Stat Comput (2008) 18: 343–373
DOI 10.1007/s11222-008-9110-y

A tutorial on adaptive MCMC

Christophe Andrieu · Johannes Thoms

Received: 23 January 2008 / Accepted: 19 November 2008 / Published online: 3 December 2008
© Springer Science+Business Media, LLC 2008

Abstract We review adaptive Markov chain Monte Carlo algorithms (MCMC) as a mean to optimise their performance. Using simple toy examples we review their theoretical underpinnings, and in particular show why adaptive MCMC algorithms might fail when some fundamental properties are not satisfied. This leads to guidelines concerning the design of correct algorithms. We then review criteria and the useful framework of stochastic approximation, which allows one to systematically optimise generally used criteria, but also analyse the properties of adaptive MCMC algorithms. We then propose a series of novel adaptive algorithms which prove to be robust and reliable in practice. These algorithms are applied to artificial and high dimensional scenarios, but also to the classic mine disaster dataset inference problem.

Keywords MCMC · Adaptive MCMC · Controlled Markov chain · Stochastic approximation

1 Introduction

Markov chain Monte Carlo (MCMC) is a general strategy for generating samples $\{X_i,\ i = 0, 1, \ldots\}$ from complex high-dimensional distributions, say π defined on a space $\mathsf{X} \subset \mathbb{R}^{n_x}$ (assumed for simplicity to have a density with respect to the Lebesgue measure, also denoted π), from which integrals of the type

$$I(f) := \int_{\mathsf{X}} f(x)\pi(x)\,dx,$$

for some π-integrable functions $\mathsf{X} \to \mathbb{R}^{n_f}$ can be approximated using the estimator

$$\hat{I}_N(f) := \frac{1}{N}\sum_{i=1}^{N} f(X_i), \tag{1}$$

provided that the Markov chain generated with, say, transition P is ergodic *i.e.* it is guaranteed to eventually produce samples $\{X_i\}$ distributed according to π. Throughout this review we will refer, in broad terms, to the consistency of such estimates and the convergence of the distribution of X_i to π as π-ergodicity. The main building block of this class of algorithms is the Metropolis-Hastings (MH) algorithm. It requires the definition of a family of proposal distributions $\{q(x,\cdot), x \in \mathsf{X}\}$ whose role is to generate possible transitions for the Markov chain, say from X to Y, which are then accepted or rejected according to the probability

$$\alpha(X,Y) = \min\left\{1, \frac{\pi(Y)q(Y,X)}{\pi(X)q(X,Y)}\right\}.$$

The simplicity and universality of this algorithm are both its strength and weakness. Indeed, the choice of the proposal distribution is crucial: the statistical properties of the Markov chain heavily depend upon this choice, an inadequate choice resulting in possibly poor performance of the Monte Carlo estimators. For example, in the toy case where $n_x = 1$ and the normal symmetric random walk Metropolis algorithm (N-SRWM) is used to produce transitions, the

C. Andrieu (✉)
School of Mathematics, University of Bristol,
Bristol BS8 1TW, UK
e-mail: c.andrieu@bristol.ac.uk
url: http://www.stats.bris.ac.uk/~maxca

J. Thoms
Chairs of Statistics, École Polytechnique Fédérale de Lausanne,
1015 Lausanne, Switzerland

density of the proposal distribution is of the form

$$q_\theta(x, y) = \frac{1}{\sqrt{2\pi\theta^2}} \exp\left(\frac{-1}{2\theta^2}(y - x)^2\right),$$

where θ^2 is the variance of the proposed increments, hence defining a Markov transition probability P_θ. The variance of the corresponding estimator $\hat{I}_N^\theta(f)$, which we wish to be as small as possible for the purpose of efficiency, is well known to be typically unsatisfactory for values of θ^2 that are either "too small or too large" in comparison to optimal or suboptimal value(s). In more realistic scenarios, MCMC algorithms are in general combinations of several MH updates $\{P_{k,\theta},\ k = 1, \ldots, n,\ \theta \in \Theta\}$ for some set Θ, with each having its own parametrised proposal distribution $q_{k,\theta}$ for $k = 1, \ldots, n$ and sharing π as common invariant distribution. These transition probabilities are usually designed in order to capture various features of the target distribution π and in general chosen to complement one another. Such a combination can for example take the form of a mixture of different strategies, *i.e.*

$$P_\theta(x, dy) = \sum_{k=1}^{n} w_k(\theta) P_{k,\theta}(x, dy), \tag{2}$$

where for any $\theta \in \Theta$, $\sum_{k=1}^{n} w_k(\theta) = 1$, $w_k(\theta) \geq 0$, but can also, for example, take the form of combinations (*i.e.* products of transition matrices in the discrete case) such as

$$P_\theta(x, dy) = P_{1,\theta} P_{2,\theta} \cdots P_{n,\theta}(x, dy).$$

Both examples are particular cases of the class of Markov transition probabilities P_θ on which we shall focus in this paper: they are characterised by the fact that they (a) belong to a family of parametrised transition probabilities $\{P_\theta, \theta \in \Theta\}$ (for some problem dependent set Θ, $\Theta = (0, +\infty)$ in the toy example above) (b) for all $\theta \in \Theta$ π is an invariant distribution for P_θ, which is assumed to be ergodic (c) the performance of P_θ, for example the variance of $\hat{I}_N^\theta(f)$ above, is sensitive to the choice of θ.

Our aim in this paper is to review the theoretical underpinnings and recent methodological advances in the area of computer algorithms that aim to "optimise" such parametrised MCMC transition probabilities in order to lead to computationally efficient and reliable procedures. As we shall see we also suggest new algorithms. One should note at this point that in some situations of interest, such as tempering type algorithms (Geyer and Thompson 1995), property (b) above might be violated and instead the invariant distribution of P_θ might depend on $\theta \in \Theta$ (although only a non θ-dependent feature of this distribution π_θ might be of interest to us for practical purposes). We will not consider this case in depth here, but simply note that most of the arguments and ideas presented hereafter generally carry on to this slightly more complex scenario *e.g.* (Benveniste et al. 1990; Atchadé and Rosenthal 2005).

The choice of a criterion to optimise is clearly the first decision that needs to be made in practice. We discuss this issue in Sect. 4.1 where we point out that most sensible optimality or suboptimality criteria can be expressed in terms of expectations with respect to the steady state-distributions of Markov chains generated by P_θ for $\theta \in \Theta$ fixed, and make new suggestions in Sect. 5 which are subsequently illustrated on examples in Sect. 6. We will denote by θ^* a generic optimal value for our criteria, which is always assumed to exist hereafter.

In order to optimise such criteria, or even simply find suboptimal values for θ, one could suggest to sequentially run a standard MCMC algorithm with transition P_θ for a set of values of θ (either predefined or defined sequentially) and compute the criterion of interest (or its derivative etc.) once we have evidence that equilibrium has been reached. This can naturally be wasteful and we will rather focus here on a technique which belongs to the well known class of processes called controlled Markov chains (Borkar 1990) in the engineering literature, which we will refer to as controlled MCMC (Andrieu and Robert 2001), due to their natural filiation. More precisely we will assume that the algorithm proceeds as follows. Given a family of transition probabilities $\{P_\theta, \theta \in \Theta\}$ defined on X such that for any $\theta \in \Theta$, $\pi P_\theta = \pi$ (meaning that if $X_i \sim \pi$, then $X_{i+1} \sim \pi, X_{i+2} \sim \pi, \ldots$) and given a family of (possibly random) mappings $\{\theta_i : \Theta \times \mathsf{X}^{i+1} \to \Theta, i = 1, \ldots\}$, which encodes what is meant by optimality by the user, the most general form of a controlled MCMC proceeds as follows:

Algorithm 1 Controlled Markov chain Monte Carlo

- Sample initial values $\theta_0, X_0 \in \Theta \times \mathsf{X}$.
- Iteration $i + 1$ ($i \geq 0$), given $\theta_i = \theta_i(\theta_0, X_0, \ldots, X_i)$ from iteration i
  1. Sample $X_{i+1} | (\theta_0, X_0, \ldots, X_i) \sim P_{\theta_i}(X_i, \cdot)$.
  2. Compute $\theta_{i+1} = \theta_{i+1}(\theta_0, X_0, \ldots, X_{i+1})$.

In Sect. 4.2 we will focus our results to particular mappings well suited to our purpose of computationally efficient sequential updating of $\{\theta_i\}$ for MCMC algorithms, which rely on the Robbins-Monro update and more generally on the stochastic approximation framework (Benveniste et al. 1990). However, before embarking on the description of practical procedures to optimise MCMC transition probabilities we will first investigate, using mostly elementary undergraduate level tools, some of the theoretical ergodicity properties of controlled MCMC algorithms.

Indeed, as we shall see, despite the assumption that for any $\theta \in \Theta$, $\pi P_\theta = \pi$, adaptation in the context of MCMC

using the controlled approach leads to complications. In fact, this type of adaptation can easily perturb the ergodicity properties of MCMC algorithms. In particular algorithms of this type will in most cases lead to the loss of π as an invariant distribution of the process $\{X_i\}$, which intuitively should be the minimum requirement to produce samples from π and lead to consistent estimators. Note also that when not carefully designed such controlled MCMC can lead to transient processes or processes such that $\hat{I}_N(f)$ is not consistent. Studying the convergence properties of such processes naturally raises the question of the relevance of such developments in the present context. Indeed it is often argued that one might simply stop adaptation once we have enough evidence that $\{\theta_i\}$ has reached a satisfactory optimal or suboptimal value of θ and then simply use samples produced by a standard MCMC algorithm using such a fixed good value $\tilde{\theta}$. No new theory should then be required. While apparently valid, this remark ignores the fact that most criteria of interest depend explicitly on features of π, which can only be evaluated with... MCMC algorithms. For example, as mentioned above most known and useful criteria can be formulated as expectations with respect to distributions which usually explicitly involve π.

Optimising such criteria, or finding suboptimal values of θ^*, thus requires one to be able to sample—perhaps approximately or asymptotically—from π, which in the context of controlled MCMC requires one to ensure that the process described above can, in principle, achieve this aim. This, in our opinion, motivates and justifies the need for such theoretical developments as they establish whether or not controlled MCMC can, again in principle, optimise such π-dependent criteria. Note that convergence of $\{\theta_i\}$ should itself not be overlooked since, in light of our earlier discussion of the univariate N-SRWM, optimisation of $\{P_\theta\}$ is our primary goal and should be part of our theoretical developments. Note that users wary of the perturbation to ergodicity brought by adaptation might naturally choose to "freeze" $\{\theta_i\}$ to a value θ_τ beyond an iteration τ and consider only samples produced by the induced Markov chain for their inference problem. A stopping rule is described in Sect. 4.2.2. In fact, as we shall see it is possible to run the two procedures simultaneously.

Finally, whereas optimising an MCMC algorithm seems a legitimate thing to do, one might wonder if it is computationally worth adapting. This is a very difficult question for which there is probably no straight answer. The view we adopt here is that such optimisation schemes are very useful tools to design or help the design of efficient MCMC algorithms which, while leading to some additional computation, have the potential to spare the MCMC user significant implementation time.

The paper is organised as follows. In Sect. 2 we provide toy examples that illustrate the difficulties introduced by the adaptation of MCMC algorithms. In Sect. 3 we discuss why one might expect vanishing adaptation to lead to processes such that $\{X_i\}$ can be used in order to estimate expectation with respect to π. This section might be skipped on a first reading. In Sect. 4 we first discuss various natural criteria which are motivated by theory, but to some extent simplified in order to lead to useful and implementable algorithms. We then go on to describe how the standard framework of stochastic approximation, of which the Robbins-Monro recursion is the cornerstone, provides us with a systematic framework to design families of mappings $\{\theta_i\}$ in a recursive manner and understand their properties. In Sect. 5 we present a series of novel adaptive algorithms which circumvent some of the caveats of existing procedures. These algorithms are applied to various examples in Sect. 6.

2 The trouble with adaptation

In this section we first illustrate the loss of π-ergodicity of controlled MCMC with the help of two simple toy examples. The level of technicality required for these two examples is that of a basic undergraduate course on Markov chains. Despite their simplicity, these examples suggest that vanishing adaptation (a term made more precise later) might preserve asymptotic π-ergodicity. We then finish this section by formulating more precisely the fundamental difference between standard MCMC algorithms and their controlled counterparts which affects the invariant distribution of the algorithm. This requires the introduction of some additional notation used in Sect. 4 and a basic understanding of expectations to justify vanishing adaptation, but does not significantly raise the level of technicality.

Consider the following toy example, suggested in Andrieu and Moulines (2006), where $\mathsf{X} = \{1, 2\}$ and $\pi = (1/2, 1/2)$ (it is understood here that for such a case we will abuse notation and use π for the vector of values of π and P_θ for the transition matrix) and where the family of transition probabilities under consideration is of the form, for any $\theta \in \Theta := (0, 1)$

$$P_\theta = \begin{bmatrix} P_\theta(X_i = 1, X_{i+1} = 1) & P_\theta(X_i = 1, X_{i+1} = 2) \\ P_\theta(X_i = 2, X_{i+1} = 1) & P_\theta(X_i = 2, X_{i+1} = 2) \end{bmatrix} = \begin{bmatrix} \theta & 1-\theta \\ 1-\theta & \theta \end{bmatrix}. \tag{3}$$

It is clear that for any $\theta \in \Theta$, π is a left eigenvector of P_θ with eigenvalue 1,

$$\pi P_\theta = \pi,$$

i.e. π is an invariant distribution of P_θ. For any $\theta \in \Theta$ the Markov chain is obviously irreducible and aperiodic, and by standard theory is therefore ergodic, *i.e.* for any starting

probability distribution μ,

$$\lim_{i\to\infty} \mu P_\theta^i = \pi$$

(with P_θ^i the i-th power of P_θ), and for any finite real valued function f

$$\lim_{N\to\infty} \frac{1}{N}\sum_{i=1}^{N} f(X_i) = \mathbb{E}_\pi(f(X)),$$

almost surely, where for any probability distribution ν, $\mathbb{E}_\nu$ represents the expectation operator with respect to ν. Now assume that θ is adapted to the current state in order to sample the next state of the chain, and assume for now that this adaptation is a time invariant function of the previous state of the MC. More precisely assume that for any $i \geq 1$ the transition from X_i to X_{i+1} is parametrised by $\theta(X_i)$, where $\theta : \mathsf{X} \to \Theta$. The remarkable property, specific to this purely pedagogical example, is that $\{X_i\}$ is still in this case a time homogeneous Markov chain with transition probability

$$\check{P}(X_i = a, X_{i+1} = b) := P_{\theta(a)}(X_i = a, X_{i+1} = b)$$

for $a, b \in \mathsf{X}$, resulting in the time homogeneous transition matrix

$$\check{P} := \begin{bmatrix} \theta(1) & 1-\theta(1) \\ 1-\theta(2) & \theta(2) \end{bmatrix}. \tag{4}$$

Naturally the symmetry of P_θ above is lost and one can check that the invariant distribution of $\check{P}$ is

$$\check{\pi} = \left(\frac{1-\theta(2)}{2-\theta(1)-\theta(2)}, \quad \frac{1-\theta(1)}{2-\theta(1)-\theta(2)}\right) \neq \pi,$$

in general. For $\theta(1), \theta(2) \in \Theta$ the time homogeneous Markov chain will be ergodic, but will fail to converge to π as soon as $\theta(1) \neq \theta(2)$, that is as soon as there is dependence on the current state. As we shall see, the principle of vanishing adaptation consists of the present toy example of making both $\theta(1)$ and $\theta(2)$ time dependent (deterministically for simplicity here), denoted $\theta_i(1)$ and $\theta_i(2)$ at iteration i, and ensure that as $i \to \infty$, $|\theta_i(1) - \theta_i(2)|$ vanishes. Indeed, while $\{\theta_i(1)\}$ and $\{\theta_i(2)\}$ are allowed to evolve forever (and maybe not converge) the corresponding transition probabilities $\{\check{P}_i := P_{\theta_i(X_i)}\}$ have invariant distributions $\{\check{\pi}_i\}$ convergent to π. We might hence expect one to recover π-ergodicity. In fact in the present case standard theory for non-homogeneous Markov chains can be used in order to find conditions on $\{\theta_i\}$ that ensure ergodicity, but we do not pursue this in depth here.

It could be argued, and this is sometimes suggested, that the problem with the example above is that in order to preserve π as a marginal distribution, θ should not depend on X_i for the transition to X_{i+1}, but on $X_0, \ldots, X_{i-1}$ only. For simplicity assume that the dependence is on X_{i-1} only. Then it is sometimes argued that since

$$\begin{aligned}
&\begin{bmatrix} \pi(X_i = 1) \\ \pi(X_i = 2) \end{bmatrix}^{\mathrm{T}} \\
&\quad \times \begin{bmatrix} P_{\theta(X_{i-1})}(X_i = 1, X_{i+1} = 1) & P_{\theta(X_{i-1})}(X_i = 1, X_{i+1} = 2) \\ P_{\theta(X_{i-1})}(X_i = 2, X_{i+1} = 1) & P_{\theta(X_{i-1})}(X_i = 2, X_{i+1} = 2) \end{bmatrix} \\
&= \begin{bmatrix} \pi(X_i = 1) \\ \pi(X_i = 2) \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} \theta(X_{i-1}) & 1-\theta(X_{i-1}) \\ 1-\theta(X_{i-1}) & \theta(X_{i-1}) \end{bmatrix} \\
&= \big[\pi(X_{i+1} = 1),\ \pi(X_{i+1} = 2)\big],
\end{aligned}$$

then $X_{i+1}, X_{i+2}, \ldots$ are all marginally distributed according to π. Although this calculation is correct, the underlying reasoning is naturally incorrect in general. This can be checked in two ways. First through a counterexample which only requires elementary arguments. Indeed in the situation just outlined, the law of X_{i+1} given $\theta_0, X_0, \ldots, X_{i-1}, X_i$ is $P_{\theta(X_{i-1})}(X_i, X_{i+1} \in \cdot)$, from which we deduce that $Z_i = (Z_i(1), Z_i(2)) = (X_i, X_{i-1})$ is a time homogeneous Markov chain with transition

$$P_{\theta(Z_i(2))}(Z_i(1), Z_{i+1}(1))\, \mathbb{I}\{Z_{i+1}(2) = Z_i(1)\},$$

where for a set A, $\mathbb{I}A$ denotes its indicator function. Denoting the states $\bar{1} := (1,1)$, $\bar{2} := (1,2)$, $\bar{3} := (2,1)$ and $\bar{4} := (2,2)$, the transition matrix of the time homogeneous Markov chain is

$$\check{P} = \begin{bmatrix} \theta(1) & 0 & 1-\theta(1) & 0 \\ \theta(2) & 0 & 1-\theta(2) & 0 \\ 0 & 1-\theta(1) & 0 & \theta(1) \\ 0 & 1-\theta(2) & 0 & \theta(2) \end{bmatrix}$$

and it can be directly checked that the marginal invariant distribution of $Z_i(1)$ is

$$\begin{aligned}
\check{\pi} &= \left(2 + \frac{\theta(2)}{1-\theta(1)} + \frac{\theta(1)}{1-\theta(2)}\right)^{-1} \\
&\quad \times \begin{bmatrix} \frac{1+\theta(2)-\theta(1)}{1-\theta(1)} & \frac{1+\theta(1)-\theta(2)}{1-\theta(2)} \end{bmatrix} \neq (1/2,\ 1/2),
\end{aligned}$$

in general. The second and more informative approach consists of considering the actual distribution of the process generated by a controlled MCMC. Let us denote $\check{\mathbb{E}}_*$ the expectation for the process started at some arbitrary $\theta, x \in \Theta \times \mathsf{X}$. This operator is particularly useful to describe the expectation of $\psi(X_i, X_{i+1}, \ldots)$ for any $i \geq 1$ and any function $\psi : \mathsf{X}^{k_\psi} \to \mathbb{R}$, $\check{\mathbb{E}}_*(\psi(X_i, X_{i+1}, \ldots, X_{i+k_\psi-1}))$. More precisely it allows one to clearly express the dependence of $\theta_i(\theta_0, X_0, \ldots, X_i)$ on the past $\theta_0, X_0, \ldots, X_i$ of the process. Indeed for any $f : \mathsf{X} \to \mathbb{R}$, using the tower property of expectations and the definition of controlled MCMC given

in the introduction, we find that

$$\begin{aligned}\check{\mathbb{E}}_*\left(f(X_{i+1})\right) &= \check{\mathbb{E}}_*\left(\check{\mathbb{E}}_*\left(f(X_{i+1})|\theta_0, X_0, \ldots, X_i\right)\right)\\ &= \check{\mathbb{E}}_*\left(\int_{\mathsf{X}} P_{\theta_i(X_0,\ldots,X_i)}(X_i, dx) f(x)\right), \qquad (5)\end{aligned}$$

which is another way of saying that the distribution of X_{i+1} is that of a random variable sampled, conditional upon $\theta_0, X_0, \ldots, X_i$, according to the random transition $P_{\theta_i(X_0,\ldots,X_i)}(X_i, X_{i+1} \in \cdot)$, where the pair $\theta_i(\theta_0, X_0, \ldots, X_i)$, X_i is randomly drawn from a distribution completely determined by the possible histories $\theta_0, X_0, \ldots, X_i$. In the case where X is a finite discrete set, writing this relation concisely as the familiar product of a row vector and a transition matrix as above would require one to determine the (possibly very large) set of values for the pair $\theta_i(\theta_0, X_0, \ldots, X_i)$, X_i (say W_i), the vector representing the probability distribution of all these pairs as well as the transition matrix from W_i to X. The introduction of the expectation allows one to bypass these conceptual and notational difficulties. We will hereafter denote

$$\varphi(\theta_0, X_0, \ldots, X_i) := \int_{\mathsf{X}} P_{\theta_i(\theta_0,X_0,\ldots,X_i)}(X_i, dx) f(x),$$

and whenever possible will drop unnecessary arguments *i.e.* arguments of φ which do not affect its values.

