The Barker proposal: combining robustness and efficiency in gradient-based MCMC

Samuel Livingstone*[1] and Giacomo Zanella†[2]

[1]Department of Statistical Science, University College London.
[2]Department of Decision Sciences, BIDSA and IGIER, Bocconi University.

May 12, 2020

Abstract

There is a tension between robustness and efficiency when designing Markov chain Monte Carlo (MCMC) sampling algorithms. Here we focus on robustness with respect to tuning parameters, showing that more sophisticated algorithms tend to be more sensitive to the choice of step-size parameter and less robust to heterogeneity of the distribution of interest. We characterise this phenomenon by studying the behaviour of spectral gaps as an increasingly poor step-size is chosen for the algorithm. Motivated by these considerations, we propose a novel and simple gradient-based MCMC algorithm, inspired by the classical Barker accept-reject rule, with improved robustness properties. Extensive theoretical results, dealing with robustness to tuning, geometric ergodicity and scaling with dimension, suggest that the novel scheme combines the robustness of simple schemes with the efficiency of gradient-based ones. We show numerically that this type of robustness is particularly beneficial in the context of adaptive MCMC, giving examples where our proposed scheme significantly outperforms state-of-the-art alternatives.

1 Introduction

The need to compute high-dimensional integrals is ubiquitous in modern statistical inference and beyond (e.g. Brooks et al. [2011], Krauth [2006], Stuart [2010]). Markov chain Monte Carlo (MCMC) is a popular solution, in which the central idea is to construct a Markov chain with a certain limiting distribution and use ergodic averages to approximate expectations of interest. In the celebrated Metropolis–Hastings algorithm, the Markov chain transition is constructed using a combination of a 'candidate' kernel, to suggest a possible move at each iteration, together with an accept-reject mechanism [Metropolis et al., 1953, Hastings, 1970]. Many different flavours of Metropolis–Hastings exist, with the most common difference being in the construction of the candidate kernel. In the Random walk Metropolis, proposed moves are generated using a symmetric distribution centred at the current point. Two more sophisticated methods are the Metropolis-adjusted Langevin algorithm [Roberts and Tweedie, 1996] and Hamiltonian/hybrid Monte Carlo [Duane et al., 1987, Neal, 2011]. Both use gradient information about the distribution of interest (the *target*) to inform proposals. Gradient-based methods are widely considered to be state-of-the-art in MCMC, and much current work has been dedicated to their study and implementation (e.g. Beskos et al. [2013], Durmus and Moulines [2017], Dalalyan [2017]).

Several measures of performance have been developed to help choose a suitable candidate kernel for a given task. One of these is high-dimensional scaling arguments, which compare

*samuel.livingstone@ucl.ac.uk
†giacomo.zanella@unibocconi.it

how the efficiency of the method decays with d, the dimension of the state space. For the random walk algorithm this decay is of the order d^{-1} [Roberts et al., 1997], while for the Langevin algorithm the same figure is $d^{-1/3}$ [Roberts and Rosenthal, 1998] and for Hamiltonian Monte Carlo it is $d^{-1/4}$ [Beskos et al., 2013]. Another measure is to find general conditions under which a kernel will produce a geometrically ergodic Markov chain. For the random walk algorithm this essentially occurs when the tails of the posterior decay at a faster than exponential rate and are suitably regular (more precise conditions are given in [Jarner and Hansen, 2000]). The same is broadly true of the Langevin and Hamiltonian schemes [Roberts and Tweedie, 1996, Livingstone et al., 2019, Durmus et al., 2017a], but here there is an additional restriction that the tails should not decay too quickly. This limitation is caused by the way in which gradients are used to construct the candidate kernel, which can result in the algorithm generating unreasonable proposals that are nearly always rejected in certain regions [Roberts and Tweedie, 1996, Livingstone et al., 2019].

