Improving Convergence of the Hastings-Metropolis Algorithm with an Adaptive Proposal

Author(s): Didier Chauveau and Pierre Vandekerkhove

Source: *Scandinavian Journal of Statistics*, Mar., 2002, Vol. 29, No. 1 (Mar., 2002), pp. 13-29

Published by: Wiley on behalf of Board of the Foundation of the Scandinavian Journal of Statistics

Stable URL: https://www.jstor.org/stable/4616696

Improving Convergence of the Hastings–Metropolis Algorithm with an Adaptive Proposal

DIDIER CHAUVEAU and PIERRE VANDEKERKHOVE

Université de Marne-la-Vallée

ABSTRACT. The Hastings–Metropolis algorithm is a general MCMC method for sampling from a density known up to a constant. Geometric convergence of this algorithm has been proved under conditions relative to the instrumental (or proposal) distribution. We present an inhomogeneous Hastings–Metropolis algorithm for which the proposal density approximates the target density, as the number of iterations increases. The proposal density at the *n*th step is a non-parametric estimate of the density of the algorithm, and uses an increasing number of i.i.d. copies of the Markov chain. The resulting algorithm converges (in *n*) geometrically faster than a Hastings–Metropolis algorithm with any fixed proposal distribution. The case of a strictly positive density with compact support is presented first, then an extension to more general densities is given. We conclude by proposing a practical way of implementation for the algorithm, and illustrate it over simulated examples.

Key words: adaptive algorithm, geometric ergodicity, Hastings–Metropolis algorithm, Markov chain Monte Carlo

1. Introduction

A Markov chain Monte Carlo (MCMC) algorithm generates an ergodic Markov chain $x^{(n)}$ with a stationary distribution with preassigned density f. In situations where direct simulation from f is not tractable, or where integrals like $\mathbb{E}_f[h]$ are not available in closed form, a MCMC method is appropriate since, for T large enough, $x^{(T)}$ is approximately f distributed, and $\mathbb{E}_f[h]$ can be approximated by ergodic averages from the chain. The most commonly used MCMC methods are the Hastings–Metropolis algorithm (Hastings, 1970), and the Gibbs sampler (first introduced by Geman & Geman, 1984; see also Gelfand & Smith, 1990). An account of their definitions and convergence properties can be found, e.g. in Robert (1996), and a good introduction to this topic is in Gelfand & Smith's (1990) seminal paper.

In this paper, we will focus on the Hastings–Metropolis algorithm. This algorithm produces a homogeneous Markov chain $(x^{(n)})$ with stationary distribution f, where f needs to be known up to a (normalizing) multiplicative constant. Each step is based on the generation of the proposed next move y from a general conditional density $q(y|x)$, called the instrumental distribution or proposal density. The particular situation where $q(y|x) = q(y)$ gives the so-called independence sampler. The algorithm starts with an initial value $x^{(0)}$ and proceeds as follows (a practical requirement is that simulations from q should be done easily).

(1) Generate $y \sim q(\cdot|x^{(u)})$.

(2) Compute $\alpha(y, x^{(n)}) = \min\left\{1, \dfrac{f(y)q(x^{(n)}|y)}{f(x^{(n)})q(y|x^{(n)})}\right\}$.

(3) Take $x^{(n+1)} = \begin{cases} y & \text{with probability } \alpha(y, x^{(n)}), \\ x^{(n)} & \text{with probability } 1 - \alpha(y, x^{(n)}). \end{cases}$

Ergodicity and convergence properties of the Hastings–Metropolis algorithm have been intensively studied in recent literature, and conditions have been given for its geometric convergence (see Mengersen & Tweedie, 1996; Roberts & Tweedie, 1996, or Robert, 1996, for

a comprehensive account). In particular, Mengersen and Tweedie proved geometric convergence in total variation of the independence sampler when the proposal density satisfies $q(y|x) = q(y) \geqslant af(y)$ for some $a \in (0, 1)$, a property which has been improved by Holden (1998), who proves (under some assumptions for f) geometric convergence with rate $(1 - a)^n$ in the relative supremum norm. This result highlights the link between the convergence rate and the proximity of q to f.

We assume here that an arbitrary proposal density q_0 such that $q_0(y) \geqslant a_0 f(y)$ is available and hence insures geometric convergence with rate $(1 - a_0)^n$ as in Holden (1998). In many set-ups, the geometric convergence alone does not mean that the chain will converge quickly in practice, due to a very bad rate of convergence, i.e. a very small value of a_0. It seems natural, in a way to improve this convergence rate, to find a minorization constant greater than a_0, i.e. to use a proposal density better than q_0, which approximates f in a better way than q_0 does. Let p^n be the density of the Hastings–Metropolis algorithm at time n. Since p^n converges geometrically to f, a non-parametric density estimator q_n of p^n should be better than q_0 if n is large enough and if it can be constructed to satisfy a minorization condition ($q_n \geqslant a_n f$). We propose to define a Hastings–Metropolis algorithm which uses a sequence of proposal densities q_n based on histogram estimates of p^n constructed from $N(n)$ i.i.d. copies of the algorithm, and suitably modified to fulfill the minorization condition. The resulting algorithm is clearly inhomogeneous. We show that, for an appropriate increasing rate of $N(n)$, it provides almost surely a rate of convergence better than $(1 - a_0)^n$ for the online algorithm (e.g. in n).

The chains used at step n to build q_n are not independent after time n since they all use q_n for the proposal density. They are not even Markov chains since they all depend on their positions at time n through q_n. Hence, theoretically, they have to be discarded to preserve i.i.d. and Markov properties between the remaining chains. As a consequence, it will be convenient in practical implementation to choose a few arbitrary switching times at which the current proposal densities will be updated by better ones "learning" f as n and $N(n)$ increase (this will be discussed in sections 3 and 6). The algorithm thus becomes a standard homogeneous Hastings–Metropolis algorithm after the last switching time, with a proposal density better than the initial one.

The case of a strictly positive density with compact support is first presented since it allows for less technical difficulties, while giving the flavour of the method (this context has been briefly exposed in Chauveau & Vandekerkhove, 1999). In section 2, the inhomogeneous Hastings–Metropolis algorithm is defined, and its geometric convergence in the relative supremum norm is given. Section 3 gives the definition and an exponential inequality for the non-parametric density estimates of p^n which will be used as proposal densities. The convergence properties of the resulting algorithm are given in section 4. Section 5 is devoted to an extension for positive densities over $\mathbb{R}^s$. Finally, section 6 suggests a way to implement this algorithm in actual situations. Its performance in comparison with a standard random walk Hastings–Metropolis algorithm and an independence sampler are illustrated on simulated examples, which show the good behaviour of the method for multimodal (hence slowly mixing) situations. Note that the implementation cost of our method for the end user is reduced since a "black-box" type computer code has been developed.

2. An inhomogeneous Metropolis algorithm

We present in this section a generic version of the Hastings–Metropolis algorithm using a sequence of proposal densities $(q_n)_{n \geqslant 0}$ (i.e. q_n is used to generate $x^{(n)}$), resulting in an

inhomogeneous algorithm with stationary distribution f. Note that different inhomogeneous algorithms with a given stationary distribution have already been used, e.g. in Green (1995) or Vandekerkhove (1998).

Let $\Omega \subseteq \mathbb{R}^s$, f be the target density, positive on Ω, and assume that the densities (q_n) are also positive on Ω. The results in this section do not require more assumptions. For a starting value $x^{(0)} \sim p^0$, the nth step $x^{(n)} \to x^{(n+1)}$ of the algorithm uses a proposal density q_n as follows.

