Submitted to Bernoulli

An adaptive multiple-try Metropolis algorithm

SIMON FONTAINE[1,2,*] and MYLÈNE BÉDARD[1]

[1] *Université de Montréal, Département de mathématiques et de statistique, 2920 chemin de la Tour, Montréal, QC, Canada, H3T 1J4.*

[2] *University of Michigan, Department of Statistics. West Hall, 1085 South University, Ann Arbor, MI, U.S.A., 48109. E-mail:* [*] simfont@umich.edu

Markov chain Monte Carlo (MCMC) methods, specifically samplers based on random walks, often have difficulty handling target distributions with complex geometry such as multi-modality. We propose an adaptive multiple-try Metropolis algorithm designed to tackle such problems by combining the flexibility of multiple-proposal samplers with the user-friendliness and optimality of adaptive algorithms. We prove the ergodicity of the resulting Markov chain with respect to the target distribution using common techniques in the adaptive MCMC literature. In a Bayesian model for loss of heterozygosity in cancer cells, we find that our method outperforms traditional adaptive samplers, non-adaptive multiple-try Metropolis samplers, and various more sophisticated competing methods.

Keywords: adaptive scaling, ergodicity, limit theorem, loss of heterozygosity, multiple candidates, random walk sampler, robust adaptation.

MSC 2010 subject classifications: Primary 60J22, Secondary 60J10, 60J20, 65C40, 62F15.

1. Introduction

Suppose we wish to find an expectation of the form $\pi(f) = \mathbb{E}_{X\sim\pi}\{f(X)\}$ for some π-integrable function $f : \mathcal{X} \to \mathbb{R}^q$ on some state space $\mathcal{X} \subseteq \mathbb{R}^d$. Monte Carlo (MC) methods make use of a law of large numbers, i.e.

$$\frac{1}{N}\sum_{n=1}^{N} f(X_n) \xrightarrow{\mathcal{C}} \pi(f)\,, \qquad n \to \infty\,, \tag{1.1}$$

for some mode of convergence $\mathcal{C} \in \{\text{in probability},\ \text{almost surely}\}$, to estimate $\pi(f)$ using the sample average. The conditions under which (1.1) holds rely on the joint distribution of $(X_n)_{n=1}^N$. For example, an iid assumption ($X_n \overset{\text{iid}}{\sim} \pi$, $n = 1, \ldots, N$) is often sufficient to verify a law of large numbers. When π is even moderately complex however, it is generally impossible to sample directly from that distribution. Markov chain Monte Carlo (MCMC) methods provide a way to produce a sample $(X_n)_{n=1}^N$ that verifies a law of large numbers for some class of functions f while not requiring direct sampling from the target distribution π.

One of the most common MCMC methods is the Metropolis-Hastings (MH) algorithm (**?**), which produces the sample $(X_n)_{n=1}^N$ sequentially from time $n = 1$ through $n = N$. At time n, instead of sampling directly from π, we sample from a proposal distribution $q(\cdot|x)$ which may or may not depend on the previous sample point $X_{n-1} = x$. Since the candidate $Y \sim q(\cdot|x)$ is sampled from q and not from π, we need to proceed to an accept/reject step to adjust for that bias. The next point X_n to be included in the sample is chosen to be equal to y with probability

$$\alpha_{\mathrm{MH}}(y|x) = \min\left\{1, \frac{\pi(y)q(x|y)}{\pi(x)q(y|x)}\right\} , \tag{1.2}$$

known as the *MH acceptance probability*, and equal to the previous observation x with probability $1-\alpha_{\mathrm{MH}}(y|x)$. Now, since the distribution of X_n only depends on the previous time point $X_{n-1} = x$, then $\mathbf{X} = (X_n)_{n\geqslant 1}$ actually forms a Markov chain.

A homogeneous Markov chain $\mathbf{X}$ on a state space $\mathcal{X}$ with Markov transition P is ergodic with respect to some distribution Π for some initial state $x \in \mathcal{X}$ if

$$\lim_{n\to\infty} \|P^n(\cdot|x) - \Pi(\cdot)\|_{\mathrm{TV}} = 0 , \tag{1.3}$$

where $\|\mu\|_{\mathrm{TV}} = \sup_{B\in\mathcal{B}(\mathcal{X})} |\mu(B)|$ denotes the total variation norm of the signed measure μ, and where

$$P^m(B|x) = \int_{\mathcal{X}} P^{m-1}(B|y)\, P(\mathrm{d}y|x) , \qquad m > 1$$

is the *iterated* Markov transition with base case $P^1(B|x) = P(B|x)$. Typically, the ergodicity of a homogeneous Markov chain with respect to a density π is established through results such as in **?**, Theorem 1 where it is required that (1) π be the stationary density of the Markov transition P with density p, (2) the chain be π-irreducible, and (3) the chain be aperiodic. While ergodicity and laws of large numbers are two different concepts, the conditions used to verify the former are sufficient to verify the latter for all π-integrable functions (**?**).

A sufficient condition for $\mathbf{X}$ to admit π as its stationary distribution is the *detailed balance* condition on the densities (**?**),

$$p(y|x)\,\pi(x) = p(x|y)\,\pi(y) , \qquad \forall\, x, y \in \mathcal{X} . \tag{1.4}$$

By construction, the chain $\mathbf{X}$ generated using a MH algorithm satisfies the first ergodicity condition since the expression of the MH acceptance probability (1.2) is specifically chosen to satisfy (1.4). A sufficient condition for aperiodicity and π-irreducibility of MH chains is the local positivity of the proposal density q,

$$\|x - y\|_2 < \delta \quad \Rightarrow \quad q(y|x) > \varepsilon , \tag{1.5}$$

for some $\delta, \varepsilon > 0$ ($\|\cdot\|_2$ is the Euclidean norm), together with the assumption that π is bounded above and away from 0 on any compact subset of the state space $\mathcal{X}$ (**?**). This

type of condition can easily be verified for the Metropolis algorithm (**?**), a special case of the MH algorithm that uses a symmetric random-walk proposal density,

$$q(y|x) = q(y-x) = q(x-y) \ .$$

Since its initial development, the MH algorithm—and most notably the Metropolis sampler—has seen multiple proposed improvements, of which we consider two here. The *Multiple-try Metropolis* (MTM) algorithm defines a variant of the Metropolis sampler where several candidates are generated in a given iteration; this technique produces a transition that is better adapted to the specific geometry of the target density, leading to an improved state space exploration. The *Adaptive Metropolis* (AM) algorithm, on the other hand, uses a random-walk proposal density at each iteration but adapts it through time to match the covariance of the target, therefore producing higher-quality candidates.

Both algorithms improve on the vanilla Metropolis sampler, but each suffers from the exact problem that the other algorithm aims at solving. The MTM sampler requires a large amount of hand-tuning that adaptive algorithms perform automatically; the AM algorithm typically uses simple proposal densities that may not be well-suited to target densities featuring complex geometries, such a multi-modal densities. In this article, we propose a novel adaptive MCMC algorithm that unites the advantages of these two samplers and therefore fixes some of their respective flaws.

Related work We review some of the recent attempts at integrating adaptation within the multiple-try framework. **?** propose the *adaptive independant sticky multiple-try Metropolis*, which uses a non-parametric independent proposal density. Multiple candidates are sampled from this non-parametric density adapted using rejected points. Being non-parametric, this method does not extend well beyond a few dimensions for full-dimensional samplers. **?** propose the *interacting multiple-try Metropolis* in which multiple parallel MTM chains interact with each other and MTM selection weights are adapted using all chains. **?** propose an *adaptive component-wise multiple-try Metropolis* algorithm that consists of a multiple-try generalization of the Metroplis-within-Gibbs where one-dimensional proposals are adapted using MTM selection proportions of the chain's past. **?** briefly mention that their proposed *adaptive correlated Metropolis-Hastings* algorithm could be extended to include multiple candidates.

Principal contributions The main contribution of this research is the proposed aMTM algorithm, which consists of an adaptive MCMC sampler with full-dimensional adapted multiple-try Metropolis proposal densities. Introducing adaptation in MTM algorithms is a natural extension to the current literature; it is surprising that nothing has yet been published on the subject. We derive an ergodicity result under the assumption that both the sample and parameter spaces are bounded. Intermediary theoretical results are readily extendable to other adaptive MCMC algorithms and provide slightly more general knowledge about MTM transitions. We provide an implementation of our proposed algorithm in a `R` package called `aMTM` available at https://github.com/fontaine618/aMTM, which mostly consists of a wrapper for the main sampling function written in `C++`.

Paper organization In Section 2, we introduce some background on MTM and adaptive algorithms. Section 3 contains a general description of our proposed algorithm along with some variants. The validity of our sampler is discussed in Section 4, where ergodicity is proven. We conduct simulation experiments in Section 5 to assess the performance of our approach. Supplement A contains additional details on particular variants of the algorithms, results and proofs omitted from the main text.