The possibly complex dependence on $\theta_i(\theta_0, X_0, \ldots, X_i)$, X_i of the transition of the process to X_{i+1} needs to be contrasted with the case of standard MCMC algorithms. Indeed, in this situation the randomness of the transition probability only stems from X_i. This turns out to be a major advantage when it comes to invariant distributions. Let us assume that for some $i \geq 1$ $\check{\mathbb{E}}_*(g(X_i)) = \mathbb{E}_\pi(g(X))$ for all π-integrable functions g. Then according to the identity in (5), for any given $\theta \in \Theta$ and $\theta_i = \theta$ for all $i \geq 0$ a standard MCMC algorithm has the well known and fundamental property

$$\begin{aligned}\check{\mathbb{E}}_*(f(X_{i+1})) &= \check{\mathbb{E}}_*\left(\varphi(\theta, X_i)\right)\\ &= \mathbb{E}_\pi\left(\varphi(\theta, X)\right)\\ &= \int_{\mathsf{X}\times\mathsf{X}} \pi(dx) P_\theta(x, dy) f(y) = \mathbb{E}_\pi\left(f(X)\right),\end{aligned}$$

where the second equality stems from the assumption $\check{\mathbb{E}}_*(g(X_i)) = \mathbb{E}_\pi(g(X))$ and the last equality is obtained by the assumed invariance of π for P_θ for any $\theta \in \Theta$. Now we turn to the controlled MCMC process and focus for simplicity on the case $\theta_i(\theta_0, X_0, \ldots, X_i) = \theta(X_{i-1})$, corresponding to our counterexample. Assume that for some $i \geq 1$ X_i is marginally distributed according to π, *i.e.* for any $g : \mathsf{X} \to \mathbb{R}$, $\check{\mathbb{E}}_*(g(X_i)) = \mathbb{E}_\pi(g(X))$, then we would like to check if $\check{\mathbb{E}}_*(g(X_j)) = \mathbb{E}_\pi(g(X))$ for all $j \geq i$. However using the tower property of expectations in order to exploit the property $\check{\mathbb{E}}_*(g(X_i)) = \mathbb{E}_\pi(g(X))$,

$$\begin{aligned}\check{\mathbb{E}}_*(f(X_{i+1})) &= \check{\mathbb{E}}_*\left(\varphi(X_{i-1}, X_i)\right)\\ &= \check{\mathbb{E}}_*\left(\check{\mathbb{E}}_*\left(\varphi(X_{i-1}, X_i)|X_i\right)\right)\\ &= \mathbb{E}_\pi\left(\check{\mathbb{E}}_*\left(\varphi(X_{i-1}, X)|X\right)\right).\end{aligned}$$

Now it would be tempting to use the stationarity assumption in the last expression,

$$\begin{aligned}\mathbb{E}_\pi\left(\varphi(X_{i-1}, X)\right) &= \int_{\mathsf{X}\times\mathsf{X}} \pi(dx) P_{\theta(X_{i-1})}(x, dy) f(y)\\ &= \mathbb{E}_\pi(f(X)).\end{aligned}$$

This is however not possible due to the presence of the conditional expectation $\check{\mathbb{E}}_*(\cdot|X)$ (which crucially depends on X) and conclude that in general

$$\begin{aligned}&\check{\mathbb{E}}_*(f(X_{i+1}))\\ &\neq \check{\mathbb{E}}_*\left(\mathbb{E}_\pi\left(\int_{\mathsf{X}} P_{\theta(\theta_0,X_0,X_{i-1})}(X, dx_{i+1}) f(x_{i+1})\right)\right).\end{aligned}$$

The misconception that this inequality might be an equality is at the root of the incorrect reasoning outlined earlier. This problem naturally extends to more general situations.

Vanishing adaptation seems, intuitively, to offer the possibility to circumvent the problem of the loss of π as invariant distribution. However, as illustrated by the following toy example, vanishing adaptation might come with its own shortcomings. Consider a (deterministic) sequence $\{\theta_i\} \subset (-1, 1)^{\mathbb{N}}$ and for simplicity first consider the non-homogeneous, and non-adaptive, Markov chain $\{X_i\}$ with transition P_{θ_i} at iteration $i \geq 1$, where P_θ is given by (3), and initial distribution $(\mu, 1-\mu)$ for $\mu \in [0, 1]$. One can easily check that for any $n \geq 1$ the product of matrices $P_{\theta_1} \times \cdots \times P_{\theta_n}$ has the simple expression

$$\begin{aligned}&P_{\theta_1} \times \cdots \times P_{\theta_n}\\ &= \frac{1}{2}\begin{bmatrix} 1 + \prod_{i=1}^n (2\theta_i - 1) & 1 - \prod_{i=1}^n (2\theta_i - 1)\\ 1 - \prod_{i=1}^n (2\theta_i - 1) & 1 + \prod_{i=1}^n (2\theta_i - 1)\end{bmatrix}.\end{aligned}$$

As a result one deduces that the distribution of X_n is

$$\frac{1}{2}\left[1 + (2\mu - 1)\prod_{i=1}^n (2\theta_i - 1) \quad 1 - (2\mu - 1)\prod_{i=1}^n (2\theta_i - 1)\right].$$

Now if $\theta_i \to 0$ (resp. $\theta_i \to 1$) and $\sum_{i=1}^\infty \theta_i < +\infty$ (resp. $\sum_{i=1}^\infty (1 - \theta_i) < +\infty$), that is convergence to either 0 or 1 of $\{\theta_i\}$ is "too fast" , then $\lim_{n\to\infty} \prod_{i=1}^n (2\theta_i - 1) \neq 0$ and as a consequence, whenever $\mu \neq 1/2$, the distribution of X_n does not converge to $\pi = (1/2, 1/2)$. Similar developments are possible for the toy adaptive MCMC algorithm given by

the transition matrix $\check{P}$ in (4), at the expense of extra technical complications, and lead to the same conclusions. This toy example points to potential difficulties encountered by controlled MCMC algorithms that exploit vanishing adaptation: whereas π-ergodicity of P_θ is ensured for any $\theta \in \Theta$, this property might be lost if the sequence $\{\theta_i\}$ wanders towards "bad" values of θ for which convergence to equilibrium of the corresponding fixed parameter Markov chains P_θ might take an arbitrarily long time.

This point is detailed in the next section, but we first turn to a discussion concerning the possibility of using vanishing adaptation in order to circumvent the loss of invariance of π by controlled MCMC.

3 Vanishing adaptation and convergence

As suggested in the previous section, vanishing adaptation, that is ensuring that θ_i depends less and less on recently visited states of the chain $\{X_i\}$ might be a way of designing controlled MCMC algorithms which produce samples asymptotically distributed according to π. In this section we provide the basic arguments and principles that underpin the validity of controlled MCMC with vanishing adaptation. We however do not provide directly applicable technical conditions here that ensure the validity of such algorithms - more details can be found in Holden (1998), Atchadé and Rosenthal (2005), Andrieu and Moulines (2006), Roberts and Rosenthal (2006), Bai et al. (2008) and Atchadé and Fort (2008). The interest of dedicating some attention to this point here is twofold. First it provides useful guidelines as to what the desirable properties of a valid controlled MCMC algorithm should be, and hence help design efficient algorithms. Secondly it points to some difficulties with the existing theory which is not able to fully explain the observed stability properties of numerous controlled algorithms, a fact sometimes overlooked.

3.1 Principle of the analysis

Existing approaches to prove that ergodicity might be preserved under vanishing adaptation all rely on the same principle, which we detail in this section. The differences between the various existing contributions lies primarily in the assumptions, which are discussed in the text. With the notation introduced in the previous section, we are interested in the behaviour of the difference

$$|\check{\mathbb{E}}_*(f(X_i)) - \mathbb{E}_\pi(f(X))|$$

as $i \to \infty$ for any $f : \mathsf{X} \to \mathbb{R}$. Although general functions can be considered (Atchadé and Rosenthal 2005; Andrieu and Moulines 2006) and (Atchadé and Fort 2008), we will here assume for simplicity of exposition that $|f| \leq 1$. The study of this term is carried out by comparing the process of interest to a process which coincides with $\{X_k\}$ up to some time $k_i < i$ but becomes a time homogeneous Markov chain with "frozen" transition probability $P_{\theta_{k_i}}$ from this time instant onwards. (We hereafter use the following standard notation $P^k[f](x) = P^k f(x)$ for any $f : \mathsf{X} \to \mathbb{R}^{n_f}$ and $x \in \mathsf{X}$ defined recursively as $P^0 f(x) = f(x)$, $Pf(x) := \int_{\mathsf{X}} P(x, dy) f(y)$ and $P^{k+1} f(x) = P[P^k f](x)$ for $k \geq 1$. In the finite discrete case this corresponds to considering powers P^k of the transition matrix P and right multiplying with a vector f.) Denoting $P^{i-k_i}_{\theta_{k_i}} f(X_{k_i})$ the expectation of f after $i - k_i$ iterations of the "frozen" time homogeneous Markov transition probability $P_{\theta_{k_i}}$ initialised with X_{k_i} at time k_i and conditional upon $\theta_0, X_0, X_1, \ldots, X_{k_i}$, this translates into the fundamental decomposition

$$\begin{aligned}
&\check{\mathbb{E}}_*(f(X_i)) - \mathbb{E}_\pi(f(X)) \\
&\quad = \check{\mathbb{E}}_* \left(P^{i-k_i}_{\theta_{k_i}} f(X_{k_i}) - \pi(f) \right) \\
&\qquad + \check{\mathbb{E}}_* \left(f(X_i) - P^{i-k_i}_{\theta_{k_i}} f(X_{k_i}) \right), \qquad (6)
\end{aligned}$$

where the second term corresponds to the aforementioned comparison and the first term is a simple remainder term.

Perhaps not surprisingly the convergence to zero of the first term, provided that $i - k_i \to \infty$ as $i \to \infty$, depends on the ergodicity of the non-adaptive MCMC chain with fixed parameter $\theta \in \Theta$, *i.e.* requires at least that for any $\theta, x \in \Theta \times \mathsf{X}$, $\lim_{k\to\infty} |P^k_\theta f(x) - \pi(f)| = 0$. However since both θ_{k_i} and X_{k_i} are random and possibly time dependent, this type of simple convergence is not sufficient to ensure convergence of this term. One could suggest the following uniform convergence condition

$$\lim_{k\to\infty} \sup_{\theta, x \in \Theta\times\mathsf{X}} |P^k_\theta f(x) - \mathbb{E}_\pi(f(X))| = 0, \qquad (7)$$

which although mathematically convenient is unrealistic in most scenarios of interest. The first toy example of Sect. 2 provides us with such a simple counterexample. Indeed, at least intuitively, convergence of this Markov chain to equilibrium can be made arbitrarily slow for values of $\theta \in (0, 1)$ arbitrarily close to either 0 or 1. This negative property unfortunately carries on to more realistic scenarios. For example the normal symmetric random walk Metropolis algorithm described in Sect. 1 can in most situations of interest be made arbitrarily slow as the variance θ^2 is made arbitrarily small or large. This turns out to be a fundamental difficulty of the "chicken and egg" type in the study of the stability of such processes, which is sometimes overlooked. Indeed in order to ensure ergodicity, $\{\theta_i\}$ should stay away from poor values of the parameter $\theta \in \Theta$, but proving the stability of $\{\theta_i\}$ might often require establishing the ergodicity of the chain $\{X_i\}$; see Andrieu and Moulines (2006) and Andrieu and Tadić (2007) where alternative conditions

are also suggested. We will come back to this point after examining the second term of the decomposition above. Note that "locally uniform" such conditions (*i.e.* where Θ in (7) is replaced by some subsets $\mathcal{K} \subset \Theta$ and the rate of convergence might be slower) are however satisfied by many algorithms—this property is exploited in Andrieu and Moulines (2006), Andrieu and Tadić (2007), Atchadé and Fort (2008) and Bai et al. (2008) although this is not explicit in the latter.

The second term in the decomposition can be analysed by "interpolating" the true process and its Markovian approximation using the following telescoping sum

$$\begin{aligned}&\check{\mathbb{E}}_*\left(f(X_i)\right) - \check{\mathbb{E}}_*\left(P_{\theta_{k_i}}^{i-k_i} f(X_{k_i})\right)\\ &\quad= \sum_{j=k_i}^{i-1} \check{\mathbb{E}}_*\left(P_{\theta_{k_i}}^{i-j-1} f(X_{j+1})\right) - \check{\mathbb{E}}_*\left(P_{\theta_{k_i}}^{i-j} f(X_j)\right),\end{aligned}$$

which can be easily understood as follows. Each term of the sum is the difference of the expectations of (a) a process that adapts up to time $j+1 > k_i$ and then freezes and "becomes Markovian" with transition probability $P_{\theta_{k_i}}$ given the history $\theta_0, X_0, X_1, \ldots, X_{j+1}$ (and hence $\theta_{k_i} = \theta_{k_i}(\theta_0, X_0, X_1, \ldots, X_{k_i})$) between time $j+1$ and time i (b) and likewise for the second term, albeit between time j and i. Hence the two terms involved only differ in that at time j the first term updates the chain with θ_j while the second term uses θ_{k_i}, which can be concisely expressed as follows (thinking about the difference between two products of matrices in the discrete case might be helpful),

$$\begin{aligned}&\check{\mathbb{E}}_*\left(P_{\theta_{k_i}}^{i-j-1} f(X_{j+1})\right) - \check{\mathbb{E}}_*\left(P_{\theta_{k_i}}^{i-j} f(X_j)\right)\\ &\quad= \check{\mathbb{E}}_*\left(P_{\theta_j} P_{\theta_{k_i}}^{i-j-1} f(X_j)\right) - \check{\mathbb{E}}_*\left(P_{\theta_{k_i}}^{i-j} f(X_j)\right)\\ &\quad= \check{\mathbb{E}}_*\left(\left(P_{\theta_j} - P_{\theta_{k_i}}\right) P_{\theta_{k_i}}^{i-j-1} f(X_j)\right).\end{aligned}$$

The role of vanishing adaptation should now be apparent. Provided that the transition probability P_θ is sufficiently smooth in θ and that the variations of $\{\theta_i\}$ vanish as $i \to +\infty$ (in some unspecified sense at this point) then we might expect

$$\sum_{j=k_i}^{i-1} \check{\mathbb{E}}_*\left(\left(P_{\theta_j} - P_{\theta_{k_i}}\right) P_{\theta_{k_i}}^{i-j-1} f(X_j)\right)$$

to vanish if the number of terms in this sum does not grow too rapidly. However as noticed when analysing the first term of the fundamental decomposition above, simple continuity cannot be expected to be sufficient in general since $\theta_{k_i}, \theta_{k_i+1}, \ldots, \theta_{i-1}$ are random and time dependent. By analogy with the analysis above one could assume some form of uniform continuity in order to eliminate the variability of θ_j and θ_{k_i} in the expression above. More precisely, denoting for any $\delta > 0$

$$\Delta(\delta) := \sup_{|g|\leq 1} \sup_{x\in\mathsf{X},\{\theta,\theta'\in\Theta:|\theta-\theta'|\leq\delta\}} |P_\theta g(x) - P_{\theta'} g(x)|,$$

one could assume,

$$\lim_{\delta\to 0} \Delta(\delta) = 0.$$

Provided that the sequence $\{\theta_i\}$ is such that its increments are bounded *i.e.* such that $|\theta_i - \theta_{i-1}| \leq \gamma_i$ for a deterministic sequence $\{\gamma_i\} \in [0,\infty)^{\mathbb{N}}$ (which is possible since the updates of $\{\theta_i\}$ are chosen by the user) then the second term of (6) can be bounded by

$$\sum_{j=k_i}^{i-1} \sum_{k=k_i+1}^{j} \Delta_{\gamma_k},$$

which can usually be easily dealt with; *e.g.* when some uniform Lipschitz continuity is assumed and $\Delta(\delta) = C\delta$ for some constant $C > 0$ (Andrieu and Moulines 2006). Unfortunately, although mathematically convenient, this condition is not satisfied in numerous situations of interest, due in particular to the required uniformity in $\theta, \theta' \in \Theta$. Other apparently weaker conditions have been suggested, but share the same practical underlying difficulties such as

$$\lim_{i\to\infty} \sup_{|g|\leq 1} \check{\mathbb{E}}_*\left[|P_{\theta_i} g(X_i) - P_{\theta_{i-1}} g(X_i)|\right] = 0$$

suggested in Benveniste et al. (1990, p. 236) and the slightly stronger condition of the type

$$\lim_{i\to\infty} \sup_{x\in\mathsf{X},|g|\leq 1} \check{\mathbb{E}}_*\left[|P_{\theta_i} g(x) - P_{\theta_{i-1}} g(x)|\right] = 0,$$

in Roberts and Rosenthal (2006).

3.2 Discussion

The simple discussion above is interesting in two respects. On the one hand, using basic arguments, it points to the primary conditions under which one might expect controlled MCMC algorithms to be π-ergodic: "expected ergodicity and continuity of the transition probabilities". On the other hand it also points to the difficulty of proposing verifiable conditions to ensure that the aforementioned primary conditions are satisfied. The problem stems from the fact that it is required to prove that the algorithm is not unlucky enough to tune θ to poor values, leading to possibly unstable algorithms. The uniform conditions suggested earlier circumvent this problem, since they suggest that there are no such arbitrarily "bad" values. This is unfortunately not the case for

numerous algorithms of practical interest. However it is often the case that such uniformity holds for subsets $\mathcal{K} \subset \Theta$. For example in the case of the first toy example of Sect. 2 the sets defined as $\mathcal{K}_\epsilon := [\epsilon, 1-\epsilon]$ for any $\epsilon \in (0,1)$ are such that for any $\epsilon \in (0,1)$ the Markov chains with parameters $\theta \in \mathcal{K}_\epsilon$ are geometrically convergent with a rate of at least $|1-2\epsilon|$, independent of $\theta \in \mathcal{K}_\epsilon$. A simple practical solution thus consists of identifying such subsets $\mathcal{K}$ and constrain $\{\theta_i\}$ to these sets by design. This naturally requires some understanding of the problem at hand, which might be difficult in practice, and does not reflect the fact that stability is observed in practice without the need to resort to such "fixed" truncation strategies. A general approach for adaptive truncation is developed and analysed in Andrieu et al. (2005) and Andrieu and Moulines (2006). It takes advantage of the fact that uniform ergodicity and continuity can be shown for families of subsets of $\{\mathcal{K}_i \subset \Theta\}$, such that $\{\mathcal{K}_i \subset \Theta\}$ is a covering of Θ such that $\mathcal{K}_i \subset \mathcal{K}_{i+1}$ for $i \geq 0$. The strategy then consists of adapting the truncation of the algorithm on the fly in order to ensure that $\{\theta_i\}$ does not wander too fast towards inappropriate values in Θ or at its boundary, hence ensuring that ergodicity can kick in and stabilise the trajectories of $\{\theta_i\}$, *i.e.* ensure that there is a random, but finite, k such that $\{\theta_i\} \subset \mathcal{K}_k$ with probability 1. The procedure has the advantage of not requiring much knowledge about what constitutes good or bad values for θ, while allowing for the incorporation of prior information and ultimately ensuring stability. It can in fact be shown, under conditions satisfied by some classes of algorithms, that the number k of reprojections needed for stabilisation is a random variable with probability distribution whose tails decay superexponentially. While this approach is general and comes with a general and applicable theory, it might be computationally wasteful in some situations and more crucially does not reflect the fact that numerous algorithm naturally present stability properties. Another possibility consists of considering mixtures of adaptive and non-adaptive MCMC proposal distributions, the non adaptive components ensuring stability *e.g.* (Roberts and Rosenthal 2007): again while this type of strategy generally ensures that the theory works, it poses the problem of the practical choice of the non-adaptive component, and might not always result in efficient strategies. In addition, as for the strategies discussed earlier, this type of approach fails to explain the observed behaviour of some adaptive algorithms.

In a number of situations of interest it is possible to show that the parameter θ stays away from its forbidden values with probability one (Andrieu and Tadić 2007). The approach establishes a form of recurrence via the existence of composite drift functions for the joint chain $\{\theta_i, X_i\}$, which in turn ensures an infinite number of visits of $\{\theta_i\}$ to some sets $\mathcal{K}$ for which the uniform properties above hold. This can be shown to result in stability under fairly general conditions. In addition the approach provides one with some insight into what makes an algorithm stable or not, and suggests numerous ways of designing updates for $\{\theta_i\}$ which will ensure stability and sometimes even accelerate convergence. Some examples are discussed in Sect. 4.2.2. In Saksman and Vihola (2008) the authors address the same problem by proving that provided that $\{\theta_i\}$ is constrained not to drift too fast to bad values, then π-ergodicity of $\{X_i\}$ is preserved. The underlying ideas are related to a general strategy of stabilisation developed for the stochastic approximation procedure, see Andradóttir (1995).

Finally we end this section with a practical implication of the developments above, related to the rate of convergence of controlled MCMC algorithms. Assume for example the existence of $\mathcal{K} \subset \Theta$, $C \in (0,\infty)$, $\rho \in (0,1)$ and $\{\gamma_i\} \in [0,\infty)^{\mathbb{N}}$ such that for all $i \geq 1$, $\theta, x \in \mathcal{K} \times \mathsf{X}$ and any $f : \mathsf{X} \to [-1,1]$

$$|P_\theta^i f(x) - \mathbb{E}_\pi(f)| \leq C\rho^i, \tag{8}$$

and for any $\theta, \theta' \in \mathcal{K}$, $x \in \mathsf{X}$ and any $f : \mathsf{X} \to [-1,1]$,

$$|P_\theta f(x) - P_{\theta'} f(x)| \leq C|\theta - \theta'|, \tag{9}$$

and such that for all $i \geq 1$, $|\theta_i - \theta_{i-1}| \leq \gamma_i$, where $\{\gamma_i\}$ satisfies a realistic assumption of slow decay (Andrieu and Moulines 2006) (satisfied for example for $\gamma_i = 1/i^\alpha, \alpha > 0$). These conditions are far from restrictive and can be shown to hold for the symmetric random walk Metropolis (SRWM) for some distributions π, the independent Metropolis-Hastings (IMH) algorithm, mixtures of such transitions etc. (Andrieu and Moulines 2006). Less restrictive conditions are possible, but lead to slower rates of convergence. Then, using a more precise form of the decomposition in (6) (Andrieu and Moulines 2006, Proposition 4), one can show that there exists a constant $C' \in (0,\infty)$ such that for all $i \geq 1$ and $|f| \leq 1$,

$$\left|\check{\mathbb{E}}_*\left[(f(X_i) - \mathbb{E}_\pi(f))\,\mathbb{I}\{\sigma \geq i\}\right]\right| \leq C'\gamma_i, \tag{10}$$

where σ is the first time at which $\{\theta_i\}$ leaves $\mathcal{K}$ (which can be infinity). The result simply tells us that while the adapted parameter does not leave $\mathcal{K}$, convergence towards π occurs at a rate of at least $\{\gamma_i\}$, and as pointed out in Andrieu and Moulines (2006) does not require convergence of $\{\theta_i\}$. This might appear to be a negative result. However it can be proved, (Andrieu 2004) and (Andrieu and Moulines 2006), that there exist constants $A(\gamma,\mathcal{K})$ and $B(\gamma,\mathcal{K})$ such that for any $N \geq 1$,

$$\sqrt{\check{\mathbb{E}}_*\left[\left|\frac{1}{N}\sum_{i=1}^N f(X_i) - \mathbb{E}_\pi(f)\right|^2 \mathbb{I}\{\sigma \geq n\}\right]} \leq \frac{A(\gamma,\mathcal{K})}{\sqrt{N}} + B(\gamma,\mathcal{K})\frac{\sum_{k=1}^N \gamma_k}{N}. \tag{11}$$

The *first term* corresponds to the Monte Carlo fluctuations while the *second term* is the price to pay for adaptation. Assuming that $\gamma_i = i^{-\alpha}$ for $\alpha \in (0,1)$, then

$$\frac{\sum_{k=1}^{N} \gamma_k}{N} \sim \frac{1}{1-\alpha} N^{-\alpha},$$

which suggests no loss in terms of rate of convergence for $\alpha \geq 1/2$. More general and precise results can be found in Andrieu and Moulines (2006, Proposition 6), including a central limit theorem (Theorem 9) which shows the asymptotic optimality of adaptive MCMC algorithms when convergence of $\{\theta_i\}$ is ensured . Weaker rates of convergence than (8) lead to a significant loss of rate of convergence, which is also observed in practice.

