a function of this acceptance rate, is proportional to*

$$(13) \qquad A_I(\ell)\left[\Phi^{-1}\left(\frac{A_I(\ell)}{2}\right)\right]^2.$$

The scaling which gives rise to the optimal limiting speed of the diffusion (*and hence the optimal asymptotic efficiency of the algorithm*) *is given by*

$$\ell_{\text{opt}} \doteq \frac{2.38}{I^{1/2}},$$

and the corresponding optimal acceptance rate is

$$A_I(\ell_{\text{opt}}) \doteq 0.234.$$

For any fixed function g, *the optimal asymptotic efficiency* e_g [*as given in* (2)] *is proportional to* $1/d$.

This theorem thus says that, for RWM algorithms on target densities of the form $\prod_{i=1}^n f(x_i)$, the efficiency of the algorithm as a function of its asymptotic acceptance rate can be explicitly described and optimized. In particular, acceptance rates approximately equal to 0.234 lead to greatest efficiency of the algorithm. Since the process Z^d had time scaled by a factor of d, the theorem also says that the convergence time of the algorithm grows with dimension like $O(d)$ [or, equivalently, the efficiency is $O(1/d)$]. Hence, if the computation time for completing each iteration of the algorithm grows with d (which appears likely for most examples), then the overall complexity of the algorithm is $O(d^2)$.

The quantity I, which is the variance of the derivative of the log density of f, can be interpreted as a measure of "roughness" of the target distribution. It measures local variation in f rather than any global scale property. For Gaussian f, I is the reciprocal of the variance of the density f, so in this case (12) reduces to

$$h(\ell) = \ell^2 \times 2\Phi\left(-\ell/2\sqrt{\text{Var}(f)}\right).$$

The first graph in Figure 3 shows the function h of Theorem 1, as a function of ℓ, for fixed I. Different I values will distort the curve, causing its maximum to appear to the left or right for lower or higher values of I, respectively. The second graph shows $h(\ell)$ as a function of $A(\ell)$, specifically as the function $a \mapsto aA_I^{-1}(a)^2 = a[\Phi^{-1}(\frac{a}{2})]^2 \times 4/I$. Note that the shape of this second curve is independent
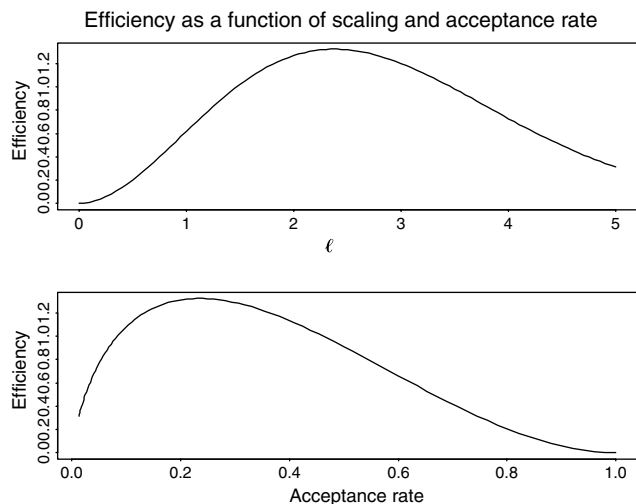
FIG. 3. *Efficiency of RWM as a function of ℓ (top) and of acceptance rate (bottom), in the infinite-dimensional limit. In this case, $I = 1$.*

FIG. 4. *Optimal scaling as a function of acceptance rate, using the minimum autocorrelation criterion, as dimension increases for the case of Gaussian target densities. This analysis comes from a simulation study on standard Gaussian target densities.*

of I and hence of f, apart from the global multiplicative factor of I^{-1}. Hence, the "optimal" acceptance rate is approximately $2\Phi(-\eta_{\text{opt}}/2) = 0.234$. Since the constant $4/I$ plays no role in the optimization of efficiency, we shall omit it when talking about the algorithm's *relative efficiency*.

REMARK. It must be stressed that Theorem 1 is an asymptotic result, valid for large d. In fact, the "appropriate" scaling in Figure 2(c), which minimizes the first-order autocorrelation of the algorithm, achieves acceptance rate 0.44; the asymptotics do not directly apply since $d = 1$ there. However, even in five dimensions, the optimal acceptance rate is so close to 0.234 as to not matter at all in practice. This can be seen in Figure 4, which is computed using the criterion of minimizing the first-order autocorrelation of the identity function for the algorithm on standard Gaussian densities. These data were taken from a simulation study which was first reported in Gelman, Roberts and Gilks (1996).

One interpretation of the efficiency curve, the second plot in Figure 3, is as follows. The number of iterations needed to reach a prescribed accuracy in estimating any function is proportional to the inverse of this efficiency. As a result, an algorithm with say 50% of the optimal efficiency would need to be run for twice as long to obtain the same accuracy of results. On the other hand, the importance of the precise optimal acceptance rate should not be overstated. The relative efficiency curve given in Figure 3 is relatively flat around 0.234, and any
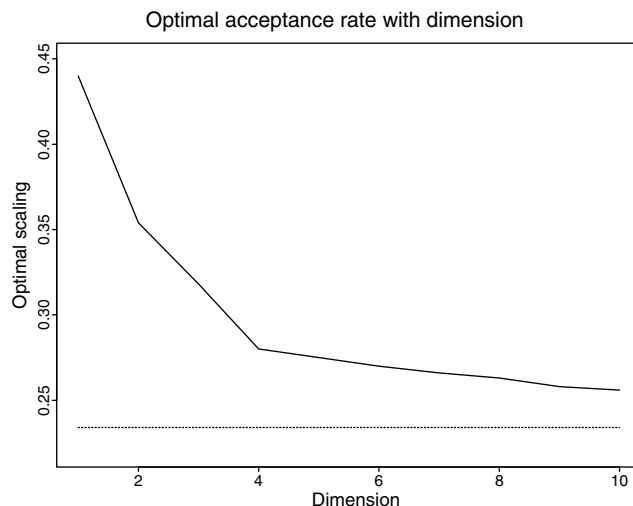
algorithm with acceptance rate between say 0.15 and 0.5 will be at least 80% efficient. It is therefore of little value to finely tune algorithms to the *exact* optimal values.

It is likely that the assumption of second-order differentiability could be relaxed to some extent, while leaving the asymptotics of Theorem 1 intact. On the other hand, the asymptotics of Theorem 1 may be completely altered if f is allowed to actually be discontinuous. For example, let f be the indicator function of $[0, 1]$. Then it is fairly straightforward to show that any scaling of the form (7) for any $\ell > 0$ will lead to acceptance rates converging to zero (in fact, exponentially quickly). In this case, optimal scaling will have to scale the jumps in each dimension by a term which converges to zero quicker than $d^{-1/2}$. Such questions are currently being investigated by Roberts and Yuen (2001); initial investigations suggest that the correct scaling in this case is $\sigma^2 = \ell^2/d^2$, with an optimal acceptance rate of approximately 0.13.

Theorem 1 applies only to the special case of a density of the form $\prod_{i=1}^d f(x_i)$, that is, corresponding to i.i.d. components. Minor generalizations are possible; for example, it suffices to have a density of the form $\prod_{i=1}^d f_i(x_i)$ where $f_i \to f$ [see Roberts, Gelman and Gilks (1997)]. However, it is reasonable to ask what happens if the different components are not identically distributed or are not independent. Such issues are examined in subsequent sections.

2.2 Assessing Efficiency of Markov Chains

Now, as already discussed in Section 1.2, the efficiency $e_g(\sigma)$ of a Markov chain can depend on the function g for which we are trying to obtain Monte Carlo estimates. Therefore, we cannot rely on integrated autocorrelation times for some particular function g to provide unambiguous criteria by which to assess efficiency.