There is clearly some tension between the different results presented above. According to the scaling arguments gradient information is preferable. The ergodicity results, however, imply that gradient-based schemes are typically less *robust* than others, in the sense that there is a smaller class of limiting distributions for which the output will be a geometrically ergodic Markov chain. It is natural to wonder whether it is possible to incorporate gradient information in such a way that this measure of robustness is not compromised. Simple approaches to modifying the Langevin algorithm for this purpose have been suggested (based on the idea of truncating gradients, e.g. Roberts and Tweedie [1996], Atchade [2006]), but these typically compromise the favourable scaling of the original method. In addition to this, it is often remarked that gradient-based methods can be difficult to tune. Algorithm performance is often highly sensitive to the choice of scale within the proposal [Neal, 2003, Fig.15], and if this is chosen to be too large in certain directions then performance can degrade rapidly. Because of this, practitioners must spend a long time adjusting the tuning parameters to ensure that the algorithm is running well, or develop sophisticated adaptation schemes for this purpose (e.g. Hoffman and Gelman [2014]), which can nonetheless still require a large number of iterations to find good tuning parameters (see Sections 5 and 6). We will refer to this issue as *robustness to tuning*.

In this article we present a new gradient-based MCMC scheme, *the Barker proposal*, which combines favourable high-dimensional scaling properties with favourable ergodicity and robustness to tuning properties. To motivate the new scheme, in Section 2 we present a direct argument showing how the spectral gaps for the random walk, Langevin and Hamiltonian algorithms behave as the tuning parameters are chosen to be increasingly unsuitable for the problem at hand. In particular, we show that the spectral gaps for commonly used gradient-based algorithms decay to zero exponentially fast in the degree of mismatch between the scales of the proposal and target distributions, while for the random walk Metropolis (RWM) the decay is polynomial. In Section 3 we derive the Barker proposal scheme beginning from a family of π-invariant continuous-time jump processes, and discuss its connections to the concept of 'locally-balanced' proposals, introduced in [Zanella, 2019] for discrete state spaces. The name *Barker* comes from the particular choice of 'balancing function' used to uncover the scheme, which is inspired by the classical Barker accept-reject rule [Barker, 1965]. In Section 4 we conduct a detailed analysis of the ergodicity, scaling and robustness properties of this new method, establishing that it shares the favourable robustness to tuning of the random walk algorithm, can be geometrically ergodic in the presence of very light tails, and enjoys the $d^{-1/3}$ scaling with dimension of the Langevin scheme. The theory is then supported by an extensive simulation study in Sections 5 and 6, including comparisons with state-of-the-art alternative sampling schemes, which highlights that this kind of robustness is particularly advantageous in the context of adaptive MCMC. The code to reproduce the experiments is available from the online repository at the link `https://github.com/gzanella/barker`. Proofs and further numerical simulations are provided in the supplement.

1.1 Basic setup and notation

Throughout we work on the Borel space $(\mathbb{R}^d, \mathcal{B})$, with $d \geq 1$ indicating the dimension. For $\lambda \in \mathbb{R}$, we write $\lambda \uparrow \infty$ and $\lambda \downarrow 0$ to emphasize the direction of convergence when this is important. For two functions $f, g : \mathbb{R} \to \mathbb{R}$, we use the Bachmann–Landau notation $f(t) = \Theta(g(t))$ if $\liminf_{t\to\infty} f(t)/g(t) > 0$ and $\limsup_{t\to\infty} f(t)/g(t) < \infty$.

The Markov chains we consider will be of the Metropolis–Hastings type, meaning that the π-invariant kernel P is constructed as $P(x, dy) := \alpha(x, y)Q(x, dy) + r(x)\delta_x(dy)$, where $Q : \mathbb{R}^d \times \mathcal{B} \to [0, 1]$ is a candidate kernel,

$$\alpha(x, y) := \min\left(1, \frac{\pi(dy)Q(y, dx)}{\pi(dx)Q(x, dy)}\right) \tag{1}$$

is the *acceptance rate* for a proposal y given the current point x (provided that the expression is well-defined, see Tierney [1998] for details here), and $r(x) := 1 - \int \alpha(x, y)Q(x, dy)$ is the average probability of rejection given that the current point is x.

2 Robustness to tuning

In this section, we seek to quantify the robustness of the random walk, Langevin and Hamiltonian schemes with respect to the mismatch between the scales of $\pi(\cdot)$ and Q in a given direction. Unlike other analyses in the MCMC literature (e.g. Roberts and Rosenthal [2001], Beskos et al. [2018]), we are interested in studying how MCMC algorithms perform when they are *not* optimally tuned, in order to understand how crucially performance depends on such design choices (e.g. the choice of proposal step-size or pre-conditioning matrix). The rationale for performing such an analysis is that achieving optimal or even close to optimal tuning can be extremely challenging in practice, especially when $\pi(\cdot)$ exhibits substantial heterogeneity. This is typically done using past samples in the chain to compute online estimates of the average acceptance rate and the covariance of π (or simply its diagonal terms for computational convenience), and then using those estimates to tune the proposal step-sizes in different directions [Andrieu and Thoms, 2008]. If the degree of heterogeneity is large, it can take a long time for certain directions to be well-explored, and hence for the estimated covariance to be representative and the tuning parameters to converge.