(1) Generate $y \sim q_n(\cdot)$.

(2) Compute $\alpha_n(y, x^{(n)}) = \min\left\{1, \dfrac{f(y)q_n(x^{(n)})}{f(x^{(n)})q_n(y)}\right\}$.

(3) Take $x^{(n+1)} = \begin{cases} y & \text{with probability } \alpha_n(y, x^{(n)}), \\ x^{(n)} & \text{with probability } 1 - \alpha_n(y, x^{(n)}). \end{cases}$

Define:

$$h_n(x,y) = \min\{f(x)q_n(y);\ f(y)q_n(x)\}, \tag{1}$$

$$Q_n(x,y) = \frac{h_n(x,y)}{f(y)} = \min\left\{q_n(x); \frac{q_n(y)f(x)}{f(y)}\right\}, \tag{2}$$

$$D^n(x) = \left(\frac{p^n(x)}{f(x)} - 1\right), \quad \text{and} \quad D_M^n = \sup_{x\in\Omega}\{|D^n(x)|\}. \tag{3}$$

The density p^n of the algorithm after n iterations satisfies the following technical lemma.

Lemma 1
If q_n is a probability density on Ω, then $\int_\Omega Q_n(x,y)\,dx \leqslant 1$ for each $y \in \Omega$, and

$$D^{n+1}(y) = D^n(y)\left(1 - \int_\Omega Q_n(x,y)\,dx\right) + \int_\Omega D^n(x)Q_n(x,y)\,dx. \tag{4}$$

Proof. Similar to the proof given in Holden (1998) for the homogeneous situation. According to the definition of the sequential Metropolis–Hastings algorithm given previously, we have:

$$p^{n+1}(y) = \int_\Omega p^n(x)q_n(y)\alpha_n(y,x)\,dx + \int_\Omega p^n(y)q_n(z)(1 - \alpha_n(z,y)\,dz. \tag{4}$$

Since $h_n(x,y)$ is symmetric and $\alpha_n(x,y) = h_n(x,y)/(f(y)q_n(x))$, we get:

$$\begin{aligned} p^{n+1}(y) &= p^n(y) + \int_\Omega (p^n(x)q_n(y)\alpha_n(y,x) - p^n(y)q_n(x)\alpha_n(x,y))\,dx \\ &= p^n(y) + \int_\Omega \left(p^n(x)q_n(y)\frac{h_n(x,y)}{f(x)q_n(y)} - p^n(y)q_n(x)\frac{h_n(y,x)}{f(y)q_n(x)}\right)dx \\ &= p^n(y) + \int_\Omega \left(\frac{p^n(x)}{f(x)} - \frac{p^n(y)}{f(y)}\right)h_n(x,y)\,dx. \end{aligned}$$

Now we can write,

$$\begin{aligned}\frac{p^{n+1}(y)}{f(y)} - 1 &= \frac{p^n(y)}{f(y)} + \int_\Omega \left(\frac{p^n(x)}{f(x)} - \frac{p^n(y)}{f(y)}\right) Q_n(x,y)\,dx - 1 \\ &= \left(\frac{p^n(y)}{f(y)} - 1\right)\left(1 - \int_\Omega Q_n(x,y)\,dx\right) \\ &\quad + \int_\Omega \left(\frac{p^n(x)}{f(x)} - 1\right) Q_n(x,y)\,dx,\end{aligned}$$

which concludes the proof.

The proposition below is crucial in assessing the geometric convergence of the Hastings–Metropolis with a sequence of proposal densities satisfying minorization conditions.

Proposition 1

If, at iteration n, $q_n(x) \geqslant a_n f(x)$ for all $x \in \Omega$, with $a_n \in (0,1)$, then:

$$\left|\frac{p^{n+1}(y)}{f(y)} - 1\right| \leqslant (1 - a_n) \sup_{x\in\Omega}\left\{\left|\frac{p^n(x)}{f(x)} - 1\right|\right\}. \tag{5}$$

Proof. Note that $q_n(x) \geqslant a_n f(x)$ implies $Q_n(x,y) \geqslant a_n f(x)$. We have

$$\begin{aligned}D^{n+1}(y) &\leqslant D_M^n - \int_\Omega D_M^n Q_n(x,y)\,dx + \int_\Omega D^n(x) Q_n(x,y)\,dx \\ &= D_M^n - \int_\Omega (D_M^n - D^n(x)) Q_n(x,y)\,dx \\ &\leqslant D_M^n - a_n \int_\Omega (D_M^n - D^n(x)) f(x)\,dx \\ &= D_M^n (1 - a_n). \end{aligned} \tag{6}$$

The above calculation remains valid for $\tilde{D}^n(x) = -D^n(x)$, then $|D^n(x)| \leqslant D_M^n(1 - a_n)$ which proves the proposition.

3. The histogram as the proposal density

As explained in the introduction, we first consider a simple situation to highlight the essential points. Thus, in this section and in section 4, we assume that Ω is compact and that f satisfies a Lipschitz condition and is such that $f(x) \geqslant \alpha$ over Ω for a (generally unknown) constant $\alpha > 0$. Consequently, f is bounded on Ω, and we set $A = \sup_{x\in\Omega} f(x)$.

The key idea of our method consists in using for q_n at iteration n a non-parametric density estimate of p^n—namely the empirical histogram—based on an adequate number of i.i.d. realizations from p^n, i.e. on $N(n)$ i.i.d. copies of the original algorithm started from the same initial distribution p^0. This set-up is often called a parallel chains method in the MCMC framework, and offers well-known advantages over single chain methods, the most important being that: (i) parallel chains provide better exploration of the support of f if the initial distribution is dispersed enough; (ii) they provide information about how well the chains are mixing, which has been used, e.g. in Gelman & Rubin (1992), or Chauveau & Diebolt (1999), to propose convergence assessment methods. These aspects of parallel chains and their impact over the implementation will be discussed further in section 6.

We choose to use the rather "naïve" histogram estimate for p^n since the kernel of the Hastings-Metropolis algorithm does not preserve the smoothness properties across iterations. This remark on the non-differentiability of the p^n densities prevents in fact the use of, e.g. a kernel estimator (which requires such smoothness assumptions to insure its consistency). Moreover, the histogram satisfies exponential inequalities at each step which will be needed in the sequel. In order to study theoretically the asymptotic behaviour of the algorithm as the time $n \to \infty$, we will need $N(n)$ to increase with n to insure the consistency of the histogram estimate. We thus assume that we have an infinite number of i.i.d. copies of the Hastings–Metropolis algorithm given in section 2, even if the practical implementation only requires a finite number of i.i.d. chains.

The sequence of proposal densities will be constructed in the following way: let $I = (t_1, \ldots, t_i, t_{i+1}, \ldots)$ be an arbitrary, increasing sequence of integers (the mutation times). We define q_n by

$$q_n(x) = H_{N(t_i)}(x)\mathbb{I}_{C(t_i)} + \tilde{H}_{N(t_i)}(x)\mathbb{I}_{\bar{C}(t_i)} \quad \text{for } t_i \leqslant n < t_{i+1},\ t_i \in \{0\} \cup I, \tag{7}$$

where $H_{N(0)}$ is the arbitrary initial density q_0 and, for $t \in I$, $H_{N(t)}$ is the histogram estimates of p^t based on $N(t)$ i.i.d. copies of the algorithm, and $\tilde{H}_{N(t)}$ is a slight alteration of $H_{N(t)}$ to be used when the condition $C(t)$, described below, is not fulfilled. Notice that from a theoretical point of view the $N(t_i)$ chains used at step t_i to define $H_{N(t_i)}$ cannot be used anymore, in order to preserve the Markov property and the independence between the remaining chains, as explained in the introduction. The proposed way to circumvent this drawback in the actual implementation will be discussed in section 6. Also, to keep the notation simple, we will assume for the theoretical study that $I = \mathbb{N}$, even if (7) is the implementation we will actually use.