However, if we look at high-dimensional problems in situations where the algorithm can be shown to converge to a diffusion process as in Theorem 1, it does not matter which function g we choose, and all functions lead to essentially the same notion of efficiency.

Indeed, suppose we let g be a function of the first variable $X^{(1)}$ of the chain. Recall that, for estimation of $\pi(g) := \int g(\mathbf{x})\pi(\mathbf{x})\,d\mathbf{x}$ for a function g using (4), a natural measure of efficiency is the inverse autocorrelation time formula (2). Define also

$$(14) \quad e_g^{(\infty)}(\ell) = \left(\lim_{T \to \infty} T\,\mathrm{Var}\left(\frac{\int_{t=0}^{T} g(Z_t)}{T} \right) \right)^{-1}$$

and the corresponding quantity for the standard Langevin diffusion

$$(15) \quad e_g^{L} = \left(\lim_{T \to \infty} T\,\mathrm{Var}\left(\frac{\int_{t=0}^{T} g(L_t)}{T} \right) \right)^{-1}.$$

Now from (14) it is easy to see that $e_g^{(\infty)}(\ell) = h(\ell)e_g^{L}$. Furthermore, from (2), for large d we can write

$$e_g^{(d)}(\ell d^{-1/2}) = \left(\lim_{T \to \infty} \lfloor Td \rfloor\,\mathrm{Var}\left(\frac{\sum_{i=1}^{\lfloor Td \rfloor} g(X_i^{(1)})}{\lfloor Td \rfloor} \right) \right)^{-1}$$

$$(16) \qquad = \left(\lim_{T \to \infty} \lfloor Td \rfloor\,\mathrm{Var}\left(\frac{\sum_{i=1}^{\lfloor Td \rfloor} g(Z_{i/d}^{d})}{\lfloor Td \rfloor} \right) \right)^{-1}$$

$$\approx \left(\lim_{T \to \infty} \lfloor Td \rfloor\,\mathrm{Var}\left(\frac{\int_0^{T} g(Z_s)\,ds}{T} \right) \right)^{-1}$$

$$\approx d^{-1}e_g^{(\infty)}(\ell),$$

so that

$$(17) \quad \lim_{d \to \infty} d\,e_g^{(d)}(\ell d^{-1/2}) = e_g^{(\infty)}(\ell) = h(\ell)e_g^{L}.$$

Since the only term on the right-hand side which depends on our scaling parameter ℓ is $h(\cdot)$, which is independent of the function of interest g, it follows that the corresponding optimization problem is independent of g as well. That is why Theorem 1 is useful regardless of the function g under investigation.

Note also that, for a diffusion process satisfying (11), the speed measure $h(\ell)$ can be understood in terms of simple autocorrelations of an arbitrary

function g. Indeed, it is easily demonstrated that, for small $\varepsilon > 0$, there is a positive constant $B_{\pi,g}$ which depends on the target density π and function of interest g, such that

$$(18) \quad \mathrm{Corr}(g(Z_\varepsilon), g(Z_0)) \sim 1 - B_{\pi,g}\varepsilon h(\ell).$$

Hence, maximizing h is equivalent to minimizing $\mathrm{Corr}(g(Z_\varepsilon), g(Z_0))$. For computational reasons, it is most convenient to just estimate single autocorrelations ρ_K of order $K > 0$ for some chosen function(s) g. We shall also translate our results in terms proportional to the number of iterations needed to estimate g to a desired accuracy, by defining

$$\text{convergence time} := \frac{-K}{\log(\rho_K)}.$$

3. RWM IN A DISCRETE STATE SPACE

Theorem 1 says that, in the limit as $d \to \infty$, certain RWM algorithms on continuous state space problems look like diffusion processes, and diffusion processes only make sense on a continuous state space. Hence, we might expect different asymptotic behavior for discrete state space problems. To investigate this, we consider the following discrete problem, which can be thought of as an analog of the continuous state space problem of Section 2.

Let $\pi\colon \{0,1\}^d \to [0,1]$ be the product measure of a collection of d independent random variables each taking the value 0 with probability p, or 1 with probability $1 - p$. Thus, π is a discrete distribution on the vertices of a d-dimensional hypercube given by

$$(19) \quad \pi(i_1, i_2, \ldots, i_d) = p^{|\{j;\, i_j=1\}|}(1-p)^{|\{j;\, i_j=0\}|}.$$

Assume without loss of generality that $p < 1/2$.

We shall consider the following *random scan* version of the RWM algorithm, for which we can find an explicit solution. Suppose that, at each iteration, the algorithm picks r sites uniformly at random, $S = \{j_1, j_2, \ldots, j_r\}$, say. The proposed move is then to change the value at each state in S and to leave all states in $\{1, \ldots, d\}\backslash S$ unaltered. From a current state $\mathbf{x}$, then, the algorithm proposes a move to state $\mathbf{y}$ with $y_j = x_j$ for $j \notin S$ and $y_j = 1 - x_j$ for $j \in S$. The acceptance rate a is given by the usual $\alpha(\mathbf{x}, \mathbf{y}) = 1 \wedge (\pi(\mathbf{y})/\pi(\mathbf{x}))$.

In this context, the notion of scaling needs to be modified: the number of sites r in S takes the place of variance in this situation. Clearly, the optimal choice of r will depend heavily on p. For p close to $1/2$, it will be possible to propose updates of large numbers of components without changing the value of π drastically. These moves will therefore be accepted with reasonable probability, and so the

optimal value of r is likely to be large. Conversely, for small values of p, if r is large, most proposed moves are likely to be rejected.

We again investigate the optimality problem in the limiting case as $d \to \infty$. Fix r (so that it does not depend on d) and let d go to infinity along odd values. Let $S^d_t = X^{(1)}_{[td]}$, so S^d is a continuous-time binary process (non-Markovian) making jumps at times which are integer multiples of $1/d$, analogous to the continuous model. The following result is taken from Roberts (1998).

THEOREM 2. (i) *Assume X_0 is distributed according to π. Then, as $d \to \infty$,*

$$S^d \Rightarrow S,$$

where S is a two-state continuous-time Markov chain with stationary distribution $(p, 1-p)$. The Q-matrix (describing its transition rates) for S has the form

$$Q = e(r) \times \begin{pmatrix} -(1-p) & 1-p \\ p & -p \end{pmatrix},$$

where $e(r)$ is available [and given in Roberts (1998) as an explicit Binomial expectation].

(ii) *Let $a(p, r)$ denote the acceptance rate for the algorithm for fixed target and algorithm parameters p and r, respectively. Let $p \nearrow 1/2$ in such a way that $\lambda = (1/2 - p)^2 r$ remains constant. Then, for p close to $1/2$, we can write*

$$e(r) \approx 2r \times a(p, r)$$

$$\approx \frac{2}{(1/2 - p)^2} \lambda \times 2\Phi(-2\sqrt{\lambda}).$$

If $p \approx 1/2$, then the optimal choice of r is approximately that achieving acceptance rate $a = 0.234$, and the efficiency curve as a function of acceptance rate converges to that of continuous RWM (as shown in Figure 3, bottom).

This theorem says that, in the discrete $\{0, 1\}^d$ model, if $p \approx 1/2$, then the optimal scaling properties are very similar to those of the continuous $\prod^d_{i=1} f(x_i)$ model of the previous section.