In such settings, algorithms that are more robust to tuning are not only easier to use when such tuning is done manually by the user, but can also greatly facilitate the process of learning the tuning parameters *adaptively* within the algorithm. We show in Sections 5 and 6 that if an algorithm is robust to tuning then this adaptation process can be orders of magnitude faster than in the alternative case, drastically reducing the overall computational cost for challenging targets. The intuition for this is that more robust algorithms will start performing well (i.e. sampling efficiently) earlier in the adaptation process (when tuning parameters are not yet optimally tuned), which in turn will speed up the exploration of the target and the learning of the tuning parameters.

2.1 Analytical framework

The most general scenario we consider is a family of target densities $\pi^{(\lambda,k)}$ indexed by $\lambda > 0$ and $k \in \{1, ..., d\}$ defined as

$$\pi^{(\lambda,k)}(x) := \lambda^{-k}\pi(x_1/\lambda, \ldots, x_k/\lambda, x_{k+1}, \ldots, x_d), \qquad x = (x_1, \ldots, x_d) \in \mathbb{R}^d, \tag{2}$$

where π is a density defined on $\mathbb{R}^d$ for which $\pi(x) > 0$ for all $x \in \mathbb{R}^d$ and $\log \pi \in C^1(\mathbb{R}^d)$. The set up allows modification of the scale of the first k components of $\pi^{(\lambda,k)}$ through the parameter λ. Our main results are presented for the case $k = 1$, and we write $\pi^{(\lambda)} := \pi^{(\lambda,1)}$ for simplicity,

before discussing extensions to the $k > 1$ setting in Section 2.5. We consider targeting $\pi^{(\lambda)}$ using a Metropolis–Hastings algorithm with fixed tuning parameters, and study performance as λ varies. Intuitively, we can think of λ as a parameter quantifying the level of heterogeneity in the problem. As a concrete example, consider a random walk Metropolis algorithm in which given the current state $x^{(t)}$ the candidate move is $y = x^{(t)} + \sigma\xi$, with $\sigma > 0$ a fixed tuning parameter and $\xi \sim N(0, \mathbb{I}_d)$, where $\mathbb{I}_d$ is the $d \times d$ identity matrix. It is instructive to take σ as the optimal choice of global scale for π, meaning when λ is far from one then σ is no longer a suitable choice for the first coordinate of $\pi^{(\lambda)}$.

In the context of the above, the $\lambda \downarrow 0$ regime is representative of distributions in which one component (in this case the first) has a very small scale compared to all others. Conversely the $\lambda \uparrow \infty$ regime reflects the case in which one component has a much larger scale than its counterparts. Studying robustness to tuning in the context of heterogeneity is particularly relevant, as highlighted above, as this is exactly the context in which tuning is more challenging. The $\lambda \downarrow 0$ regime is particularly interesting and has been recently considered in Beskos et al. [2018], where the authors study the behaviour of the random walk Metropolis for 'ridged' densities for different values of k using a diffusion limit approach. The focus in that work, however, was on the finding optimal tuning parameters for the algorithm as a function of λ, whereas the present paper is concerned with the regime in which the tuning parameters are fixed (as discussed above).

The above framework could be equivalently formulated by keeping the target distribution π fixed and instead rescaling the first component of the candidate kernel by a factor $1/\lambda$. This is indeed the formulation we mostly use in the proofs of our theoretical results. A proof of the mathematical equivalence between the two formulations can be found in the supplement.