We recall briefly here the definition of the histogram in our set-up. A complete description of this estimator, together with its classical convergence properties, may be found in Bosq & Lecoutre (1987, ch. 6). Without loss of generality, we assume Ω to be $[0, 1]^s$. Define a rectangular grid of $[0, 1]^s$ by

$$\pi_{N(n),r} = [(r_1 - 1)h_{N(n)}, r_1 h_{N(n)}[\times \cdots \times [(r_s - 1)h_{N(n)}, r_s h_{N(n)}[,$$

where $r = (r_1, \ldots, r_s) \in R = \{t \in \mathbb{N}^s:\ \pi_{N(n),t} \cap \Omega \neq \varnothing\}$ and $h_{N(n)}$ is a positive real value which represents the side width of each class (we will require $h_{N(n)} \to 0$ for the histogram to converge to p^n). The histogram is then defined by

$$H_{N(n)}(x) = \sum_{r \in R} \frac{\mu_{N(n)}(\pi_{N(n),r})}{N(n)h_{N(n)}^s} \mathbb{I}_{\pi_{N(n),r}}(x), \tag{8}$$

where $\mu_{N(n)}(B)$ is the number of observations in the $N(n)$-sample from p^n belonging to B. In order to apply proposition 1, we cannot use the histogram directly as a proposal density in situations which result in at least one empty class (since then the minorization condition is not fulfilled). These situations are detected in (7) by the condition $\bar{C}(n)$, where $C(n) = \{\inf_{x \in \Omega}(H_{N(n)}(x)) > 0\}$. Whenever $\bar{C}(n)$ occurs, we propose to use, instead of $H_{N(n)}$, a modified version $\tilde{H}_{N(n)}$ for which the zero value over empty classes are replaced by some $\varepsilon > 0$, with a suitable renormalization over the other classes. The suitable choice for ε is defined in the proof of the following proposition.

Proposition 2

Let f be a Lipschitzian density on Ω. For the sequence of proposal densities (q_n) given in (7), there exists a sequence (a_n), $a_n \in (0, 1)$ for $n \in \mathbb{N}$, such that $q_n(x) \geqslant a_n f(x)$, $x \in \Omega$, and, if $N(n)h_{N(n)}^s \to \infty$ as $n \to \infty$,

$$|D^{n+1}(y)| \leqslant \prod_{k=1}^{n}(1 - a_k)D_M^0 \leqslant \left(1 - \frac{1}{AN(n)h_{N(n)}^s}\right)^n D_M^0. \tag{9}$$

Proof. Clearly, a minorization condition holds for q_n at each iteration n; the first majorization then comes from the iterative application of (5) from proposition 1. For the second majorization, it suffices to note that, if when $\bar{C}(n)$ occurs we replace each empty class of $H_{N(n)}$ by $\varepsilon \geqslant 1/N(n)h_{N(n)}^s$, then we always have $q_n(x) \geqslant 1/N(n)h_{N(n)}^s$, and the "worst" choice $a_n = 1/AN(n)h_{N(n)}^s$ ensures $q_n \geqslant a_n f$ with a sequence (a_n) decreasing to zero.

Note that this sequence (q_n) may be worse than q_0. However, if $N(n)h_{N(n)}^s = o(n)$, we still have that $\prod_{k=1}^{n}(1 - a_k) \to 0$ as $n \to \infty$, which will be used below. The next proposition gives an exponential inequality for the deviation between $H_{N(n)}$ and p^n, for any n and $N(n)$ large enough. This bound will be used in the next section. To simplify the notation in the sequel, we will sometimes write N, H_N and h_N instead of $N(n)$, $H_{N(n)}$ and $h_{N(n)}$.

Proposition 3

Let f be a C-Lipschitzian density on Ω such that $f(x) \geqslant \alpha > 0$ on Ω, H_N the histogram estimate given by (8), and $\varepsilon > 0$. Define in addition

$$\delta_{N,n} = 2A\left(1 - \frac{1}{ANh_N^s}\right)^n D_M^0 + \sqrt{s}h_N C.$$

Then, if $h_N \to 0$, $Nh_N^s \to \infty$ as $n \to \infty$, $Nh_N^s = o(n)$ and

$$Nh_N^{3s} \geqslant (20/(\varepsilon - \delta_{N,n})^2) \quad \text{for } N > N_0,\ n > n_0, \tag{10}$$

where n_0 and N_0 are such that $(\varepsilon - \delta_{N_0,n_0}) > 0$ and $(\varepsilon - \delta_{N_0,n_0})h_{N_0}^s \leqslant 1$, then we have, for $n > n_0$ and $N > N_0$:

$$\mathbb{P}\left(\sup_{x\in\Omega}|H_N(x) - p^n(x)| > \varepsilon\right) \leqslant 3\exp(-Nh_N^{2s}(\varepsilon - \delta_{N,n})^2/25). \tag{11}$$

Proof. Assume as before that $\Omega \equiv [0, 1]^s$ without loss of generality. We have

$$\sup_{x\in\Omega}|H_N(x) - p^n(x)| \leqslant \sup_{x\in\Omega}|H_N(x) - \mathbb{E}[H_N(x)]| + \sup_{x\in\Omega}|\mathbb{E}[H_N(x)] - p^n(x)|.$$

To control the first term on the right hand side, we establish an exponential inequality for the deviation between the histogram and its mean (lemma 3), based on the following exponential inequality for the deviations of the multinomial distribution (Bosq & Lecoutre, 1987, p. 174):

Lemma 2

Let $(N_1, \ldots, N_k) \sim \mathcal{M}(N; (p_1, \ldots, p_k))$ and $\gamma \in (0, 1)$. If the number of classes satisfies $k \leqslant N\gamma^2/20$, then:

$$\mathbb{P}\left(\sum_{q=1}^{k}|N_q - N_{pq}| \geqslant N\gamma\right) \leqslant 3\exp(-N\gamma^2/25).$$

The histogram on Ω is just a realization from a multinomial distribution with $k = k_N = 1/h_N^s, H_n(x) = N_r/Nh_N^s$ and $\mathbb{E}[H_N(x)] = p_r/h_N^s$ for $x \in \pi_{N,r}$, where

$$N_r = \mu_N(\pi_{N,r}), \quad p_r = \int_{\pi N,r} p^n(x)\,dx.$$

For $\gamma \in (0,1)$, we obtain from lemma 2:

$$\mathbb{P}\left(\sup_{x\in\Omega} |H_N(x) - \mathbb{E}[H_N(x)| > \gamma/h_N^s\right) \leqslant 3\exp(-N\gamma^2/25). \tag{12}$$

For $\xi > 0$, we apply (12) with the sequence $\gamma_N = \xi h_N^s$ and $N > N_0$, where N_0 needs to be large enough to ensure $\gamma_{N_0} < 1$. The condition from lemma 2 becomes here $Nh_N^{3s} \geqslant 20/\xi^2$ (which requires in particular that $Nh_N^{2s} \to +\infty$, i.e. $h_N \to 0$ slowly enough). We thus have shown:

Lemma 3
Let $\xi > 0$, and N_0 such that $\xi h_{N_0}^s < 1$. For each $N > N_0$ and $Nh_N^{3s} \geqslant 20/\xi^2$,

$$\mathbb{P}\left(\sup_{x\in\Omega} |H_N(x) - \mathbb{E}[H_N(x)]| \geqslant \xi\right) \leqslant 3\exp\left(-Nh_N^{2s}\xi^2/25\right). \tag{13}$$