4. MALA ALGORITHM

As in the RWM case, we shall again consider the case where the target density takes the product form (5). We will consider the MALA proposal given by

$$(20) \quad \mathbf{Y}_{n+1} \sim N\left(\mathbf{X}_n + \frac{\sigma^2_d}{2} \nabla \log \pi(\mathbf{X}_n), \sigma^2_d I_d\right).$$

This proposal is chosen so as to mimic a Langevin diffusion for the density $\pi(\mathbf{y})$ and, therefore, provide a "smarter" choice of proposal. In particular, since the proposal (20) tends to move the chain in the direction of $\nabla \log \pi(\mathbf{X}_n)$, it tends to move toward larger values of π, which tends to help the chain converge to $\pi(\cdot)$ faster. For more details, see, for example, Roberts and Tweedie (1996) and Roberts and Rosenthal (1998).

We again ask how the optimal value of σ^2_d should depend on d for large d and investigate how these optimal scalings can be characterized in practically useful ways. As a first attempt to solve this problem, we might again set $\sigma^2_d = \ell^2/d$ as for the RWM case and define

$$Z^d_t = X^{(1)}_{[td]}.$$

Using this approach, it turns out that the overall acceptance rate converges to 1 and $Z^d \Rightarrow Z$ as $d \to \infty$, where, in this case,

$$(21) \qquad dZ_t = \ell \, dB_t + \frac{\ell^2 \nabla \log \pi(Z_t)}{2} \, dt.$$

The speed of this limiting algorithm is ℓ^2, which is unbounded, and therefore the limiting optimization problem is ill posed. The fact that the speed is unbounded, as we choose arbitrarily large ℓ, suggests that larger variances for proposals should be adopted in this case.

The scaling $\sigma^2_d = O(d^{-1/3})$ was suggested in the physics literature [Kennedy and Pendleton (1991)]. This turns out to be the correct scaling rate and defines a limiting optimal scaling. The following result is taken from Roberts and Rosenthal (1998).

THEOREM 3. *Consider a Metropolis–Hastings chain $\mathbf{X}_0, \mathbf{X}_1, \ldots$ for a target distribution having density π and with MALA proposals as in (20). Suppose π satisfies (5). Let $\sigma^2_d = \ell^2/d^{1/3}$ and set*

$$Z^d_t = X^{(1)}_{[d^{1/3} t]},$$

where $X^{(1)}_n$ is the first component of $\mathbf{X}_n$. Then, assuming various regularity conditions on the densities f [described in detail in Roberts and Rosenthal (1998)], as $d \to \infty$, we have the following:

(i) *Z^d converges weakly to the continuous-time process Z satisfying*

$$dZ_t = g(\ell)^{1/2} \, dB_t + \frac{g(\ell) \nabla \log \pi(Z_t)}{2} \, dt,$$

where

$$(22) \qquad g(\ell) = 2\ell^2 \Phi(-J\ell^3),$$

and J is given by

(23) $\quad J = \sqrt{\mathsf{E}\left(\dfrac{5(\log f)'''(X)^2 - 3(\log f)''(X)^3}{48}\right)},$

where the expectation is with respect to f.

(ii) *The acceptance rate a of the algorithm is given by $2\Phi(-J\ell^3)$, and the scaling which gives optimal asymptotic efficiency is that having asymptotic acceptance rate equal to 0.574.*

Thus, as for the RWM case, the optimality can be characterized in terms of the acceptance rate in a manner which is otherwise independent of properties of f. The MALA algorithm thus has a smaller convergence time [$O(d^{1/3})$ instead of $O(d)$] and larger optimal asymptotic acceptance rate (0.574 instead of 0.234), as compared to RWM algorithms. [Balanced against this, of course, is the need to compute $\nabla \log \pi(\mathbf{X}_n)$ at each step of the algorithm.] Note that, similar to the RWM case, there is no need to calculate or estimate J, since the optimality result is stated in terms of asymptotic acceptance rate only.

The regularity conditions needed for this result rely again on smoothness conditions on the function $\log f$. In Roberts and Rosenthal (1998), the existence of eight continuous derivatives of $\log f$ is assumed, though it is clear that these conditions can be relaxed to some extent at least.

5. A GIBBS RANDOM-FIELD MODEL

One can try to relax the independence assumptions in Section 2. The following result is an informal version of a statement taken from Breyer and Roberts (2000) in the context of finite-range Gibbs random-field target distributions, which gives a flavor of the problems encountered.

Suppose that $\mu(dx)$ is a continuous probability measure on $\Re$. We define a Gibbs measure on $\Re^{\mathbb{Z}^r}$ which has a density with respect to $\prod_{k \in \mathbb{Z}^r} \mu(dx_k)$ given by

(24) $\qquad \exp\left\{-\sum_{k \in \mathbb{Z}^r} U_k(\mathbf{x})\right\},$

where U_k is a finite-range potential (depending only on a finite number of neighboring terms of k), which is everywhere finite. [Formally, this framework only defines finite-dimensional conditional distributions; therefore, (24) needs to be interpreted in terms of conditional distributions.]

EXAMPLE. We give a simple Gaussian example of a Markov random field. Suppose μ denotes the measure $\mu(dx) \propto \exp\{-\tau x^2/2\}\, dx$ and define the

neighborhood structure on $\mathbb{Z}^r$ by $l \sim m$ if and only if $|l - m| = 1$; that is, l and m differ in only one coordinate, and in that coordinate by only one. Define

$$U_k(\mathbf{x}) = -\frac{\rho \sum_{l \sim k} x_k x_l}{4r}.$$

The one-dimensional full conditionals are given by

$$x_k | x_{-k} \sim N\left(\frac{\rho \bar{x}_{\sim k}}{\tau}, \frac{1}{\tau}\right),$$

where $\bar{x}_{\sim k}$ denotes the mean of the x values at neighboring states.

Now we can consider RWM on π_i with Gaussian proposals with variance given by (6) with proposal variance σ_i^2. We are now ready to informally state the following result which is taken from Breyer and Roberts (2000). We take $\{A_i\}$ to be a sequence of hyperrectangular grids increasing to $\mathbb{Z}^r$ as $i \to \infty$.

THEOREM 4. *Consider RWM with Gaussian proposals with variance $\sigma_i^2 = \ell^2/|A_i|$. Call this chain $X^{(i)}$ and consider the speeded-up chain*

(25) $\qquad\qquad Z_t^{(i)} = X_{[t|A_i|]}^{(i)},$

speeded up by a factor $|A_i|$. Then, under certain technical conditions discussed below, $Z^{(i)}$ converges weakly to a limiting infinite-dimensional diffusion on $\Re^{\mathbb{Z}^r}$. Moreover, the relative efficiency curve as a function of acceptance rate is again that given by Figure 2 and the corresponding optimal scaling problem for ℓ has a solution which can again be characterized as that which achieves acceptance rate 0.234.

This theorem thus says that, under appropriate technical conditions, RWM (with Gaussian proposal distributions) on Markov random-field target distributions again has the same optimality properties (including the 0.234 optimal acceptance rate) as RWM on the target distributions given by product densities $\prod_{i=1}^d f(x_i)$.

The most important technical condition assumed for Theorem 4 is that the random field has no phase transition; in fact, the field's correlations decrease exponentially quickly as a function of distance. [This avoids the multiplicity of distributions satisfying (24).] It should be emphasized that phase-transition behavior is indeed possible for these types of distributions, and the consequences for practical MCMC are very important. In order for the algorithm to mix in this case, the proposal needs to make considerably larger jumps, and the optimal acceptance rate will then necessarily converge to 0. In practice, this makes the algorithm prohibitively

slow (typically taking a time exponential in d to mix adequately). Therefore, for all intents and purposes, problems with heavy dependence structure (sufficient to mimic phase transition) will not yield practically useful algorithms.

It is interesting to note that phase-transition behavior of this type does not happen for $r = 1$, where ergodicity ensures that (24) uniquely specifies the full distribution. In this case, for large i, algorithms are considerably more robust to dependence between components. This supports (for example) the prolific success of MCMC algorithms in hierarchical models [see, e.g., Smith and Roberts (1993)] where dependence is propagated through a one-dimensional hierarchical structure only. On the other hand, for a multidimensional Markov random field, if significant phase-transition-type behavior does occur, then the optimal scaling properties can be very different than the above.