2.2 Measure of performance

Our measure of performance for the various algorithms will be the spectral gap of the resulting Markov chains. Consider the space of functions

$$L^2_{0,1}(\pi) = \{f : \mathbb{R}^d \to \mathbb{R} \mid \mathbb{E}_\pi[f] = 0, \mathrm{Var}_\pi[f] = 1\}.$$

Note that any function g with $\mathbb{E}_\pi g^2 < \infty$ can be associated with an $f \in L^2_{0,1}(\pi)$ through the map $f = (g - \mathbb{E}_\pi g)/\sqrt{\mathrm{Var}_\pi g}$, and that if $X^{(t)} \sim \pi(\cdot)$ and $X^{(t+1)}|X^{(t)} \sim P(X^{(t)}, \cdot)$ then $\mathrm{Corr}\{g(X^{(t)}), g(X^{(t+1)})\} = \mathrm{Corr}\{f(X^{(t)}), f(X^{(t+1)})\}$. The (right) spectral gap of a π-reversible Markov chain with transition kernel P is

$$\mathrm{Gap}(P) = \inf_{f \in L^2_{0,1}(\pi)} \frac{1}{2} \int (f(y) - f(x))^2 \, \pi(dx) P(x, dy). \tag{3}$$

The expression inside the infimum is called a *Dirichlet form*, and can be thought of as the 'expected squared jump distance' for the function f provided the chain is stationary. This can in turn be re-written as $1 - \mathrm{Corr}\{f(X^{(t)}), f(X^{(t+1)})\}$. Maximising the spectral gap of a reversible Markov chain can therefore be understood as minimising the *worst-case* first-order auto-correlation among all possible square-integrable test functions.

The spectral gap allows to bound the variances of ergodic averages (see Proposition 1 of Rosenthal, 2003). Also, a direct connection between the spectral gap and mixing properties of the chain can be made if the operator $Pf(x) := \int f(y) P(x, dy)$ is positive on $L^2(\pi)$. This will always be the case if the chain is made lazy, which is the approach taken in Woodard et al. [2009], and the same adjustment can be made here if desired.

2.3 The small λ regime

In this section we assess the robustness to tuning of the random walk, Langevin and Hamiltonian schemes as $\lambda \downarrow 0$. This corresponds to the case in which the proposal scale is chosen to be too

large in the first component of $\pi^{(\lambda)}$. The results in this section will support the idea that classical gradient-based schemes pay a very high price for any direction in which this tuning parameter is chosen to be too large, as already noted in the literature (e.g. Neal, 2003, page 738), while the random walk Metropolis is less severely affected by such issues.

2.3.1 Random walk Metropolis

In the random walk Metropolis (RWM), given a current point $x \in \mathbb{R}^d$, a proposal y is calculated using the equation

$$y = x + \sigma\xi, \tag{4}$$

with $\sigma > 0$ and $\xi \sim \mu(\cdot)$ for some centred symmetric distribution μ. The resulting candidate kernel Q^R is given by $Q^R(x, dy) = q^R(x, y)dy$ with $q^R(x, y) = \sigma^{-d}\mu((y - x)/\sigma)$, where $\mu(\xi)$ for $\xi \in \mathbb{R}^d$ denotes the density of μ. Following Section 2.1, we consider Metropolis–Hastings algorithms with proposal Q^R and target distribution $\pi^{(\lambda)}$ defined in (2), and denote the resulting transition kernels as P^R_λ.

We impose the following mild regularity conditions on the density $\mu(\xi)$. These are satisfied for most popular choices of μ, as shown in the subsequent proposition.

Condition 2.1. *There exists $\lambda_0 > 0$ such that for any $x, y \in \mathbb{R}^d$ and $\lambda < \lambda_0$we have $\mu(\delta_\lambda) \geq \mu(\delta)$, where $\delta = y - x$ and*

$$\delta_\lambda := (\lambda(y_1 - x_1), y_2 - x_2, ..., y_d - x_d)\,. \tag{5}$$

In addition, $\sup_{\xi_1 \in \mathbb{R}} \mu_1(\xi_1) < \infty$, where $\mu_1(\xi_1) = \int_{\mathbb{R}^{d-1}} \mu(\xi_1, \xi_2, \ldots, \xi_d)d\xi_2 \ldots d\xi_d$ is the marginal distribution of ξ_1 under $\xi \sim \mu$.

Proposition 2.1. *Denoting the usual p-norm as $\|x\|_p = (\sum_{i=1}^d x_i^p)^{1/p}$, Condition 2.1 holds in each of the below cases:*
(i) $q^R(x, y) = (2\pi\sigma^2)^{-d/2} \exp\left(-\|x - y\|_2^2/(2\sigma^2)\right)$ (Gaussian)
(ii) $q^R(x, y) = 2^{-d} \exp\left(-\|x - y\|_1\right)$ (Laplace)
(iii) $q^R(x, y) \propto (1 + \|y - x\|_2^2/\nu)^{-(\nu+d)/2}$ for $\nu \in \{1, 2, ...\}$ (Student's t)

We conclude the section with a characterization of the rate of convergence to zero of the spectral gap for the Random Walk Metropolis as $\lambda \downarrow 0$.