2. Background

2.1. Multiple-try Metropolis

A natural extension to the MH algorithm is to consider K candidate points per iteration instead of a single one (**?**). The resulting *multiple-try Metropolis* sampler must therefore include an additional step that randomly selects a proposal among the set of K candidates $Y^{(1:K)} \sim q(\cdot|x)$ according to some positive sampling weight function $w^{(k)}(\cdot|x)$, which may depend on the index k of the candidate and on the previous state x of the chain. Throughout the text, exponents in parentheses $^{(k)}$ index candidates with the convention that $^{(1:K)}$ selects all candidates while $^{(-k)}$ omits the k-th candidate. Standardizing these weights, we obtain the probability of choosing $y^{(k)} = y$ as the official candidate:

$$\bar{w}^{(k)}(y, y^{(-k)}|x) = \frac{w^{(k)}(y^{(k)}|x)}{\sum_{j=1}^{K} w^{(j)}(y^{(j)}|x)} \, .$$

Once an official candidate $k \in \{1, \ldots, K\}$ is selected, the proposed value $y^{(k)} = y$ must go through an accept/reject step in order to become the next state of the chain. As before, the acceptance probability is chosen such as to satisfy the detailed balance condition (1.4), which basically requires that the trajectory produced by a stationary Markov chain be equally probable when run forward or backward in time. To satisfy this condition, we thus need to generate a shadow sample $x_*^{(j)}$, $j = 1, \ldots, K$, that mimics the generation of a candidate set, and then select the official candidate x for going from y to x (instead of from x to y). That is, we let $X_*^{(k)} = x$ and sample $X_*^{(-k)} \sim q^{(-k)}(\cdot|y)$, where $q^{(-k)}$ is the conditional proposal distribution given the k-th component $X_*^{(k)} = x$. Using the following *MTM acceptance probability*

$$\alpha_{\text{MTM}}\left(y, y^{(-k)}|x, x_*^{(-k)}\right) = \min\left\{1, \frac{\pi(y)q^{(k)}(x|y)\bar{w}^{(k)}(x, x_*^{(-k)}|y)}{\pi(x)q^{(k)}(y|x)\bar{w}^{(k)}(y, y^{(-k)}|x)}\right\} \tag{2.1}$$

is then sufficient to verify (1.4), assuming that the marginal density of the k-th candidate, $q^{(k)}(\cdot|\cdot)$, satisfies

$$q^{(k)}(x|y) > 0 \qquad \Leftrightarrow \qquad q^{(k)}(y|x) > 0 \, , \tag{2.2}$$

for all $k = 1, \ldots, K$ (see Supplement A, Proposition 3.1).

The MTM design and the detailed balance condition's verification do not impose any restriction on the *joint* distribution of candidates, only on their *marginal* distributions. The set of candidates can thus be generated in any way and choosing appropriate correlation structures within the candidate set can greatly improve the algorithm's performance. The simplest choice is to generate candidates independently, but this does not make use of the sampler's full potential. Indeed, nothing prevents two candidates from being very close to one another, which does not improve the state space exploration.

Extremely antithetic (EA) candidates are generated so that their pairwise Euclidean distances be maximized. **?**, Section 3.1 achieve this by introducing a correlation of $\rho = -1/(K-1)$ between the K candidates. For example, if marginal Gaussian proposal distributions with unit spherical covariances I_d are used, this yields the joint covariance matrix

$$\mathrm{Var}\left(Y^{(1:K)}\right) = \begin{pmatrix} I_d & \cdots & \rho I_d \\ \vdots & \ddots & \vdots \\ \rho I_d & \cdots & I_d \end{pmatrix} \in \mathbb{R}^{dK \times dK} . \tag{2.3}$$

To produce candidates using different covariance matrices, we can simply generate K d-dimensional Gaussian observations using (2.3) and transform them using a Cholesky decomposition. To produce the shadow sample, we need to compute the conditional distribution of $Y^{(-k)}$ given $Y^{(k)} = y^{(k)}$. In the unit spherical covariance case, we can show (**?**, Section 5.3.4.2) that this corresponds to a Gaussian distribution with some specific mean and the following joint covariance,

$$\mathrm{Var}\left(Y^{(-k)}|y^{(k)}\right) = (1-\rho) \begin{pmatrix} (1+\rho)I_d & \cdots & \rho I_d \\ \vdots & \ddots & \vdots \\ \rho I_d & \cdots & (1+\rho)I_d \end{pmatrix} \in \mathbb{R}^{d(K-1)\times d(K-1)} .$$

Randomized quasi-Monte Carlo (RQMC) methods are constructed using a (random) regularly-spaced grid on the unit hyper-cube before going through a probability integral transform. For example, **?**, Section 3.2 construct such a grid using a Korobov rule. The RQMC candidates were generalized by **?** to *common random number* candidates, where the regular grid assumption is removed. We refer the reader to Supplement A, Section 1.1, for a summary on how to perform these various sampling schemes with a multivariate Gaussian random walk proposal density.

The detailed balance condition (1.4) only requires that weight functions $w^{(k)}$ be positive everywhere. Users therefore have the freedom to choose functions that favor some particular behaviour. To encourage large jumps for instance, $w^{(k)}$ could be chosen to contain the factor $\|y-x\|_2$ (see, e.g., **?**). We refer the reader to **?** for an extensive study of different weight functions. The two most common choices—and the ones that seem to perform best empirically—are importance weights,

$$w^{(k)}_{\mathrm{imp}}(y|x) = \frac{\pi(y)}{q^{(k)}(y|x)} , \tag{2.4}$$

and weights proportional to the target density,

$$w_{\text{prop}}^{(k)}(y|x) = \pi(y) \ . \tag{2.5}$$

The ergodicity of MTM chains can be verified using the same conditions as for the Metropolis algorithm. The assumption that (1.5) holds for each proposal density $q^{(k)}$, $k = 1, \ldots, K$, is sufficient to ensure π-irreducibility and aperiodicicity as long as both π and $w^{(k)}(\cdot|x)$ are bounded above and below on any compact subset of $\mathcal{X}$ for each k and for each fixed x (see Supplement A, Proposition 3.2). The same conditions are also sufficient to establish a strong law of large numbers for all π-integrable functions.

Algorithm 1 summarizes the MTM sampler in its most general form—the joint proposal density and weight functions are left completely free.

Algorithm 1 Multiple-try Metropolis (MTM)

Input Target density π with support $\mathcal{X} \subseteq \mathbb{R}^d$, MC sample size N, joint proposal distribution q with marginals $q^{(k)}$ and conditionals $q^{(-k)}$, $k = 1, \ldots, K$, weight functions w^k, $k = 1, \ldots, K$.

Procedure

1. *Initialization.* Initialize the state to $x_0 \in \mathcal{X}$.
2. *MCMC iteration.* For $n = 0, \ldots, N-1$, do:
   (a) *Candidates generation.* Sample $y^{(1:K)} \sim q(\cdot|x_n)$;
   (b) *Weights.* Compute $w^{(k)}(y^{(k)}|x_n)$, $k = 1, \ldots, K$;
   (c) *Proposal selection.* Sample $k \in \{1, \ldots, K\}$ with probability proportional to weights $w^{(k)}$, $k = 1, \ldots, K$, and set $y = y^{(k)}$;
   (d) *Shadow sample.* Sample $x_*^{(-k)} \sim q^{(-k)}(\cdot|y^{(k)}, x_n)$ and set $x_*^{(k)} = x_n$;
   (e) *Reverse weights.* Compute $w^{(k)}(x_*^{(k)}|y)$, $k = 1, \ldots, K$;
   (f) *Acceptance probability.* Compute $\alpha_{\text{MTM}}(y, y^{(-k)}|x, x_*^{(-k)})$ using (2.1);
   (g) *Acceptance.* Accept the proposal $(x_{n+1} = y)$ with probability α_{MTM}; otherwise reject the proposal $(x_{n+1} = x_n)$.

Output The MC sample $\{x_n\}_{n=1}^N$.

2.2. Optimal scaling of MCMC

While a law of large numbers (1.1) guarantees that the sample average converges toward the desired expected value, it does not provide any insight about the estimation error for finite samples. The advantage of *central limit theorems* (CLTs) is that they provide information about the asymptotic distribution of Monte Carlo estimates. A Markov chain $\mathbf{X} \subseteq \mathcal{X}$ satisfies a CLT for a function f if there exists a constant $\sigma_f^2 < \infty$, known as the *asymptotic variance*, such that

$$\sqrt{N}\left[\frac{1}{N}\sum_{n=1}^{N} f(X_n) - \pi(f)\right] \xrightarrow{\mathcal{D}} \mathcal{N}\left(0, \sigma_f^2\right) \ , \qquad N \to \infty \ , \tag{2.6}$$

MTM optimal acceptance probability

	Number of candidates (K)				
Sampling scheme	1	2	3	4	5
Independent	0.23	0.32	0.37	0.39	0.41
EA	0.23	0.46	0.52	0.54	0.55

Table 1. Optimal acceptance probability of the MTM with spherical multivariate Gaussian candidates and weights proportional to π, a target with iid components (**?**).

where $\xrightarrow{\mathcal{D}}$ denotes convergence in distribution. In MCMC contexts, CLTs are often used to produce Monte Carlo standard errors by applying the asymptotic result to a finite sample. Replacing convergence with approximation, (2.6) suggests, for large N,

$$\frac{1}{N}\sum_{n=1}^{N} f\left(X_n\right) \approx \mathcal{N}\left(\pi(f), \sigma_f^2/N\right) \ ,$$

so we may use an estimate of $\sigma_f/\sqrt{N}$ as the Monte Carlo standard error.

Now, if we fix the function of interest f and the target density π, we find that different Markov transitions yield different values for the asymptotic variance σ_f^2, hence different Monte Carlo estimation precisions. Therefore, choosing a transition producing a small—ideally the smallest—asymptotic variance is an important aspect of MCMC theory.

We note that the transition of Metropolis-Hastings samplers is entirely defined by the proposal density q. In the simple case of iid targets and Metropolis proposal densities with spherical Gaussian steps, **?** showed that, asymptotically as the dimension $d \to \infty$, the optimal scaling of the proposal variance $\sigma_d^2 I_d$ is given by $\sigma_d^2 = (2.38)^2/d$ and is associated to an optimal acceptance probability of 0.234. Similar results were eventually obtained for more general target densities (**????????**), for other algorithms (**?????**), and also for finite-dimensional targets (**??**). These results provide guidelines for MCMC users to choose near-optimal proposal densities.