4 Vanishing adaptation: a framework for consistent adaptive MCMC algorithms

In the previous section we have given arguments that suggest that vanishing adaptation for MCMC algorithms might lead to algorithms from which expectations with respect to a distribution of interest π can be consistently estimated. However neither criteria nor ways of updating the parameter θ were described. The main aim of this section is to point out the central role played by stochastic approximation and the Robbins-Monro recursion (Robbins and Monro 1951) in the context of vanishing or non-vanishing adaptation. While a complete treatment of the theoretical aspects of such controlled MCMC algorithms is far beyond the scope of this review, our main goal is to describe the principles underpinning this approach that have a practical impact and to show the intricate link between criteria and algorithms. Indeed, as we shall see, while the stochastic approximation framework can be used in order to optimise a given criterion, it can also help understand the expected behaviour of an updating algorithm proposed without resorting to grand theory, but by simply resorting to common sense.

4.1 Criteria to optimise MCMC algorithms and a general form

Since our main aim is that of optimising MCMC transition probabilities, the first step towards the implementation of such a procedure naturally consists of defining what is meant by optimality, or suboptimality. This can be achieved through the definition of a cost function, which could for example express some measure of the statistical performance of the Markov chain in its stationary regime *e.g.* favour negative correlation between X_i and X_{i+l} for some lag l and $i = 0, 1, \ldots$. In what follows we will use the convention that an optimum value θ_* corresponds to a root of the equation $h(\theta) = 0$ for some function $h(\theta)$ closely related to the aforementioned cost function.

Since the main use of MCMC algorithms is to compute averages of the form $\hat{I}_N^\theta(f)$ given in (1) in order to estimate $\mathbb{E}_\pi(f(X))$, in situations where a central limit theorem holds, *i.e.* in scenarios such that for any $\theta \in \Theta$

$$\sqrt{N}\,(\hat{I}_N^\theta(f) - \mathbb{E}_\pi(f(X))) \to_{\mathcal{D}} \mathcal{N}(0, \sigma_\theta^2(f)),$$

it might seem natural to attempt to optimise the constant $\sigma_\theta^2(f)$. This however poses several problems. The first problem is computational. Indeed, for a given $\theta \in \Theta$ and $f : \mathsf{X} :\to [-1,1]$ (for simplicity) and when it exists, $\sigma_\theta^2(f)$ can be shown to have the following expression

$$\begin{aligned}\sigma_\theta^2(f) &= \mathbb{E}_\pi(\bar{f}^2(X_0)) + 2\sum_{k=1}^{+\infty} \mathbb{E}^\theta(\bar{f}(X_0)\bar{f}(X_k)) \\ &= \mathbb{E}_\pi(\bar{f}^2(X_0)) + 2\mathbb{E}^\theta\left(\sum_{k=1}^{+\infty} \bar{f}(X_0)\bar{f}(X_k)\right), \end{aligned} \tag{12}$$

with $\bar{f}(x) := f(x) - \mathbb{E}_\pi(f(X))$ and $\mathbb{E}^\theta$ the expectation associated to the Markov chain with transition probability P_θ and such that $X_0 \sim \pi$. This quantity is difficult to estimate and optimise (since for all $\theta \in \Theta$ it is the expectation of a non-trivial function with respect to an infinite set of random variables) although some solutions exist (Vladislav Tadić, personal communication, see also Richard Everitt's Ph.D. thesis) and truncation of the infinite sum is also possible (Andrieu and Robert 2001; Pasarica and Gelman 2003), allowing for example for the recursive estimation of the gradient of $\sigma_\theta^2(f)$ with respect to θ. In Pasarica and Gelman (2003), maximising the expected mean square jump distance is suggested, *i.e.* here in the scalar case and with $\bar{X}_i = X_i - \mathbb{E}_\pi(X)$ for $i = 0, 1$,

$$\begin{aligned}\mathbb{E}^\theta\left((X_0 - X_1)^2\right) &= \mathbb{E}^\theta\left(\left(\bar{X}_0 - \bar{X}_1\right)^2\right) \\ &= 2\left(\mathbb{E}_\pi\left(\bar{X}^2\right) - \mathbb{E}^\theta\left(\bar{X}_0\bar{X}_1\right)\right)\end{aligned} \tag{13}$$

which amounts to minimising the term corresponding to $k = 1$ in (12) for the function $f(x) = x$ for all $x \in \mathsf{X}$. Another difficulty is that the criterion depends on a specific function f, and optimality for a function f might not result in optimality for another function g. Finally it can be argued that although optimising this quantity is an asymptotically desirable criterion, at least for a given function, this criterion can in some scenarios lead to MCMC samplers that are slow to reach equilibrium (Besag and Green 1993).

Despite the difficulties pointed out earlier, the criterion above should not be totally discarded, but instead of trying to optimise it directly and perfectly, suboptimal optimisation through proxies that are amenable to simple computation and efficient estimation might be preferable.

Such a simple criterion, which is at least completely supported by theory in some scenarios (Roberts et al. 1997; Sherlock and Roberts 2006; Roberts and Rosenthal 1998; Bédard 2006) and proves to be more universal in practice, is the expected acceptance probability of the MH algorithm for random walk Metropolis algorithms or Langevin based MH updates. The expected acceptance probability is more formally defined as the jump rate of a MH update in the stationary regime

$$\bar{\alpha}_\theta := \int_{\mathsf{X}^2} \min\left\{1, \frac{\pi(y)\, q_\theta(y,x)}{\pi(x)\, q_\theta(x,y)}\right\} \pi(x)\, q_\theta(x,y)\, dxdy$$
$$= \mathbb{E}_{\pi\otimes q_\theta}\left(\min\left\{1, \frac{\pi(Y)\, q_\theta(Y,X)}{\pi(X)\, q_\theta(X,Y)}\right\}\right). \tag{14}$$

This criterion has several advantages. The first one is computational, since it is much simpler an expectation of a much simpler function than $\sigma_\theta^2(f)$. In such cases it has the double advantage of being independent of any function f and to provide a good compromise for $\sigma_\theta^2(f)$ for all functions f. A less obvious advantage of this criterion, which we illustrate later on in Sect. 5, is that where some form of smoothness of the target density is present it can be beneficial in the initial stages of the algorithm in order to ensure that the adaptive algorithm actually starts exploring the target distribution in order to "learn" some of its features.

The aforementioned theoretical results tell us that optimality of $\sigma_\theta^2(f)$ (in terms of θ) or proxy quantities related to this quantity (truncation, asymptotics in the dimension) is reached for a specific value of the expected acceptance probability $\bar{\alpha}_\theta$, denoted α^* hereafter: 0.234 for the random walk Metropolis algorithm for some specific target distributions and likewise 0.574 for Langevin diffusion based MH updates (Roberts and Rosenthal 1998).

In some situations, Gelman et al. (1995) have shown that the "optimal" covariance matrix for a multivariate random walk Metropolis algorithm with proposal $\mathcal{N}(0,\Sigma)$ is $\Sigma := (2.38^2/n_x)\Sigma_\pi$, where Σ_π is the covariance matrix of the target distribution π

$$\Sigma_\pi = \mathbb{E}_\pi\left(XX^{\mathrm{T}}\right) - \mathbb{E}_\pi(X)\,\mathbb{E}_\pi^{\mathrm{T}}(X).$$

The covariance is unknown in general situations and requires the numerical computation of the pair

$$\left(\mathbb{E}_\pi(X), \mathbb{E}_\pi\left(XX^{\mathrm{T}}\right)\right) = \mathbb{E}_\pi\left((X,\ XX^{\mathrm{T}})\right). \tag{15}$$

As pointed out in Andrieu and Moulines (2006, Sect. 7), this can also be interpreted as minimising the Kullback-Leibler divergence

$$\int_{\mathsf{X}} \pi(x) \log\frac{\pi(x)}{\mathcal{N}(x;\mu,\Sigma)}dx = \mathbb{E}_\pi\left(\log\frac{\pi(X)}{\mathcal{N}(X;\mu,\Sigma)}\right),$$

which suggests generalisations consisting of minimising

$$\mathbb{E}_\pi\left(\log\frac{\pi(X)}{q_\theta(X)}\right), \tag{16}$$

in general, for some parametric family of probability distributions $\{q_\theta, \theta\in\Theta\}$. Section 7 of Andrieu and Moulines (2006) is dedicated to the development or an on-line EM algorithm and a theoretical analysis of an adaptive independent MH algorithm where q_θ is a general mixture of distributions belonging to the exponential family. We will come back to this strategy in Sect. 5.2.2 where we show that this procedure can also be used in order to cluster the state-space X and hence define locally adaptive algorithms.

Before turning to ways of optimising criteria of the type described above, we first detail a fundamental fact shared by all the criteria described above and others, which will allow us to describe a general procedure for the control of MCMC algorithms. The shared characteristic is naturally that all the criteria developed here take the form of an expectation with respect to some probability distribution dependent on $\theta\in\Theta$. In fact as we shall see optimality can often be formulated as the problem of finding the root(s) of an equation of the type

$$h(\theta) := \mathbb{E}^\theta\left(H(\theta, X_0, Y_1, X_1, \ldots)\right) = 0 \tag{17}$$

(remember that $\{Y_i\}$ is the sequence of proposed samples) for some function $\Theta\times\mathsf{X}^{\mathbb{N}} :\to \mathbb{R}^{n_h}$ for some $n_h\in\mathbb{N}$, with in many situations $n_h = n_\theta$ (but not always). The case of the coerced acceptance probability corresponds to

$$H(\theta, X_0, Y_1, X_1, \ldots) = \min\left\{1, \frac{\pi(Y_1)\, q_\theta(Y_1, X_0)}{\pi(X_0)\, q_\theta(X_0, Y_1)}\right\} - \alpha^*,$$

which according to (14) results in the problem of finding the zero(s) of $h(\theta) = \bar{\alpha}_\theta - \alpha^*$. The moment matching situation corresponds to

$$H(\theta, X) = (X, XX^{\mathrm{T}}) - (\mu, \Sigma)$$

for which it is sought to find the zeros of $h(\theta) = (\mu_\pi, \Sigma_\pi) - (\mu, \Sigma)$ *i.e.* simply (μ_π, Σ_π) (naturally assuming that the two quantities exist). It might not be clear at this point how optimising the remaining criteria above might amount to finding the zeros of a function of the form (17). However, under smoothness assumptions, it is possible to consider the gradients of those criteria (note however that one might consider other methods than gradient based approaches in order to perform optimisation). In the case of the Kullback-Leibler divergence, and assuming that differentiation and integration can be swapped, the criterion can be expressed as

$$\mathbb{E}_\pi\left(\nabla_\theta \log\frac{\pi(X)}{q_\theta(X)}\right) = 0 \tag{18}$$

that is

$$H(\theta, X) = \nabla_\theta \log \frac{\pi(X)}{q_\theta(X)}$$

and in the more subtle case of the first order autocovariance minimisation one can invoke a standard score function argument and find the zeros of (in the scalar case for simplicity)

$$\nabla_\theta \mathbb{E}^\theta \left(\bar{X}_0 \bar{X}_1\right) = \mathbb{E}_\theta \left(\frac{\nabla_\theta P_\theta(X_0, X_1)}{P_\theta(X_0, X_1)} X_0 X_1 \right) = 0.$$

Similarly, under smoothness assumptions, one can differentiate $\sigma_\theta^2(f)$ and obtain a theoretical expression for $\nabla_\theta \sigma_\theta^2(f)$ of the form (17). Note that when q_θ is a mixture of distribution belonging to the exponential family, then it is possible to find the zeros (assumed here to exist) of (18) using an on-line EM algorithm (Andrieu and Moulines 2006).

Note that all the criteria described above are "steady state" criteria and explicitly involve π, but that other criteria such as the minimisation of return times to a given set $C \subset \mathsf{X}$ (Andrieu and Doucet 2003), namely

$$\tau = \check{\mathbb{E}}_\lambda^\theta \left[\sum_{i=1}^{\infty} \mathbb{I}\{X_i \notin C\} \right]$$

with λ a probability measure concentrated on C, do not enter this category. Such criteria seem however difficult to optimise in practice and we do not pursue this.

4.2 The stochastic approximation framework

We dedicate here a section to the Robbins-Monro update, which although not the only possibility to optimise criteria of the type (17) appears naturally in most known adaptive algorithms and provides us with a nice framework naturally connected to the literature on controlled Markov chains in the engineering literature. The reason for its ubiquity stems from the trivial identity: $\theta_{i+1} = \theta_i + \theta_{i+1} - \theta_i$. This turns out to be a particularly fruitful point of view in the present context. More precisely, it is well suited to sequential updating of $\{\theta_i\}$ and makes explicit the central role played by the updating rule defining the increments $\{\theta_{i+1} - \theta_i\}$. In light of our earlier discussion $\{\theta_{i+1} - \theta_i\}$ should be vanishing, and when convergence is of interest their cumulative sums should also vanish (in some probabilistic sense) in the vicinity of optimal values θ^*. Naturally, although convenient, this general framework should not prevent us from thinking "outside of the box".

4.2.1 Motivating example

Consider the case where $\mathsf{X} = \mathbb{R}$ and a symmetric random walk Metropolis (SRWM) algorithm with normal increment distribution $\mathcal{N}(z; 0, \exp(\theta))$, resulting in a transition probability P_θ^{NSRW}. We know that in some situations (Roberts et al. 1997) the expected acceptance probability should be in a range close to $\bar{\alpha}^* = 0.44$. We will assume for simplicity that $\bar{\alpha}_\theta$ in (14) is a non-increasing function of θ (which is often observed to be true, but difficult to check rigourously in practice and can furthermore be shown not to hold in some situations, Hastie 2005). In such situations one can suggest the following intuitive algorithm. For an estimate $\theta_i \in \Theta$ obtained after $i \times L$ iterations of the controlled MCMC algorithm, one can simulate L iterations of the transition probability $P_{\theta_i}^{NSRW}$ and estimate the expected acceptance probability for such a value of the parameter for the i-th block of samples $\{X_{iL+1}, Y_{iL+1}, \ldots, X_{iL+L}, Y_{iL+L}, k = 1, \ldots, L\}$ (initialised with X_i)

$$\hat{\alpha}_{\theta_i} = \frac{1}{L} \sum\nolimits_{k=1}^{L} \min \left\{ 1, \frac{\pi(Y_{iL+k})}{\pi(X_{iL+k-1})} \right\}$$

and update θ_i according to the following rule, motivated by our monotonicity assumption on $\bar{\alpha}_\theta$: if $\hat{\alpha}_{\theta_i} > \bar{\alpha}^*$ then θ_i is probably ($\hat{\alpha}_{\theta_i}$ is only an estimator) too small and should be increased while if $\hat{\alpha}_{\theta_i} < \bar{\alpha}^*$ then θ_i should be decreased. There is some flexibility concerning the amount by which θ_i should be altered and depends either on the criterion one wishes to optimise or more heuristic considerations. However, as detailed later, this choice will have a direct influence on the criterion effectively optimised and in light of the discussion of Sect. 3 concerning diminishing adaptation, this amount of change should diminish as $i \to \infty$ in order to either ensure that π-ergodicity of $\{X_i\}$ is ensured or that "approximate convergence" of $\{\theta_i\}$ is ensured. The intuitive description given above can suggest the following updating rules (see also Gilks et al. 1998, Andrieu and Robert 2001, Atchadé and Rosenthal 2005 for similar rules)

$$\theta_{i+1} = \theta_i + \gamma_{i+1} \left(\mathbb{I}\left\{\hat{\alpha}_{\theta_i} - \bar{\alpha}^* > 0\right\} - \mathbb{I}\left\{\hat{\alpha}_{\theta_i} - \bar{\alpha}^* \leq 0\right\} \right) \tag{19}$$

or

$$\theta_{i+1} = \theta_i + \gamma_{i+1} \left(\hat{\alpha}_{\theta_i} - \bar{\alpha}^* \right), \tag{20}$$

where $\{\gamma_i\} \subset (0, +\infty)^{\mathbb{N}}$ is a sequence of possibly stochastic stepsizes which ensures that the variations of $\{\theta_i\}$ vanish. The standard approach consists of choosing the sequence $\{\gamma_i\}$ deterministic and non-increasing, but it is also possible to choose $\{\gamma_i\}$ random *e.g.* such that it takes values in $\{\delta, 0\}$ for some $\delta > 0$ and such that $\mathbb{P}(\gamma_i = \delta) = p_i$ where $\{p_i\} \subset [0, 1]^{\mathbb{N}}$ is a deterministic and non-increasing sequence (Roberts and Rosenthal 2007), although it is not always clear what the advantage of introducing such an additional level of randomness is. A more interesting choice in practice consists of choosing $\{\gamma_i\}$ adaptively, see Sect. 4.2.2,

but for simplicity of exposition we focus here on the deterministic case.

We will come back to the first updating rule later on, and now discuss the second rule which as we shall see aims to set (14) equal to $\bar{\alpha}^*$. Notice first that if $L \to \infty$ and the underlying Markov chain is ergodic, then $\hat{\alpha}_\theta \to \bar{\alpha}_\theta$ and the recursion becomes deterministic

$$\theta_{i+1} = \theta_i + \gamma_{i+1} \left(\bar{\alpha}_{\theta_i} - \bar{\alpha}^* \right) \tag{21}$$

and is akin to a standard gradient algorithm, which will converge under standard conditions. Motivated by this asymptotic result, one can rewrite the finite L recursion (20) as follows

$$\theta_{i+1} = \theta_i + \gamma_{i+1} \left(\bar{\alpha}_{\theta_i} - \bar{\alpha}^* \right) + \gamma_{i+1} \left(\hat{\alpha}_{\theta_i} - \bar{\alpha}_{\theta_i} \right). \tag{22}$$

Assuming for simplicity that there exists $\theta^* \in \overset{\circ}{\Theta}$, the interior of Θ, such that $\bar{\alpha}_{\theta^*} = \bar{\alpha}^*$ and that $\hat{\alpha}_{\theta_i}$ is unbiased, at least as $i \to \infty$. Then, since $|\theta_{i+1} - \theta_i| \leq \gamma_{i+1} \to 0$ as $i \to \infty$, and provided that $\hat{\alpha}_\theta - \bar{\alpha}_\theta$ is smooth in terms of $\theta \in \Theta$ the sequence of noise terms $\{\hat{\alpha}_{\theta_i} - \bar{\alpha}_{\theta_i}\}$ is expected to average out to zero (*i.e.* statistically, positive increments are compensated by negative increments) and we expect the trajectory of (22) to oscillate about the trajectory of (21), with the oscillations vanishing as $i \to \infty$. This is the main idea at the core of the systematic analysis of such recursions which, as illustrated below, has an interest even for practitioners. Indeed, by identifying the underlying deterministic recursion which is approximated in practice, it allows one to understand and predict the behaviour of algorithms, even in situations where the recursion is heuristically designed and the underlying criterion not explicit. Equation (20) suggests that stationary points of the recursion should be such that $\bar{\alpha}_{\theta^*} = \bar{\alpha}^*$. The stationary points of the alternative recursion (19) are given in the next subsection.

In general most of the recursions of interest can be recast as follows,

$$\theta_{i+1} = \theta_i + \gamma_{i+1} H_{i+1} \left(\theta_i, X_0, \ldots, Y_i, X_i, Y_{i+1}, X_{i+1} \right) \tag{23}$$

where $H_{i+1}(\theta, X_0, \ldots, Y_i, X_i, Y_{i+1}, X_{i+1})$ takes its values in Θ. Typically in practice $\{H_{i+1}\}$ is a time invariant sequence of mappings which in effect only depends on a fixed and finite number of arguments through time invariant subsets of $\{Y_i, X_i\}$ (*e.g.* the last L of them at iteration i, as above). For simplicity we will denote this mapping H and include all the variables $\theta_i, X_0, \ldots, Y_i, X_i, Y_{i+1}, X_{i+1}$ as an argument, although the dependence will effectively be on a subgroup. Considering sequences $\{H_i(\theta, X_0, \ldots, Y_i, X_i, Y_{i+1}, X_{i+1})\}$ with a varying numbers of arguments is possible (and needed when trying to optimise (12) directly), but at the expense of additional notation and assumptions.

4.2.2 Why bother with stochastic approximation?

In this subsection we point to numerous reasons why the standard framework of stochastic approximation can be useful in order to think about controlled MCMC algorithms: as we shall see motivations range from theoretical to practical or implementational, and might help shed some lights on possibly heuristically developed strategies. Again, although this framework is very useful and allows for a systematic approach to the development and understanding of controlled MCMC algorithms, and despite the fact that this framework encompasses most known procedures, it should however not prevent us from thinking differently.