6. EXTENSIONS TO MORE GENERAL TARGET DISTRIBUTIONS

Theorems 1 and 3 assume that the target density π is of the special form $\pi(\mathbf{x}) = \prod_{i=1}^{d} f(x_i)$, consisting of i.i.d. components. This assumption is obviously very restrictive. However, the essential result, that the optimal acceptance rate should be about 0.234 for random-walk Metropolis (RWM) algorithms, and about 0.574 for Langevin (MALA) algorithms, appears to be considerably more robust. Thus, in this section, we consider somewhat more general target distributions and optimal scaling properties in a broader context. We have already demonstrated robustness of the 0.234 rule to different dependence structures in Section 5. Next we shall consider the notion of heterogeneity of scale between different components.

We assume that π is of the form

$$(26) \qquad \pi(\mathbf{x}) = \prod_{i=1}^{d} C_i f(C_i x_i),$$

where f is again a fixed one-dimensional density (satisfying the same regularity conditions as for Theorem 1), but where now there is an arbitrary scaling factor $C_i > 0$ in each component. We again assume that the proposals are given by $q(\mathbf{y})\mu(d\mathbf{y}) \sim N(0, I_d \sigma_d^2)$, a normal distribution with variance σ_d^2 times the identity. To make a reasonable limiting theory as $d \to \infty$, we assume that the values C_i are i.i.d. from some distribution having mean 1 and finite variance. Then we have the following.

THEOREM 5. *Consider RWM with target density of the form* (26), *where* $\{C_i\}$ *are i.i.d. with* $E(C_i) = 1$

and $E(C_i^2) \equiv b < \infty$ *and with proposal distribution* $N(0, I_d \sigma_d^2)$, *where* $\sigma_d^2 = \ell^2/d$ *for some* $\ell > 0$. *Let* $W_t^d = C_1 X_{[td]}^{(1)}$. *Then, as* $d \to \infty$, W_t^d *converges to a limiting diffusion process* W_t *satisfying*

$$dW_t = \frac{1}{2} g'(W_t)(C_1 s)^2 \, dt + (C_1 s) \, dB_t,$$

where B_t *is standard Brownian motion and where*

$$
\begin{aligned}
(27) \quad s^2 &= 2\ell^2 \Phi(-\ell b^{1/2} I^{1/2}/2) \\
&= \frac{1}{b} \times 2(\ell^2 b)\Phi\bigl(-(\ell^2 b)^{1/2} I^{1/2}/2\bigr),
\end{aligned}
$$

with $I = E_f[(g'(X))^2]$. *Hence, the efficiency of the algorithm* (*when considering functionals of the first coordinate only*), *as a function of acceptance rate, is identical to that of Theorem* 1, *except multiplied by the global factor of* C_1^2/b. *In particular, the optimal acceptance rate is still equal to* 0.234. *For a fixed function* f, *the optimal asymptotic efficiency is proportional to* C_1^2/bd.

This result does not appear in any of the MCMC scaling literature, so we have sketched a proof which appears in the Appendix.

This theorem shows that the optimal acceptance properties for RWM on the "inhomogeneous" target distribution (26) are identical to the case where the target satisfies (5). However, at least if $C_1 = 1$, the efficiency of the algorithm is slowed down by a factor b, where $b = E(C_i^2)/E(C_i)^2$. Of course, if the C_i are constant, then there is no slow-down effect at all. However, if the C_i are nonconstant, then we will have $b > 1$, and the algorithm will be slower than the algorithm for the corresponding homogeneous target distribution.

On the other hand, if the values of C_i are known, then, instead of using the proposal distribution $q \sim N(0, \sigma_d I_d)$, one could use a proposal which scales proportional to C_i in each component. In this case, the scalings C_i would be the same for the target and the proposal, so they would "cancel out," and the resulting algorithm would be precisely equivalent to running the original RWM algorithm on the corresponding target distribution satisfying (5), that is, to setting $C_i = 1$ for each i. By Theorem 5, this would be more efficient by a factor of b. We therefore have the following.

THEOREM 6. *Consider running a Metropolis algorithm on a target density of the form* (26) *to estimate some function of the first component only. Suppose we use either homogeneous proposals of the form* $q \sim N(0, \sigma_d^2 I_d)$ *or inhomogeneous proposals of the form* $q \sim N(0, \bar{\sigma}_d^2 \operatorname{diag}(C_1, \ldots, C_d))$. *Then, in either case, it is optimal to tune the algorithm to an*

asymptotic acceptance rate of 0.234, and the asymptotic efficiency is proportional to d^{-1}. (On the other hand, note that the optimal value of $\bar{\sigma}_d^2$ itself is not equal to the optimal value of σ_d^2.) However, the optimal inhomogeneous-proposal algorithm will have asymptotic relative efficiency which is larger than that of the optimal homogeneous-proposal algorithm by a factor of bC_1^2, where $b = E(C_i^2)/E(C_i)^2$.

This theorem suggests that, if the target density is even *approximately* of the form (26), with significantly varying values of C_i, then it may be worthwhile to obtain a preliminary estimate of the C_i and then use inhomogeneous proposals of the form $q \sim N(0, \bar{\sigma}_d^2 \operatorname{diag}(C_1, \ldots, C_d))$ when running the Metropolis algorithm.

REMARK. On examining the proof of Theorem 5, we see that, in the infinite-dimensional limit, the "rejection penalty" for trying big moves depends only on the components $2, \ldots, d$. Hence, if we were merely interested in the first component, it would make sense to make bigger jumps in that component than others. In fact, best of all would be to just update the first component and ignore all the others. However, this argument uses strongly the independence between components. In practice, where dependence between components is present, it will be necessary for all components to converge rapidly to have confidence in the stationarity of any one component.

REMARK. In fact, Theorem 5 extends readily to target densities whose first component has a different form of density to all other components, that is, specifically target densities of the form

$$(28) \qquad \pi(\mathbf{x}) = f_1(x_1) \prod_{i=2}^{d} C_i f(C_i x_i),$$

provided that the function f_1 does not depend on d; see Section 7.

Finally, the arguments above can be mimicked for investigating the effect of heterogeneity of components of the target density on MALA. In this case, the following result holds (we omit the proof).

THEOREM 7. *Consider a Metropolis–Hastings chain $\mathbf{X}_0, \mathbf{X}_1, \ldots$ for a target distribution having density π, with MALA proposals as in (20). Suppose π satisfies (26), let $\sigma_d^2 = \ell^2/d^{1/3}$ and set*

$$Z_t^d = X_{[d^{1/3}t]}^{(1)}.$$

Assume the same regularity conditions on the densities f as in Roberts and Rosenthal (1998).

(i) *Z^d converges weakly to*

$$dZ_t = h(\ell)^{1/2}\, dW_t + \frac{h(\ell)\nabla \log \pi(Z_t)}{2}\, dt$$

as $d \to \infty$, where

$$(29) \qquad h(\ell) = 2\ell^2 \Phi(-Jk\ell^3),$$

J is given by (23), and with

$$(30) \qquad k = \sqrt{E(C_i^6)/E(C_i)^6}.$$

(ii) *The asymptotic acceptance rate of the algorithm is given by $2\Phi(-kJ\ell^3)$, and the optimal scaling is that having asymptotic acceptance rate 0.574.*

(iii) *The efficiency of this algorithm is reduced by a factor of $k^{1/3}$, compared to the corresponding homogeneous algorithm with $C_i = 1$ for all i.*

Therefore, Langevin algorithms are considerably more sensitive to scale changes than Metropolis algorithms: the cost of heterogeneity is governed by the L^6 norm of the $\{C_i\}$, rather than the L^2 norm as in the Metropolis case.