? studied MTM samplers and obtained asymptotically optimal scaling results ($d \to \infty$) for each fixed number of candidates $K = 1, \ldots, 5$. They considered various sampling schemes, including independent and EA candidates; their results, partially summarized in Table 1, apply to iid targets with spherical multivariate Gaussian candidates and weights proportional to the target. As K increases, we notice a growth in the optimal acceptance probability, which indicates that the chain has access to higher-quality selected proposals. Extremely antithetic candidates lead to acceptance rates that are significantly larger than those of independent candidates, meaning that an adequate correlation structure may substantially improves the quality of the selected proposal.

2.3. Adaptive MCMC

Optimal scaling results such as those presented in Section 2.2 implicitly require that the covariance of the target distribution be known. Indeed, the Gaussian random walk

proposal uses a covariance that should be a multiple of the true covariance. In practice, the true covariance is never known—recall that we wish to compute some expectation $\pi(f)$ and that variance is the special case $f = (I - \pi(I))^2$, $I(x) = x$—so it is a dubious assumption to make.

To use optimal scaling results without knowledge of the true covariance, **?** propose the *adaptive Metropolis* sampler, which learns the true covariance over time using the growing sample. Suppose that the time-n estimate of the covariance is Σ_n. A Metropolis iteration is performed using $s_d\Sigma_n$ as the proposal variance of the Gaussian random walk, where $s_d = (2.38)^2/d$ (from optimal scaling results). Once x_{n+1} is selected as either the proposal y or the previous state x_n, we define Σ_{n+1} as the empirical covariance of the sample $(x_i)_{i=1}^{n+1}$ (to which a small multiple of the identity matrix is added to ensure non-singularity). Simple recursions allow the computation of Σ_{n+1} from Σ_n without much work so this extra step is computationally cheap.

Since transitions change at every iteration, the chain is no longer homogeneous. The ergodicity property (1.3) is not defined properly for inhomogeneous chains so we require different definitions for studying the convergence of adaptive MCMC: we will use the setting in **?**, Section 2. Furthermore, since the proposal variance depends on all past samples, the Markovian property of the chain is destroyed. Still, under some mild conditions, **?** verified that AM chains satisfy a strong law of large numbers for bounded functions. The same conditions also imply convergence of the adaptive proposal covariance to the true target covariance (up to the small identity matrix added). **?**, see also **?** later weakened the sufficient conditions for a strong law of large numbers, for an expanded class of functions, and provided a central limit theorem.

Following the work of **?**, **?** developed a simple yet powerful framework to study the ergodicity of adaptive MCMC in a more general setting. They showed that two main conditions are sufficient to verify both the ergodicity of adaptive chains and a weak law of large numbers. To state these sufficient conditions, let $\mathcal{Y}$ be some indexing set for the family of possible transitions $\{P_\gamma : \gamma \in \mathcal{Y}\}$ and let Γ_n denote the (random) index of the chain's transition at time $n \geqslant 0$. When the transitions are all within the same parametric family, the indexing set corresponds to the parameter space where the Γ_n's lie. For example, the indexing of the AM transition may be performed using the adapted covariance Σ_n.

The first condition, termed *diminishing adaptation* (DA), requires that subsequent transitions change less, in probability, as the chain progresses:

$$\operatorname{dist}(P_{\Gamma_n}, P_{\Gamma_{n+1}}) \xrightarrow{\mathbb{P}} 0 \ , \qquad n \to \infty \ , \tag{2.7}$$

where $\xrightarrow{\mathbb{P}}$ denotes convergence in probability and where

$$\operatorname{dist}(P_{\gamma_n}, P_{\gamma_{n+1}}) := \sup_{x \in \mathcal{X}} \left\| P_{\gamma_n}(\cdot|x) - P_{\gamma_{n+1}}(\cdot|x) \right\|_{\mathrm{TV}}$$

is the distance between two consecutive transitions. In particular, convergence of transitions is not a necessary condition, even though it is often satisfied and desired.

The second condition, coined *bounded convergence* (BC, also called *containment*), states that all transitions are individually ergodic with respect to the target density and that their convergence rates do not degenerate, at least in probability. For all $\varepsilon > 0$, the process $\{M_\varepsilon(X_n, \Gamma_n)\}_{n \geqslant 0}$ is bounded in probability conditionally on the initial state $X_0 = x_*$ and initial transition index $\Gamma_0 = \gamma_*$, where

$$M_\varepsilon(x, \gamma) := \inf_m \left\{ m \geqslant 1 : \left\| P_\gamma^m(\cdot|x) - \pi(\cdot) \right\|_{\mathrm{TV}} \right\} \tag{2.8}$$

is the ε-convergence time of the homogeneous chain using transition P_γ with parameter $\gamma \in \mathcal{Y}$ and starting at $x \in \mathcal{X}$. Following the work of **?**, most adaptive MCMC algorithms have their ergodicity verified using DA and BC, or some derivatives of these conditions.

3. Description of the algorithm

3.1. Motivation

Before introducing our proposed algorithm, we consider a toy example that exhibits some of the shortcomings of the AM and MTM samplers taken separately. Let π be a two-dimensional mixture of two Gaussian densities with weights $w_1 = 0.3$ and $w_2 = 0.7$, with means $\mu_1 = (20, 0)^\top$ and $\mu_2 = (0, 8)^\top$, and with covariance matrices $\Sigma_1 = \mathrm{diag}(9, 1)$ and $\Sigma_2 = \mathrm{diag}(1, 9)$. An iid sample of size $N = 10,000$ from that density may be found in Figure 1(a).

Multimodal densities are notoriously hard to sample using simple algorithms. Indeed, two types of moves are required to adequately explore the whole support of such distributions: local moves to explore a given mode and global ones to jump between modes. A single proposal density generally cannot do both efficiently because of the different scales on which they lie (see, e.g., a Metropolis sampler in Figure 1(b) featuring a very low acceptance rate). Furthermore, using adaptation to find an optimal proposal covariance matrix leads to the optimization of a single type of moves. Depending on the initialization, the AM algorithm will either converge to the covariance of one mode (Figure 1(c)) or to the global covariance of the target (Figure 1(d)). While the latter yields decent results, the acceptance rate is still small for a two-dimensional target density.

MTM samplers are better suited for multimodal densities because the different proposal densities can model various types of jumps. When proposal densities are well adjusted to the target, the resulting chain may offer good performances (e.g., Figure 1(e)). The tuning of proposal densities must however be done by hand, which rapidly becomes impractical with increasing dimensions.

Hence, adaptation of the MTM's proposal densities could bring the best of both worlds together: automatic tuning of the proposal distributions and better fit to the target's distinct characteristics. Our proposed method, whose description follows, can achieve improved performance with minimal tuning (see Figure 1(f)).

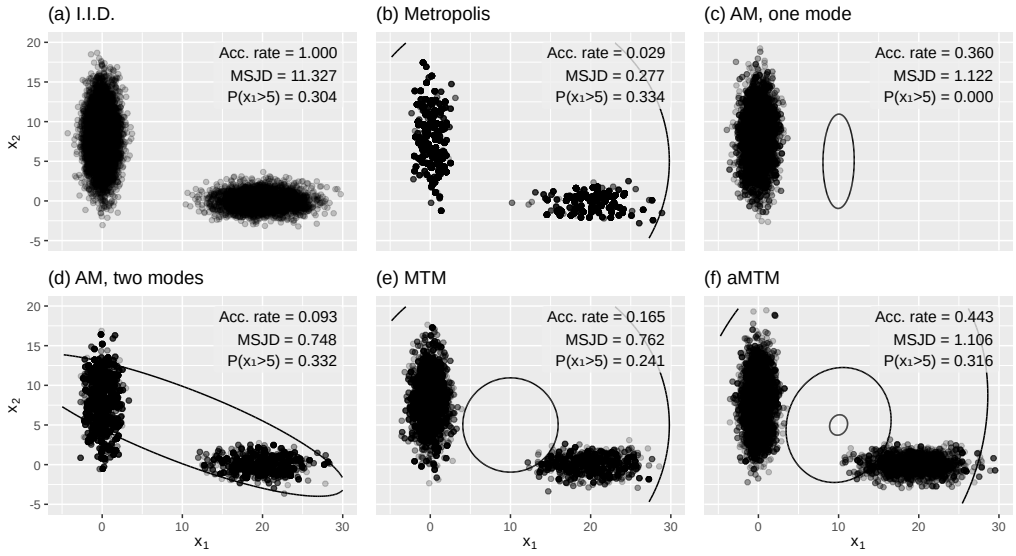

Figure 1. Samples for the bimodal density of Section 3.1 obtained using different samplers and parameters. Ellipses show the final proposal covariance(s); $P(x_1 > 5)$ is the proportion of points with $x_1 > 5$; MSJD is the mean squared jumping distance.

3.2. General algorithm

Adaptive MCMC algorithms are essentially defined by two components: a family of transitions and a way to move from one transition to another. The results of **?** suggest a way to construct ergodic adaptive MCMC algorithms: we define both components so that the DA and BC conditions be easy to verify. The BC condition is mostly related to the family of transitions while the DA condition is related to the agreement between successive transitions.

With that in mind, we propose to equip the MTM sampler with adaptation in the following way. The family of transitions is chosen to be MTM transitions in which candidates are generated using Gaussian random walks with a fixed, common correlation structure between the candidates and a fixed weight function. Keeping the correlation structure and weight function fixed will help verifying the DA condition since switching between correlation structures or weight functions would create discontinuous jumps between transitions. We will consider the four correlation structures described in Section 2.1 (independent, EA, RQMC, common random variable) as well as the weight functions (2.4) and (2.5). Algorithm 2 contains an abstract description of our proposed algorithm with arbitrary adaptation.