A standardized framework for programming and analysis Apart from the fact that the standard form (23) allows for systematic ways of coding the recursions, in particular the creation of "objects", the approach allows for an understanding of the expected behaviour of the recursion using simple mathematical arguments as well as the development of a wealth of very useful variations, made possible by the understanding of the fundamental underlying nature of the recursions. As suggested above with a simple example (21)–(22) the recursion (23) can always be rewritten as

$$\theta_{i+1} = \theta_i + \gamma_{i+1} h \left(\theta_i \right) + \gamma_{i+1} \xi_{i+1}, \tag{24}$$

where $h(\theta)$ is the expectation in steady state for a fixed $\theta \in \Theta$ of $H(\theta, X_0, \ldots, Y_i, X_i, Y_{i+1}, X_{i+1})$, *i.e.*

$$h(\theta) := \mathbb{E}^\theta \left(H \left(\theta, X_0, \ldots, Y_i, X_i, Y_{i+1}, X_{i+1} \right) \right)$$

and $\xi_{i+1} := H(\theta_i, X_0, \ldots, Y_i, X_i, Y_{i+1}, X_{i+1}) - h(\theta_i)$ is usually referred to as the "noise". The recursion (24) can therefore be thought of as being a noisy gradient algorithm. Intuitively, if we rearrange the terms in (24)

$$\frac{\theta_{i+1} - \theta_i}{\gamma_{i+1}} = h \left(\theta_i \right) + \xi_{i+1},$$

we understand that provided that the noise increments ξ_i "cancel out on average", then a properly rescaled continuous interpolation of the recursion $\theta_0, \theta_1, \ldots$ should behave more or less like the solutions $\theta(t)$ of the ordinary differential equation

$$\dot{\theta} \left(t \right) = h \left(\theta \left(t \right) \right), \tag{25}$$

whose stationary points are precisely such that $h(\theta) = 0$. The general theory of stochastic approximation consists of establishing that the stationary points of (24) are related to the stationary points of (25) and that convergence occurs provided that some conditions concerning $\{\gamma_i\}, h(\theta)$ and $\{\xi_i\}$ are satisfied. While this general theory is rather involved, it nevertheless provides us with a useful recipe to

try to predict and understand some heuristically developed algorithms. For example it is not clear what criterion is actually optimised when using the updating rule (19). However the "mean field" approach described above can be used to compute

$$\begin{aligned} h(\theta) &= \mathbb{E}^{\theta}\left(H\left(\theta, X_0, \ldots, Y_i, X_i, Y_{i+1}, X_{i+1}\right)\right) \\ &= \mathbb{E}^{\theta}\left(\mathbb{I}\left\{\hat{\alpha}_\theta - \bar{\alpha}^* > 0\right\} - \mathbb{I}\left\{\hat{\alpha}_\theta - \bar{\alpha}^* \leq 0\right\}\right) \\ &= \mathbb{P}^{\theta}\left(\hat{\alpha}_\theta - \bar{\alpha}^* > 0\right) - \mathbb{P}^{\theta}\left(\hat{\alpha}_\theta - \bar{\alpha}^* \leq 0\right). \end{aligned}$$

Its zeros (the possible stationary points of the recursion) are such that $\mathbb{P}^{\theta}(\hat{\alpha}_\theta - \bar{\alpha}^* > 0) = \mathbb{P}^{\theta}(\hat{\alpha}_\theta - \bar{\alpha}^* \leq 0) = 1/2$, *i.e.* the stationary points θ^* are such that $\bar{\alpha}^*$ is the median of the distribution of $\hat{\alpha}_\theta$ in steady-state, which seems reasonable when this median is not too different from $\bar{\alpha}_{\theta^*}$ given our initial objective. In addition this straightforward analysis also tells us that the algorithm will have the desired gradient like behaviour when $\mathbb{P}^{\theta}(\hat{\alpha}_\theta - \bar{\alpha}^* > 0)$ is a non-increasing function of θ. Other examples of the usefulness of the framework to design and understand such recursions are given later in Sect. 5, in particular Sect. 5.2.2.

In addition to allowing for an easy characterisation of possible stationary points of the recursion (and hence of the "ideal" optimal values θ_*) the decomposition (24) points to the role played by the deterministic quantity $h(\theta)$ to ensure that the sequence $\{\theta_i\}$ actually drifts towards optimal values θ_*, which is the least one can ask from such a recursion, and the fact that the noise sequence $\{\xi_i\}$ should also average out to zero for convergence purposes. This latter point is in general very much related to the ergodicity properties of $\{X_i\}$, which justifies the study of ergodicity even in situations where it is only planned to use the optimised MCMC algorithm with a fixed and suboptimal parameter $\tilde{\theta}$ obtained after optimisation. This in turn points to the intrinsic difficulty of ensuring and proving such ergodicity properties before $\{\theta_i\}$ wanders towards "bad" values, as explained in Sect. 2. Recent progress in Andrieu and Tadić (2007), relying on precise estimates of the dependence in θ of standard drift functions for the analysis of Markov chains allows one to establish that $\{\theta_i\}$ stays away from such "bad" values, ensuring in turn ergodicity and a drift of $\{\theta_i\}$ towards the set of values of interest θ^*. Similar results are obtained in Saksman and Vihola (2008), albeit using totally different techniques.

Finally note that the developments above stay valid in the situation where $\{\gamma_i\}$ is set to a constant, say γ. In such situations it is possible to study the distribution of θ_i around a deterministic trajectory underlying the ordinary differential equation, but it should be pointed out that in such situations $\{X_i\}$ is not π-stationary, and one can at most hope for π_γ-stationarity for a probability distribution π_γ such that $\pi_\gamma \to \pi$ in a certain sense as $\gamma \to 0$.

The connection between stochastic approximation and the work of Haario et al. (2001) and the underlying generality was realised in Andrieu and Robert (2001), although it is mentioned in particular cases in Geyer and Thompson (1995) and Ramponi (1998), the latter reference being probably the first rigourous analysis of the stability and convergence properties of a particular implementation of controlled MCMC for tempering type algorithms.

A principled stopping rule As pointed out earlier, and although ergodicity is intrinsically related to the sequence $\{\theta_i\}$ approaching the zeroes of $h(\theta)$ and hence taking "good values", one might be more confident in using samples produced by a standard MCMC algorithm that would use an optimal or suboptimal value of θ. This naturally raises the question of the stopping rule to be used. In the ubiquitous case of the Robbins-Monro updating rule, and given the clear interpretation in terms of the root finding of $h(\theta)$, one can suggest monitoring the average of the field

$$\frac{1}{n}\sum_{i=1}^{n} H(\theta_i, X_{i+1})$$

and stop, for example, when its magnitude is less than a preset threshold ϵ for a number m of consecutive iterations. More principled statistical rules relying on the CLT can also be suggested, but we do not expand on this here.

Boundedness and convergence The dependence of the ergodicity properties of P_θ can lead to some difficulties in practice. Indeed these ergodicity properties are rarely uniform in $\theta \in \Theta$ and tend to degrade substantially for some values, typically on the boundary $\partial\Theta$ of Θ. For example for the toy example of Sect. 2, both values $\partial\Theta = \{0, 1\}$ are problematic. For $\theta = 0$ aperiodicity is lost whereas for $\theta = 1$ irreducibility is lost. This can result in important problems in practice since π-ergodicity can be lost as pointed out in Sect. 2 through the aforementioned toy example when the sequence $\{\theta_i\}$ converges to $\partial\Theta$ too quickly. In fact, as pointed out to us by Y.F. Atchadé, an example in Winkler (2003) shows that even in the situation where $\theta_i(1) = \theta_i(2) = 1 - 1/i$, the sequence $\{n^{-1}\sum_{i=1}^{n} X_i - 3/2\}$ does not vanish (in the mean square sense) as $i \to \infty$. This problem of possible loss of ergodicity of P_θ and its implications for controlled Markov chains has long been identified, but is often ignored in the current MCMC related literature. For example a normal symmetric random walk Metropolis (N-SRWM) algorithm loses ergodicity as its variance (or covariance matrix) becomes either too large or too small and an algorithm with poor ergodicity properties does not learn features of the target distribution π. In the case of a random scan MH within Gibbs algorithm as given in (2), it is possible to progressively lose irreducibility whenever a weight drifts towards 0. Several cures are possible. The first and obvious one consists of truncating Θ in order to ensure the existence of some uniform ergodicity properties of the family

of transitions $\{P_\theta\}$. While this presumes that one knows by how much one can truncate Θ without affecting the ergodicity properties of $\{P_\theta\}$ significantly, this is not a completely satisfactory solution since stability is actually observed in numerous situations.

In Andrieu and Tadić (2007), using explicit dependence of the parameters of well known drift conditions for MCMC algorithms on the tuning parameter θ, general conditions on the transition probability P_θ and the updating function $H(\theta, x)$ that ensure boundedness of $\{\theta_i\}$ are derived. As a result π-ergodicity of $\{X_i\}$. and convergence to optimal or suboptimal values of θ are automatically satisfied without the need ro resort to fixed or adaptive truncations for example One aspect of interest of the results is that they suggest some ways of designing fully adaptive and stable algorithms.

For example by noting that the zeroes of $h(\theta)$ are also the zeroes of $h(\theta)/(1+|\theta|^\alpha)$ for example, one can modify the standard recursion in order to stabilise the update, resulting in the alternative updating rule

$$\theta_{i+1} = \theta_i + \gamma_{i+1} H(\theta_i, X_{i+1})/(1+|\theta_i|^\alpha).$$

One can also add regularisation terms to the recursion. For example, assuming for example that we learn optimal weights for a mixture of transition probabilities as in (2), the recursion

$$w_{i+1}^k = w_i^k + \gamma_{i+1} H_k(w_i, X_{i+1})$$

(with $w_i = (w_i^1, w_i^2, \ldots, w_i^n)$) can be for example modified to

$$w_{i+1}^k = w_i^k + \gamma_{i+1} H_k(w_i, X_{i+1}) + \gamma_{i+1}^{1+\lambda} \left(\frac{\alpha + (w_i^k)^{-\beta}}{\sum_{j=1}^n \alpha + (w_i^j)^{-\beta}} - w_i^k \right)$$

for some $\alpha, \beta, \lambda > 0$. Note that since the sum over k of the fields is 0, the weights still sum to 1 after the update and also that due to the boundedness of the additional term it vanishes as $i \to \infty$. Finally in Andrieu et al. (2005) and Andrieu and Moulines (2006), following Chen et al. (1988), an algorithm with adaptive truncation boundaries is suggested and a general theory developed that ensures that both boundedness and convergence of $\{\theta_i\}$ is ensured. Although requiring an intricate theory, the conditions under which boundedness and convergence hold cover a vast number of situations, beyond the situations treated in Andrieu and Tadić (2007). In Saksman and Vihola (2008) a different approach to prove stability is used, and consists of proving that provided that $\{\theta_i\}$ does not drift too fast to bad values, then the algorithm preserves ergodicity. In fact the analysis performed by the authors can be directly used to study the general stabilisation strategy of Andradóttir (1995) (see also reference therein) for stochastic approximation.

Finally, under more restrictive conditions, detailed in Benveniste et al. (1990) and Andrieu and Atchadé (2007, Theorem 3.1), which include the uniqueness of θ^* such that $h(\theta^*) = 0$ and conditions (8)–(9) for $\theta \in \mathcal{K} \subset \Theta$, it is possible to show that for a deterministic sequence $\{\gamma_i\}$, there exists a finite constant C such that for all $i \geq 1$,

$$\check{\mathbb{E}}_* \left[|\theta_i - \theta^*|^2 \mathbb{I}\{\sigma \geq i\} \right] \leq C\gamma_i,$$

where σ is the first exit time from $\mathcal{K}$, meaning that while θ_i remains in $\mathcal{K}$ (where locally uniform conditions of the type (8)–(9) hold), then the rate of convergence towards θ^* is given by $\{\gamma_i\}$.

Automatic choice of the stepsizes The stochastic approximation procedure requires the choice of a stepsize sequence $\{\gamma_i\}$. A standard choice consists of choosing a deterministic sequence satisfying $\sum_{i=1}^\infty \gamma_i = \infty$ and $\sum_{i=1}^\infty \gamma_i^{1+\lambda} < \infty$ for some $\lambda > 0$. The former condition somehow ensure that any point of Θ can eventually be reached, while the second condition ensures that the noise is contained and does not prevent convergence. Such conditions are satisfied by sequences of the type $\gamma_i = C/i^\alpha$ for $\alpha \in ((1+\lambda)^{-1}, 1]$. We tend in practice to favour values closer to the lower bound in order to increase convergence of the algorithm towards a neighbourhood of θ^*. This is at the expense of an increased variance of $\{\theta_i\}$ around θ^* however.

A very attractive approach which can be useful in practice, and for which some theory is available, consists of adapting $\{\gamma_i\}$ in light of the current realisation of the algorithm—this proves very useful in some situations see Andrieu and Jasra (2008). The technique was first described in Kesten (1958) and relies on the remark that, for example, an alternating sign for $\{\hat{\alpha}_{\theta_i} - \bar{\alpha}^*\}$ in (22) is an indication that $\{\theta_i\}$ is oscillating around (a) solution(s), whereas a constant sign suggests that $\{\theta_i\}$ is, roughly speaking, still far from the solution(s). In the former case the stepsize should be decreased, whereas in the later it should, at least, be kept constant. More precisely consider a function $\gamma : [0, +\infty) \to [0, +\infty)$. The standard scenario corresponding to a predetermined deterministic schedule consists of taking $\{\gamma_i = \gamma(i)\}$. The strategy suggested by Kesten (1958) and further generalised to the multivariate case in Delyon and Juditsky (1993) suggests to consider for $i \geq 2$ the following sequence of stepsizes

$$\gamma_i = \gamma \left(\sum_{k=1}^{i-1} \mathbb{I}\{\langle H(\theta_{k-1}, X_k), H(\theta_k, X_{k+1}) \rangle \leq 0\} \right)$$

where $\langle u, v \rangle$ is the inner product between vector u and v. Numerous generalisations are possible in order to take into

account the magnitudes of $\{H(\theta_i, X_{i+1})\}$ in the choice of $\{\gamma_i\}$ (Plakhov and Cruz 2004) (and references therein),

$$\gamma_i = \gamma\left(\sum_{k=1}^{i-1} \phi\left(\langle H(\theta_{k-1}, X_k), H(\theta_k, X_{k+1})\rangle\right)\right)$$

for some function $\phi : \mathbb{R} \to [0, +\infty)$. Numerous generalisations of these ideas are naturally possible and we have found that in numerous situations a componentwise choice of stepsize can lead to major acceleration (Andrieu and Jasra 2008), *i.e.* consider for example for $j = 1, \ldots, n_\theta$

$$\gamma_i^j = \gamma\left(\sum_{k=1}^{i-1} \mathbb{I}\left\{\left\langle H_j(\theta_{k-1}, X_k), H_j(\theta_k, X_{k+1})\right\rangle \le 0\right\}\right)$$

where $H_j(\theta, X)$ is the j-th component of $H(\theta, X)$, but care must be taken to ensure that important properties of θ (such as positivity if it is a covariance matrix) are preserved. Finally note that this idea needs to be handled with care in the unlikely situations where (here in the scalar case for simplicity) $h(\theta) \ge 0$ as well as $H(\theta, x)$ for all $\theta, x \in \Theta \times \mathsf{X}$ and the solution to our problem is on the boundary of Θ.

4.2.3 Some variations

The class of algorithms considered earlier essentially rely on an underlying time homogeneous Markov chain Monte Carlo algorithm with target distribution π. It is however possible to consider non-homogeneous versions of the algorithms developed above. More precisely one can suggest defining a sequence $\{\pi_i, i \ge 1\}$ of probability distributions on X such that $\pi_i \to \pi$ in some sense, *e.g.* total variation distance, and select associated MCMC transition probabilities $\{P_{i,\theta}\}$ such that for any $i \ge 1$ and $\theta \in \Theta$ $\pi_i P_{i,\theta} = \pi_i$. Then the controlled MCMC algorithm defined earlier can use P_{i+1,θ_i} at iteration $i+1$ instead of P_{θ_i}. This opens up the possibility for example to use tempering ideas, *i.e.* choose $\pi_i(x) \propto \pi^{\beta_i}(x)$ for $\beta_i \in (0, 1)$, allowing for the accumulation of useful information concerning the distribution of interest π, while exploring "simpler" distributions. This type of strategy can be useful in order to explore multimodal distributions.

Another possibility, particularly suitable to two stage strategies where adaptation is stopped, consists of removing the vanishing character of adaptation. In the context of stochastic approximation this means for example that the sequence $\{\gamma_i\}$ can be set to a constant small value γ. As a result, in light of the examples of the first section, one expects that under some stability assumptions the chain $\{X_i\}$ will produce samples asymptotically distributed according to an approximation π_γ of π (such that $\pi_\gamma \to \pi$ in some sense) and optimise an approximate criterion corresponding to the standard criterion where π is replaced by π_γ. This strategy can offer some robustness properties.

5 Some adaptive MCMC procedures

In this section we present combinations of strategies, some of them original,[1] which build on the principles developed in previous sections. Note that in order to keep notation simple and ensure readability we present here the simplest versions of the algorithms but that additional features described in Sect. 4.2.2, such as the modification of the mean field to favour stability, the automatic choice of the stepsize (componentwise or not) or Rao-Blackwellisation etc., can easily be incorporated.

5.1 Compound criteria, transient and starting to learn

As pointed out earlier desirable asymptotic criteria and associated optimisation procedures can easily be defined. However it can be observed in practice that the algorithm can be slow to adapt, in particular in situations where the initial guess of the parameter θ is particularly bad, resulting for example in a large rejection probability. More generally the MH algorithm has this particular rather negative characteristic that if not well tuned it will not explore the target distribution and hence will be unable to gather information about it, resulting in a poor learning of the target distribution, and hence algorithms that adapt and behave badly. We describe in this section some strategies that circumvent this problem in practice.

We focus here on the symmetric increments random-walk MH algorithm (hereafter SRWM), in which $q(x, y) = q(x - y)$ for some symmetric probability density q on $\mathbb{R}^{n_x}$, referred to as the *increment distribution*. The transition probability of the Metropolis algorithm is then given for $x, A \in \mathsf{X} \times \mathcal{B}(\mathsf{X})$ by

$$\begin{aligned} &P_q^{SRWM}(x, A) \\ &\quad = \int_{A-x} \alpha(x, x+z) q(z)\, dz \\ &\qquad + \mathbb{I}(x \in A) \int_{\mathsf{X}-x} (1 - \alpha(x, x+z))\, q(z)\, dz, \\ &\quad x \in \mathsf{X}, A \in \mathcal{B}(\mathsf{X}), \end{aligned} \tag{26}$$

where $\alpha(x, y) := 1 \wedge \pi(y)/\pi(x)$. A classical choice for the proposal distribution is $q(z) = \mathcal{N}(z; 0, \Sigma)$, where $\mathcal{N}(z; \mu, \Sigma)$ is the density of a multivariate Gaussian with mean μ and covariance matrix Σ. We will later on refer to this algorithm as the N-SRWM. It is well known that either too small or too large a covariance matrix will result in highly positively correlated Markov chains, and therefore estimators $\hat{I}_n^\Sigma(f)$ with a large variance. In Gelman et al.

[1] First presented at the workshop Adapski'08, 6–8 January 2008, Bormio, Italy.

(1995) it is shown that the "optimal" covariance matrix (under restrictive technical conditions not given here) for the N-SRWM is $(2.38^2/n_x)\Sigma_\pi$, where Σ_π is the true covariance matrix of the target distribution. In Haario et al. (2001) (see also Haario et al. 1999) the authors have proposed to "learn Σ_π on the fly", whenever this quantity exists. It should be pointed out here that in situations where this quantity is not well defined, one should resort to "robust" type estimates in order to capture the dependence structure of the target distribution; we do not consider this here. Denoting $P^{SRWM}_{\mu_i,\Sigma_i}$ the transition probability of the N-SRWM with proposal distribution $\mathcal{N}(0,\lambda\Sigma)$ for some $\lambda > 0$. With $\lambda = 2.38^2/n_x$, the algorithm in Haario et al. (2001) can be summarised as follows,

Algorithm 2 AM algorithm

- Initialise X_0, μ_0 and Σ_0.
- At iteration $i+1$, given X_i, μ_i and Σ_i
  1. Sample $X_{i+1} \sim P^{SRWM}_{\mu_i,\Sigma_i}(X_i,\cdot)$.
  2. Update

$$\begin{aligned}\mu_{i+1} &= \mu_i + \gamma_{i+1}(X_{i+1}-\mu_i),\\ \Sigma_{i+1} &= \Sigma_i + \gamma_{i+1}((X_{i+1}-\mu_i)(X_{i+1}-\mu_i)^{\mathrm{T}} - \Sigma_i).\end{aligned} \tag{27}$$

This algorithm has been extensively studied in Andrieu and Moulines (2006), Atchadé and Fort (2008), Bai et al. (2008) and Andrieu and Tadić (2007). We now detail some simple improvements on this algorithm.