We also need to bound the deterministic term $\sup_{x\in\Omega}|\mathbb{E}[H_N(x)] - p^n(x)|$. For each class $\pi_{N,r}$,

$$\begin{aligned}\sup_{x\in\pi_{N,r}} |\mathbb{E}[H_N(x)] - p^n(x)| &\leqslant \sup_{x,y\in\pi_{N,r}} |p^n(x) - p^n(y)| \\ &\leqslant 2\sup_{x\in\Omega} |p^n(x) - f(x)| + \sup_{x,y\in\pi_{N,r}} |f(x) - f(y)| \\ &\leqslant 2A\left(1 - \frac{1}{ANh_N^s}\right)^n D_M^0 + \sqrt{s}h_N C \end{aligned} \tag{14}$$

where (14) comes from proposition 2 and from the Lipchitz condition for f. Finally we have, for $\varepsilon > 0$:

$$\mathbb{P}\left(\sup_{x\in\Omega} |H_N(x) - p^n(x)| \geqslant \varepsilon\right) \leqslant \mathbb{P}\left(\sup_{x\in\Omega} |H_N(x) - \mathbb{E}[H_N(x)]| \geqslant \varepsilon - \delta_{N,n}\right)$$

and the application of lemma 3 with $\xi_{N,n} = \varepsilon - \delta_{N,n}$ leads to an exponential bound for the right hand side, providing that n being larger than n_0 (and $N > N_0$ to satisfy the condition of lemma 3) such that $\varepsilon - \delta_{N_0,n_0} > 0$, which is always feasible since $\delta_{N,n}$ decreases to zero when both n and N increase and h_N decreases with the rates given in the proposition.

4. Convergence of the algorithm

We show in this section that our adaptive algorithm converges almost surely with a rate better than $(1 - a_0)^n$. To simplify the notation, we will assume here that the proposal density is updated at each step, i.e. that $I = \mathbb{N}$. At the nth iteration of the algorithm, q_n may be either $H_{N(n)}$, or $\tilde{H}_{N(n)}$. Define

$$A_n = \left\{\mathbb{1}_{C(n)} = 0\right\}, \quad B_n = \left\{a_n < a_0 | A_n^c\right\}, \quad \text{and} \quad C_n = \{a_n < a_0\}, \tag{15}$$

where A_n is equivalent to the event "the modified histogram $\tilde{H}_{N(n)}$ is used at iteration n", B_n means that "q_0 is a better proposal density (with respect to (5)) than the q_n computed at step n, knowing that the histogram $H_{N(n)}$ is strictly positive at step n", and C_n is the event "q_0 is better than q_n". The result below states that the algorithm will *not* use i.o. a proposal density worse than q_0 (i.e. $a_n < a_0$). A more comprehensive way of stating this result is by considering the time

$$T(a_0) = \inf\{t: \forall n \geqslant t, a_n > a_0\},$$

after which any homogeneous single chain Hastings–Metropolis algorithm using q_t with $t > T(a_0)$ would be an online algorithm faster than the initial one using the arbitrary proposal density q_0.

Theorem 1

If the number $N(n)$ of i.i.d. copies of the algorithm, and the class width $h_{N(n)}$ used to compute $H_{N(n)}$ satisfy conditions from proposition 3, and

$$N(n)h_{N(n)}^{2s} \geqslant c\log(n), \tag{16}$$

where $c = c(\alpha)$ is a constant, then

$$\mathbb{P}(\overline{\lim}\, A_n) = \mathbb{P}(\overline{\lim}\, B_n) = \mathbb{P}(\overline{\lim}\, C_n) = 0,$$

and

$$\mathbb{P}(T(a_0) < \infty) = 1. \tag{17}$$

This result means that after an almost surely finite time, a single chain issued from our adaptive algorithm converges faster then a single chain using any fixed initial proposal density.

Proof. We first prove $\mathbb{P}(\overline{\lim}\, A_n) = 0$ using a Borel–Cantelli lemma. The event A_n means that H_N has at least one empty class, which implies

$$A_n \subset \left\{\sup_{x\in\Omega} |H_N(x) - f(x)| > \alpha\right\},$$

and this latter event can be expressed as a probability of deviation between the histogram and p^n since p^n converges to f by:

$$\mathbb{P}(A_n) \leqslant \mathbb{P}\left(\sup_{x\in\Omega} |\mathrm{H}_{N(n)}(x) - p^n(x)| > \varepsilon_n(\alpha)\right), \tag{18}$$

where the deviation takes the form

$$\varepsilon_n(\alpha) = \alpha - A\prod_{k-1}^{n-1}(1 - a_k)\, D_M^0,$$

and $\varepsilon_n(\alpha) \to \alpha$ when $n \to \infty$ as in proposition 3. We then apply (11) from proposition 3 with $\varepsilon_n(\alpha)$. To insure the convergence of $\sum_{n\geqslant 0} \mathbb{P}(A_n)$, we need the increasing rate of $N(n)$ to satisfy the condition

$$N(n)h_{N(n)}^{2s} \geqslant \frac{25\beta}{(\varepsilon_n(\alpha) - \delta_{N,n})^2}\log(n), \quad \beta > 1, \tag{19}$$

which is condition (16), where $c \approx 25\beta/\alpha^2$ for n and N large enough.

Notice that we can in fact show that for all $\epsilon > 0$, the sequence of events $E_n(\epsilon) = \{\sup_{x\in K} |H_N(x) - f(x)| > \epsilon\}$ satisfies $\sum_{n\geqslant 0} \mathbb{P}(E_n(\epsilon)) < \infty$, which implies that $\sup_K |H_N - f|$ goes to 0 almost surely as n goes to infinity, for any compact set $K \subseteq \Omega$. This will be used in section 5. The same reasoning holds for B_n since the non-modified histogram satisfies a minorization condition $H_N(x) \geqslant a_{N,n} f(x)$, where

$$a_{N,n} = 1 - \frac{1}{\alpha}\left(\sup_{x\in\Omega} |\mathrm{H}_{N(n)}(x) - p^n(x)| + \prod_{k=1}^{n-1}(1 - a_k)\, D_M^0\right).$$

Obviously, this minorization constant will not beat a_0 at the first iterations (it will even be negative), but $a_{N,n}$ will not be worse that a_0 i.o. since

$$\mathbb{P}(B_n) = \mathbb{P}\left(\sup_{x\in\Omega} |\mathrm{H}_{N(n)}(x) - p^n(x)| > \alpha(1-a_0) - \prod_{k=1}^{n-1}(1-a_k)\, D_M^0\right), \tag{20}$$

and an increasing rate for $N(n)$ similar to (19) insures $\sum_{n\geqslant 0}\mathbb{P}(B_n) < \infty$. The Borel–Cantelli lemma holds also for C_n since

$$\mathbb{P}(C_n) \leqslant \mathbb{P}(A_n) + (1 - \mathbb{P}(A_n))\mathbb{P}(B_n).$$

Finally, (17) is easily deduced since events $\{a_n < a_0\}$ cannot occur i.o.

5. Extension to general densities

We propose here an extension of our algorithm to general density functions ($\Omega = \mathbb{R}^s$), and derive some results about its good asymptotic behaviour under a theoretical hypothesis. The following algorithm will take benefit of the behaviour of our method on compact sets of $\mathbb{R}^s$, the intuitive idea being that if we have a good approximation of the tails of f out of a compact set, then our algorithm will provide a convergence rate associated to the minorization constant coming from this approximation. This approximation of the tails of f may not be easy to find in complex enough situations, even if the analytical form of f is available in the Hastings–Metropolis context. In these situations, one can either use the previous adaptive method on a compact large enough, so that the tails become negligible, or use a proposal as below, built from the histogram over a compact set and a generic function g allowing some exploration on the tails (without a precise approximation of f). The result given below must be understand as an ideal situation.