Algorithm 2 Adaptive multiple-try Metropolis (aMTM)

Input Target density π, MC sample size N, adaptation procedure $q_n \mapsto q_{n+1}$ inside some family of proposal densities $\mathcal{Q}$, weight functions w^k, $k = 1, \ldots, K$.

Procedure

1. *Initialization.* Initialize the state to $x_0 \in \mathbb{R}^d$ and the joint proposal density to $q_0 \in \mathcal{Q}$.
2. *MCMC iteration.* For $n = 0, \ldots, N-1$, do:
   (a) *MTM sampling.* Generate x_{n+1} from x_n using the current joint proposal distribution q_n and the weight functions w^k, $k = 1, \ldots, K$, and according to one MTM sampling iteration (Algorithm 1, Step 2).
   (b) *Adaptation.* Update q_n to q_{n+1} according to the specified adaptation procedure.

Output The MC sample $\{x_n\}_{n=1}^N$.

3.3. Adaptation variants

When the family of proposal densities is chosen to be multivariate Gaussian random walks, then the joint proposal density q also is a (possibly singular) multivariate Gaussian density. In particular, each marginal density is uniquely determined by the covariance matrix $\Sigma^{(k)}$, $k = 1, \ldots, K$. With a fixed correlation structure, the correlations between the candidates are all known given the marginal distributions. Hence, adaptation of q is reduced to adaptation of $\Sigma^{(k)}$, $k = 1, \ldots, K$; our proposed aMTM algorithm updates only one of these covariance matrices in a given iteration, namely, the one that was used to generate the selected candidate. The adaptation between two consecutive marginal covariances $\Sigma^{(k)}$ is inspired from existing schemes used to improve upon the Metropolis algorithm. Additional adaptation variants are also discussed in Supplement A, Section 1.2.

AM updates A first update rule is given by the AM updates of **?**. At time n, the k-th proposal covariance is $s_d \Sigma_n^{(k)}$ for some scale $s_d > 0$. The recursion for the update $\Sigma_n^{(k)}$ satisfies

$$\mu_{n+1}^{(k)} = \mu_n^{(k)} + \gamma_{n+1} \left(x_{n+1} - \mu_n^{(k)} \right) , \tag{3.1}$$

$$\Sigma_{n+1}^{(k)} = \Sigma_{n+1}^{(k)} + \gamma_{n+1} \left[\left(x_{n+1} - \mu_n^{(k)} \right) \left(x_{n+1} - \mu_n^{(k)} \right)^\top - \Sigma_{n+1}^{(k)} \right] , \tag{3.2}$$

where $\mu_n^{(k)}$ is the running mean of the k-th component and $\gamma_{n+1} > 0$ is the adaptation step.

ASWAM updates Optimal scaling results provide guidelines about the choice of s_d given the dimension d of the target density. The AM algorithm uses that information to directly scale the proposal covariance. Now, these results also provide an optimal acceptance rate which can be used, instead of the scale itself, to tune the marginal covariances. Empirical evidence shows that the optimal acceptance rate is much less

sensitive to a change of target density than the optimal scale. Thus, aiming at an optimal acceptance rate rather than an optimal scaling is a more robust adaptation principle.

Based on this argument, a second update rule is provided by the *adaptive scaling within adaptive Metropolis* (ASWAM) updates of **?**. The idea is to compute the running mean and covariance as in the AM updates (3.1) and (3.2), but to also adapt the scale s_d toward a value that yields an acceptance rate approaching some target rate $\alpha_* \in [0, 1]$. The marginal covariance for candidate k at time n is $\lambda_n^{(k)}\Sigma_n^{(k)}$, where the scale $\lambda_n^{(k)}$ is updated using

$$\log\left(\lambda_{n+1}^{(k)}\right) = \log\left(\lambda_n^{(k)}\right) + \gamma_{n+1}\left[\alpha_{\mathrm{MTM}}\left(y, y^{(-k)}|x, x_*^{(-k)}\right) - \alpha_*\right] .$$

RAM updates An alternative to ASWAM updates is the *robust adaptive Metropolis* (RAM) of **?**, whose updates are better suited to target densities with no finite second moment. In a single step, the marginal covariance is updated to approach both the (pseudo-)covariance of the target and a target acceptance rate. Given a square root decomposition $\Sigma_n^{(k)} = S_n^{(k)} S_n^{(k)\top}$, the next marginal covariance is given by

$$\Sigma_{n+1}^{(k)} = S_n^{(k)}\left\{I_d + \gamma_{n+1}\left[\alpha_{\mathrm{MTM}}\left(y, y^{(-k)}|x, x_*^{(-k)}\right) - \alpha_*\right]\frac{z_n^{(k)} z_n^{(k)\top}}{\left\|z_n^{(k)}\right\|_2^2}\right\} S_n^{(k)\top} ,$$

where $z_n^{(k)} = (S_n^{(k)})^{-1}[y - x_n]$ is the standardized proposed step.

4. Validity

The aMTM sampler is first and foremost an adaptive algorithm. The regularity assumptions that are imposed to verify the theoretical properties of the aMTM (ergodicity, LLN) are therefore very similar to other adaptive methods, such as the AM of **?**.

Although Section 3 describes several variants of the aMTM sampler, it is not possible to simultaneously consider all these variants when proving the ergodicity of this algorithm. In what follows, we consider some general results that can be used in verifying the ergodicity, but we relegate the details applicable to specific instances of the algorithm to the supplementary material. For example, the simple case of independent candidates and weights proportional to the target density requires no further assumption; other variants may require stronger assumptions, which are discussed in Supplement A, Section 3.3.

Before pursuing, let us introduce some more notation. The target distribution has a density π with respect to Lebesgue measure, with support $\mathcal{X} = \{x \in \mathbb{R}^d \,|\pi(x) > 0\} \subseteq \mathbb{R}^d$. We are interested in the expectation of a function $f : \mathbb{R}^d \to \mathbb{R}^p$ such that $\pi(|f|) < \infty$. At time $n \in \mathbb{N}$, the K proposal covariance matrices of the MTM kernel are $\Sigma = (\Sigma^{(1)}, \ldots, \Sigma^{(K)})$ with $\Sigma^{(k)} \in \mathcal{C}_d^+$, $k = 1, \ldots, K$, where $\mathcal{C}_d^+$ denotes the cone of symmetric positive-definite $d \times d$ matrices. The parameter space therefore is $\Theta = (\mathcal{C}_d^+)^K$

and the marginal proposal densities are $q_\theta^{(k)}(y|x) = \varphi(y|x, \Sigma^{(k)})$, where $\varphi(\cdot|\mu, \Sigma)$ denotes a d-dimensional normal density with mean μ and covariance Σ.

We assume that $\mathcal{X}$ and Θ both are compact; these assumptions are similar to those in **?** and greatly simplify proofs in Section 4.1, where the algorithm's ergodicity is studied. Generalizations to unbounded cases are discussed in Section 4.2, while limit theorems are considered in Section 4.3.

4.1. Ergodicity

First let us recall a result from **?**, which will be the basis of the aMTM algorithm ergodicity's analysis.

Theorem 4.1 (**?**, Theorem 2)**.** *Consider an adaptive MCMC algorithm using a family of Markov transitions $\{P_\theta\}_{\theta\in\Theta}$ and let $x_0 \in \mathcal{X}$ and $\theta_0 \in \Theta$ be the initial state and transition index, respectively. Suppose each transition P_θ admits the target density π as its stationary distribution and suppose the algorithm satisfies the diminishing adaptation (2.7) and the bounded convergence (2.8) conditions for these initial values. Then, the algorithm is ergodic to the target density for these initial values.*

In this result, there are three main conditions to verify: stationarity, diminishing adaptation and bounded convergence. In our context, the family of Markov transitions $\{P_\theta\}_{\theta\in\Theta}$ consists in MTM transitions with some fixed correlation structure and fixed weight function. The index θ corresponds to the collection of marginal covariances Σ.

4.1.1. Stationary distribution

As mentioned in Section 2.1, a sufficient condition for P_θ to admit π as its stationary distribution is to satisfy (2.2). Conveniently, any symmetrical proposal density meets this requirement; in particular, a multivariate Gaussian random walk assumption is sufficient to establish the stationarity of π. The proof of (2.2)'s sufficiency (in Supplement A, Proposition 3.1) is valid for any choice of correlation structure and any weight function.

4.1.2. Bounded convergence

Our study of the aMTM's bounded convergence relies on **?**, Proposition 23:

Proposition 4.1 (**?**, Proposition 23)**.** *Consider an adaptive MCMC algorithm using a family of Markov transitions $\{P_\theta\}_{\theta\in\Theta}$ such that Θ is compact, and such that each P_θ admits the target density π as stationary distribution and is Harris-ergodic to π. If, for all $n \geqslant 1$, the application $(x, \theta) \mapsto \Delta_n(x, \theta)$ with*

$$\Delta_n(x, \theta) := \|P_\theta^n(\cdot|x) - \pi(\cdot)\|_{\mathrm{TV}}$$

is jointly continuous in (x, θ) for all $(x, \theta) \in \mathcal{X} \times\Theta$ and if $\{X_n\}_{n\geqslant 1}$ is bounded in probability, then the algorithm satisfies the bounded convergence condition (2.8).

Based on the previous result, the verification of the bounded convergence further requires (1) that each MTM transition be Harris-ergodic with respect to π, (2) that the parameter space Θ be compact, (3) that $\Delta_n(x, \theta)$ be continuous for each n, and (4) that $\{X_n\}_{n\geqslant 1}$ be bounded in probability. The intuition behind this result is that each transition is ergodic with some rate of convergence. We then suppose that this rate of convergence varies continuously on the compact set $\mathcal{X} \times \Theta$ (at least in probability) so that it remains bounded, whence "bounded convergence".