7. EXAMPLES AND SIMULATIONS

In this section, we provide various examples and simulations to further illustrate the theorems presented above.

7.1 Inhomogeneous Multivariate Gaussian Examples

We performed a large simulation study to demonstrate examples of Theorems 5 and 6 working in practice. We considered three classes of distributions, as follows.

1. Independent and identically distributed normal components, $\pi(\mathbf{x}) = \exp\{-\sum_{i=1}^{d} x_i^2/2\}$; that is, if f is a standard normal density in (26), then we take $C_i = 1$ for all i.
2. Normal components with the first having smaller scale, $\pi(\mathbf{x}) = \exp\{-2x_1^2 - \sum_{i=2}^{d} x_i^2/2\}$; that is, we take $C_i = 1$ for all $i \neq 1$ and take $C_1 = 2$.
3. Normal components with $C_1 = 1$ but with the other C_i's being chosen from the Exponential(1) distribution.

For each of these three cases, we performed runs in dimensions $5, 10, 15, \ldots, 50$, together with 100 and 200. For each dimension, we used spherically symmetric Gaussian proposals with various values of the proposal variance, and the algorithm was run for 100,000 iterations. We concentrated on analysis of the first component for simplicity. In each case, we estimated this component's integrated autocorrelation time (called convergence time for short in

what follows) using the empirical lag-k autocorrelation $\hat{\rho}_k$ of the first component:

$$\hat{\tau} \approx \frac{-k}{\log(\hat{\rho}_k)}.$$

This estimate becomes increasingly accurate as the diffusion limit is approached—as the arguments at the end of Section 2 indicate.

Figure 5 summarizes the results of our simulations for the first two distributions. We have plotted the convergence time of the algorithm divided by dimension (since by Theorem 5 this ought to be stable as $d \to \infty$) against the acceptance rate of the algorithm. The plotting symbol indicated the dimension in which the simulation was performed in each case.

According to Theorem 5, the optimal (i.e., minimal) convergence time is proportional to b/C_1^2. Therefore, we should expect that the optimal convergence time of distribution 1 to be 4 times that of distribution 2. This is seen quite clearly in the minima achieved in each case: just below 1.5 in distribution 1 and around 0.38 for distribution 2. Note that, as expected, the optimal values are achieved at acceptance rates between 0.2 and 0.25 although it is difficult to be precise about optimal acceptance rates because the functions are inevitably rather flat around the optimal values and there is still some Monte Carlo error in our simulations.

Figure 6 shows the results of our simulation in distribution 3. The plotting symbol used in each case refers to a particular collection of random C_i's. Here the additional randomness of the C_i's means that even in 50 dimensions, the optimal scaling for the algorithm can vary quite appreciably, depending on the particular values of the C_i's. However, the robustness of the optimal acceptance rate 0.234 remains intact despite this.

If $C_i \sim \mathrm{Exp}(1)$, then b is easily computed to be 2, so, according to Theorem 5, we ought to lose a factor of 2 in efficiency between distribution 2 and distribution 1. Our results show very good agreement with Theorem 5. For instance, Figure 5 suggests that a minimum convergence time is around $3d/2$, which is around 300 for $d = 200$. Figure 6(d) shows that the minimized convergence time in 200 dimensions is around 600 for distribution 3, a loss of efficiency by a factor of around 2.

7.2 A Multimodel Example

For example, consider a target density of the form

$$\pi = \frac{1}{2}N(-M_d, I_d) + \frac{1}{2}N(M_d, I_d),$$

where $M_d = (m_d, 0, 0, \ldots, 0)$. This distribution has half its mass near $(m_d, 0, \ldots, 0)$ and the other half near $(-m_d, 0, \ldots, 0)$.

Strictly speaking, this distribution is not of the form (26), since the first coordinate is not merely
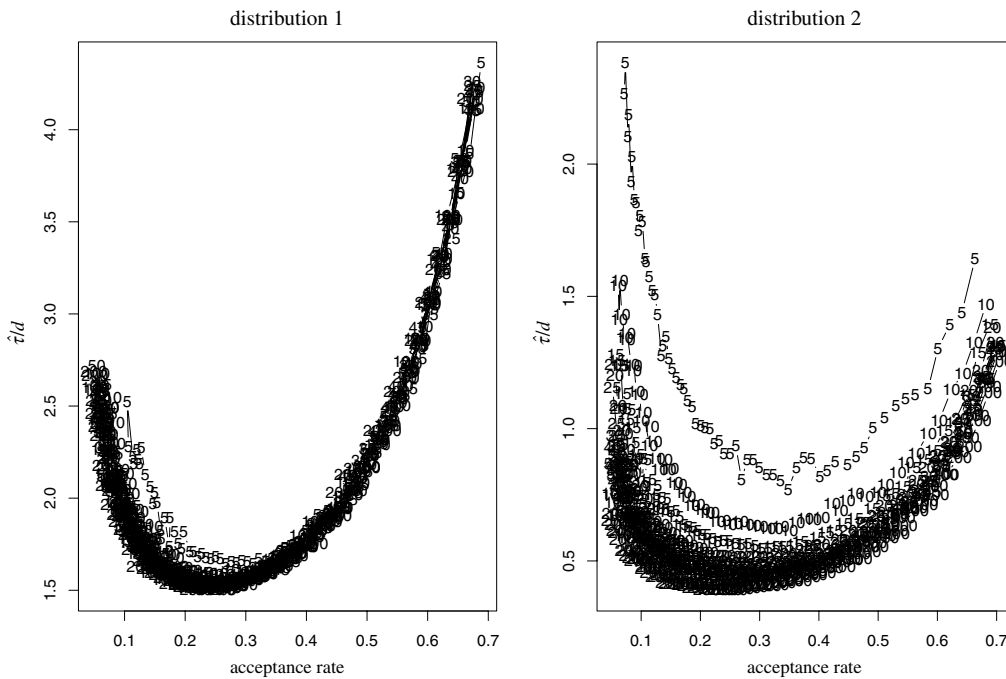