Theorem 2.1. *Assume Condition 2.1 and* $\mathrm{Gap}(P^R_1) > 0$. *Then it holds that*

$$\mathrm{Gap}(P^R_\lambda) = \Theta(\lambda)\,, \quad \text{as } \lambda \downarrow 0\,.$$

Note that Theorem 2.1 requires very few assumptions on the target π other than $\mathrm{Gap}(P^R_1) > 0$. Note also that the lower bound is of the form $\mathrm{Gap}(P^R_\lambda) \geq \lambda\mathrm{Gap}(P^R_1)$, see proof of Theorem 2.1 for details. No dependence on the dimension of the problem other than that intrinsic to $\mathrm{Gap}(P^R_1)$ is therefore introduced.

2.3.2 The Langevin algorithm

In the Langevin algorithm (or more specifically the Metropolis-adjusted Langevin algorithm, MALA), given the current point $x \in \mathbb{R}^d$, a proposal y is generated by setting

$$y = x + \frac{\sigma^2}{2}\nabla \log \pi^{(\lambda)}(x) + \sigma\xi, \tag{6}$$

for some $\sigma > 0$ and $\xi \sim N(0, \mathbb{I}_d)$. In this case the proposal is no longer symmetric and so the full Hastings ratio (1) must be used. The proposal mechanism is based on the overdamped Langevin stochastic differential equation $dX_t = \nabla \log \pi^{(\lambda)}(X_t)dt + \sqrt{2}dB_t$. We write Q^M_λ for the

corresponding candidate distribution and P_λ^M for the Metropolis–Hastings kernel with proposal Q_λ^M and target $\pi^{(\lambda)}$.

We present results for the Langevin algorithm in two settings. Initially we consider more restrictive conditions under which our upper bound on the spectral gap depends on the tail behaviour of π in a particularly explicit manner, and then give a broader result.

Condition 2.2. *Assume the following:*

(i) π has a density of the form $\pi(x) = \pi_1(x_1)\pi_{2:n}(x_2, ..., x_d)$, for some densities π_1 and $\pi_{2:n}$ on $\mathbb{R}$ and $\mathbb{R}^{d-1}$, respectively.

(ii) For some $q \in [0, 1)$, it holds that

$$\left| \frac{d}{dx_1} \log \pi_1(x_1) \right| = \Theta\left(|x_1|^q\right) \qquad \text{as } |x_1| \uparrow \infty. \tag{7}$$

Theorem 2.2. *If Condition 2.2 holds, then there is a $\gamma > 0$ such that*

$$\mathrm{Gap}(P_\lambda^M) \leq \Theta\left(e^{-\gamma\lambda^{-(1+q)} + q\log(\lambda)}\right) \qquad \text{as } \lambda \downarrow 0\,.$$

When compared with the random walk algorithm, Theorem 2.2 shows that the Langevin scheme is much less robust to heterogeneity. Indeed, the spectral gap decays *exponentially fast* in $\lambda^{-(1+q)}$, meaning that even small errors in the choice of step-size can have a large impact on algorithm efficiency, and so practitioners must invest considerable effort tuning the algorithm for good performance, as shown through simulations in Sections 5 and 6. Theorem 2.2 also illustrates that the Langevin algorithm is more sensitive to λ when the tails of $\pi(\cdot)$ are lighter. This is intuitive, as in this setting gradient terms can become very large in certain regions of the state space.

Remark 2.1. *Theorem 2.2 (and Theorem 2.4 below) could be extended to the case $q \geq 1$ in (7), however in these cases samplers typically fail to be geometrically ergodic when λ is small [Roberts and Tweedie, 1996, Livingstone et al., 2019] meaning the spectral gap is typically 0 and the theorem becomes trivial.*

Remark 2.2. *Condition 2.2 (ii) could be replaced with the simpler requirement that $|\nabla \log \pi_1(x_1)| \uparrow \infty$, with the corresponding bound $\mathrm{Gap}(P_\lambda^M) \leq \Theta(e^{-1/\lambda})$.*

A different set of conditions, which hold much more generally, and corresponding upper bound are presented below.