We can show that Harris erogodicity of MTM transitions only requires the verification of π-irreducibility and aperiodicity. Indeed, we have the following result.

Proposition 4.2. *Let P be a MTM transition for a target density π. If P is π-irreducible, then P is Harris-recurrent.*

Proof. The complete argument may be found in Supplement A, Proposition 3.3, but is almost identical to the proof by **?**, Corollary 2 in the Metropolis case. □

We note that a similar implication exists for Metropolis-Hastings transitions. It will then be convenient to observe that π-irreducibility and aperiodicity follow from the condition stated in the following result, which is reminiscent of similar results for Metropolis-Hastings algorithms with extra assumptions on the weight functions.

Proposition 4.3. *Let P be a MTM transition for a target density π with connected support $\mathcal{X}$. Suppose that π is bounded above on $\mathcal{X}$ and below on any compact subset of $\mathcal{X}$. Suppose that, for each $k = 1, \ldots, K$, there exist $\delta, \varepsilon > 0$ such that the marginal proposal densities are symmetric and satisfy*

$$q^{(k)}(y|x) > \varepsilon \ , \qquad \forall\, x,\, y \in \mathcal{X} : \|y - x\|_2 < \delta \ .$$

Suppose that, for all fixed $x \in \mathcal{X}$, the weights $w^{(k)}(\cdot|x)$ are bounded above and below on any compact set. Then, the kernel P is π-irreducible and aperiodic.

Proof. The complete argument may be found in Supplement A, Proposition 3.2, but is completely analogous to the proof by **?**, Lemma 7.6 in the MH case. □

The verification of Δ_n's continuity is inspired from a proof by **?**, Corollary 11 in the case of a Metropolis-Hastings sampler. We refer the reader to Supplement A, Section 3.3.4, for a complete proof, which requires no further assumption other than those already mentioned.

Under the assumption that $\mathcal{X}$ is compact, we directly have that $\{X_n\}_{n\geqslant 1}$ is bounded in probability. The more general case where $\mathcal{X}$ is unbounded requires more care and will be discussed in Section 4.2.

4.1.3. Diminishing adaptation

The diminishing adaptation condition (2.7) is easier to verify in the context of stochastic approximations. In particular, we recognize the covariance updates described in Section 3.3 as those of a Robbins-Monro algorithm (**?**), which consists of updates taking the following form:

$$\theta_{n+1} = \theta_n + \gamma_{n+1} H(\theta_n, (k, y^{(1:K)}, x_*^{(1:K)})) \ , \tag{4.1}$$

for some function $H : \Theta \times \{1, \dots, K\} \times \mathcal{X}^{2K} \to \mathbb{R}^{d_\theta}$ with $\Theta \subseteq \mathbb{R}^{d_\theta}$. In the case where running means $\mu_n^{(k)}$ or scales $\lambda_n^{(k)}$ are used, we add them to θ and augment Θ accordingly.

Following the work of **?** and **?**, we can prove the following result for adaptive MCMC algorithms.

Proposition 4.4. *Suppose that the transition update function $H_\theta(\cdot) = H(\theta, \cdot)$ is 1-Lipschitz in θ, that is, there exists $C < \infty$ such that for every pair $(\theta, \theta') \in \Theta \times \Theta$ and for every bounded function f we have*

$$\|P_\theta f - P_{\theta'} f\|_1 \leqslant C \|f\|_1 \|\theta - \theta'\|_2 \ , \tag{4.2}$$

where $\|\cdot\|_1$ is defined for a function $f : \mathcal{X} \to \mathbb{R}^p$ by $\|f\|_1 = \sup_{x \in \mathcal{X}} \|f(x)\|_2$ and where

$$P_\theta f(z) = \int f(x) P_\theta(x|z) \, \mathrm{d}x.$$

Suppose that $\{\theta_n\}_{n \geqslant 1}$ in bounded in probability; if

$$\sup_{\theta \in \Theta} \|H_\theta\|_1 < \infty \ , \qquad \forall \, x \in \mathcal{X} \ , \, \theta \in \Theta \ , \tag{4.3}$$

and if the sequence of adaptation steps converges to 0, $\gamma_n \to 0$, then the adaptive MCMC algorithm satisfies the diminishing adaptation condition (2.7).

Proof. See Supplement A, Proposition 3.4, for a stronger statement of which Proposition 4.4 is a special case when Θ is compact. □

The previous result reduces the verification of the diminishing adaptation to the verification of the Lipschitz transitions condition (4.2) and the bounded updates condition (4.3). Verifying the Lipschitz transitions condition (4.2) does not require any additional assumption under the simple case of independent proposals and weights proportional to the target density. For other aMTM variants, the verification of the Lipschitz transition must be made on a case by case basis; see Supplement A, Section 3.3.2, for a discussion. Verifying the bounded updates condition (4.3) is trivial under the assumption that $\mathcal{X}$ is compact because any update rule described in Section 3.3 will only involve bounded quantities (Supplement A, Section 3.3.3).

4.2. Generalizations to unbounded spaces

The major assumptions made in Section 4.1 were the compactness of the state space $\mathcal{X}$ and parameter space Θ. These conditions greatly simplify the verification of the algorithm's ergodicity, but also substantially restrict the theoretical applicability of the proposed sampler. In this section, we discuss different approaches that could be used to relax or even remove these assumptions.

Assuming $\mathcal{X}$ to be compact may seem a major impediment to the practical use of the algorithm since target densities often have unbounded supports. One might then worry about the fact that the theoretical results of the previous section only apply to a very restricted class of target densities. Now, a simple workaround is to consider the target $\tilde{\pi} = \pi|_{\widetilde{\mathcal{X}}}$, a version of the initial π restricted to a compact set $\widetilde{\mathcal{X}} \subset \mathcal{X}$, which can be chosen arbitrarily large. In that case, the expectation $\tilde{\pi}(f)$ of the resulting MC estimate will be virtually indistinguishable from the original expectation $\pi(f)$ provided that $\widetilde{\mathcal{X}}$ is chosen large enough. In practice, this approach corresponds to rejecting any proposal that lies outside of $\widetilde{\mathcal{X}}$.

In contrast, the compactness of Θ does not reduce the scope of theoretical results; in reality, it only restricts the family of MTM transitions on which adaptation can be performed. In the aMTM algorithm, the space Θ lies within the product of K convex cones of symmetric positive definite matrices. We can therefore simply choose Θ compact by bounding the eigenvalues of the covariance matrices inside some interval. Since users typically have some idea of their problem's scaling, it is easy to find reasonable bounds so that Θ contains the most efficient MTM transitions. In any case, practical implementations are subject to program and machine limitations so any symmetric positive definite matrix will lie in some definitive compact set when stored.

4.2.1. Compact coverages

It is important to note that, because of the extensive similarities between MH and MTM algorithms, results applicable to the AM sampler are expected to have aMTM counterparts holding under fairly similar conditions. Here is an example of a construction used to prove the ergodicity of the AM algorithm on unbounded domains.

Dating back to the work of **?**, *compact coverages* or *troncation at randomly varying bounds* or *sequentially constrained adaptive MCMC algorithms* is the idea of performing a Robbins-Monro stochastic approximation—which covers most adaptive MCMC algorithms as a special case—within some compact set and expanding that set when necessary. More explicitly, a compact coverage of Θ is a sequence of compact sets $\{\mathcal{K}_r\}_{r\geqslant 0}$ increasing to Θ, i.e. such that $\cup_{r\geqslant 0}\mathcal{K}_r = \Theta$ and $\mathcal{K}_r \subset \operatorname{int}(\mathcal{K}_{r+1})$. Then, the adaptation step of the sampler is modified so that the parameter θ_{n+1} is updated only if the new value lies in $\mathcal{K}_{n+1}$.

Subject to some regularity conditions on the target density π, **?**, Section 5 show that the sequentially constrained AM algorithm is ergodic with respect to π for $\mathcal{X}$ and Θ unbounded (a generalization of **?**, Theorem 2). Furthermore, **?**, Section 5 extends these results to the ASWAM sampler and **?**, Theorem 6 uses results from **?** to verify the

ergodicity of the RAM algorithm on unbounded domains.

Unfortunately, the *compact coverages* method of proof relies heavily on the geometry of the acceptation and rejection regions around the current state. In the case of MTM transitions, these regions become incredibly complicated because of the multiple candidates and shadow points; it is therefore far from easy to extend these proofs to the aMTM case.

4.2.2. Bounded adaptation and combocontinuity

Another setting under which the ergodicity of adaptive MCMC algorithms with unbounded state space can be studied is that of *bounded adaptation*, introduced by **?**. Consider $\widetilde{\mathcal{X}}$, a compact subset of $\mathcal{X}$, and let us modify the adaptive MCMC as follows: whenever the current state is outside of $\widetilde{\mathcal{X}}$, a fixed transition is used and the parameter θ is not updated. We also assume *bounded jumps*; this means that there exists $D < \infty$ such that the probability of moving from $x \in \mathcal{X}$ to a point that is at most D away is 1 uniformly in $\theta \in \Theta$. This can be enforced by construction, by using proposal densities truncated beyond D. **?**, Theorem 21 then show that the AM algorithm verifies the bounded convergence condition, provided that the sampler features a continuous transition (or a continuous proposal in the case of MH algorithms). Note that we still require the compactness of the parameter space Θ in that case. **?** extend this result to more general adaptive MCMC algorithms that verify a *combocontinuity* condition, i.e. samplers using a transition density that can be written as a finite combination of continuous densities. In particular, MTM transitions fall under the scope of combocontinuity assuming that all proposals are continuous densities.