FIG. 5. *Convergence times for Metropolis algorithms as a function of their acceptance rates. The plotting symbol indicates the dimension of the simulation.*
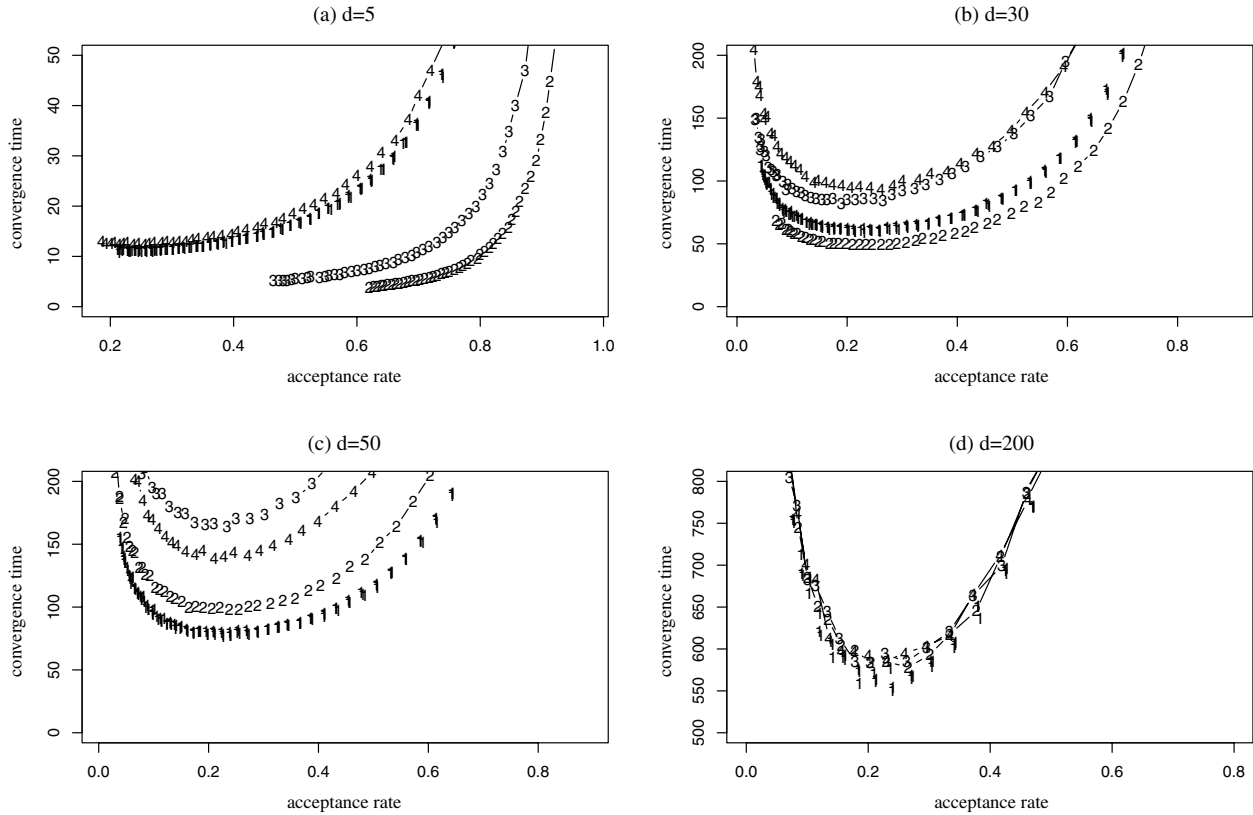
FIG. 6. *Convergence times of RWM in the heterogeneous environment of distribution* 3 *in dimensions* 5, 30, 50 *and* 200. *Here the plotting number indicates a particular random collection of* C_i's.

a rescaling of the other coordinates. However, if m_d does not depend on d, then the asymptotic results of Theorem 5 still apply. Figure 7 shows such a simulation in various dimensions, with $m_d = 3$ throughout. This is demonstrated in Figure 7(a) where the convergence time is $O(d)$ as expected.

On the other hand, suppose now that $m_d = \sqrt{d}/2$ [which is equivalent to having the two normal components centered at $(0, 0, \ldots, 0)$ and $(1, 1, \ldots, 1)$, respectively]. In this case, since m_d is growing with d, then the asymptotic results of Theorem 5 do not apply. Indeed, in this case, for large d the chain will tend to spend long amounts of time in one of the two components of π before moving to the other. Hence, there is no limiting diffusion at all. Indeed, computing autocorrelations in this case can be quite deceptive, since the autocorrelations could be small even though the chain spends all its time in just one of the two components, and a Langevin diffusion limit is not a good description of the process. Figure 7(b) shows an estimate of convergence time from autocorrelations, scaled by d^2. In fact, the true convergence time is at least exponential in d in this example, and this can be proved by the use of capacitance ideas [see, e.g., Sinclair

and Jerrum (1989)], although we do not attempt to show that in this paper.

7.3 Example: Multivariate Normal with Correlations

Suppose now that π is a d-dimensional multivariate normal distribution, but with nontrivial covariance matrix Σ. Since Σ must be symmetric (and hence self-adjoint), it follows that we can find an orthogonal rotation of the axes which transforms π into another multivariate normal, but this one having independent components, that is, a diagonal variance/covariance matrix of the form $\Sigma' = \mathrm{diag}(\lambda_1, \ldots, \lambda_d)$, where $\{\lambda_i\}$ are the eigenvalues of Σ.

Now, if we knew Σ or could estimate it, then we could use proposals of the form $q \sim N(\mathbf{x}, \sigma^2 \Sigma)$. This would be equivalent to starting with a target distribution satisfying (5) and running standard RWM. By Theorem 6, this would be better than using a fixed $N(\mathbf{x}, \sigma^2 I_d)$ distribution. This provides theoretical justification for the commonly used strategy of running random walk Metropolis with increment distribution having covariance structure which is set to be proportional to the empirically estimated
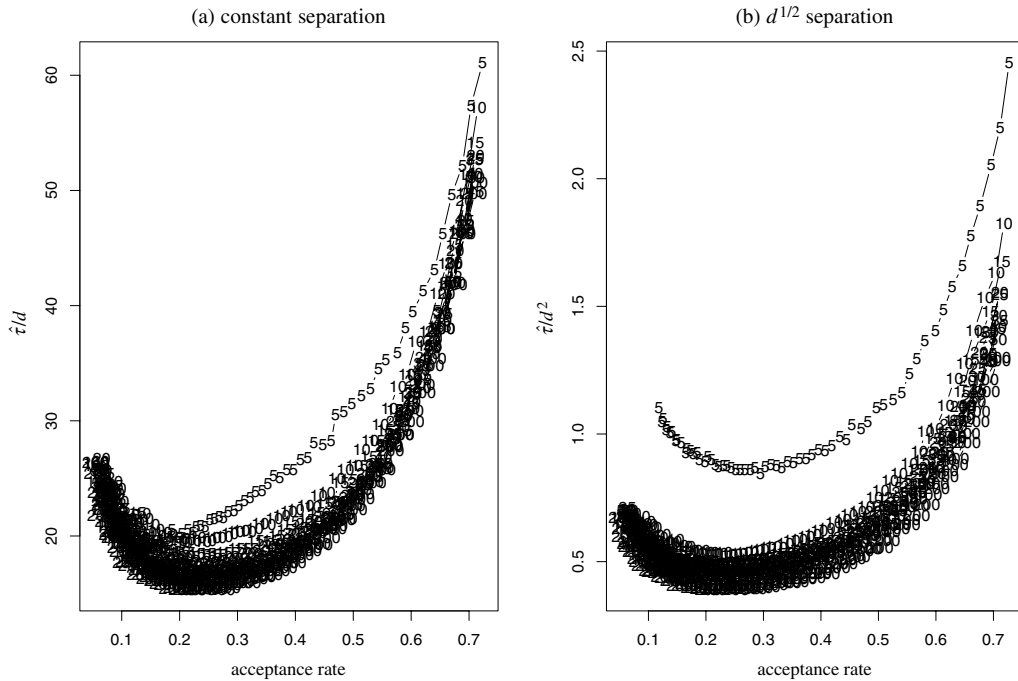
FIG. 7. (a) *Convergence time as a function of acceptance rate for RWM on the bimodal example, with $m_d = 3$ throughout, and (b) $\hat{\tau}/d^2$ as a function of acceptance rate in the case $m_d = \sqrt{d}/2$. Note that in (b) $\hat{\tau}$ does not well estimate the convergence time. In dimension 100 and 200, the algorithm in (b) fails to leave its starting mode at all, so that convergence has certainly not occurred, even though the plot of $\hat{\tau}$ seems to suggest convergence ought to occur within the 500,000 iterations for which the algorithm was run. The problem is that, since the separation distance is increasing with d, there is no longer a diffusion limit and Theorem 5 does not apply.*

correlation structure of the target density; see, for example, Tierney (1994).

In any case, this suggests that the behavior of ordinary RWM on a multivariate normal distribution is governed by the eigenvalues of Σ.