In order to simulate a general density function f defined on $\mathbb{R}^s$, we propose to use the Hastings–Metropolis algorithm described in section 2 with a convenient sequence of proposal densities $(q_n)_{n\geqslant 0}$. We have seen in sections 3 and 4 the good behaviour of the histogram used as a proposal distribution when Ω were a compact set. Suppose now that for $0 < a < 1$ and $1 < b$ fixed, close to one, we can find a compact set $K \subset \mathbb{R}^s$ and a function $g \in L^1$, such that:

$$\text{for all } x \in \mathbb{R}^s\backslash K\text{: } bf(x) \geqslant g(x) \geqslant af(x), \tag{21}$$

which means that for K large enough the tails of f outside K are close to the tails of a generic (easy to simulate) function g. We denote in addition the weights of f and g's tails by:

$$m_g = \int_{\mathbb{R}^s\backslash K} g(x)\,dx, \quad m_f = \int_{\mathbb{R}^s\backslash K} f(x)\,dx.$$

Without loss of generality K will be identified to $[0, 1]^s$ to simplify the notation. We propose to build the proposal density at time n in the following way: let $R(n)$ be the random variable denoting the number of i.i.d. copies of the algorithm remaining in K at time n within the $N(n)$ used to explore p^n. We then define q_n^* by:

$$q_n^*(x) = (H^*_{R(t_i)}(x)\mathbb{I}_{C^*(t_i)} + \tilde{H}^*_{R(t_i)}(x)\mathbb{I}_{\bar{C}^*_{(t_i)}})\mathbb{I}_{[t_i,t_{i+1}[}(n), \tag{22}$$

where

$$H^*_{R(n)}(x) = H_{N(n)}(x)\mathbb{I}_K(x) + \frac{N(n)-R(n)}{N(n)m_g}g(x)\mathbb{I}_{\mathbb{R}^s\backslash K}(x), \tag{23}$$

and $\bar{C}^*(n) = \{\inf_{x\in K}(H^*_{R(n)}(x)) \leqslant 0\}$ is a situation where $H^*_{R(n)}$ has to be changed to $\tilde{H}^*_{R(n)}$ in K in order to insure $q^*_n(x) > 0$ for all $x \in \mathbb{R}^s$. Clearly, $\int_{\mathbb{R}^s} q^*_n(x)\,dx = 1$ since $\int_K H_{N(n)}(x)\,dx = R(n)/N(n)$.

Proposition 4

If the number $N(n)$ of i.i.d. copies of the algorithm, and the class width $h_{N(n)}$ used to compute $H^*_{R(n)}$ satisfy conditions from theorem 1, then

$$\sup_{x\in K} |H_{N(n)}(x) - f(x)| \to 0 \quad a.s., \text{ when } n \to \infty,$$

$$\frac{N(n) - R(n)}{N(n)} \to \int_{\mathbb{R}^s\setminus K} f(x)\,dx \quad a.s., \text{ when } n \to \infty,$$

and the sequence (a_n) insuring at each step $q^*_n(x) \geqslant a_n f(x)$ tends a.s. to $\bar{a} = a(m_f/m_g)$, where $a/b \leqslant \bar{a} \leqslant 1$.

Proof. The first result is a consequence of the a.s. uniform convergence of $H_{N(n)}$ to f on any compact set $K \subset \mathbb{R}^s$, noticed in the proof of theorem 1. This remark implies obviously the a.s. convergence of the integrals $\int_K H_{R(n)}(x)\,dx = R(n)/N(n)$ to $\int_K f(x)\,dx$, and thus the second result. Finally the sequence (a_n) insuring at each step $q^*_n(x) \geqslant a_n f(x)$ is asymptotically driven by the behaviour of q^* on $\mathbb{R}^s\setminus K$, which converges to $(m_f/m_g)g$ and thus implies the conclusion since $bm_f \geqslant m_g \geqslant am_f$.

6. Implementation and examples

6.1. *Practical implementation of the algorithm*

As pointed out in the introduction, the application of our method in practice will not be done through a straightforward implementation of the theoretical framework. A good way of taking advantage of the convergence improvements provided by our algorithm in comparison with its classical counterparts (e.g. a single chain Hastings–Metropolis with a random walk proposal distribution), is to consider its two interesting features: (1) the use of parallel chains started from a dispersed initial distribution, which provide a good exploration of the support of f, hopefully discovering all its modes; (2) the generation of the next move from a "good" proposal density after some exploration stages, which does attribute the appropriate weight to each (discovered) modes of f, avoiding under or over-estimations of distant modes and resulting in a good acceptance ratio.

Advantages of point (1) need to be clarified: of course, we can never be sure that the parallel chains started using a dispersed enough initial distribution will discover all the regions of interest of f. However, reasonably, this setting should do better than a single chain running for as many iterations as the iterations of all the i.i.d. chains, simply because a single chain may require a long time to escape from a modal region. Actually, in many MCMC situations, a prior exploration of the target distribution through "mode hunting methods" (as the one proposed by Gelman & Rubin, 1992) is recommended, as pointed out by Brooks & Roberts (1998). To some extent, our method can be considered as an exploratory method, with the particular feature that it uses this exploration step to produce a Hastings–Metropolis algorithm with an improved kernel, i.e. using a "good" proposal density. Point (2) will be illustrated in the simulated example, which provides some evidence that a parallel chains method using a random walk Hastings-Metropolis algorithm can lead to dramatically wrong

over-estimations of modes when a fraction of the parallel chains get trapped in some distant region.

A reasonable way of using our algorithm is to implement an exploration scheme which quickly visits the state space to locate the regions of interest for f, using some sets of parallel chains across a small number of mutations, and to end up with a single chain using a good proposal density resulting from this exploratory and adaptive stage. This is in accordance with the theoretical result given in section 4, which says that after some finite time, a single chain issued from our method is faster then a single chain using the initial fixed proposal density. More precisely, we suggest to:

(i) select a small number k of mutation times $I = (t_1, \ldots, t_k)$, which rapidly occur (i.e. the t_is are small) and use sets of an increasing number of chains $N(t_1), \ldots, N(t_k)$ to build the proposal densities at each mutation as in (7);
(ii) start with a total number $N = 1 + \sum_{i=1}^{k} N(t_i)$ of parallel chains, initialized from a uniform-like distribution over a compact large enough (typically to encompass the regions of interest for f while omitting the tails);
(iii) discard after each mutation time t_i the $N(t_i)$ chains used to build q_{t_i} from the histogram (to preserve Markov property and independence, as discussed in section 3);
(iv) after time t_k run the single (and only remaining) Hastings–Metropolis chain with the proposal density q_{t_k} issued from the last mutation.

Since the exploratory stage of our method (parallel chains, mutations and elimination of used chains) is costly in simulation jumps, we want it to be as short as possible. The idea is that if the last proposal density is close enough to f, then we do not need parallel chains anymore to insure a good exploration. The question of how to choose the tuning parameters of this preliminary stage (namely k, the t_is and the $N(t_i)$s) leads us to a trade-off between precision for the resulting single chain algorithm and simulation cost. From the end user point of view, the implementation cost is reduced since we propose a "black-box" type computer code implementing this exploratory stage i.e. the parallel simulation and the histograms construction in a multidimensional setting. (This black-box computer code in `ANSI C` is available upon request to the first author.) This black-box algorithm has been used to run the examples. The only specific implementation tasks were to plug the definitions of the target densities for each model into this computer code, and to set the tuning parameters k, t_is and $N(t_i)$s defined above.