4.3. Limit theorems

Two other interesting characteristics of MC estimates are satisfying a law of large numbers and a central limit theorem. Indeed, ergodicity guarantees that the marginal distribution of the chain converges to the target distribution, but does not directly inform us on the properties of the estimate itself.

Under ergodicity, it is not hard to verify a weak law of large numbers for any bounded function: **?**, Theorem 23 show that Bounded Convergence and Diminishing Adaptation are sufficient conditions in that case. However, extending the result to a strong LLN or broadening the class of functions over which it applies generally are trickier tasks.

Typically, strong LLN for adaptive MCMC algorithms require some sort of V*-ergodicity* condition, where V is some test function that ultimately controls the convergence rate. The obtained results thus apply to any V^α-bounded function f for some $\alpha \in [0, 1)$, i.e. such that $\sup_{x\in\mathcal{X}} |f(x)|/V^\alpha(x) < \infty$ (**?**, Theorem 8). In our context, this method of proof however requires $V \equiv 1$, which then only allows bounded functions. Now, the context of compact coverages described in Section 4.2.1 could potentially enable verifying a strong LLN for the aMTM algorithm as was done for the AM algorithm (**?**, Theorem 10), the ASWAM algorithm (**?**, Proposition 5), and the RAM algorithm (**?**, Theorem 6), all of which allow π^{-1}-bounded functions.

The story is very similar when it comes to obtaining a central limit theorem for the aMTM sampler. **?**, Theorem 9 provide a CLT for adaptive MCMC using compact coverages assuming $\pi^{-\alpha}$-ergodicity, which holds for any $\pi^{-\alpha/2}$-bounded function with $\alpha \in [0, 1)$. **?**, Theorem 18 derive a similar result for the specific case of the AM algorithm with $\alpha = 1$.

5. Numerical experiments

5.1. Simulation experiments

5.1.1. Summary of findings

? contains multiple simulation experiments investigating the many variants of the aMTM algorithm. For the MTM sampling component of the algorithm, the user can specify a correlation structure, a weight function and the number of candidates; for the adaptation component, the user can specify the update scheme, the target acceptance rate and the step-size sequence. We refer the reader to **?** for the details, but report here our main observations.

Extremely antithetic and randomized quasi-Monte Carlo candidates tend to perform slightly better than independent proposals and significantly better than common random variable proposals. These two correlation structures encourage a better spread of candidates over the sample space and it is therefore not surprising to record better mixing and space exploration. These results seem consistent both with theory—EA candidates have larger optimal acceptance rates (**?**)—and with practice (**?**). Experimenting with importance weights and weights proportional to the target did not yield a clear favorite; this seems to agree with similar experiments (see, e.g., the extensive analysis of **?**.)

It is important to keep in mind that the complexity of the aMTM scales with $2K-1$ as the target evaluation is generally the computational bottleneck of each iteration. Striking a balance between more efficient and costlier iterations is thus a crucial problem. Gladly, we find that a small number of candidates—between 2 and 5, depending on the target—is generally enough to obtain the largest improvements in performance compared to single-candidate samplers. The following section contains an empirical study of number of candidates, computational cost and performance.

In terms of adaptation, we find that updates using a target acceptance rate (RAM, ASWAM) outperform the simple AM and that RAM updates seem to improve marginally on the ASWAM in some cases. As for the target acceptance rate, we observe that rates relatively smaller than the optimal ones (**?**) perform best: these optimal rates are obtained for well-behaved targets, so it not surprising to find that smaller rates are preferable. Generally, we find that rates in the range $[0.2, 0.5]$ produce results with somewhat uniform performances. Finally, for step sizes of the form $n^{-\gamma}$, we observe that AM and ASWAM updates tend not to be significantly affected by the value of $\gamma \in [0.5, 1]$, while RAM updates seem to benefit from values closer to the lower bound.

5.1.2. Computation time and number of proposals

As mentioned previously, one major drawback of MTM algorithms is their increased computational cost: each MTM sampling step requires $2K - 1$ target evaluations. In comparison, the adaptation step within the aMTM algorithm is typically computationally cheap. Indeed, full scale samplers such as the aMTM can only be reasonably used on small to moderate dimensions, so the covariance update usually is insignificant compared to additional target evaluations.

Setting. We perform a numerical experiment to study how performance and computational cost are affected by the number of candidates. To this end, we compare the MSJD and the multivariate effective sample size (mESS, **?**) of the output chain, divided by computing time (/CPU) or by the number of evaluations per step (/NbEval). The mESS leads to natural comparisons in terms of equivalent iid samples; as with other MCMC estimates however, this value can be made arbitrarily large by prolonging the chain. It is then convenient to report the statistics mESS/CPU and mESS/NbEval, which tell us how many equivalent iid samples are produced every second and every target evaluation, respectively.

We consider a variation on the famous "banana" target example (see, among many others, **?????**). The 5-dimensional target density is expressed as

$$\pi(x) \propto \exp\left\{-\frac{1}{2}\left[\frac{x_1^2}{a_1^2} + (x_2 - B_1 x_1^2)^2 + x_3^2 + \frac{x_4^2}{a_2^2} + (x_5 - B_2 x_4^2)^2\right]\right\} \tag{5.1}$$

with $a_1 = 1$, $a_2 = 1$, $B_1 = 3$, and $B_2 = 1$. In particular, the pairs of components $(1, 2)$ and $(4, 5)$ each marginally forms a "banana", as can be seen from the iid sample[1] in Figure S1 of Supplement A. This target has 5 connected but fairly different regions (defined in Table S2 and depicted in Figure S1, both in Supplement A) that may require different proposal densities to be explored efficiently. We use these regions to construct a lower bound on the TV distance by comparing the weight of each region in the MCMC sample to the weight in an iid sample. Since we have access to iid samples, the mESS will be computed using the true target covariance (estimated from a large iid sample). As the target density is cheap to compute— we would not see the impact of K on CPU—, we artificially slow down its evaluation by repeating the computations 20 times.

We consider four adaptation variants (None, AM, ASWAM, and RAM) and vary the number of candidates K between 1 and 10. This therefore includes the Metropolis algorithm (None; $K = 1$), MTM algorithms (None; $K > 1$), and adaptive Metropolis algorithms (AM, ASWAM, RAM; $K = 1$). Initial values (or fixed values in the case of no adaptation) for proposal covariances are chosen to be somewhat adjusted to the target, but not so much as to give an unrealistic advantage to non-adaptive samplers: in particular, all proposal covariance matrices are chosen to be scalar multiples of $\mathrm{diag}(1, 3, 1, 1, 3)$,

[1] This density emerges as the transformation

$$(z_1, z_2, z_3, z_4, z_5) \mapsto (a_1 z_1, z_2 + B_1 z_1^2, z_3, a_2 z_4, z_5 + B_2 z_4^2)$$

where the z_i are iid standard normals.

where the scales are a log-spaced sequence of length K between 10^{-2} and 10^2. Each algorithm is run for 200,000 iterations and the first half of the chain is discarded as burn-in.

Results. Results are illustrated in Figure 2. First, we note that for an expensive target, the computing time (CPU, panel (d)) is virtually unaffected by the adaptation; the computational cost is mostly explained by the number of target evaluations, as well as a fixed cost independent of K.

In terms of TV distance (panel (a)), we find that the best adjustment to the target density is obtained using adaptive algorithms (especially AM or ASWAM) and around 4 to 8 proposals. Interestingly, we find that the TV distance increases beyond 7 proposals for adaptive samplers: a possible explanation is that each proposal is getting updated less often as K increases, leading to candidates of worse quality. Non-adaptive samplers struggle to adequately sample from regions 1–4, regardless of the number of candidates.

The mESS statistic (panel (b)) indicates that adaptive samplers do not gain much efficiency beyond the first 5 proposals; the MSJD statistic (panel (c)) shows a similar trend, where the increase seems to slow down around the same number of proposals. Once we take into account computing time (panels (e) and (f)), we observe a small decrease in efficiency with the number of candidates for both mESS and MSJD and for all adaptation schemes, but we have to keep in mind that fewer than 4 candidates leads to inadequate samples. In terms of target evaluations (panels (g) and (h)), we see a sharp decrease in efficiency starting from the 2nd candidate—this is not surprising as the increase in mESS or MSJD is insufficient to counteract the division by 3, 5, etc. instead of by 1.

Overall, this experiment indicates that there is a sweet spot balancing the efficiency (fewer proposals is always preferable) and the adjustment to the target (more proposals is generally better). In this instance, the target has 5 regions with varying local covariances and we find that around the same number of candidates works best.

5.2. Loss of heterozygosity in esophageal cancer cells

Problem description. During a cancer's progression, affected cells undergo genetic changes such as loss of chromosomes sections. This abnormality, called *loss of heterozygosity* (LOH), can be detected in laboratory: the Seattle Barrett's Esophagus research project (**?**) collected LOH rates for 40 different regions of the genome. For each region $i = 1, \ldots, 40$, we denote by X_i the number of cells with detected LOH and by N_i the total number of cells analyzed; Figure 3 (top) contains an histogram of the proportions of LOH within each region.