5.1.1 *Rao-Blackwellisation and square root algorithms*

Following (Ceperley et al. 1977) and (Frenkel 2006), we note that, conditional upon the previous state X_i of the chain and the proposed transition Y_{i+1}, the vector $f(X_{i+1})$ (for any function $f : \mathsf{X} \to \mathbb{R}^{n_f}$) can be expressed as

$$\begin{aligned}f(X_{i+1}) := {} & \mathbb{I}\{U_{i+1} \le \alpha(X_i, Y_{i+1})\} f(Y_{i+1})\\ & + \mathbb{I}\{U_{i+1} > \alpha(X_i, Y_{i+1})\} f(X_i),\end{aligned} \tag{28}$$

where $U_{i+1} \sim \mathcal{U}(0,1)$. The expectation of $f(X_{i+1})$ with respect to U_{i+1} conditional upon X_i and Y_{i+1} leads to

$$\begin{aligned}\overline{f(X_{i+1})} := {} & \alpha(X_i, Y_{i+1}) f(Y_{i+1})\\ & + (1-\alpha(X_i, Y_{i+1})) f(X_i).\end{aligned} \tag{29}$$

For example $\bar{X}_{i+1} := \alpha(X_i, Y_{i+1})Y_{i+1} + (1-\alpha(X_i, Y_{i+1}))X_i$ is the "average location" of state X_{i+1} which follows X_i given Y_{i+1}. This can be incorporated in the following "Rao-Blackwellised" AM recursions

$$\begin{aligned}\mu_{i+1} &= \mu_i + \gamma_{i+1}\left[\alpha(X_i, Y_{i+1})(Y_{i+1}-\mu_i) + (1-\alpha(X_i, Y_{i+1}))(X_i-\mu_i)\right],\\ \Sigma_{i+1} &= \Sigma_i + \gamma_{i+1}\left[\alpha(X_i, Y_{i+1})(Y_{i+1}-\mu_i)(Y_{i+1}-\mu_i)^{\mathrm{T}} + (1-\alpha(X_i, Y_{i+1}))(X_i-\mu_i)(X_i-\mu_i)^{\mathrm{T}} - \Sigma_i\right].\end{aligned}$$

Using, for simplicity, the short notation (29) a Rao-Blackwellised AM algorithm can be described as follows:

Algorithm 3 Rao-Blackwellised AM algorithm

- Initialise X_0, μ_0 and Σ_0.
- At iteration $i+1$, given X_i, μ_i and Σ_i
  1. Sample $Y_{i+1} \sim \mathcal{N}(X_i, \Sigma_i)$ and set $X_{i+1} = Y_{i+1}$ with probability $\alpha(X_i, Y_{i+1})$, otherwise $X_{i+1} = X_i$.
  2. Update

$$\begin{aligned}\mu_{i+1} &= \mu_i + \gamma_{i+1}(\bar{X}_{i+1}-\mu_i),\\ \Sigma_{i+1} &= \Sigma_i + \gamma_{i+1}[\overline{(X_{i+1}-\mu_i)(X_{i+1}-\mu_i)^{\mathrm{T}}} - \Sigma_i].\end{aligned} \tag{30}$$

Note that it is not clear that this scheme is always advantageous in terms of asymptotic variance of the estimators, as shown in Delmas and Jourdain (2007), but this modification of the algorithm might be beneficial during its transient whenever the acceptance probability is not too low naturally.

It is worth pointing out that for computational efficiency and stability one can directly update the Choleski decomposition of Σ_i, using the classical rank 1 update formula

$$\begin{aligned}\Sigma_{i+1}^{1/2} = {} & (1-\gamma_{i+1})^{1/2}\Sigma_i^{1/2}\\ & + \frac{\sqrt{1+\frac{\gamma_{i+1}}{1-\gamma_{i+1}}\|\Sigma_i^{-1/2}(X_{i+1}-\mu_i)\|^2}-1}{\|\Sigma_i^{-1/2}(X_{i+1}-\mu_i)\|^2}\\ & \times (1-\gamma_{i+1})^{1/2}(X_{i+1}-\mu_i)(X_{i+1}-\mu_i)^{\mathrm{T}}\Sigma_i^{-\mathrm{T}/2}\end{aligned}$$

where $A^{\mathrm{T}/2}$ is a shorthand notation for $(A^{1/2})^{\mathrm{T}}$ whenever this quantity is well defined. This expression can be simplified through an expansion (requiring $\gamma_{i+1} \ll 1$) and modified to enforce a lower triangular form as follows

$$\begin{aligned}\Sigma_{i+1}^{1/2} = {} & \Sigma_i^{1/2} + \gamma_{i+1}\Sigma_i^{1/2}\mathcal{L}\\ & \times\left(\Sigma_i^{-1/2}(X_{i+1}-\mu_i)(X_{i+1}-\mu_i)^{\mathrm{T}}\Sigma_i^{-\mathrm{T}/2} - I\right),\end{aligned}$$

where $\mathcal{L}(A)$ is the lower triangular part of matrix A. Note again the familiar stochastic approximation form of the re-

cursion, whose mean field is

$$\mathcal{L}\left(\Sigma^{-1/2}\left(\Sigma_\pi + (\mu-\mu_\pi)(\mu-\mu_\pi)^{\mathrm{T}}\right)\Sigma^{-\mathrm{T}/2} - I\right),$$

and whose zeros (together with those of the recursion on the mean) are precisely any square root of Σ_π. The operator ensures that the recursion is constrained to lower triangular matrices. Note that this is only required if one wishes to save memory. Rank r updates can also be used when the covariance matrix is updated every r iterations only. In what follows, whenever covariance matrices are updated, recursions of this type can be used although we will not make this explicit for notational simplicity.

5.1.2 Compound criterion: global approach

As pointed out earlier, in the case of the N-SRWM algorithm the scaling of the proposal distribution is well understood in specific scenarios and intuitively meaningful for a larger class of target distributions. A good rule of thumb is to choose $\lambda = (2.38^2/n_x)\Sigma_\pi$, where Σ_π is the covariance matrix of π. We have shown above that following (Haario et al. 2001) one can in principle estimate Σ_π from the past of the chain. However the difficulties that lead to the desire to develop adaptive algorithms in the first place, including the very poor exploration of the target distribution of π, also hinder learning about the target distribution in the initial stages of an adaptive MCMC algorithm when our initial value for the estimator of Σ_π is a poor guess. Again if $\lambda\Sigma_i$ is either too large in some directions or too small in all directions the algorithm has either a very small or a very large acceptance probability, which results in a very slow learning of Σ_π since the exploration of the target's support is too localised. This is a fundamental problem in practice, which has motivated the use of delayed rejection for example (Haario et al. 2003), and for which we present here an alternative solution which relies on the notion of composite criterion.

While theory suggests a scaling of $\lambda = 2.38^2/n_x$ we propose here to adapt this parameter in order to coerce the acceptance probability to a preset and sensible value (*e.g.* 0.234), at least in the initial stages of the algorithm. Indeed, while this adaptation is likely not to be useful in the long-run, this proves very useful in the early stages of the algorithm (we provide a detailed illustration in Sect. 6.3) where the pathological behaviour described above can be detected through monitoring of the acceptance probability, and corrected.

As a consequence in what follows the proposal distribution of the adaptive N-SRWM algorithm we consider is $q_\theta(z) = \mathcal{N}(z; 0, \lambda\Sigma)$ where here $\theta := (\lambda, \mu, \Sigma)$. Assuming that for any fixed covariance matrix Σ the corresponding expected acceptance probability $\bar{\alpha}_\lambda$ (see (14)) is a non-increasing function of λ, one can naturally suggest the recursion $\log\lambda_{i+1} = \log\lambda_i + \gamma_{i+1}[\alpha(X_i, Y_{i+1}) - \bar{\alpha}_*]$, which following the discussion of Sect. 4 is nothing but a standard Robbins-Monro recursion. Now when the covariance matrix Σ_π needs to be estimated, one can suggest the following "compound criterion" or "multicriteria" algorithm:

Algorithm 4 AM algorithm with global adaptive scaling

- Initialise X_0, μ_0 and Σ_0.
- At iteration $i+1$, given X_i, μ_i, Σ_i and λ_i

1. Sample $Y_{i+1} \sim \mathcal{N}(X_i, \lambda_i\Sigma_i)$ and set $X_{i+1} = Y_{i+1}$ with probability $\alpha(X_i, Y_{i+1})$, otherwise $X_{i+1} = X_i$.
2. Update

$$\begin{aligned}
\log(\lambda_{i+1}) &= \log(\lambda_i) + \gamma_{i+1}[\alpha(X_i, Y_{i+1}) - \bar{\alpha}_*],\\
\mu_{i+1} &= \mu_i + \gamma_{i+1}(X_{i+1} - \mu_i),\\
\Sigma_{i+1} &= \Sigma_i + \gamma_{i+1}[(X_{i+1}-\mu_i)(X_{i+1}-\mu_i)^{\mathrm{T}} - \Sigma_i].
\end{aligned} \tag{31}$$

Again the interest of the algorithm is as follows: whenever our initial guess Σ_0 is either two large or two small, this will be reflected in either a large or small acceptance probability, meaning that learning of Σ_π is likely to be slow for a fixed scaling parameter. However this measure of performance of the algorithm can be exploited as illustrated above: if $\alpha(X_i, Y_{i+1}) - \bar{\alpha}_* < 0$ for most transition attempts then λ_i should be decreased, while if on the other hand $\alpha(X_i, Y_{i+1}) - \bar{\alpha}_* \geq 0$ for most transition attempts, then λ_i should be increased. As a result one might expect a more rapid exploration of the target distribution following a poor initialisation. Although this strategy can improve the performance of the standard AM algorithm in practice, we show in the next section that it is perfectible.

5.1.3 Compound criterion: local approach

As we shall now see, the global approach described in the previous subsection might be improved further. There are two reasons for this. First it should be clear that adjusting the global scaling factor ignores the fact that the scaling of $\lambda_i\Sigma_i$ might be correct in some directions, but incorrect in others. In addition, in order to be efficient, such *bold* updates require in general some good understanding of the dependence structure of the target distribution, in the form of a reasonable estimate of Σ_π, which is not available in the initial stages of the algorithm. These problems tend to be amplified in scenarios involving a large dimension n_x of the space X since innocuous approximations in low dimensions tend to accumulate in larger cases. Inspired by Haario et al. (2005), we suggest the following componentwise update

strategy which consists of a mixture of *timid* moves whose role is to attempt simpler transitions better able to initiate the exploration of π. Note, however, that in contrast with (Haario et al. 2005) our algorithm uses the notion of compound criterion, which in our experience significantly improves performance. With e_k the vector with zeroes everywhere but for a 1 on its k-th row and a sensible $\bar{\alpha}_{**} \in (0, 1)$ *e.g.* 0.44:

Algorithm 5 Componentwise AM with componentwise adaptive scaling

- Initialise X_0, μ_0, Σ_0 and $\lambda_0^1, \ldots, \lambda_0^{n_x}$.
- At iteration $i+1$, given μ_i, Σ_i and $\lambda_i^1, \ldots, \lambda_i^{n_x}$

1. Choose a component $k \sim \mathcal{U}\{1, \ldots, n_x\}$.
2. Sample $Y_{i+1} \sim X_i + e_k \mathcal{N}(0, \lambda_i^k [\Sigma_i]_{k,k})$ and set $X_{i+1} = Y_{i+1}$ with probability $\alpha(X_i, Y_{i+1})$, otherwise $X_{i+1} = X_i$.
3. Update

$$\begin{aligned} \log(\lambda_{i+1}^k) &= \log(\lambda_i^k) + \gamma_{i+1}[\alpha(X_i, Y_{i+1}) - \bar{\alpha}_{**}], \\ \mu_{i+1} &= \mu_i + \gamma_{i+1}(X_{i+1} - \mu_i), \\ \Sigma_{i+1} &= \Sigma_i + \gamma_{i+1}[(X_{i+1} - \mu_i)(X_{i+1} - \mu_i)^{\mathrm{T}} - \Sigma_i] \end{aligned} \tag{32}$$

and $\lambda_{i+1}^j = \lambda_i^j$ for $j \neq k$.

One might question the apparently redundant use of both a scaling λ_i^k and the marginal variance $[\Sigma_i]_{k,k}$ in the proposal distributions above, and one might choose to combine both quantities into a single scaling factor. However the present formulation allows for a natural combination (*i.e.* a mixture or composition) of the recursion above and variations of the standard AM algorithm (Algorithm 2) such as Algorithm 4. Such combinations allow one to circumvent the shortcomings of bold moves, which require extensive understanding of the structure of π, in the early iterations of the algorithm. The timid moves allow the procedure to start gathering information about π which might then be used by more sophisticated and more global updates.

We now turn to yet another version of the AM algorithm (Algorithm 2) which can be understood as being a version of Algorithm 4 which exploits the local scalings computed by Algorithm 5 instead of a single global scaling factor. It consists of replacing the proposal distribution $\mathcal{N}(X_i, \lambda_i \Sigma_i)$ in Algorithm 4 with $\mathcal{N}(X_i, \Lambda_i^{1/2} \Sigma_i \Lambda_i^{1/2})$, where

$$\Lambda_i := \operatorname{diag}\left(\lambda_i^1, \ldots, \lambda_i^{n_x}\right).$$

As we now show, such an update can be combined with Algorithm 5 into a single update. For a vector V we will denote $V(k)$ its k-th component and e_k the vector with zeroes everywhere but for a 1 on its k-th row. We have,

Algorithm 6 Global AM with componentwise adaptive scaling

- Initialise X_0, μ_i, Σ_i and $\lambda_i^1, \ldots, \lambda_i^{n_x}$.
- Iteration $i+1$

1. Given μ_i, Σ_i and $\lambda_i^1, \ldots, \lambda_i^{n_x}$, sample $Z_{i+1} \sim \mathcal{N}(0, \Lambda_i^{1/2} \Sigma_i \Lambda_i^{1/2})$ and set $X_{i+1} = X_i + Z_{i+1}$ with probability $\alpha(X_i, X_i + Z_{i+1})$, otherwise $X_{i+1} = X_i$.
2. Update for $k = 1, \ldots, n_x$

$$\begin{aligned} \log(\lambda_{i+1}^k) &= \log(\lambda_i^k) + \gamma_{i+1}[\alpha(X_i, X_i + Z_{i+1}(k)e_k) \\ &\qquad - \bar{\alpha}_{**}], \\ \mu_{i+1} &= \mu_i + \gamma_{i+1}(X_{i+1} - \mu_i), \\ \Sigma_{i+1} &= \Sigma_i + \gamma_{i+1}[(X_{i+1} - \mu_i)(X_{i+1} - \mu_i)^{\mathrm{T}} - \Sigma_i]. \end{aligned} \tag{33}$$

It is naturally possible to include an update for a global scaling parameter, but we do not pursue this here. This algorithm exploits the fact that a proposed sample $X_i + Z_{i+1}$ provides us with information about scalings in various directions through the "virtual" componentwise updates with increments $\{Z_{i+1}(k)e_k\}$ and their corresponding directional acceptance probabilities. This strategy naturally requires $n_x + 1$ evaluations of π, which is equivalent to one update according to Algorithm 4 and n_x updates according to Algorithm 5.

5.2 Fitting mixtures, clustering and localisation

As pointed out in Andrieu and Moulines (2006, Sect. 7) the moment matching criterion corresponding to the recursion (27) can be understood as minimising the Kullback-Leibler divergence

$$KL_\theta(\pi, q_\theta) := \mathbb{E}_\pi\left(\log \frac{\pi(X)}{\breve{q}_\theta(X)}\right) \tag{34}$$

where $\breve{q}_\theta(x) = \mathcal{N}(x; \mu, \Sigma)$ (but using $q_\theta(z) = \mathcal{N}(z; 0, \lambda\Sigma)$ as a proposal distribution for the increments of a N-SRWM update). This remark leads to the following considerations, of varying importance.

The first remark is that $\breve{q}_\theta$ could be used as the proposal distribution of an independent MH (IMH) update, as in Andrieu and Moulines (2006) or Giordani and Kohn (2006). Although this might be a sensible choice when $\breve{q}_\theta(x)$ is a good approximation of π, this might fail when θ is not close to θ^* (in the transient for example) or simply because the chosen parametric form is not sufficiently rich. In addition such a bad behaviour is generally exacerbated by large dimensions as illustrated by the following toy example.

Example 1 The target distribution is $\pi(x) = \mathcal{N}(x; 0, I)$ with $x \in \mathbb{R}^{n_x}$ and proposal distribution $q(x) = \mathcal{N}(x; \varepsilon \times$

$e, I)$ for some $\varepsilon > 0$ with $e = (1, 1, 1, \ldots)^{\mathrm{T}}$. The importance sampling weight entering the acceptance ratio of an IMH algorithm is

$$\frac{\pi(x)}{q(x)} = \exp\left(\frac{1}{2}\varepsilon^2 n_x - \varepsilon e^{\mathrm{T}} x\right) = \exp\left(\frac{-1}{2}\varepsilon^2 n_x - \varepsilon n_x^{1/2} n_x^{-1/2} \sum_{i=1}^{n_x} (x(i) - \varepsilon)\right),$$

which is not bounded, hence preventing geometric ergodicity. The distribution of $n_x^{-1/2} \sum_{i=1}^{n_x} (x(i) - \varepsilon)$ is precisely $\mathcal{N}(0, 1)$, which results in a variance for the weights of

$$\exp\left(\varepsilon^2 n_x\right) - 1.$$

This is known to result in poorly performing importance sampling algorithms, but will also have an impact on the convergence of IMH algorithms which will get stuck in states x with arbitrarily large weights x as n_x increases, with non negligible probability.

IMH updates hence fall in the category of "*very bold*" updates which require significant knowledge of the structure of π and do not usually form the base for reliable adaptive MCMC algorithms.

The second remark, which turns out to be of more interest, is that one can consider other parametric forms for $\breve{q}_\theta$, and use such approximations of π to design proposal distributions for random walk type algorithms, which are likely to perform better given their robustness. It is suggested in Andrieu and Moulines (2006, Sect. 7) to consider mixtures, finite or infinite, of distributions belonging to the exponential family (see also Cappé et al. 2007 for a similar idea in the context of importance sampling/population Monte Carlo). This has the advantage of leading to an elegant optimisation algorithm which relies on an on-line version of the EM algorithm and results in a marginal additional computational overhead.

In this section we first detail two particular cases of this procedure: mixture of normal distributions and Student t-distributions.

5.2.1 *Updates for fitting mixtures in the exponential family*

We first briefly review how, given samples $\{X_i\}$, it is possible to iteratively fit a mixture

$$\breve{q}_\theta(x) = \sum_{k=1}^{n} w^k \mathcal{N}(x; \mu^k, \Sigma^k), \tag{35}$$

with $\theta = (w, \mu, \Sigma)$ with $w = (w^1, w^2, \ldots, w^n)$, in order to minimise (34). For the purpose of describing the algorithm it is convenient to introduce the missing data z such that $\breve{q}_\theta(x, z = k) := w^k \mathcal{N}(x; \mu^k, \Sigma^k)$ and hence for $k \in \{1, \ldots, n\}$

$$\breve{q}_\theta(k|x) = \frac{w^k \mathcal{N}(x; \mu^k, \Sigma^k)}{\breve{q}_\theta(x)} = \frac{w^k \mathcal{N}(x; \mu^k, \Sigma^k)}{\sum_{l=1}^{n} w^l \mathcal{N}(x; \mu^l, \Sigma^l)}.$$

Now for any $k \in \{1, \ldots, n\}$ and $i \geq 0$ the recursions are, with

$$\breve{q}_{\theta_i}(Z_{i+1} = k|x) := \frac{w_i^k \mathcal{N}(x; \mu_i^k, \Sigma_i^k)}{\breve{q}_{\theta_i}(x)},$$

$$\begin{aligned} \mu_{i+1}^k &= \mu_i^k + \gamma_{i+1}\, \breve{q}_{\theta_i}(Z_{i+1} = k|X_{i+1})(X_{i+1} - \mu_i^k), \\ \Sigma_{i+1}^k &= \Sigma_i^k + \gamma_{i+1}\, \breve{q}_{\theta_i}(Z_{i+1} = k|X_{i+1}) \\ &\quad \times [(X_{i+1} - \mu_i^k)(X_{i+1} - \mu_i^k)^{\mathrm{T}} - \Sigma_i^k], \\ w_{i+1}^k &= w_i^k + \gamma_{i+1}(\breve{q}_{\theta_i}(Z_{i+1} = k|X_{i+1}) - w_i^k). \end{aligned} \tag{36}$$

Note that the standard EM framework suggests various acceleration techniques, which we do not consider here for brevity.

It is also possible to consider a mixture of multivariate Student-t distributions, which is a mixed continuous/discrete mixture of normals. More precisely consider the case where

$$\breve{q}_\theta(x) = \sum_{k=1}^{n} w^k \mathcal{T}_\nu(x; \mu^k, \Sigma^k)$$

where

$$\mathcal{T}_\nu(x; \mu, \Sigma) = \frac{\Gamma(\frac{\nu+n_x}{2})\, |\Sigma|^{-1/2}}{(\pi\nu)^{\frac{1}{2}n_x}\Gamma(\frac{\nu}{2})(1 + \frac{1}{\nu}(x-\mu)^{\mathrm{T}}\Sigma^{-1}(x-\mu))^{\frac{1}{2}(\nu+n_x)}}.$$

We consider here for simplicity the case "one ν for all" since we are not interested in a very precise fit of the target distribution. Note that as $\nu \to \infty$ the mixture converges to a mixture of normal distributions which coincides with that described above. The on-line EM algorithm relies on the standard fact that $\breve{q}_\theta(x)$ can be seen as the marginal distribution of

$$\breve{q}_\theta(k, u, x) = \frac{w_k u^{n_x}}{\sqrt{|2\pi\Sigma_k|}} \exp\left(\frac{-u}{2}(x - \mu_k)^{\mathrm{T}}\Sigma_k^{-1}(x - \mu_k)\right) \times \frac{(\nu/2)^{\nu/2}}{\Gamma(\nu/2)} u^{\nu/2-1} \exp\left(-\frac{\nu}{2}u\right) \mathbb{I}\{u \geq 0\}.$$

We denote

$$\breve{q}_{\theta_i}(Z_{i+1} = k|X_{i+1}) := \frac{w_i^k \mathcal{T}_\nu(x; \mu_i^k, \Sigma_i^k)}{\breve{q}_{\theta_i}(x)}$$

and introduce the conditional expectation of U given X and Z

$$\bar{u}(k,X) := \mathbb{E}^{\theta}[U|k,X] = \frac{\nu + n_x}{\nu + (X-\mu_k)^{\mathrm{T}}\Sigma_k^{-1}(X-\mu_k)}.$$

The required recursions are

$$\begin{aligned}
\mu_{i+1}^k &= \mu_i^k + \gamma_{i+1}\,\bar{u}(k,X_{i+1})\breve{q}_{\theta_i}(Z_{i+1}=k|X_{i+1}) \\
&\quad \times (X_{i+1}-\mu_i^k), \\
\Sigma_{i+1}^k &= \Sigma_i^k + \gamma_{i+1}\,\bar{u}(k,X_{i+1})\breve{q}_{\theta_i}(Z_{i+1}=k|X_{i+1}) \\
&\quad \times [(X_{i+1}-\mu_i^k)(X_{i+1}-\mu_i^k)^{\mathrm{T}} - \Sigma_i^k], \\
w_{i+1}^k &= w_i^k + \gamma_{i+1}(\breve{q}_{\theta_i}(Z_{i+1}=k|X_{i+1}) - w_i^k), \\
\bar{u}_{i+1}^k &= \bar{u}_i^k + \gamma_{i+1}\left(\bar{u}(k,X_{i+1})\breve{q}_{\theta_i}(Z_{i+1}=k|X_{i+1}) - \bar{u}_i^k\right).
\end{aligned}$$

This later choice is closely related to the "fast K-mean" algorithm used in Giordani and Kohn (2006) (although the algorithm developed there is not on-line, whereas the algorithm developed here is computationally very efficient) which is beneficial in the initial stages of the algorithm in order to start the learning process. In practice we suggest that when fitting a mixture of normal distributions, the recursions for the Student t-distributions be used with a parameter $\nu_i \to \infty$ with the iterations.