Two examples in one and two-dimensional settings follow. Note that the comparisons are not quite easy to do at first sight, since the classical Hastings–Metropolis methods use single chains, and our adaptive method uses parallel chains in essence. To be fair, we always allow the classical methods a number of iterations equal to the total amount of iterations used by the corresponding adaptive method to build its histograms across the parallel simulations. This means that the experiments are comparable from this point of view.

6.2. A simple example

Our purpose in this section is to illustrate on a toy but meaningful example how our method may perform better in practice than a classical Hastings–Metropolis algorithm. Since our algorithm is expected to behave well in multimodal or slowly mixing situations, we selected a trimodal target density f issued from a mixture of three univariate Gaussian distributions,

$$f(x) = \sum_{i=1}^{3} \alpha_i \varphi(x; \mu_i, \sigma_i^2), \tag{24}$$

where $\varphi(\cdot;\ \mu, \sigma^2)$ denotes the p.d.f. of the Gaussian $\mathcal{N}(\mu, \sigma^2)$. The parameters are $\alpha_1 = 0.7$, $\alpha_2 = 0.05$, $\mu_1 = 0, \mu_2 = 15$, $\mu_3 = -6$, $\sigma_1^2 = 1, \sigma_2^2 = 0.1$ and $\sigma_3^2 = 2$, resulting in two "close" modes located in -6 and 0, and a smaller "distant" mode located in 15. The true p.d.f. with this setting is depicted in solid line on the figures.

We ran our algorithm with an exploratory scheme. It has been initialized with 231 parallel chains, and has performed four mutations at times $I = (1, 3, 5, 7)$, with sets of parallel chains of sizes $N(\cdot) = (40, 50, 60, 80)$. The single chain using q_7 was simulated up to $n = 2000$. This algorithm uses a total of 3050 jumps, 1050 of which were used to build the four proposal densities.

We tried to compare our method against a popular and often used random walk Hastings–Metropolis algorithm (RWHM), which uses the Gaussian proposal density $q(\cdot|x) = \mathcal{N}(x, \sigma^2)$ with a variance parameter which is often crucial for the performance of the algorithm, and is usually tuned by trial and error. Since our method uses parallel chains (at least during its exploration stage), we believe that for a fair comparison, the RWHM needs to be run in both the single and parallel implementation. We thus ran the RWHM algorithm using (i) a single chain for 3050 iterations, and (ii) 50 parallel chains running each for 61 iterations, resulting in a total number of 3050 jumps.

All the methods were initialized using the same uniform distribution over a compact set $\Omega = [-15;\ 20]$ encompassing the three modes, and our method used this distribution for its arbitrary starting proposal density q_0. Notice that the q_ns are positive in Ω only, even if the support of f is the real line, but we believe that this approximation omitting the tails is good enough since $\int_{\mathbb{R}\setminus\Omega} f(x)\,dx \approx 10^{-11}$ here. We give the results in terms of the estimated p.d.f. (histogram computed using all the iterations for both methods) against the true p.d.f. (24).

For the single chain RWHM (Fig. 1, top), the quality of the recovering heavily depends (as expected) on the tuning of the variance parameter: for too small variance parameter $\sigma^2 \leqslant 15$, the algorithm "misses" the distant mode during the first 3050 iterations. The setting $\sigma^2 \approx 20$ allows for the best recovering of f. After that, the estimation deteriorates for too large σ^2 (it runs approximately like an independence sampler with a uniform proposal density).

For the RWHM based on 50 i.i.d. chains, the recovering of the target density f was even worse (see Fig. 1, bottom). Here, a fraction of the parallel chains started uniformly over a compact large enough actually discovered the distant mode, but because of the random walk proposal distribution, the chains started within the distant mode could not escape from this modal region (at least not during the first 61 iterations of each single run). Several attempts with different values for σ^2 up to $\sigma^2 = 50$ did not give better results and Fig. 1, bottom, represents typically what we obtained.

Our exploration scheme with four mutations provided a reasonably good fit of the target density f, as shown in Fig. 2: it is comparable to the estimation obtained by the random walk with the best choice for the variance parameter. The appealing feature of our algorithm is that it does not require such sharp, data-driven tuning of technical parameters (like the variance of the random walk) provided that the initial compact being large enough. It delivers roughly the same approximation of f for several comparable definitions of the exploratory stage. The successive proposal densities for this example are in Fig. 2. The fourth mutation (Fig. 2, bottom right) illustrates the way q_7 appropriately reflects the global shape of f.

6.3. A multidimensional example

To show that our method can be implemented, and also improve convergence in multi-dimensional settings, we used it to recover a four-component mixture of bivariate Gaussian distributions. The true p.d.f. has been defined to generate four spread modes, with the true parameters

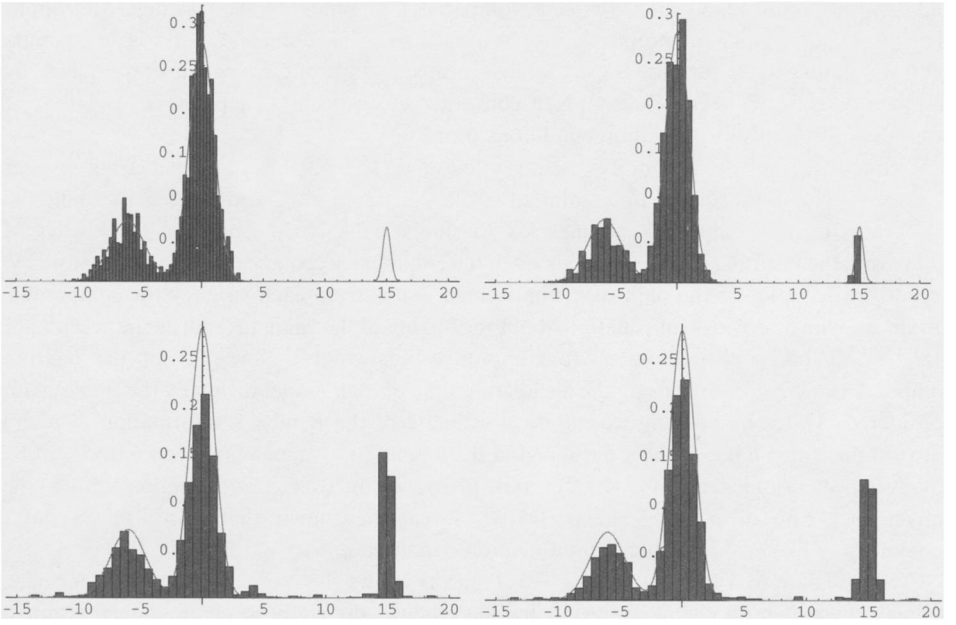

Fig. 1. Random walk Hastings–Metropolis for a total of 3050 iterations. Top left: for a single chain, with random walk variance $\sigma^2 = 8$. Top right: for a single chain with $\sigma^2 = 20$; this setting for σ^2 gives the best fit. Bottom left: for 50 parallel chains, $n = 61$ steps, and $\sigma^2 = 8$. Bottom right: for 50 parallel chains, $n = 61$ steps, and $\sigma^2 = 20$. Each plot gives the histogram of the estimated density over the iterations, against the true p.d.f. (solid line).