In particular, suppose that Σ has 1's on the diagonal and ρ off the diagonal (corresponding to a collection of normal random variables with unit variance and pairwise correlation ρ). Then the eigenvalues of Σ are $\lambda_1 = d\rho + 1 - \rho$ (with multiplicity 1 and eigenvector $\bar{x}$) and $\lambda_2 = \cdots = \lambda_d = 1 - \rho$ (with multiplicity $d-1$ and eigenvectors of the form $x_i - \bar{x}$ for any i). This corresponds to $C_1 = 1/\sqrt{\lambda_1} = 1/\sqrt{d\rho + 1 - \rho}$ and $C_2 = \cdots = C_d = 1/\sqrt{1 - \rho}$. Here f is the standard normal density, so that $g'(x) = -x$. Hence, the limiting diffusion satisfies $dW_t = -\frac{1}{2}C_1^2 W_t s^2 \, dt + C_1 s \, dB_t$, which can be solved explicitly (as an Ornstein–Uhlenbeck process) to give that $W_t = e^{-C_1^2 s^2 t^2/2} W_0 + N(0, (1 - e^{-C_1 st}))$. Since $C_1 = 1/\sqrt{d\rho + 1 - \rho}$, for large d we see that $C_1^2 = O(d^{-1})$ and is therefore very small, so the diffusion converges very slowly.

For this example, Theorem 5 does not strictly apply, although, since it describes the algorithm as being $O(C_1^2/d)$, it therefore suggests that, in this case, the algorithm is $O(d^2)$. In fact, by speeding up time by d^2 instead of d an analogous weak con-

vergence result can be demonstrated to show that, in this case, the algorithm is in fact $O(d^2)$.

On the other hand, the above analysis is for functions of the first eigenvector only, that is, for functions of $\bar{x}$. Suppose instead we renumber the eigenvalues of Σ, so that the large eigenvalue is numbered λ_d (or λ_2) instead of λ_1. This corresponds to considering functions which are orthogonal to $\bar{x}$, that is, which are functions of $x_i - \bar{x}$. In this case, we obtain $C_1^2 = 1/(1 - \rho)$ which is not small, so the diffusion converges much faster.

We conclude that, for RWM with normal proposals on the multivariate normal target distribution with Σ as above, the algorithm will converge quickly for functions orthogonal to $\bar{x}$, but will converge much more slowly [$O(d^2)$] for functions of $\bar{x}$ itself. This is illustrated in Figure 8.

7.4 Some Simulations with MALA

Figure 9 shows the $O(d^{1/3})$ performance of MALA on product form densities of the type in (5). These simulations were performed using distribution 1. Again notice the stability of these curves, even in relatively small dimensional problems. The asymptotically optimal acceptance rate 0.574 performs excellently even in five dimensions.
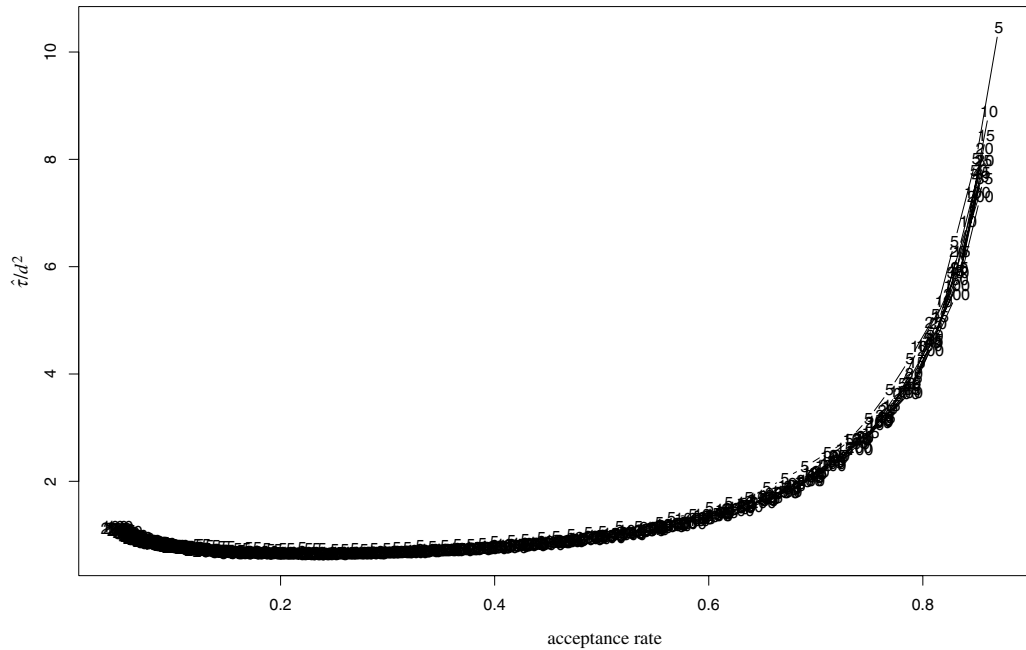
FIG. 8. *Convergence time as a function of acceptance rate for RWM on the exchangeable normal example using $\bar{x}$ as the function of interest. Note that the convergence time is scaled by d^2 in this case, demonstrating the $O(d^2)$ behavior.*

8. DISCUSSION

In this paper, we have summarized and extended recent work on scaling of RWM and MALA algorithms. The remarkable thing about the acceptance rate criteria for scaling of algorithms is their robustness to different types of target distributions, to dependence in the target density and to homogeneity in scale between different components. However, as we show in our results and examples, the efficiency of the algorithm itself can still depend critically on properties of the target density.

How should these results be used in practice? The asymptotic efficiency curve given in Figure 3 (bottom) shows that there is very little to be gained
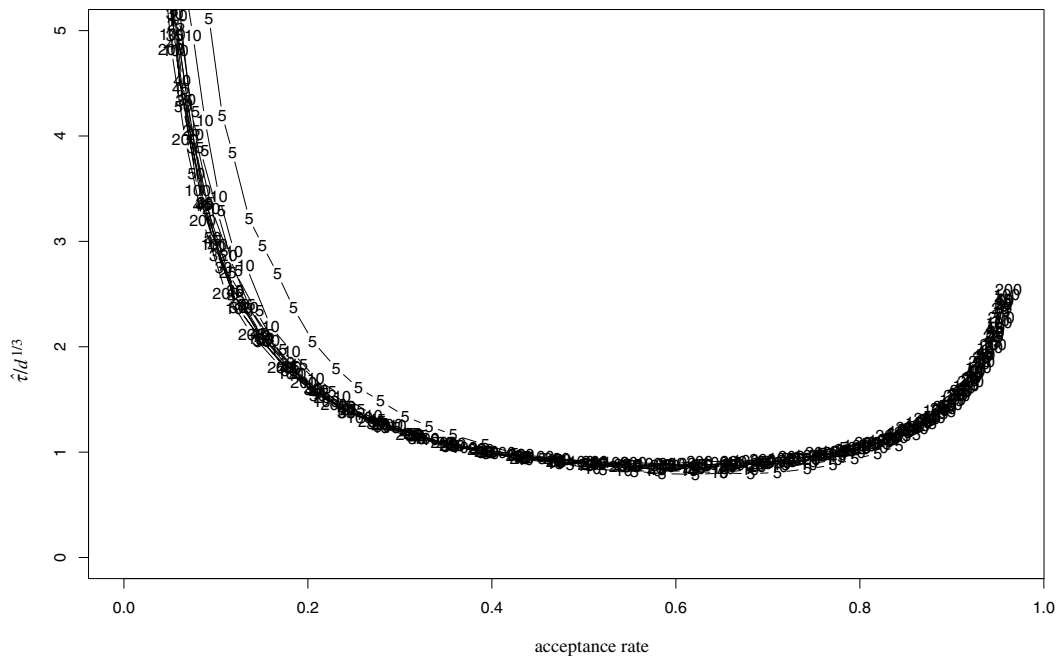

FIG. 9. *MALA Langevin algorithm, showing its optimality at around 0.574 and its convergence time being $O(d^{1/3})$.*

from fine tuning of acceptance rates. For RWM on smooth densities, any acceptance rate between 0.1 and 0.4 ought to perform close to optimal. One surprising feature of this is that very low acceptance rates of, say, 0.1 can be very close to optimal, even in highly regular problems. Similar general comments can be made about MALA algorithms and RWM on discontinuous densities.