5.2.2 *Localised random walk Metropolis updates*

The Metropolis-Hastings algorithm in its simplest form offers the possibility for local adaptation given the possible dependence of its family of proposal distributions $\{q(x,\cdot), x \in \mathsf{X}\}$ on the current state of the Markov chain. Obvious examples include the Langevin algorithm or self-targeting schemes (Stramer and Tweedie 1999). This dependence is exploited further in Green (1995) where the weights of a mixture of MH updates are allowed to depend on the current state x of the Markov chain, hence offering the possibility to select a particular update depending on the region currently visited by, say, state $X_i = x$.

We now describe an original use of the information about π contained in the approximation $\breve{q}_\theta(x)$ of π which allows for some localisation of the adaptation in the spirit of a suggestion in Andrieu and Robert (2001) concerned with Voronoi tesselations (for which the Linde-Buzo-Gray, an EM like algorithm, could be used here). We however restrict here the presentation to that of an algorithm for which $\breve{q}_\theta(x)$ is a mixture of normal distributions—other cases are straightforward extensions. Note that another form of localisations has been suggested in Roberts and Rosenthal (2006), which is more in line with the ideas of Stramer and Tweedie (1999), and can lead to interesting algorithms.

The algorithm we suggest here is a mixture of N-SRWM algorithms—one should bear in mind that such an algorithm will in general be a component of a much larger mixture or part of a composition of updates. The interest of our approach is that following (Green 1995) we allow the weights of the mixture to depend on the current state of the chain. More precisely one can suggest for example using

$$P_\theta(x,dy) = \sum_{k=1}^{n} \breve{q}_\theta(k|x)\,P_{\theta,k}^{NSRWM}(x,dy)$$

where $\theta = (\mu^{1:n}, \Sigma^{1:n}, w^{1:k}, \lambda^{1:k})$ and the transition $P_{\theta,k}^{NSRW}(x,dy)$ is a random walk Metropolis algorithm with proposal distribution, here in the normal case, $\mathcal{N}(y;x, \lambda^k\Sigma^k)$. Note that other choices than the weights $\breve{q}_\theta(k|x)$ can be chosen in order to ensure, in particular in the early stages of the algorithm, that all components are being used. In the case of a mixture of normals one can for example suggest using the conditional distribution $\breve{q}_\theta(k|x)$ for a mixture of Student t-distributions with a parameter $\nu_i \to \infty$ as $i \to \infty$. The choice of λ^k is made adaptive in order to achieve a preset acceptance probability, according to (31). The motivations for this algorithm are twofold: (a) first the weight $\breve{q}_\theta(k|x)$, or a function of this quantity, favours association of x to relevant components of the mixture of distributions $\breve{q}_\theta(x)$, that is for example the local linear dependencies present among the components of x through the covariance matrices $\Sigma^{1:n}$ (b) secondly $\breve{q}_\theta(x)$ can be used in order to cluster states of X and associate local criteria to each cluster (here a local expected acceptance probability but other choices are possible) which in turn can be locally adapted using a rule of our choice. Note the advantage of this algorithm in terms of innovation (or exploration in machine learning speak) over a simple IMH algorithm that would try to sample and learn from its own samples.

The algorithm can be summarised with the following pseudo-code in the case where a mixture of normal distributions is used in order to map the state-space.

Algorithm 7 Localised N-SRWM algorithm

- Initialise $X_0, \mu_0^{1:n}, \Sigma_0^{1:n}, w_0^{1:n}$ and $\lambda_0^{1:n}$.
- Iteration $i+1$, given $X_i, \mu_i^{1:n}, \Sigma_i^{1:n}, w_i^{1:n}$ and $\lambda_i^{1:n}$
  1. $Z_{i+1} \sim \breve{q}_{\theta_i}(Z=k|X_i)$, $Y_{i+1} \sim \mathcal{N}(X_i, \lambda_i^{Z_{i+1}}\Sigma_i^{Z_{i+1}})$ and set $X_{i+1} = Y_{i+1}$ with probability $\min\{1, \frac{\pi(Y_{i+1})\breve{q}_{\theta_i}(k|Y_{i+1})}{\pi(X_i)\breve{q}_{\theta_i}(k|X_i)}\}$, otherwise $X_{i+1} = X_i$.
  2. Update $\mu_i^{1:n}, \Sigma_i^{1:n}, w_i^{1:n}$ and $\lambda_i^{1:k}$ to $\mu_{i+1}^{1:n}, \Sigma_{i+1}^{1:n}, w_{i+1}^{1:n}$ and $\lambda_{i+1}^{1:n}$, according to (36) and (37).

The localised nature of the algorithm, in the spirit of the state dependent mixtures of updates of Green (1995), requires some care. Firstly note the form of the acceptance

probability required in order to ensure that the underlying "fixed θ" transition probability is in detailed balance with π

$$\alpha_k(x,y) := \min\left\{1, \frac{\pi(y)\breve{q}_\theta(k|y)}{\pi(x)\breve{q}_\theta(k|x)}\right\}.$$

Secondly, updating of the parameters requires some attention. Indeed, given that component k is chosen, we wish to adjust the conditional expected acceptance probability of this component in order to reach an expected acceptance rate $\bar{\alpha}_*$. In mathematical terms we wish to set the following mean field $h(\theta)$ with components $h_k(\theta)$ to zero,

$$\begin{aligned} h_k(\theta) &:= \int_{\mathsf{X}^2} \frac{\pi(x)\breve{q}_\theta(k|x)\mathcal{N}(y;x,\lambda^k\Sigma^k)}{\int_{\mathsf{X}}\pi(x)\breve{q}_\theta(k|x)dx}\alpha_k(x,y)dxdy - \bar{\alpha}_* \\ &= \frac{\int_{\mathsf{X}^2}\pi(x)\breve{q}_\theta(k|x)\mathcal{N}(y;x,\lambda^k\Sigma^k)\,(\alpha_k(x,y)-\bar{\alpha}_*)\,dxdy}{\int_{\mathsf{X}}\pi(x)\breve{q}_\theta(k|x)dx}, \end{aligned}$$

where the fraction on the first line is the density of the conditional steady state distribution $\mathbb{P}^\theta(X\sim\pi, Y\sim\mathcal{N}(X,\lambda^k\Sigma^k)|Z=k)$. Finding the zeros of $h_k(\theta)$ hence amounts to finding the zeros of the top of the last fraction, which can be written as an expectation

$$\begin{aligned} &\sum_{m=1}^n \int_{\mathsf{X}^2}\pi(x)\breve{q}_\theta(m|x)\mathbb{I}\{m=k\} \\ &\qquad\times\mathcal{N}(y;x,\lambda^k\Sigma^k)\,(\alpha_k(x,y)-\bar{\alpha}_*)\,dxdy \\ &= \sum_{m=1}^n \int_{\mathsf{X}^2}\pi(x)\breve{q}_\theta(m|x)\mathcal{N}(y;x,\lambda^m\Sigma^m) \\ &\qquad\times\mathbb{I}\{m=k\}\,(\alpha_k(x,y)-\bar{\alpha}_*)\,dxdy. \end{aligned}$$

The second form of $h_k(\theta)$ above (and since the denominator does not (in general) affect the zeros of $h_k(\theta)$) suggests the following recursions to update $\{\lambda_i^k\}$ and compute the components' running expected acceptance probabilities $\{\alpha_i^k\}$

$$\begin{aligned} \log(\lambda_{i+1}^k) &= \log(\lambda_i^k) + \gamma_{i+1}\mathbb{I}\{Z_{i+1}=k\} \\ &\qquad\times\left[\alpha_k(X_i,Y_{i+1})-\bar{\alpha}_*\right], \\ \alpha_{i+1}^k &= \alpha_i^k + \gamma_{i+1}\mathbb{I}\{Z_{i+1}=k\}\left[\alpha_k(X_i,Y_{i+1})-\alpha_i^k\right]. \end{aligned} \tag{37}$$

Naturally we do not address here the choice of the number n of components of the mixture. Although the use of simple information criteria could be advocated in order to choose n, even in a crude way, we believe that although feasible this might lead to additional complications at this stage. We here simply argue that choosing $n>1$ should in general be beneficial compared to the use of a plain N-SRWM for which $n=1$. Alternatively one can suggest the possibility of fitting simultaneously several mixtures with each its own number of components.

5.3 Block sampling and principal directions

For large dimensional problems, updating the whole vector X in one block might lead to a poorly performing algorithm which fails to explore the distribution π of interest. It is standard in practice to attempt to update subblocks of X conditional upon the corresponding complementary subblock, which in practice facilitates the design of better proposal distributions. The choice of such subblocks is however in practice crucial while far from obvious in numerous situations. Indeed it is well known and easy to understand that variables that are highly dependent components of X (under π) should in practice be updated simultaneously as it can otherwise lead to algorithms that are slow to converge, and produce samples with poor statistical properties. Identifying such subblocks of dependent components can be very difficult in practice, and it is natural to ask if it is possible to automatise this task in practice.

A possible suggestion is to consider an MCMC update that takes the form of a mixture of MCMC updates

$$P_\theta(x,dy) = \sum_{k=1}^n \omega_k(\theta)\,P_{k,\theta}(x,dy), \tag{38}$$

where for any $\theta\in\Theta$, $\sum_{k=1}^n\omega_k(\theta)=1$, $\omega_k(\theta)\geq 0$ and $\{P_{i,\theta}, i=1,\ldots,n\}$ is a family of "partial" updates which correspond to all the possible partitions of vector X. Then one can suggest updating the weights $\{\omega_k(\theta)\}$ according to some criterion. This is of course not realistic in practice as soon as the dimension n_x of X is even moderate, and can lead to very poorly mixing algorithms since intuitively all the possible transitions should be tried in order to assess their efficiency. This might be inefficient as we expect only a restricted number of these transitions to be of real interest.

Instead we suggest here a simple alternative which relies on principal component analysis and a natural and well known generalisation able to handle multi-modal distributions. Our algorithms rely on the recursive diagonalisation of either the estimates $\{\Sigma_i\}$ of the covariance matrix Σ_π or the covariance matrices $\{\Sigma_i^k, k=1,\ldots,n\}$ used to approximate the target distribution π, *e.g.* using a mixture of normal or Student t-distributions. We will focus here on the former scenario for simplicity, the extension to the mixture of distributions case is straightforward.

5.3.1 Updates description

Formally this update is of the form (38) where $P_{k,\theta}(x,dy)$ is a one-dimensional random walk Metropolis update along eigenvector k of the covariance matrix Σ_π, with a scaling factor $\ell(k)$ which ensures a predetermined acceptance probability. We will describe below the recursive estimation of the first m ($\leq n_x$) eigenvectors of Σ_π, which we assume

form the columns of an $n_x \times m$ matrix W (the columns being denoted $w(l), l = 1, \ldots, m$) and the corresponding eigenvalues $\rho(l)$. We denote hereafter $\bar{\rho}(l) := \rho(l)/\sum_{p=1}^{m} \rho(p)$ the normalised eigenvalues and let $d(1), \ldots, d(m)$ denote an arbitrary distribution on the first m positive integers. The update at iteration i proceeds as follows:

Algorithm 8 Principal components Metropolis update

- At iteration $i+1$, given X_i and (ℓ_i, ρ_i, W_i)
  1. Sample an update direction $l \sim (d(1), d(2), \ldots, d(m))$.
  2. Sample $Z_{i+1} \sim \mathcal{N}(0, \ell_i(l)\rho_i(l))$, set $Y_{i+1} = X_{i+1} + Z_{i+1}\, w(l)$.
  3. Set $X_{i+1} = Y_{i+1}$ with probability $\min\{1, \pi(Y_{i+1})/\pi(X_i)\}$, otherwise $X_{i+1} = X_i$.
  4. Update (ℓ_i, ρ_i, W_i) to $(\ell_{i+1}, \rho_{i+1}, W_{i+1})$ in light of X_{i+1}.

In words, at every iteration one of the available principal direction l is randomly selected, here according to the probability $d(1), d(2), \ldots, d(m)$ ($d(j) = \bar{\rho}(j)$ being a possibility) but other choices are possible, and a "univariate" random walk update in the direction $w(l)$ is then attempted with an increment drawn from $\mathcal{N}(0, \ell_i(l)\rho_i(l))$, where $\ell(l)$ is a directional scaling factor adjusted to ensure that updates in direction l have a preset acceptance probability—it uses an update of the type (22). This enables finer scaling in every principal direction. Note that this update might correspond to a reducible algorithm when $m < n_x$, but that this should not be a difficulty when used in combination with other updates.

We now turn to the description of an on-line algorithm for the computation of the m first eigenvectors of the covariance matrix of samples $\{X_i\}$. The algorithm relies on an on-line EM algorithm for the popular *probabilistic PCA* (PPCA) algorithm.

5.3.2 Online PCA recursion

The basis for PPCA was laid by Tipping and Bishop (1999) who endowed the problem with a linear Gaussian model. Even though the possibility of using an EM-algorithm is mentioned, it is Roweis (1997) who extends the formalism more specifically to the application of such a scheme. The approach suffers however from a rotational ambiguity in the latent variable space, since the returned vector set is a linear superposition of the principal components, inducing a need for post-processing. This drawback is overcome by Ahn and Oh (2003) through the introduction of a constrained EM-algorithm that corresponds to using several coupled models, rather than a single model. These papers assume that all observations are present initially, whereas the adaptive algorithm presented in this project needs to determine the principal eigenvectors on-line. Ghasemi and Sousa (2005) reformulate the constrained EM in order to achieve this. Roweis (1997) mentions an on-line version, but it was not further explored here because of the inherent rotational ambiguity.

The structure that is employed in PPCA is closely related to factor analysis. This linear model is founded on the assumption that the d-dimensional data can be explained by a m-dimensional unobservable variable Z and an additive noise ϵ,

$$X_i = W Z_i + \epsilon, \tag{39}$$

where W is a $n_x \times m$ real valued matrix of *factor loadings*, $Z_i \overset{\text{iid}}{\sim} \mathcal{N}(0, I_m)$ and $\epsilon \sim \mathcal{N}(0, R)$, where usually $R = \sigma^2 I_{n_x}$ for some $\sigma^2 > 0$. It can be shown that the ML estimator of W contains the eigenvectors and the latent variables structure suggests the use of an EM-algorithm. It is possible to alter this problem in order to exactly remove the rotational ambiguity, leading to the following recursion (for $\sigma^2 = 0$)

$$W_{i+1} = \Sigma_{i+1} W_i \mathcal{L}(W_i^{\mathrm{T}} W_i)^{-1} \times \mathcal{U}\left(\mathcal{L}(W_i^{\mathrm{T}} W_i)^{-1} W_i^{\mathrm{T}} \Sigma_{i+1} W_i \mathcal{L}(W_i^{\mathrm{T}} W_i)^{-1}\right)^{-1}$$

where Σ_i is an estimate of the covariance matrix of π at iteration i of the algorithm, while for a square matrix $\mathcal{L}(A)$ (resp. $\mathcal{U}(A)$) is the lower (resp. upper) part of A. Note the computationally interesting feature of this recursion where the inversion of triangular, rather than full, matrices is required. Roweis (1997) also provides an EM-algorithm for PPCA without taking the zero-error limit, in what is called *Sensible PCA*.

5.4 Discrete valued random vectors: the Gaussian copula approach

So far we have implicitly assumed that π has a density with respect to the Lebesgue measure and *de facto* excluded the case where π is the probability distribution of a discrete valued vector. This problem has been largely overlooked in the literature, with however the exception of Nott and Kohn (2005). We here briefly describe a strategy proposed in Andrieu and Moffa (2008) which, as we shall see, offers the possibility to exploit the tools developed for the purely continuous case in earlier sections. It differs from the work presented so far in this paper in that, as we shall see, the distribution π or interest is embedded in an extended probability model, which needs to be adapted.

In order to simplify presentation we will focus on the case where $\mathsf{X} = \{0, 1\}^{n_x}$, the generalisation to scenarios involving a larger number of discrete states or a mixture of continuous

and discrete valued random variables being straightforward. Note that this simple scenario is of interest in the context of variable selection, but also in the context of inference in Ising models. The strategy consists of embedding the discrete valued problem into a continuous framework by means of an auxiliary variable z taking its values in $\mathsf{Z} := \mathbb{R}^{n_x}$. More precisely, consider the following distribution

$$\tilde{\pi}_\mu(x,z) := \pi(x)\prod_{i=1}^{n_x}\frac{\mathcal{N}(z(i);\mu(i),\Sigma(i,i))}{\Phi_{\Sigma(i,i)}(\mu(i))^{x(i)}\left(1-\Phi_{\Sigma(i,i)}(\mu(i))\right)^{1-x(i)}} \times \mathbb{I}\{z\in\mathcal{I}_x\},$$

where $\Phi_{\sigma^2}(u)$ is the cumulative distribution function of the univariate centered normal distribution with variance σ^2, $\mu, \Sigma \in \mathbb{R}^{n_x} \times \mathbb{R}^{n_x \times n_x}$ and $\mathcal{I}_x := I_{x(1)} \times I_{x(2)} \times \cdots \times I_{x(n_x)}$ with $I_0 := (-\infty, 0]$ and $I_1 := (0, +\infty)$. Note that the evaluation of $\Phi_{\sigma^2}(u)$ is routine, and that whenever $\pi(x)$ can be evaluated pointwise up to a normalising constant so can $\tilde{\pi}_\mu(x,z)$, therefore allowing the use of standard sampling algorithms. One can notice that marginally $\tilde{\pi}_\mu(x) = \pi(x)$ but also that

$$\tilde{\pi}_\mu(x(i)|z) \propto \mathbb{I}\{z(i)\in I_{x(i)}\},$$

a type of deterministic relationship between z and x. These properties suggest that the problem of sampling from $\pi(x)$ can be replaced by that of effectively sampling the continuous component $z \sim \tilde{\pi}_\mu(z)$ followed by the determination of the unique x satisfying $\mathbb{I}\{z \in \mathcal{I}_x\} = 1$.

Naturally not all choices of $\mu \in \mathbb{R}^{n_x}$ will lead to efficient sampling algorithms for a given distribution π and we note in addition that the component z does not capture the dependence structure of $\pi(x)$. We shall see now how adaptive procedures can be of great help here. Consider the following distribution and denote $\theta := \{\mu, \breve{\mu}, \Sigma, \breve{\Sigma}\}$ for some $\breve{\mu}, \breve{\Sigma} \in \mathbb{R}^{n_x} \times \mathbb{R}^{n_x \times n_x}$

$$\breve{q}_\theta(x,z) := \mathcal{N}(z;\breve{\mu},\breve{\Sigma})\,\mathbb{I}\{z\in\mathcal{I}_x\},$$

whose marginal $\breve{q}_\theta(x)$ is often used to model the distribution of multivariate discrete valued random vectors *e.g.* in the context of multivariate probit regression models. A particular strength of the model is that the dependence structure of $\breve{q}_\theta(x)$ is parametrised by the pair $\breve{\mu}, \breve{\Sigma}$ and that sampling from $\breve{q}_\theta(x)$ is straightforward. However $\breve{q}_\theta(x)$ is usually intractable, precluding its direct use to approximate $\pi(x)$. A natural suggestion here is simply to work on the extended space $\mathsf{X}\times\mathsf{Z}$ and approximate $\tilde{\pi}_\theta(x,z) := \tilde{\pi}_\mu(x,z)$ with $\breve{q}_\theta(x,z)$. For example, with the natural choice $\mu = \breve{\mu}$ and $\Sigma = \breve{\Sigma}$ one could suggest minimising the following Kullback-Leibler divergence

$$\int_{\mathsf{X}\times\mathsf{Z}} \tilde{\pi}_\theta(x,z)\log\frac{\tilde{\pi}_\theta(x,z)}{\breve{q}_\theta(x,z)}dxdz.$$

Given the structure of $\tilde{\pi}_\theta(x,z)$, it is clear that the resulting $\breve{q}_\theta(x,z)$ is meant to "learn" both the marginals and the dependence structure of $\pi(x)$. Assuming that this can be achieved, even approximately, $\breve{q}_\theta(x,z)$ and its parameters can be used in multiple ways in order to sample from $\tilde{\pi}_\theta(x,z)$. Following the ideas developed in earlier sections, one could suggest to use $\breve{q}_\theta(x,z)$ as a proposal distribution in an IMH algorithm targeting $\tilde{\pi}_\theta(x,z)$. Indeed sampling from $\breve{q}_\theta(x,z)$ is straightforward since it only requires one to sample from $Z \sim \mathcal{N}(\mu,\Sigma)$ and to determine X such that $\mathbb{I}\{Z\in\mathcal{I}_X\} = 1$. However, as argued earlier, using the IMH sampler is not always a good idea, and one could instead suggest a more robust random walk Metropolis type algorithm. For example, for some $\lambda > 0$ and θ, we have

Algorithm 9 The Gaussian copula SRWM

- At iteration $i+1$, given X_i
  1. Sample $Z = Z_i + W$ with $W \sim \mathcal{N}(0, \lambda\breve{\Sigma})$.
  2. Determine X such that $\mathbb{I}\{Z\in\mathcal{I}_X\} = 1$.
  3. Set $(X_{i+1}, Z_{i+1}) = (X, Z)$ with probability
  $$\min\left\{1, \frac{\tilde{\pi}_\theta(X,Z)}{\tilde{\pi}_\theta(X_i,Z_i)}\right\},$$
  otherwise $(X_{i+1}, Z_{i+1}) = (X_i, Z_i)$.

The problem of effectively determining $\breve{\Sigma}$ can be addressed by using an adaptive algorithm, and in particular by using recursions of the type (31) or (32) in the case of an update component by component for example. More generally all the strategies developed earlier in this paper for the continuous case can be adapted to the discrete setup (Andrieu and Moffa 2008). Note however that the target distribution now depends on θ and that a slight modification of the convergence theory outlined earlier is required in this scenario.