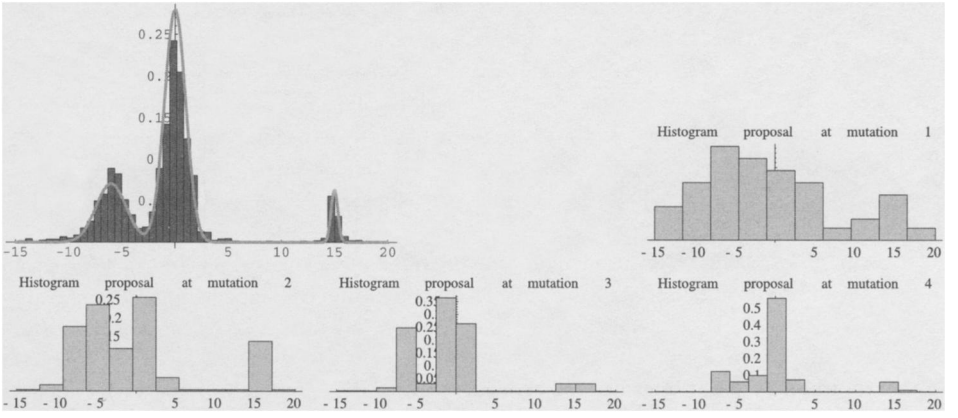

Fig. 2. Hastings–Metropolis algorithm with adaptive proposal density (four mutations), resulting in an overall number of iterations of 3050. Top left: the estimated and true p.d.f. Top right and bottom: the q_ts at mutation times.

$$\text{weights:}\quad \alpha_1 = 0.5,\ \alpha_2 = 0.3,\ \alpha_3 = 0.15, \alpha_4 = 0.05,$$

$$\text{means:}\quad \mu_1 = \begin{pmatrix} +10 \\ -10 \end{pmatrix},\ \mu_2 = \begin{pmatrix} 15 \\ 15 \end{pmatrix},\ \mu_3 = \begin{pmatrix} -15 \\ -15 \end{pmatrix},\ \mu_4 = \begin{pmatrix} -12 \\ +07 \end{pmatrix},$$

$$\text{variances:}\ \Sigma_1 = \Sigma_2 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix},\ \Sigma_3 = \begin{pmatrix} 0.5 & 0 \\ 0 & 3 \end{pmatrix},\ \Sigma_4 = \begin{pmatrix} 0.5 & 0 \\ 0 & 1 \end{pmatrix},$$

and is shown in Fig. 3, left. We have compared an homogeneous Hastings–Metropolis algorithm with a uniform proposal density over a compact domain $C \in \mathbb{R}^2$ large enough, against our adaptive algorithm. Since the mass located outside of C is negligible, we essentially compare here two Markov chains which converge geometrically (the proposal densities for both methods satisfy minorization conditions on C).

We ran our method with four mutations at times $I = (1, 3, 6, 10)$ with an increasing number of parallel chains, resulting in a total of 2850 individual realizations used to build the histograms (other comparable settings led to similar results). We propose three ways of comparing the resulting chains. First, through the empirical acceptance ratio for each strategy. Second, with a plot of the path of a single chain issued from each strategy for comparable durations, which provides information about the mixing of the chain (as well as the acceptance rate). These paths also allow us to compute estimates of the weights of the mixture components α_i, $i = 1, \ldots, 4$ using the occupation time of each modal region by the single chain considered. Third, by plotting an empirical estimate of the Kullback information $K(p^n, f)$ between the n-step p.d.f. of the algorithm and the target. This estimate has been introduced by Chauveau & Vandekerkhove (2000), and provides an overall information about the convergence rate of p^n to f. In particular, it can be shown that $K(p^n, f)$ decreases geometrically like $(1 - a)^n$ when a minorization condition $q \geqslant af$ holds.

The results are in Table 1 and Figs 3–5. Figure 3 shows the estimate of $K(p^n, f)$ for each strategy. Even if both chains are geometrically ergodic, the adaptive chain converges much faster than the homogeneous one. The single chain paths in Fig. 4 show that the adaptive chain mixes in a better way than the homogeneous chain: the first two plots compare the two strategies during $n = 1000$ iterations. The rightmost plot gives the path of the homogeneous chain (uniform proposal density) for $n = 3850$ iterations, which corresponds to 1000 iterations

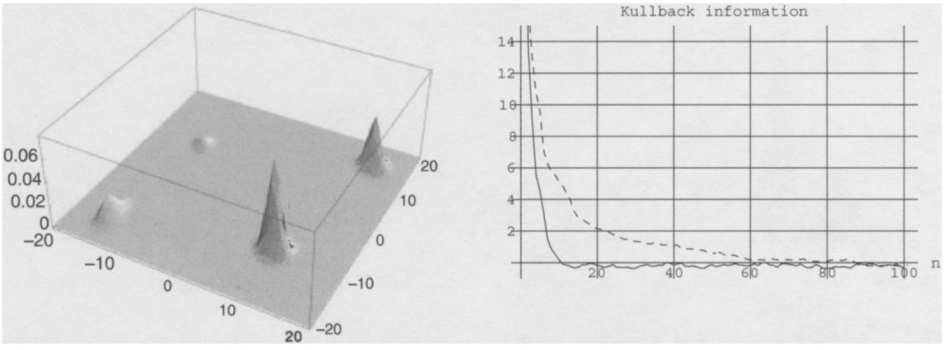

Fig. 3. Left: true p.d.f. Right: estimated Kullback information $K(p^n, f)$ for the adaptive chain (solid line) and the homogeneous chain (dashed).

Table 1. *Table of the true weights of the mixture components $\alpha_i, i = 1, \ldots, 4$, and their estimates based on a single chain for each strategy, in per cent*

	true	adaptive $n = 1000$	uniform $n = 1000$	uniform $n = 3850$
α_1	50	45	30.5	35.7
α_2	30	27.7	55.4	37.2
α_3	15	23.5	14	25.6
α_4	5	3.9	0.1	1.4

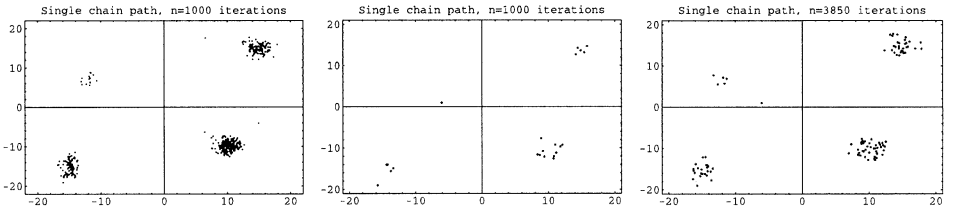

Fig. 4. Single chain path. Left: for $n = 1000$ steps of the adaptive chain. Centre: for $n = 1000$ steps of the homogeneous chain. Right: for $n = 3850$ steps of the same homogeneous chain.