It is hypothesized that there are two causes for LOH in cancer cells: a "background" LOH, possibly caused by the cancer's progression, and a "systematic" LOH, due to the presence of tumor suppressor genes (TSGs). Regions with higher rates of LOH are therefore suspected to contain TSGs: modeling LOH rates is thus of interest for cancer researchers in order to identify regions with such TSGs. The existence of these two regimes suggests modeling LOH rates using mixture models: component membership

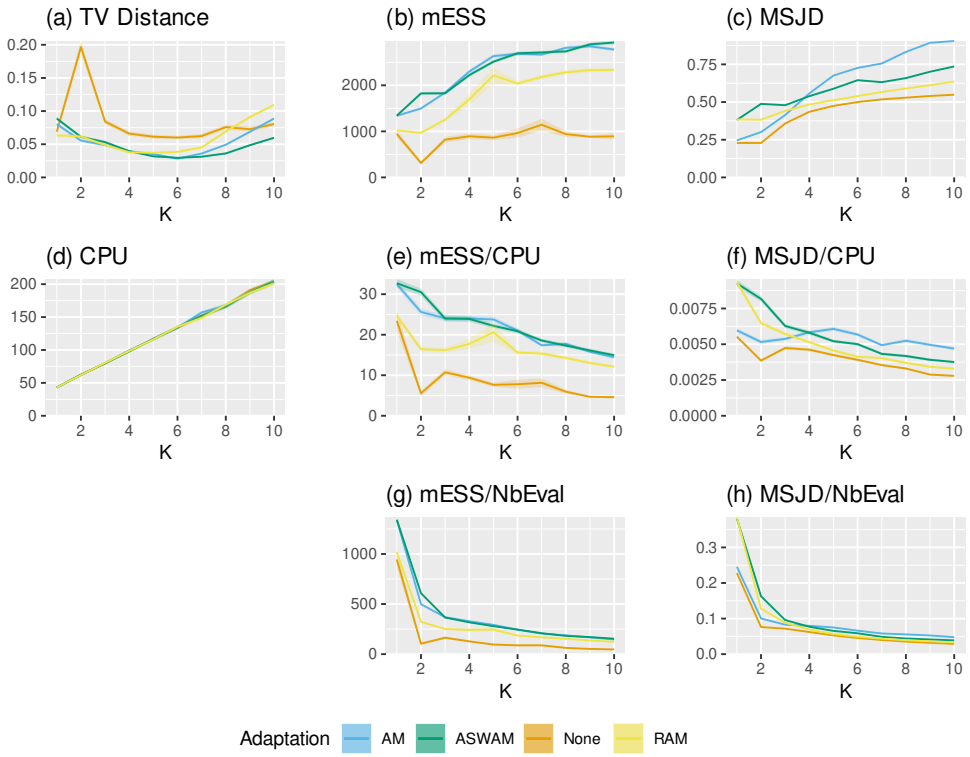

Figure 2. MCMC chain statistics for the banana target (5.1). TV distance is a lower bound on the total variation distance of the empirical distributions between a MCMC sample and an iid sample using regions defined in Table S1; mESS is the multivariate effective sample size using a target covariance estimated from an iid sample; MSJD is the mean squared jumping distance; CPU is the computing time in seconds; NbEval is the number of target evaluations per MCMC iteration (i.e. $2K-1$ where K is the number of candidates). Statistics are shown as means (line) with one standard error (band) over 100 random initializations.

probabilities may provide insight on the presence of TSGs. For more information on localization of TSGs and on LOH, we refer to **?** and references therein.

? suggests several two-components mixture models for LOH in cancer cells, where each component is either a binomial or a beta-binomial distribution. Following the analysis of this dataset by **?**, we consider a mixture of a binomial component and a beta-binomial component. The likelihood is given by

$$X_i \mid N_i, \eta, \pi_1, \pi_2, \gamma \sim \eta \text{Binomial}\left(N_i, \pi_1\right) + (1-\eta)\text{BetaBinomial}\left(N_i, \pi_2, \gamma\right) \ ,$$

where $\eta \in [0,1]$ controls the mixture weights, $\pi_1, \pi_2 \in [0,1]$ control the center of each component, and $\gamma \in \mathbb{R}$ controls the (logit) spread of the Beta-binomial component ($\gamma \to -\infty$

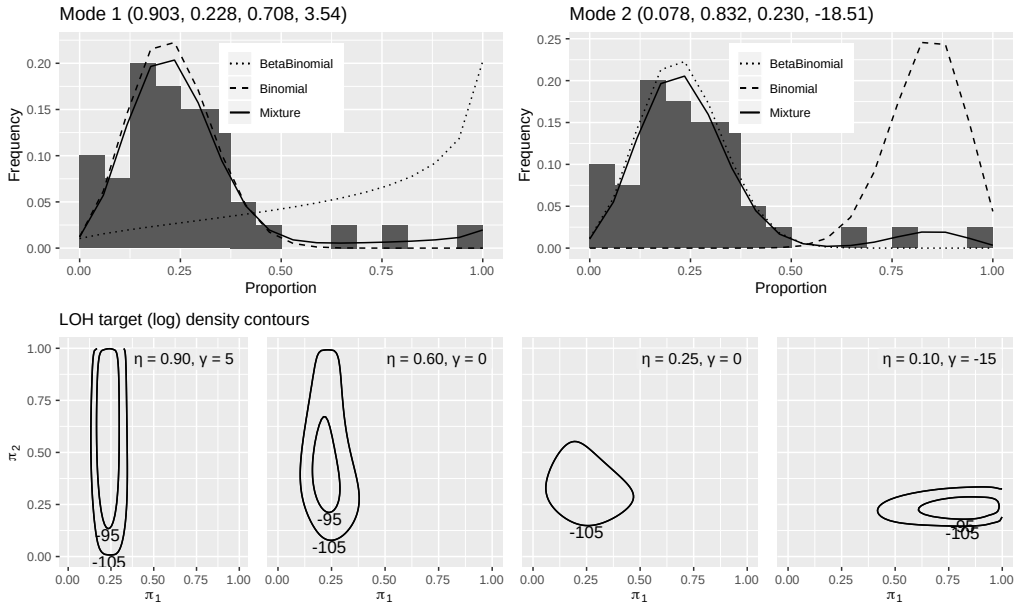

Figure 3. (Top) Histogram of observed proportions of LOH (**?**) together with the distribution of each mixture component (dotted: Beta-Binomial; dashed: Binomial) and of the resulting mixture (solid) for the two posterior modes $(\eta, \pi_1, \pi_2, \gamma)$. (Bottom) Contours of the posterior density plots in (π_1, π_2) for some fixed values of (η, γ).

corresponds to a binomial distribution and $\gamma \to \infty$ to a discrete uniform distribution). The prior on the four model parameters is taken to be uniform over a set of plausible values:

$$(\eta, \pi_1, \pi_2, \gamma) \sim \text{Uniform}\left([0,1]^3 \times [-30, 30]\right) \ .$$

Depending on which component is used to model each regime, the mixture model features some clear multi-modality. Indeed, the posterior distribution exhibits a first mode (Figure 3, top left) where the binomial component models the lower (background) LOH rates and the beta-binomial component models the higher (systematic) LOH rates; the second mode (Figure 3, top right) exchanges this assignment. A few cross-sections of the posterior distribution are depicted in Figure 3 (bottom).

Experiment. Obtaining samples from a multi-modal posterior is a notoriously hard task for standard MCMC algorithms and multiple samplers have been proposed to approach such problems. We detail some proposals that were applied to this very LOH mixture model problem. First, **?** proposes the *normal kernel coupler* (NKC), which uses interacting chains forming a normal kernel density estimate of the target. Conveniently,

in their analysis of the LOH data, they provide global and per-mode posterior means computed using numerical integration (*adaptive quadrature*, AQ) against which we can compare our results. Second, a variety of MCMC algorithms using regional adaptation have been proposed: Mixed RAPT (**?**), RAPTOR (**?**), and OPRA (**?**). Third, **?** propose to sample from multimodal distributions using *parallel sampling* (PS). Fourth, **?** propose an *interacting multiple-try Metropolis* algorithm and apply it to the LOH model, but no numerical results are reported.

We compare our proposed aMTM algorithm to these methods both in terms of estimate quality and in terms of MCMC chain metrics. Across the different analyses of the LOH model posterior in the literature, we find multiple reportings such as means, standard deviations and quantiles, either calculated globally or restricted to each of the two modes; we will provide all those estimates for comparison. Inspecting slices of the posterior density (Figure 3, bottom) and various scatter plots emerging from other samplers (**???**), we define the two regions of interest. Specifically, by dividing the space using the boundary $\pi_1 = 0.4$ we find that each region contains one mode, with Mode 1 being assigned to the region $\pi_1 < 0.4$. In addition to parameter estimates, we will be interested in estimating the weight of each mode: numerical integration (**?**) indicates that Mode 2 carries 3.0% of the target weight. Furthermore, we provide some chain statistics: the *mean squared jumping distance* (MSJD), the acceptance rate and the average marginal autocorrelation ($\overline{|\text{ACF}|}$) as described in **?**. [2]

For comparative purposes, we include AM, ASWAM, and RAM as single-candidate samplers as well as a non-adaptive MTM sampler with $K = 3$ candidates. The MTM samplers' proposal covariances are chosen to be (1) all equal to an estimate of the target's covariance (*Common*), (2) a downscaled version of that same global covariance (factors of 1, 0.1 and 0.01, *Scaled*), and (3) adjusted to each mode (a global component, plus one component for each mode, *Oracle*). We produce chains for different tuning parameters (target acceptance rate, adaptation parameter, etc.) and report the best instances. For our aMTM algorithm, we use RAM updates with a target acceptance rate of $\alpha = 0.2$ and proposal initialized using the *scaled* covariances. In all cases, we produce chains of length $N = 10,000$ with a burn-in of $1,000$ iterations; we proceed to 100 replications with random initializations of the chain uniform on the support and we report means and standard errors across those replications.