6 Examples of applications

6.1 Erratic normal distribution

In this section we first demonstrate the practical interest of the idea of compound criteria developed in Sect. 5.1 which aims to accelerate the learning of features of the target distribution by the algorithm. Following Roberts and Rosenthal (2006) we consider a normal distribution $\mathcal{N}(0, \Sigma_\pi = MM^{\mathrm{T}})$ defined on $\mathsf{X} = \mathbb{R}^{n_x}$ with M a $n_x \times n_x$ matrix with i.i.d. entries sharing the distribution $\mathcal{N}(0,1)$—we focus here on the case $n_x = 50$. The algorithm we use consists of a mixture of Algorithms 4–5 and 8. Comparison with the standard AM algorithm is provided in Figs. 1–3 for a realisation of each of the algorithm and for the same number of

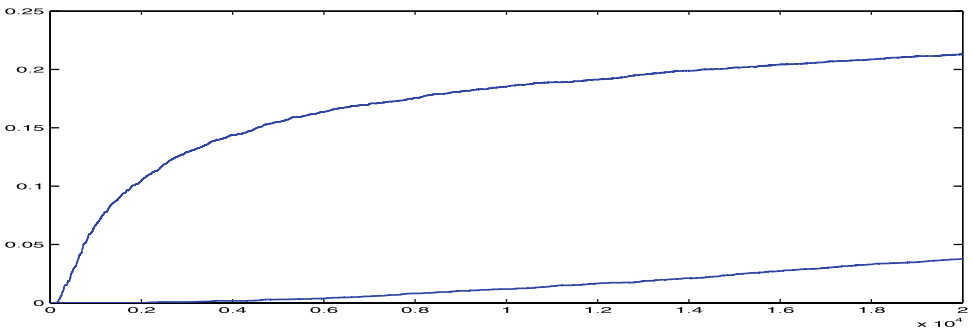

Fig. 1 Comparison of the expected acceptance probability of the standard AM algorithm (*bottom*) and the corresponding global update used by the multi-criteria algorithm as a function of the iterations

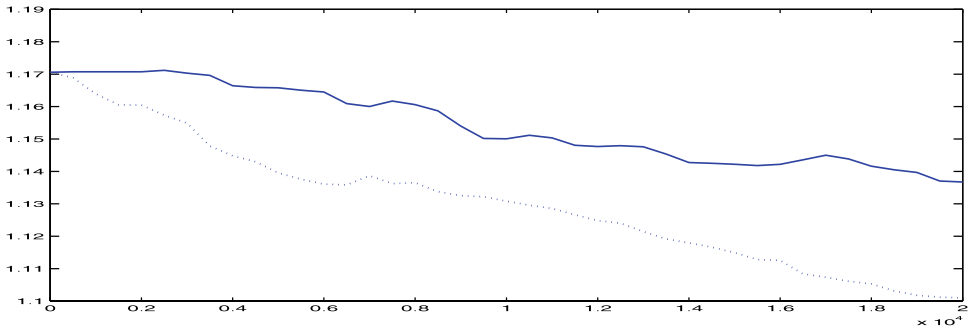

Fig. 2 Comparison of the R coefficient for the standard AM algorithm (*top*) and the multi-criteria algorithm (*bottom*) as a function of the iterations

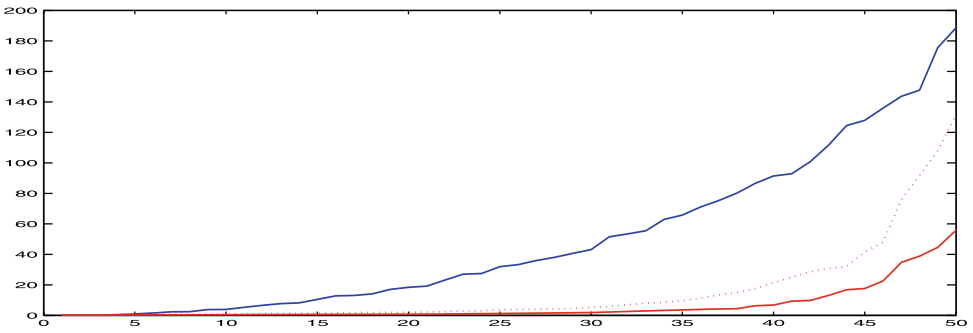

Fig. 3 Comparison of the 50 ordered estimated eigenvalues after 50,000 iterations. *Top*: truth. *Middle*: multi-criteria algorithm. *Bottom*: standard AM algorithm

evaluations of π. The coefficient $R \geq 1$ is precisely defined in Roberts and Rosenthal (2006). It is a measure of mismatch between Σ_π and any arbitrary covariance matrix Σ related to the asymptotic performance of the N-SRWM, the value $R = 1$ corresponding to optimality. Although potentially useful the comparison of the eigenvalues alone might be misleading without a comparison of the quality of the eigenvectors—the R coefficient does this.

The gains are clear in terms of the number of evaluations of the target density, whose computational cost will in general dominate that of the additional recursions needed for adaptation.

Table 1 Summaries (mean + / − std): Norm of the first moment's estimator, based on 100 runs. Different banana-shaped Gaussian distributions are used and compared to the results of the adaptive Metropolis sampler (AM) presented by Haario et al. (1999). Since the target is centered, the norm's correct value is zero

n_x	Norm $\|\mathbb{E}_\pi[X]\|$			
	$\pi_1 = \mathcal{B}_{0.03}$		$\pi_2 = \mathcal{B}_{0.1}$	
	Multi-criteria	AM	Multi-criteria	AM
2	1.13 ± 0.74	1.10 ± 0.67	2.80 ± 1.47	2.62 ± 1.61
4	1.33 ± 0.79	1.27 ± 0.77	5.20 ± 5.69	5.13 ± 12.85
8	1.17 ± 0.67	1.31 ± 0.72	4.99 ± 3.99	4.85 ± 4.20

6.2 The banana example

The banana distribution, introduced in Haario et al. (1999) and Haario et al. (2001) is a popular example to test adaptive algorithms since it presents the advantage of analytical tractability of numerous characteristics, while allowing for a non-linear dependency between its components. Formally it is the distribution of a normally distributed multivariate normal random $X \sim \mathcal{N}(0, \Sigma)$ for $n_x \geq 2$ which undergoes the transformation

$$[X_1, X_2 + b(X_1^2 - 100), X_3, \ldots, X_{n_x}],$$

Table 2 Empirical quantiles of adaptive MCMC output based on 25 runs of length 80,000 (burn-in: 60,000 lags), Banana-shaped target $\mathcal{B}_{0.03}$ in n_x dimensions and an adaptive mixture of three Gaussian distributions, used as proposal, maximum deviation per dimension in red

n_x	Banana-shaped target $\pi_1 = \mathcal{B}_{0.03}$								
	Quantile (in %)								
	10	20	30	40	50	60	70	80	90
2	9.60 ± 0.60	19.54 ± 0.74	29.29 ± 1.07	39.52 ± 1.34	49.63 ± 1.58	59.78 ± 1.85	70.14 ± 1.87	80.38 ± 1.65	90.22 ± 1.24
5	9.55 ± 0.75	19.33 ± 1.08	29.05 ± 1.36	39.17 ± 1.67	49.32 ± 1.98	59.42 ± 2.18	69.37 ± 1.97	79.65 ± 1.80	89.93 ± 1.25
7	9.79 ± 0.81	19.49 ± 1.28	29.43 ± 1.56	39.46 ± 2.09	49.51 ± 2.27	59.58 ± 2.18	69.66 ± 2.04	79.87 ± 1.62	90.15 ± 1.18
9	9.69 ± 1.12	19.58 ± 1.87	29.57 ± 2.33	39.47 ± 2.54	49.47 ± 2.60	59.56 ± 2.43	69.71 ± 1.95	79.47 ± 1.62	90.22 ± 1.22
15	10.27 ± 1.14	20.46 ± 1.84	30.81 ± 2.23	40.77 ± 2.39	50.83 ± 2.31	60.95 ± 2.00	70.95 ± 1.81	80.86 ± 1.54	90.41 ± 1.03

Table 3 Empirical quantiles of adaptive MCMC output based on 25 runs of length 80,000 (burn-in: 60,000 lags), Banana-shaped target $\mathcal{B}_{0.1}$ in n_x dimensions and an adaptive mixture of three Gaussian distributions, used as proposal

n_x	Banana-shaped target $\pi_2 = \mathcal{B}_{0.1}$								
	Quantile (in %)								
	10	20	30	40	50	60	70	80	90
2	9.60 ± 0.60	19.54 ± 0.74	29.29 ± 1.07	39.52 ± 1.34	49.63 ± 1.58	59.78 ± 1.85	70.14 ± 1.87	80.38 ± 1.65	90.22 ± 1.24
5	9.55 ± 0.75	19.33 ± 1.08	29.05 ± 1.36	39.17 ± 1.67	49.32 ± 1.98	59.42 ± 2.18	69.37 ± 1.97	79.65 ± 1.80	89.93 ± 1.25
7	9.79 ± 0.81	19.49 ± 1.28	29.43 ± 1.56	39.46 ± 2.09	49.51 ± 2.27	59.58 ± 2.18	69.66 ± 2.04	79.87 ± 1.62	90.15 ± 1.18

and we denote hereafter $\mathcal{B}_b(\Sigma)$ the distribution of this random vector, and simply $\mathcal{B}_b$ when Σ is the identity matrix, except for the top left element which is 100. We compare the performance of a mixture of updates based on Algorithm 7 which uses for each of the mixture component either Algorithm 6 or Algorithm 8 with that of the AM algorithm (Haario et al. 1999), for 10,000 iterations for $\mathcal{B}_{0.03}$ and 20,000 iterations for $\mathcal{B}_{0.1}$ and $n = 3$ components for the fitted mixture. The results are summarised in Table 1 seem comparable, although our algorithm seems to be more robust in the difficult situation where $\pi = \mathcal{B}_{0.1}$.

We further tested the ability of the algorithm to properly sample from the target distribution by comparing empirical and exact quantiles. The results and methodology are summarised in Tables 2 and 3.

The fitted mixture makes it possible to estimate the normalising constant of the target distribution π, using the so-called "harmonic mean" estimator, which relies on the identity

$$\int_{\mathsf{X}} \frac{\breve{q}_\theta(x)}{\tilde{\pi}(x)} \pi(x)dx = \frac{1}{\int_{\mathsf{X}} \tilde{\pi}(x)dx} =: 1/Z, \tag{40}$$

where $\tilde{\pi}(x)$ is proportional to $\pi(x)$, but unormalised. Note that $\breve{q}_\theta(x)$ provides us with a potentially reasonable instrumental distribution since it is adapted to fit π and might have thinner tails than π. This is estimator is notoriously known to be unstable whenever the variance of $\breve{q}_\theta(x)/\tilde{\pi}(x)$ under π is large and the suggested approach might in some situations remedy this. In the case of the banana shaped distribution we choose $\tilde{\pi}(x)$ such that

$$Z = \det(2\pi\,\Sigma)^{1/2}.$$

Table 4 Harmonic mean estimator of the normalizing constant of centered spherical Gaussian distributions and banana-shaped distributions $F_{0.03}(X)$, obtained by applying to a Gaussian $\mathcal{N}(0, S)$ with $\mathrm{diag}(S) = [100, 1, \ldots, 1]$ in d dimensions; Z is the partition function's analytical value

n_x	$\pi_1 = \mathcal{N}(0, I_{n_x})$		$\pi_2 = \mathcal{B}_{0.03}$	
	$\hat{Z}$	Z	$\hat{Z}$	Z
2	6.27 ± 0.01	6.28	68.9 ± 15.5	62.831
5	97.53 ± 0.23	98.96	1013.7 ± 178.8	989.577
7	601.72 ± 2.57	621.77	6204.6 ± 1337.6	6217.696

We present results for both $\mathcal{B}_{0.03}$ and $\mathcal{N}(0, I_{n_x})$, based on 50 runs, in Table 4.

For each of them the chain was run with three adaptive Gaussian components for 100,000 iterations. The estimator $\hat{Z}$ was calculated according with the harmonic mean estimator after a burn-in period of 80,000 iterations. The estimation of the Gaussian target's partition function is very accurate. In seven dimensions Z is somewhat underestimated suggesting that the chain was not run long enough to reach its stationary regime. The second target's non-linearity leads to a significant deterioration of the simulation results. While the sample mean is close to the partition function's true value

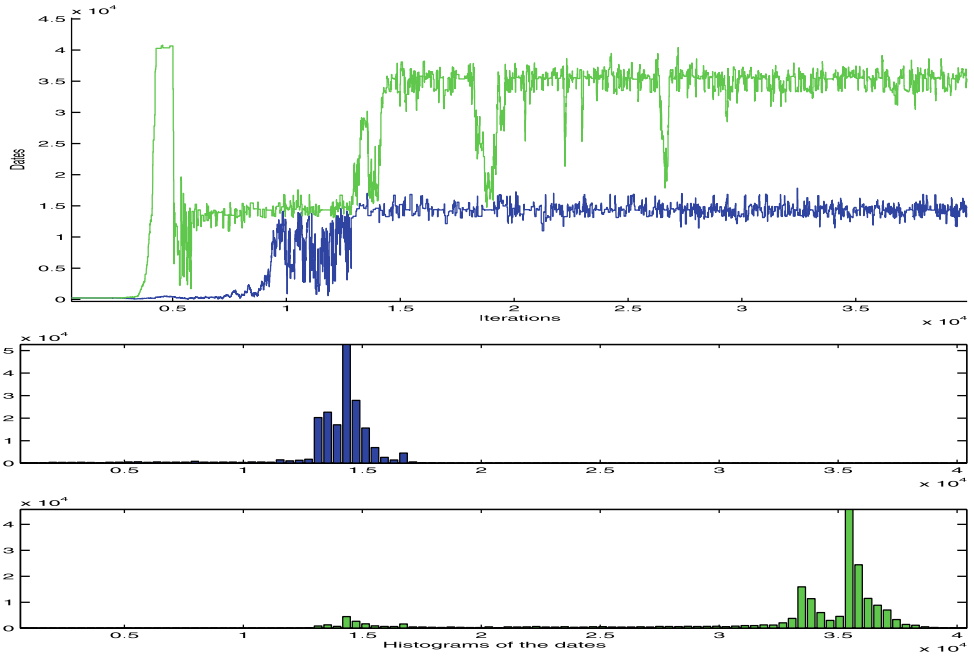

Fig. 4 *Top*: Trace of the positions s_1, s_2 for $k = 2$ corresponding to iterations $1, \ldots, 40{,}000$. *Bottom*: Histograms of the dates for $k = 2$ after 200,000 iterations (with the first 10,000 samples discarded)

the sample deviation is very large. A possible explanation is that the chain has to be run much longer in this setting to ensure convergence of the harmonic mean estimator.

A possible use of this result is that of the estimation of posterior model probabilities.

6.3 Mine disaster data

The dataset of this classic example consists of the recorded dates (in days) $\{y(i)\}$ at which mine disaster have occurred over a period covering 1851–1962. The data is modelled as a Poisson process with intensity $x(t)$ modelled as a step function consisting of $k+1$ plateaux with starting positions $s(0) = 0 < s(1) < s(2) < \cdots < s(k+1) = T$ and heights $h(0), h(1), \ldots, h(k)$ that is

$$x(t) = \sum_{i=1}^{k+1} h(i-1)\mathbb{I}\{s(i-1) \le t < s(i)\}.$$

The unknowns are therefore $k, s := \{s(i)\}$ and $h := \{h(i)\}$. With the priors of Green (1995) the log-posterior distribution, $\log \pi(x)$, is the sum of the three following terms with

$$-\Lambda + k\log(\Lambda) - \log(\Gamma(k+1)) + \log\Gamma(2(k+1)) - (2k+1)\log T,$$

$$(k+1)\,(\alpha\log\beta - \log\Gamma(\alpha)) + (\alpha-1)\sum_{i=1}^{k+1}\log(h(i-1)) - \beta\sum_{i=1}^{k+1}h(i-1) + \sum_{i=1}^{k+1}\log(s(i)-s(i-1)),$$

$$\sum_{i=1}^{k+1}\log h(i-1)\sum_{j=1}^{n}\mathbb{I}\{s(i-1) \le y(j) < s(i)\} - \sum_{i=1}^{k+1}h(i-1)(s(i)-s(i-1)).$$

In our numerical experiments we took $\alpha = 1.0$, $\beta = 200$ and $\Lambda = 3$, which is in line with Green (1995) and Hastie (2005), and simply provided our adaptive algorithm, a combination of the components described in Sect. 5, *i.e.* a mixture of Algorithms 4–6 and 8, with the log-posterior above. We report here the results obtained using one normal component, and did not observe any significant different with 2 or 3 components. One of the difficulty with the posterior distribution of interest is that it involves very different scales and various dependence patterns between the parameters. We ran the algorithm for fixed $k = 1, 2, 3, 4, 5, 6$. In all scenarios the components of h and s were initialised at 1000 and the initial value for the estimate of the covariance matrix of π was set to $10 \times I_{2k+1}$. In order to comment on the behaviour of the adaptive procedure, we primarily focus on the case $k = 2$ in order to maintain the legibility of the various figures. In Figs. 4–6 and 7 we present the traces and in relevant cases histograms for $\{s_i\}$, $\{h_i\}$, $\{\lambda_i\}$ (the scaling coefficient of the global RWM update), the corresponding running expected acceptance probabilities, $\{(\lambda_i^1, \ldots, \lambda_i^{2k+1})\}$ (the scaling coefficients of the componentwise RWM updates) and their corresponding running expected acceptance probabilities. In this case the algorithm was ran for 200,000 iterations. The reported robust behaviour of the algorithm is typical of what we have systematically observed for all the realisations

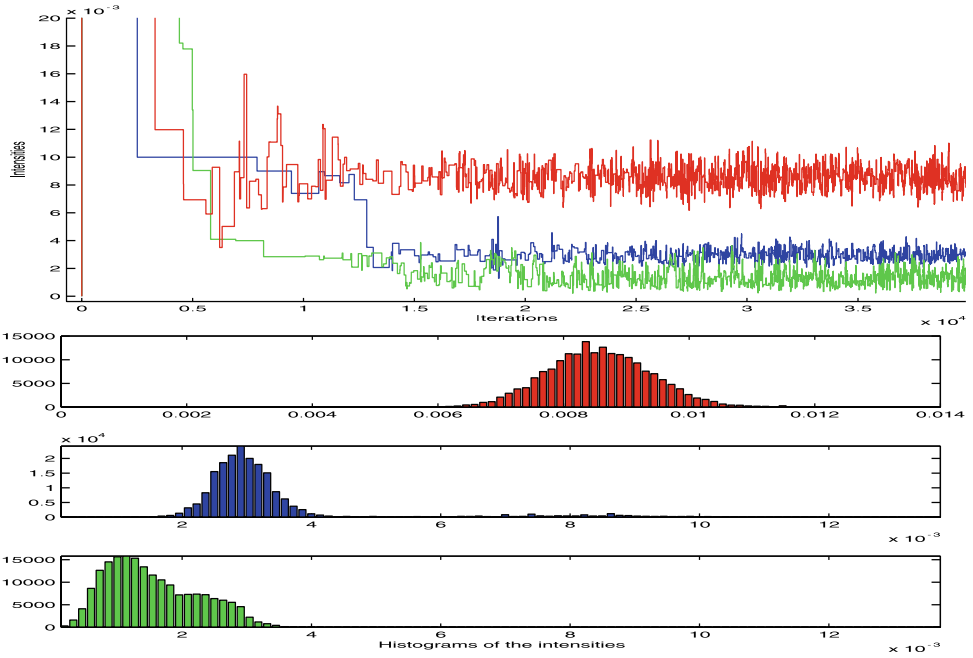

Fig. 5 *Top*: Trace of the intensities h_0, h_1, h_2 for $k = 2$ corresponding to iterations $1, \ldots, 40{,}000$. *Bottom*: Histogram of the intensities for $k = 2$ after 200,000 iterations (with the first 10,000 samples discarded)

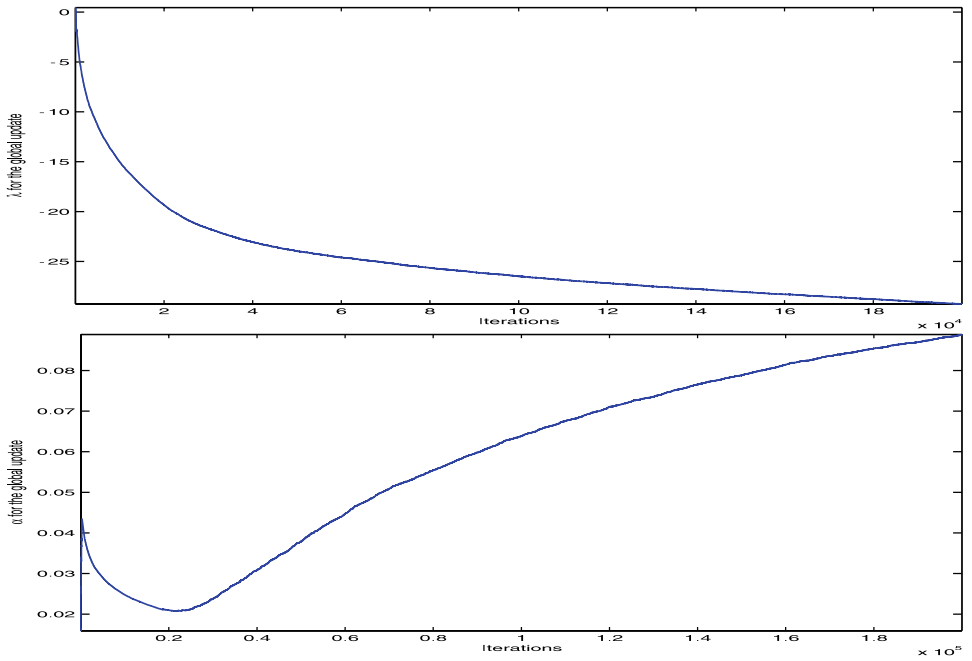

Fig. 6 *Top*: Trace of the "global" RWM update's $\log(\lambda)$. *Bottom*: Running expected acceptance probability of the "global" RWM update

of the algorithm that we have run. Despite poor initialisations of s, h and the parameters of the algorithm (observe in particular the high rejection rate during the first 10,000 iterations particularly visible in Fig. 5) the procedure manages to rapidly recover. The histograms show that our results are in accordance with the results found in Hastie (2005). The behaviour of $\{\lambda_i\}$ and $\{(\lambda_i^1, \ldots, \lambda_i^{2k+1})\}$ and their corresponding running expected acceptance probabilities demonstrate both the interest of adapting these parameters in the initial phase of the algorithm, and the notion of bold and timid moves: small acceptance probabilities prompt the use of smaller scaling factors in order to improve exploration and timid moves seem to improve their performance faster than bold moves (whose expected acceptance probabilities is multiplied by 5 in the course of the first 200,000 iterations). Naturally we observed that not all the parameters of the algorithm seem to have converged, or stabilised around fixed values, whereas the histograms for s and h seem to be in agreement with previously reported results (*e.g.* Hastie 2005). The observed performance of the algorithm is in our