In high-dimensional problems, tuning acceptance rates can be quite difficult. Therefore, the results described here on the way optimal proposal variances scale in high-dimensional problems are practically useful as guidelines. For instance, when performing MALA on a Gaussian image analysis problem on a 100×100 square grid, suppose we have already tuned a proposal on a 20×20 subgrid to give a variance σ^2, say. Then using the fact that optimal variances for MALA scale like $d^{-1/3}$, we should start to look for the optimal variance for the larger problem by choosing variances around $\sigma^2 \times (100/20)^{-2/3} \approx 0.34\sigma^2$. We would always advocate monitoring acceptance rates in addition to using scaling rules, since heterogeneity between components may distort the simple picture provided by just dimension-dependent scaling.

Although there are many clean mathematical statements that can be made in this area, as with many Monte Carlo problems, there is a point where the practitioner needs to extrapolate beyond the constraints imposed by the theorem's regularity conditions. We hope we have demonstrated using our examples that the conclusions we can draw from theoretical results can be extended with confidence to situations where the theorem does not strictly apply, and that the application of these results in real problems is often straightforward. It is not necessary for the practitioner to understand the detailed mathematical arguments behind the scaling rules we apply, but it is important to understand where and when they can be applied.

All these results rely on the assumption of light-tailed proposals (satisfying some kind of moment constraints). We have stated most of these results for Gaussian proposals but it is not difficult to generalize this. However, the assumption of light-tailed proposals is not merely a technicality. Most of the scaling results described here (and all of those in Euclidean space) are based on diffusion approximations of the algorithm in high dimensions. The entire character of the algorithms changes when heavy-tailed proposals are used [see, e.g., Jarner and Roberts (2001)], and the algorithms tend to make sudden jerky movements followed by periods of inactivity as opposed to the diffusion-type behavior we see here.

APPENDIX: PROOF OF THEOREM 5

Most of the proof is very similar to the corresponding proof of Theorem 1 in Roberts, Gelman and Gilks (1997). The key point of departure is the computation of the quantity $f(\mathbf{y})/f(\mathbf{x})$ used in the accept/reject ratio. Here $y_i = x_i + \sigma Z_i$ are the proposal values (with $\{Z_i\}$ i.i.d. standard normal). Setting $g = \log f$, we compute that

$$
\begin{aligned}
\frac{f(\mathbf{y})}{f(\mathbf{x})} &= \exp\left[\sum_{i=2}^{d} \big(g(C_i y_i) - g(C_i x_i)\big)\right] \\
&\quad \times \exp\{g(C_1 y_1) - g(C_1 x_1)\} \\
&= \exp\left[\sum_{i=2}^{d} \big(g(C_i x_i + C_i \sigma Z_i) - g(C_i x_i)\big)\right] \\
&\quad \times \exp\{g(C_1 y_1) - g(C_1 x_1)\} \\
&\approx \exp\left[\sum_{i=2}^{d} \Big(C_i \sigma Z_i g'(C_i x_i)\right. \\
&\qquad\qquad \left.+ \frac{1}{2}(C_i \sigma Z_i)^2 g''(C_i x_i)\Big)\right] \\
&\quad \times \exp\{g(C_1 y_1) - g(C_1 x_1)\}.
\end{aligned}
$$

Now, as $d \to \infty$, using laws of large numbers, the quantity inside the square brackets is seen to converge in distribution to the $N(-\frac{1}{2}\sigma^2 bR\,\mathbf{1}, \sigma^2 bR\,I)$ distribution, where $R = \sum_{i=2}^{d}\big(g'(C_i x_i)\big)^2$, so that, for large d, $R = O(d-1)$. It then follows [again, as in Roberts, Gelman and Gilks (1997)] that the asymptotic acceptance rate is equal to $2\Phi(-\frac{1}{2}\ell b^{1/2} I^{1/2})$, by applying this Taylor series expansion to the expression in (1). The result then follows similar lines to that described in Roberts, Gelman and Gilks (1997), where a further Taylor expansion is performed, this time just of the first component term, and limits of expectations are taken in order to calculate the infinitesimal change in expected functions in small time periods. These calculations then characterize the limiting process according to the powerful mathematical theory of weak convergence of Markov processes. The precise mathematical framework for, these calculations involves computing the *generator* of the limiting process, which in, turn characterizes the dynamics of the limiting process; see Roberts, Gelman and Gilks (1997) for details. This leads to the expression (27). Now letting $A(\ell)$ denote the asymptotic acceptance rate a, using proposal variance ℓ^2/d, we can reexpress (27) as

$$
s^2 = \frac{4}{bI}\left(\Phi^{-1}\left(\frac{A(\ell)}{2}\right)\right)^2 \times A(\ell),
$$

so that the maximization problem is expressed solely in terms of $A(\ell)$ and the other constants b, I etc. only affect the maximized value itself.

ACKNOWLEDGMENT

REFERENCES

BESAG, J. E. (1994). Comment on "Representations of knowledge in complex systems" by U. Grenander and M. I. Miller. *J. Roy. Statist. Soc. Ser. B* **56** 591–592.

BREYER, L. and ROBERTS, G. O. (2000). From Metropolis to diffusions: Gibbs states and optimal scaling. *Stochastic Process. Appl.* **90** 181–206.

GELMAN, A., ROBERTS, G. O. and GILKS, W. R. (1996). Efficient Metropolis jumping rules. *Bayesian Statist.* **5** 599–608.

GILKS, W. R, RICHARDSON, S. and SPIEGELHALTER, D. J. eds. (1996). *Markov Chain Monte Carlo in Practice*. Chapman and Hall, London.

JARNER, S. F. and ROBERTS, G. O. (2001). Convergence of heavy tailed Metropolis algorithms. Available at www.statslab.cam.ac.uk/mcmc.

KENNEDY, A. D. and PENDLETON, B. (1991). Acceptances and autocorrelations in hybrid Monte Carlo. *Nuclear Phys. B* **20** 118–121.

ROBERTS, G. O. (1998). Optimal Metropolis algorithms for product measures on the vertices of a hypercube. *Stochastics Stochastic Rep.* **62** 275–283.

ROBERTS, G. O., GELMAN, A. and GILKS, W. R. (1997). Weak convergence and optimal scaling of random walk Metropolis algorithms. *Ann. Appl. Probab.* **7** 110–120.

ROBERTS, G. O. and ROSENTHAL, J. S. (1998). Optimal scaling of discrete approximations to Langevin diffusions. *J. Roy. Statist. Soc. Ser. B* **60** 255–268.

ROBERTS, G. O. and TWEEDIE, R. L. (1996). Exponential convergence of Langevin diffusions and their discrete approximations. *Biometrika* **2** 341–363.

ROBERTS, G. O. and YUEN. W. K. (2001). Optimal scaling of Metropolis algorithms for discontinuous densities. Unpublished manuscript.

SINCLAIR, A. J. and JERRUM, M. R. (1989). Approximate counting, uniform generation, and rapidly mixing Markov chains. *Inform. and Comput.* **82** 93–133.

SMITH, A. F. M. and ROBERTS, G. O. (1993). Bayesian computation via the Gibbs sampler and related Markov chain Monte Carlo methods (with discussion). *J. Roy. Statist. Soc. Ser. B* **55** 3–24.

TIERNEY, L. (1994). Markov chains for exploring posterior distributions (with discussion). *Ann. Statist.* **22** 1701–1762.