Results. Table 2 contains chain statistics obtained from all of our methods, along with results from other sources. One of the hardest elements to get right while sampling multimodal distributions using MCMC samplers is the respective weight of each mode. We observe that only MTM methods achieve weight estimates that are close to the truth (0.030), while single candidate samplers often find a single mode. This phenomenon also explains why AM and ASWAM exhibit higher MSJD and acceptance rates as sampling from a unimodal distribution leads to better mixing. Inspecting the non-adaptive MTM samplers, we find that relatively well-adjusted proposals can lead to decent chain properties: using proposals on varying scales yields large MSJD and acceptance rates while

[2] The authors mention averaging the first 40 absolute lag-correlations while their code averages the first 1600; we use the latter here to obtain comparable results.

LOH Binomial-BetaBinomial mixture model: Chain statistics				
Algorithm	Details	MSJD	Acc. rate	$\mathbb{P}$(mode 2)
AM		1.338 (0.043)	0.191 (0.006)	0.280 (0.045)
ASWAM	$\alpha = 0.4$	1.674 (0.043)	0.312 (0.006)	0.262 (0.044)
RAM	$\alpha = 0.3$	0.257 (0.006)	0.034 (0.001)	0.086 (0.025)
MTM($K = 3$)	Common proposals	0.104 (0.002)	0.011 (0.000)	0.032 (0.008)
	Scaled proposals	0.858 (0.002)	0.259 (0.000)	0.047 (0.008)
	Oracle proposals	0.381 (0.003)	0.041 (0.000)	0.028 (0.009)
aMTM($K = 3$)	RAM, $\alpha = 0.2$	0.864 (0.005)	0.265 (0.001)	0.038 (0.010)
NKC				0.047
RAPTOR			0.194	

Table 2. MCMC chain statistics for the LOH mixture model. MSJD is the mean squared jumping distance; Acc. rate is the acceptance rate of the chain; $\mathbb{P}$(mode 2) is the proportion of points in Region 2 (defined by $\pi_1 > 0.4$). Statistics are shown as mean (standard error) over 100 random initializations.

these samplers still spend an appropriate amount of time in each mode. Our aMTM sampler, which does not require such fine tuning, achieves similar if not better chain statistics and maintains accurate estimates of the modes' respective weights.

Turning to the global parameter estimates presented in Table 3, we compare our method to other proposed algorithms. We find that estimating correctly the modes' weights greatly improves the accuracy of the estimates. Indeed, the NKC slightly oversamples from the smaller mode which introduces a fairly large bias, especially for quantile estimates with probabilities that are close to the smaller mode's weight. Our estimates agree with those obtained from numerical integration and those from RAPTOR, and seem to improve on the estimates obtained from NKC, Mixed RAPT, OPRA, and PS. Furthermore, the mixing of the marginal chains, evaluated through $\overline{|\text{ACF}|}$, seems to be slightly better in aMTM chains than in the RAPTOR chain, which is not surprising given the larger acceptance rate of aMTM (26.5 % vs. 19.4 %).

When restricting the estimates to either of the two regions (Table 4), we find that our method yields estimates that agree more closely with numerical integration than NKC and Mixed RAPT, especially for the smaller mode.

6. Discussion

The proposed adaptive multiple-try Metropolis algorithm is a natural extension of both the adaptive Metropolis (AM) sampler (**?**) and the multiple-try Metropolis (MTM) algorithm (**?**). It combines the flexibility of the MTM and the ease of use of adaptive samplers. Indeed, in our multimodal LOH example, the aMTM sampler produces accurate samples with limited tuning: in terms of mixing, space exploration and estimates, our method outperforms non-adaptive MTM and single candidate adaptive samplers, and is at least on par with more sophisticated methods such as RAPTOR (**?**).

The flexibility induced by the multiple proposals and the adaptation makes it an

LOH Binomial-BetaBinomial mixture model: Global estimates

Parameter (AQ mean)	Method	Mean	Std dev.	$Q_{0.025}$	$Q_{0.975}$	$\overline{\lvert\text{ACF}\rvert}$
η (0.832)	aMTM	0.823 (0.008)	0.124 (0.008)	0.507 (0.025)	0.963 (0.000)	0.105 (0.015)
	NKC	0.82		0.075	0.965	
	RAPTOR	0.828	0.155			0.15
	Mixed RAPT	0.838				
	OPRA	0.901				
	PS	0.816 (0.001)				
π_1 (0.246)	aMTM	0.252 (0.006)	0.062 (0.008)	0.193 (0.000)	0.408 (0.025)	0.092 (0.017)
	NKC	0.257		0.193	0.829	
	RAPTOR	0.248	0.106			0.19
	Mixed RAPT	0.275				
	OPRA	0.230				
	PS	0.299 (0.001)				
π_2 (0.617)	aMTM	0.613 (0.005)	0.170 (0.002)	0.293 (0.004)	0.906 (0.001)	0.104 (0.009)
	NKC	0.612		0.230	0.912	
	RAPTOR	0.614	0.174			0.05
	Mixed RAPT	0.679				
	OPRA	0.729				
	PS	0.678 (0.002)				
γ (12.82)	aMTM	12.542 (0.317)	11.321 (0.205)	−13.479 (0.963)	29.119 (0.030)	0.106 (0.010)
	NKC	12.3		−21.2	29.3	
	RAPTOR	12.732	11.561			0.09
	Mixed RAPT	13.435				
	OPRA	12.401				
	PS	9.49 (0.51)				

Table 3. Global estimates for the LOH mixture model. Q_p denotes the p-th quantile; $\overline{\lvert\text{ACF}\rvert}$ is the average marginal autocorrelation as defined in **?**. Statistics are shown as mean (standard error) over 100 random initializations for the aMTM sampler. See text for a description of the methods.

interesting option for MCMC users dealing with target exhibiting multimodality or, more generally, complex geometry. The computational overhead of this method mostly emerges from the multiple target evaluations, but performance improvement can be observed with just a few additional proposals. For well-behaved targets, theory tells us that as few as $K = 2$ candidates provides the largest efficiency increase; for the more complex target of Section 5.1.2, $K \approx 5$ produced the best balance between accuracy and efficiency. In additional synthetic experiments (**?**), we also find that few candidates, in the range 2–5, is generally enough, with lower values being preferable for simpler targets and larger values for more complicated targets.

The diminishing adaptation and bounded convergence conditions provide simple guidelines to follow when proposing adaptive MCMC algorithms. Indeed, our proposed adaptation scheme is one of many that can be imagined for MTM transitions. We experimented with other valid adaptations described in Supplement A, Section 1.2, which did not significantly improve on aMTM. Still, our proposal may not be the optimal way to introduce adaptation in MTM samplers. In particular, as experimental results suggest, updating

LOH Binomial-BetaBinomial mixture model: Mode estimates					
Parameter (AQ mean)	Method	Mean	Std dev.	Q 0.025	Q 0.975
Mode 1 estimates					
η (0.854)	aMTM	0.852 (0.008)	0.082 (0.008)	0.649 (0.025)	0.964 (0.000)
	NKC	0.856		0.656	0.966
	Mixed RAPT	0.897			
π_1 (0.229)	aMTM	0.229 (0.000)	0.019 (0.000)	0.193 (0.000)	0.267 (0.000)
	NKC	0.229		0.192	0.266
	Mixed RAPT	0.229			
π_2 (0.629)	aMTM	0.628 (0.002)	0.162 (0.001)	0.317 (0.002)	0.907 (0.001)
	NKC	0.631		0.319	0.913
	Mixed RAPT	0.714			
γ (13.73)	aMTM	13.709 (0.105)	10.401 (0.105)	−9.519 (0.825)	29.175 (0.021)
	NKC	13.7		−4.97	29.3
	Mixed RAPT	15.661			
Mode 2 estimates					
η (0.091)	aMTM	0.089 (0.001)	0.044 (0.001)	0.021 (0.001)	0.188 (0.003)
	NKC	0.084		0.017	0.219
	Mixed RAPT	0.079			
π_1 (0.825)	aMTM	0.814 (0.006)	0.062 (0.008)	0.694 (0.000)	0.930 (0.025)
	NKC	0.832		0.741	0.914
	Mixed RAPT	0.863			
π_2 (0.232)	aMTM	0.231 (0.000)	0.018 (0.000)	0.198 (0.000)	0.267 (0.001)
	NKC	0.23		0.199	0.261
	Mixed RAPT	0.237			
γ (-16.28)	aMTM	−16.559 (0.296)	7.422 (0.124)	−28.743 (0.162)	−4.303 (0.328)
	NKC	−17.5		−29.5	−4.11
	Mixed RAPT	−14.796			

Table 4. Per-mode estimates for the LOH mixture model. Q_p denotes the p-th quantile. Statistics are shown as mean (standard error) over 100 random initializations for the aMTM sampler. See text for a description of the methods.

only the selected proposal may not be the most efficient adaptation method: on average, proposals are updated only once every K iterations.

While we provided significant theoretical guarantees about the validity of our proposed algorithm, some improvements still are desirable. In particular, the state space and parameter space compactness assumptions are fairly restrictive, but we remind the reader about the promising avenues that could extend our results to unbounded spaces and that are discussed in Section 4.2. Additionally, although we did not provide a central limit theorem for our MCMC algorithm, we still expect one to hold considering related work discussed in Section 4.3.

Acknowledgements

This work has been supported by the Natural Sciences and Engineering Research Council of Canada.

Supplementary Material

Supplement A: Supplement to "An adaptive multiple-try Metropolis algorithm"
(doi: TBD; .pdf). Additional details on aMTM variants and on experiments. Intermediary results and proofs.

Supplement B: Package `aMTM`: Adaptive multiple-try Metropolis algorithm
(; R package). The main sampling routine is implemented in `C++`; the `R` package consists of a wrapper for the sampler as well as some utility functions for chain statistics and plotting. The output is compatible with the `R` package `coda`.

Supplement C: Code producing the results
(; R code). Contains the `R` code defining and running the experiments, processing the results and generating the output.

Received December 2020 and revised July 2021