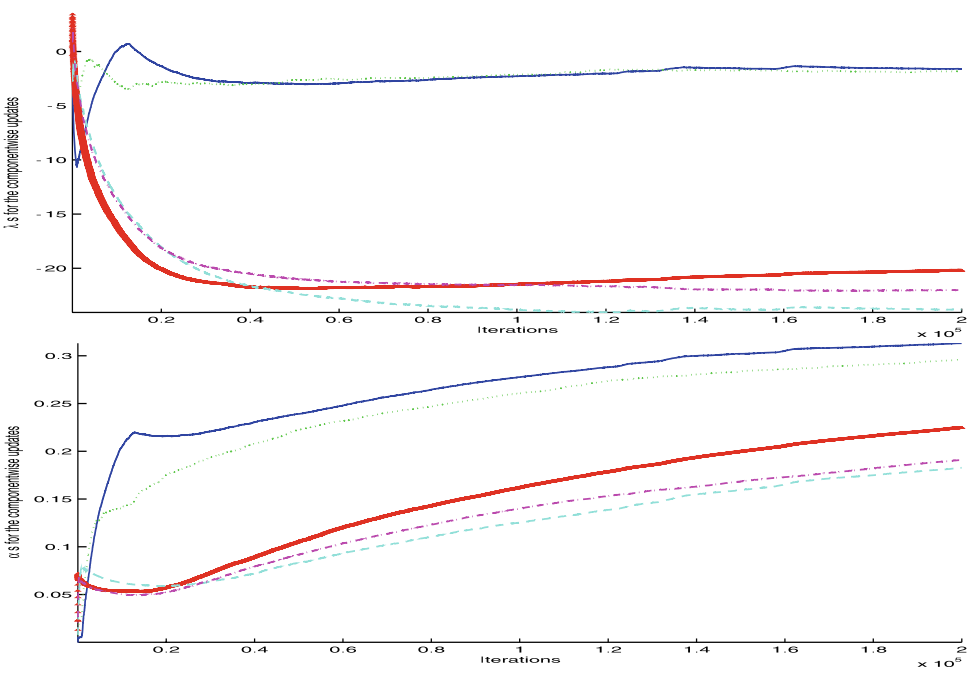

Fig. 7 $k = 2$: *Top*: Trace of the "local" RWM updates' $\log(\lambda)$'s. *Bottom*: Running expected acceptance probability of the "local" RWM updates

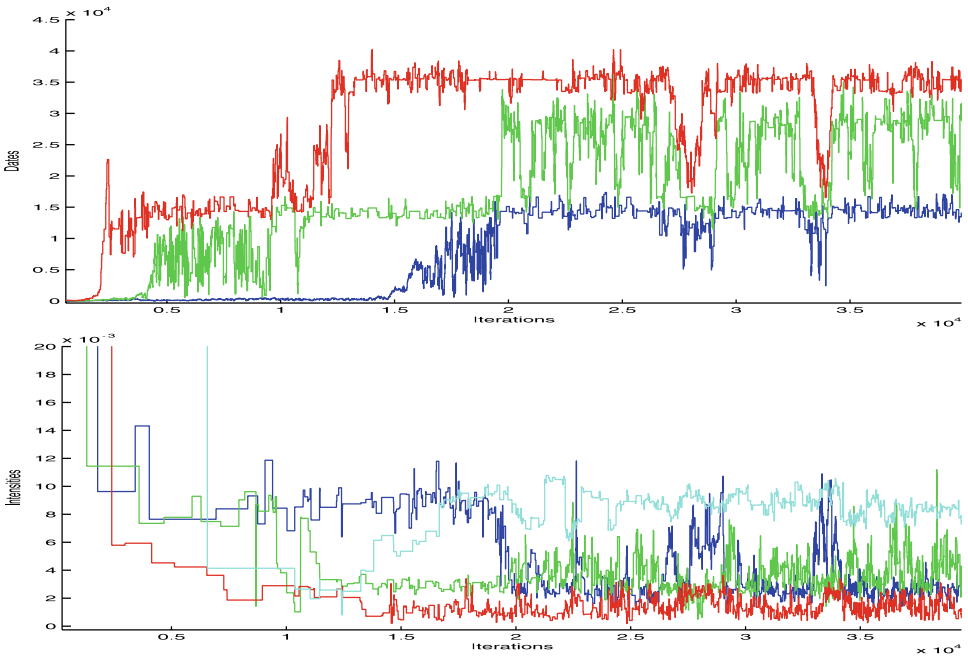

Fig. 8 *Top*: Trace of the positions s_1, s_2, s_3 for $k = 3$ corresponding to iterations $1, \ldots, 40{,}000$. *Bottom*: Trace of the intensities h_0, h_1, h_2, h_3 for $k = 3$ corresponding to iterations $1, \ldots, 40{,}000$

view illustrative of three crucial points discussed earlier in the paper:

1. Vanishing adaptation does not require convergence to ensure that ergodic averages are asymptotically correct,
2. but at the same time the study of the convergence properties of $\{\theta_i\}$ is fundamental since it ensures that this sequence is guaranteed to eventually approach the optimal values defined by our criteria. It is indeed the convergence properties of $\{\theta_i\}$ which explain both the observed good behaviour of $\{\lambda_i\}$ and $\{(\lambda_i^1, \ldots, \lambda_i^{2k+1})\}$ in Figs. 6 and 7. As a result, and provided that we are ready to run the algorithm longer then one can expect to be able to obtain "better" values for the tuning parameter.
3. The user might decide to use this run as a preliminary run to determine a satisfactory tuning parameter θ which can then be used in a standard non-adaptive MCMC algorithm, for which none of the ergodicity problems discussed earlier exist. Effectively, if this is the choice made, this preliminary run is simply an optimisation procedure,

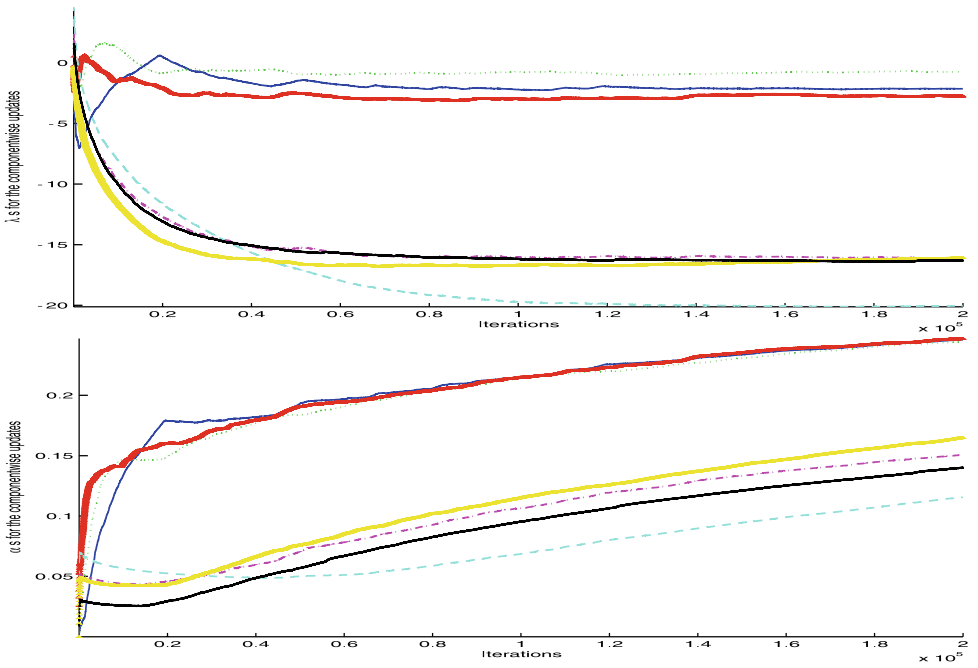

Fig. 9 $k = 3$: *Top*: Trace of the "local" RWM updates' $\log(\lambda)$'s. *Bottom*: Running expected acceptance probability of the "local" RWM updates

which however requires the use of samples at least approximately distributed according to the posterior distribution π, therefore justifying the study of the ergodicity properties of such algorithms.

We report the corresponding results for the case $k = 3$ in Figs. 8–9. Due to the positivity constraints the harmonic mean estimator in (40) cannot be mathematically exact. Despite finding results similar to those of Green (2003) and Hastie (2005) we cannot in this approach as a reliable one.

Acknowledgements The authors are very grateful to the editor and associate editor for their great patience. They would like to thank the reviewers, David Hastie and Arnaud Doucet for very useful comments which have helped to improve the manuscript.

References

Ahn, J.-H., Oh, J.-H.: A constrained EM algorithm for principal component analysis. Neural Comput. **15**, 57–65 (2003)

Andradóttir, S.: A stochastic approximation algorithm with varying bounds. Oper. Res. **43**(6), 1037–1048 (1995)

Andrieu, C.: Discussion of Haario, H., Laine, M., Lehtinen, M., Saksman, E.: Markov chain Monte Carlo methods for high dimensional inversion in remote sensing (December 2003). J. R. Stat. Soc. Ser. B **66**(3), 497–813 (2004)

Andrieu, C., Atchadé, Y.F.: On the efficiency of adaptive MCMC algorithms. Electron. Commun. Probab. **12**, 336–349 (2007)

Andrieu, C., Doucet, A.: Discussion of Brooks, S.P., Giudici, P., Roberts, G.O.: Efficient construction of reversible jump Markov chain Monte Carlo proposal distributions. Part 1. J. R. Stat. Soc. B **65**, 3–55 (2003)

Andrieu, C., Jasra, A.: Efficient and principled implementation of the tempering procedure. Tech. Rep. University of Bristol (2008)

Andrieu, C., Moffa, G.: A Gaussian copula approach for adaptation in discrete scenarios (2008, in preparation)

Andrieu, C., Moulines, É.: On the ergodicity properties of some adaptive MCMC algorithms. Ann. Appl. Probab. **16**(3), 1462–1505 (2006)

Andrieu, C., Robert, C.P.: Controlled MCMC for optimal sampling. Tech. Rep. 0125, Cahiers de Mathématiques du Ceremade, Université Paris-Dauphine (2001)

Andrieu, C., Tadić, V.B.: The boundedness issue for controlled MCMC algorithms. Tech. Rep. University of Bristol (2007)

Andrieu, C., Moulines, É., Priouret, P.: Stability of stochastic approximation under verifiable conditions. SIAM J. Control Optim. **44**(1), 283–312 (2005)

Atchadé, Y.F.: An adaptive version for the Metropolis adjusted Langevin algorithm with a truncated drift. Methodol. Comput. Appl. Probab. **8**, 235–254 (2006)

Atchadé, Y.F., Fort, G.: Limit Theorems for some adaptive MCMC algorithms with subgeometric kernels. Tech. Rep. (2008)

Atchadé, Y.F., Liu, J.S.: The Wang-Landau algorithm in general state spaces: applications and convergence analysis. Technical report Univ. of Michigan (2004)

Atchadé, Y.F., Rosenthal, J.S.: On adaptive Markov chain Monte Carlo algorithms. Bernoulli **11**, 815–828 (2005)

Bai, Y., Roberts, G.O., Rosenthal, J.S.: On the Containment Condition for Adaptive Markov Chain Monte Carlo Algorithms. Tech. Rep. University of Toronto (2008)

Bédard, M.: Optimal acceptance rates for metropolis algorithms: moving beyond 0.234. Tech. Rep. University of Montréal (2006)

Bédard, M.: Weak convergence of metropolis algorithms for non-i.i.d. target distributions. Ann. Appl. Probab. **17**, 1222–1244 (2007)

Bennet, J.E., Racine-Poon, A., Wakefield, J.C.: MCMC for nonlinear hierarchical models. In: MCMC in Practice. Chapman & Hall, London (1996)

Benveniste, A., Métivier, M., Priouret, P.: Adaptive Algorithms and Stochastic Approximations. Springer, Berlin (1990)

Besag, J., Green, P.J.: Spatial statistics and Bayesian computation. J. R. Stat. Soc. Ser. B Stat. Methodol. **55**, 25–37 (1993)

Borkar, V.S.: Topics in Controlled Markov Chains. Longman, Harlow (1990)

Browne, W.J., Draper, D.: Implementation and performance issues in the Bayesian and likelihood fitting of multilevel models. Comput. Stat. **15**, 391–420 (2000)

Cappé, O., Douc, R., Gullin, A., Marin, J.-M., Robert, C.P.: Adaptive Importance Sampling in General Mixture Classes. Preprint (2007)

Ceperley, D., Chester, G.V., Kalos, M.H.: Monte Carlo simulation of a many fermion study. Phys. Rev. B **16**(7), 3081–3099 (1977)

Chauveau, D., Vandekerkhove, P.: Improving convergence of the Hastings-Metropolis algorithm with an adaptive proposal. Scand. J. Statist. **29**(1), 13–29 (2001)

Chen, H.F., Guo, L., Gao, A.J.: Convergence and robustness of the Robbins-Monro algorithm truncated at randomly varying bounds. Stoch. Process. Their Appl. **27**(2), 217–231 (1988)

Chib, S., Greenberg, E., Winkelmann, R.: Posterior simulation and Bayes factors in panel count data models. J. Econ. **86**, 33–54 (1998)

de Freitas, N., Højen-Sørensen, P., Jordan, M., Russell, S.: Variational MCMC. In: Proceedings of the 17th Conference in Uncertainty in Artificial Intelligence, pp. 120–127. Morgan Kaufman, San Mateo (2001). ISBN:1-55860-800-1

Delmas, J.-F., Jourdain, B.: Does waste-recycling really improve Metropolis-Hastings Monte Carlo algorithm? Tech. Rep. Cermics, ENPC (2007)

Delyon, B.: General results on the convergence of stochastic algorithms. IEEE Trans. Automat. Control **41**(9), 1245–1256 (1996)

Delyon, B., Juditsky, A.: Accelerated stochastic approximation. SIAM J. Optim. **3**(4), 868–881 (1993)

Douglas, C.: Simple adaptive algorithms for cholesky, LDL^T, QR, and eigenvalue decompositions of autocorrelation matrices for sensor array data. In: Signals, Systems and Computers, 2001, Conference Record of the Thirty-Fifth Asilomar Conference, vol. 21, pp. 1134–1138 (2001)

Erland, S.: On Adaptivity and Eigen-Decompositions of Markov Chains. Ph.D. thesis Norwegian University of Science and Technology (2003)

Frenkel, D.: Waste-recycling Monte Carlo. In: Computer Simulations In Condensed Matter: from Materials to Chemical Biology. Lecture Notes in Physics, vol. 703, pp. 127–138. Springer, Berlin (2006)

Gåsemyr, J.: On an adaptive Metropolis-Hastings algorithm with independent proposal distribution. Scand. J. Stat. **30**(1), 159–173 (2003). ISSN 0303-6898

Gåsemyr, J., Natvig, B., Nygård, C.S.: An application of adaptive independent chain Metropolis–Hastings algorithms in Bayesian hazard rate estimation. Methodol. Comput. Appl. Probab. **6**(3), 293–302(10) (2004)

Gelfand, A.E., Sahu, S.K.: On Markov chain Monte Carlo acceleration. J. Comput. Graph. Stat. **3**(3), 261–276 (1994)

Gelman, A., Roberts, G., Gilks, W.: Efficient Metropolis jumping rules. In: Bayesian Statistics, vol. 5. Oxford University Press, New York (1995)

Geyer, C.J., Thompson, E.A.: Annealing Markov chain Monte Carlo with applications to ancestral inference. J. Am. Stat. Assoc. **90**, 909–920 (1995)

Ghasemi, A., Sousa, E.S.: An EM-based subspace tracker for wireless communication applications. In: Vehicular Technology Conference. VTC-2005-Fall. IEEE 62nd, pp. 1787–1790 (2005)

Gilks, W.R., Roberts, G.O., George, E.I.: Adaptive direction sampling. The Statistician **43**, 179–189 (1994)

Gilks, W.R., Roberts, G.O., Sahu, S.K.: Adaptive Markov chain Monte Carlo through regeneration. J. Am. Stat. Assoc. **93**, 1045–1054 (1998)

Giordani, P., Kohn, R.: Efficient Bayesian inference for multiple change-point and mixture innovation models. Sveriges Riksbank Working Paper No. 196 (2006)

Green, P.J.: Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. Biometrika **82**, 711–732 (1995)

Green, P.J.: Trans-dimensional Markov chain Monte Carlo. In: Green, P.J., Hjort, N.L., Richardson, S. (eds.) Highly Structured Stochastic Systems. Oxford Statistical Science Series, vol. 27, pp. 179–198. Oxford University Press, London (2003)

Green, P.J., Mira, A.: Delayed rejection in reversible jump Metropolis-Hastings. Biometrica **88**(3) (2001)

Haario, H., Saksman, E., Tamminen, J.: Adaptive proposal distribution for random walk Metropolis algorithm. Comput. Stat. **14**(3), 375–395 (1999)

Haario, H., Saksman, E., Tamminen, J.: An adaptive Metropolis algorithm. Bernoulli **7**(2), 223–242 (2001)

Haario, H., Laine, M., Mira, A., Saksman, E.: DRAM: Efficient adaptive MCMC (2003)

Haario, H., Laine, M., Lehtinen, M., Saksman, E.: Markov chain Monte Carlo methods for high dimensional inversion in remote sensing. J. R. Stat. Soc. Ser. B **66**(3), 591–607 (2004)

Haario, H., Saksman, E., Tamminen, J.: Componentwise adaptation for high dimensional MCMC. Comput. Stat. **20**, 265–274 (2005)

Hastie, D.I.: Towards automatic reversible jump Markov chain Monte Carlo. Ph.D. thesis Bristol University, March 2005

Holden, L.: Adaptive chains. Tech. Rep. Norwegian Computing Center (1998)

Holden, L. et al.: History matching using adaptive chains. Tech. Report Norwegian Computing Center (2002)

Kesten, H.: Accelerated stochastic approximation. Ann. Math. Stat. **29**(1), 41–59 (1958)

Kim, S., Shephard, N., Chib, S.: Stochastic volatility: likelihood inference and comparison with ARCH models. Rev. Econ. Stud. **65**, 361–393 (1998)

Laskey, K.B., Myers, J.: Population Markov chain Monte Carlo. Mach. Learn. **50**(1–2), 175–196 (2003)

Liu, J., Liang, F., Wong, W.H.: The use of multiple-try method and local optimization in Metropolis sampling. J. Am. Stat. Assoc. **95**, 121–134 (2000)

Mykland, P., Tierney, L., Yu, B.: Regeneration in Markov chain samplers. J. Am. Stat. Assoc. **90**, 233–241 (1995)

Nott, D.J., Kohn, R.: Adaptive sampling for Bayesian variable selection. Biometrika **92**(4), 747–763 (2005)

Pasarica, C., Gelman, A.: Adaptively scaling the Metropolis algorithm using the average squared jumped distance. Tech. Rep. Department of Statistics, Columbia University (2003)

Plakhov, A., Cruz, P.: A stochastic approximation algorithm with step-size adaptation. J. Math. Sci. **120**(1), 964–973 (2004)

Ramponi, A.: Stochastic adaptive selection of weights in the simulated tempering algorithm. J. Ital. Stat. Soc. **7**(1), 27–55 (1998)

Robbins, H., Monro, S.: A stochastic approximation method. Ann. Math. Stat. **22**, 400–407 (1951)

Robert, C.P., Casella, G.: Monte Carlo Statistical Methods. Springer, Berlin (1999)

Roberts, G.O., Rosenthal, J.: Optimal scaling of discrete approximation to Langevin diffusion. J. R. Stat. Soc. B **60**, 255–268 (1998)

Roberts, G.O., Rosenthal, J.S.: Examples of adaptive MCMC. Technical Report University of Toronto (2006)

Roberts, G.O., Rosenthal, J.S.: Coupling and ergodicity of adaptive MCMC. J. Appl. Probab. **44**(2), 458–475 (2007)

Roberts, G.O., Gelman, A., Gilks, W.: Weak convergence and optimal scaling of random walk Metropolis algorithms. Ann. Appl. Probab. **7**, 110–120 (1997)

Roweis, S.: EM algorithms for PCA and SPCA. Neural Inf. Process. Syst. **10**, 626–632 (1997)

Sahu, S.K., Zhigljavsky, A.A.: Adaptation for self regenerative MCMC. Available from http://www.maths.soton.ac.uk/staff/Sahu/research/papers/self.html

Saksman, E., Vihola, M.: On the ergodicity of the adaptive Metropolis algorithm on unbounded domains (2008). arXiv:0806.2933

Sherlock, C., Roberts, G.O.: Optimal scaling of the random walk Metropolis on elliptically symmetric unimodal targets. Tech. Rep. University of Lancaster (2006)

Sims, C.A.: Adaptive Metropolis-Hastings algorithm or Monte Carlo kernel estimation. Tech. report Princeton University (1998)

Spall, J.C.: Adaptive stochastic approximation by the simultaneous perturbation method. IEEE Trans. Automat. Control **45**(10), 1839–1853 (2000)

Stramer, O., Tweedie, R.L.: Langevin-type models II: self-targeting candidates for MCMC algorithms. Methodol. Comput. Appl. Probab. **1**(3), 307–328 (1999)

Sutton, R.S., Barto, A.G.: Reinforcement Learning: An Introduction. MIT Press, Cambridge (1998)

Tierney, L., Mira, A.: Some adaptive Monte Carlo methods for Bayesian inference. Stat. Med. **18**, 2507–2515 (1999)

Tipping, M.E., Bishop, C.M.: Probabilistic principal component analysis. J. R. Stat. Soc. Ser. B Stat. Methodol. **61**, 611–622 (1999)

Winkler, G.: Image Analysis, Random Fields and Markov Chain Monte Carlo Methods: A Mathematical Introduction. Stochastic Modelling and Applied Probability. Springer, Berlin (2003)