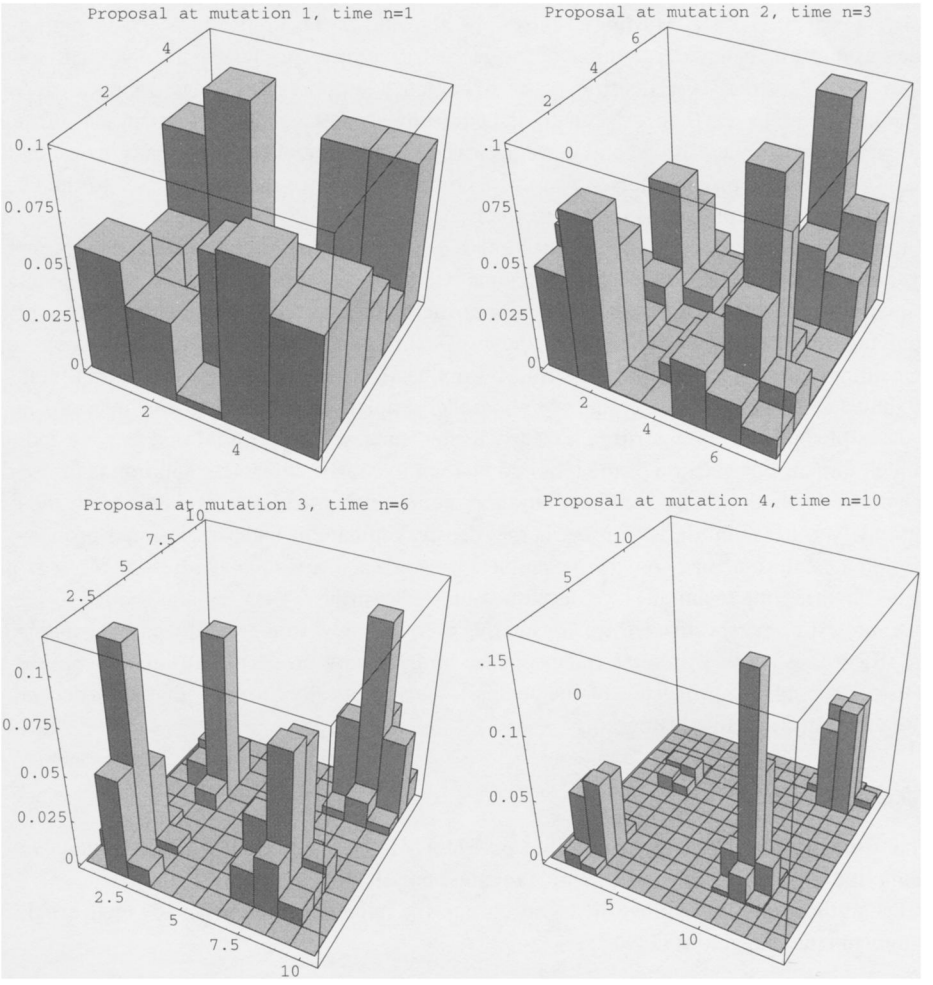

Fig. 5. Successive proposal densities at mutation times.

plus the total number of iterations used by the adaptive chain to build its histograms. We see that, even with that many iterations, the way the homogeneous chain visits each mode is not as good as the visit of the adaptive chain. Here, the empirical acceptance ratios were about 3% for the homogeneous chain, and up to 35% with the last proposal density (mutation 4) for the adaptive chain. We give in Table 1 the estimates of the weights of the mixture components.

Clearly, the adaptive chain gives the best estimates, particularly for the smallest mode ($\alpha_4 = 5\%$). The independence sampler with the uniform proposal density underestimates this smallest mode, and even inverts the weights of the first two modes. Finally, Fig. 5 gives the successive updates of the histogram proposal densities. The last histogram is clearly a "good" proposal density, even if it has many empty cells (and has consequently been modified as in (7) to give a positive density).

7. Conclusion

In this paper, we have defined a new inhomogeneous Hastings–Metropolis algorithm taking advantage of parallel chains and of an adaptive sequence of proposal densities to improve the convergence of classical Hastings–Metropolis algorithms. Theoretically, we have shown its good asymptotic behaviour in terms of the geometric convergence rate of a single chain issued from this algorithm: for densities over a compact support, this single chain converges asymptotically faster than any chain issued from an homogeneous Hastings–Metropolis independence sampler; for general densities (over $\mathbb{R}^s$), the rate of convergence is driven by the quality of some approximation of f out of a compact set, since the convergence to the target density inside the compact is related to the previous result.

Intuitively, this adaptive algorithm should behave well in multimodal or slowly mixing situations, and we tried to provide some evidence of this by suggesting a practical implementation of the method which mimics the theoretical (asymptotic) situation, and by applying it to simulated examples. For these typical examples in one and two dimensions, our algorithm performs better than classical Metropolis algorithms in a comparable total number of simulations (i.e. taking account of the parallel simulations required by our method).

The drawback of our algorithm is clearly its implementation cost, that may become tedious in high dimensions because of the required histogram construction. One solution is to use the generic code we developed for the multidimensional setting, which is available from the first author. Another solution is to use a kernel density estimate in high dimensional problems if the implementation turns out to be simpler, since even if a similar theoretical result is not available using the techniques we developed here, the intuitive idea remains.

A interesting perspective we are considering is to find how to recycle the parallel chains we are discarding here to preserve independence and Markov property. An answer may be to consider the mixing properties of the chains to asymptotically "forget" that they depend at some time on the same histogram.

Acknowledgements

This work has been partially supported by EU TMR Network ERB-FMRX-CT96-0095 on Computational and Statistical methods for the analysis of spatial data.

The authors wish to express their gratitude to the referees of this paper for their insightful comments and suggestions.

References

Bosq, D. & Lecoutre, J. P. (1987). *Théorie de l'estimation fonctionnelle*. Economica, Paris.

Brooks, S. P. & Roberts, G. (1998). Assessing convergence of Markov Chain Monte Carlo algorithms. *Statist. Comput.* **8**, 319–335.

Chauveau, D. & Diebolt, J. (1999). An automated stopping rule for MCMC convergence assessment. *Comput. Statist.* **14**, 419–442.

Chauveau, D. & Vandekerkhove, P. (1999). Un algorithme de Hastings–Metropolis avec apprentissage séquentiel. *C. R. Acad. Sci. Paris Sér. I Math.* **329**, 173–176.

Chauveau, D. & Vandekerkhove, P. (2000). *Comparaison de vitesse de convergence d'algorithmes de Hastings–Metropolis basée sur l'entropie*. Actes des XXXIIèmes Journées de Statistique, Fès, Maroc.

Gelfand, A. E. & Smith, A. F. M. (1990). Sampling based approaches to calculating marginal densities. *J. Amer. Statist. Assoc.* **85**, 398–409.

Gelman, A. & Rubin, D. B. (1992). Inference from iterative simulation using multiple sequences. *Statist. Sci.* **7**, 457–511.

Geman, S. & Geman, D. (1984). Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. *IEEE Trans. Pattern Anal. Mach. Intell.* **6**, 721–741.

Green, P. J. (1995). Reversible jump MCMC computation and Bayesian model determination. *Biometrika* **82**, 711–732.

Hastings, W. K. (1970). Monte Carlo sampling methods using Markov Chains and their applications. *Biometrika* **57**, 97–109.

Holden, L. (1998). Geometric convergence of the Metropolis–Hastings simulation algorithm. *Statist. Probab. Lett.* **39**, 371–377.

Mengersen, K. L. & Tweedie, R. L. (1996). Rates of convergence of the Hastings and Metropolis algorithms. *Ann. Statist.* **24**, 101–121.

Robert, C. P. (1996). *Méthodes de Monte Carlo par chaînes de Markov*. Economica, Paris.

Roberts, G. O. & Tweedie, R. L. (1996). Geometric convergence and central limit theorems for multidimensional Hastings and Metropolis algorithms. *Biometrika* **83**, 95–110.

Vandekerkhove, P. (1998). A sequential Metropolis–Hastings algorithm. Preprint, Università di Pavia, Italy.

Received June 2000, in final form February 2001

Didier Chauveau, Université de Marne-la-Vallée, Analyse et Mathématiques Appliquées, 5 Bd. Descartes, Champs-sur-Marne, 77454 Marne-la-Vallée Cedex 2, France.
E-mail: chauveau@math.univ-mlv.fr