Condition 2.3. *Assume the following:*

(i) There is a $\gamma > 0$ such that

$$\liminf_{|x_1| \to \infty} \left(\inf_{(x_2,\ldots,x_d) \in \mathbb{R}^{d-1}} \left| \frac{\partial \log \pi(x)}{\partial x_1} \right| \|x\|_2^\gamma \right) > 0\,, \tag{8}$$

(ii) Given $X \sim \pi$ there is a $\beta > 0$ such that

$$\mathbb{P}\left(\|X\|_2 > t\right) \leq \Theta\left(e^{-t^\beta}\right) \qquad \text{as } t \to \infty\,. \tag{9}$$

Theorem 2.3. *If Condition 2.3 holds, then*

$$\mathrm{Gap}(P_\lambda^M) \leq \Theta(e^{-\lambda^{-\alpha}}) \qquad \text{as } \lambda \downarrow 0\,.$$

for some $\alpha > 0$, which can be taken as $\alpha = \min\{\beta/2, \beta/\gamma, 2/3\}$.

We expect Condition 2.3 to be satisfied in many commonly encountered scenarios, with the exception of particularly heavy-tailed models. In the exponential family class $\pi(x) \propto \exp\{-\alpha\|x\|_2^\beta\}$, for example, Condition 2.3 holds for any α and $\beta > 0$ (see proof in the supplement).

2.3.3 Hamiltonian Monte Carlo

In Hamiltonian Monte Carlo (HMC) we write the current point $x \in \mathbb{R}^d$ as $x(0)$, and construct the proposal $y := x(L)$ for some prescribed integer L using the update

$$x(L) = x(0) + \sigma^2 \left(\frac{L}{2} \nabla \log \pi^{(\lambda)}(x(0)) + \sum_{j=1}^{L-1} (L-j) \nabla \log \pi^{(\lambda)}(x(j)) \right) + L\sigma\xi(0), \tag{10}$$

where each $x(j)$ is defined recursively in the same manner, and $\xi(0) \sim N(0, \mathbb{I}_d)$. The transition is based on numerically solving Hamilton's equations for the Hamiltonian system $H(x, \xi) = -\log \pi^{(\lambda)}(x) + \xi^T \xi/2$ for $L\sigma$ units of time. The decision of whether or not the proposal is accepted is taken using the acceptance probability $\min(1, \pi^{(\lambda)}(y)/\pi^{(\lambda)}(x) \times e^{-\xi(L)^T \xi(L)/2 + \xi(0)^T \xi(0)/2})$, where

$$\xi(L) = \xi(0) + \frac{\sigma}{2} \left(\nabla \log \pi^{(\lambda)}(x(0)) + \nabla \log \pi^{(\lambda)}(x(L)) \right) + \sigma \sum_{j=1}^{L-1} \nabla \log \pi^{(\lambda)}(x(j)).$$

A more detailed description is given in Neal [2011]. We write P_λ^H for the corresponding Metropolis–Hastings kernel with proposal mechanism as above and target $\pi^{(\lambda)}$. Here we present a heterogeneity result under Condition 2.2 of the previous subsection.

Theorem 2.4. *If Condition 2.2 holds, then there is a $\gamma > 0$ such that*

$$\mathrm{Gap}(P_\lambda^H) \leq \Theta\left(e^{-\gamma \lambda^{-(1+q)} + q \log(\lambda)} \right) \qquad \text{as } \lambda \downarrow 0 \,.$$

It is no surprise that Theorem 2.4 is comparable to Theorem 2.2, since setting $L = 1$ equates the Langevin and Hamiltonian methods.

2.4 The large λ regime

In this section we briefly discuss the $\lambda \uparrow \infty$ regime, where σ is chosen to be too small for the first component of $\pi^{(\lambda)}$, arguing that all samplers under consideration behave similarly in this regime and pay a similar price for too small tuning parameters in a given direction. The intuition for this is that as $\lambda \uparrow \infty$ the gradient-based proposal mechanisms discussed here all tend towards that of the random walk sampler in the first coordinate. For example, if we consider one-dimensional models, for any $x \in \mathbb{R}$ we can write $\nabla \log \pi^{(\lambda)}(x) = \lambda^{-1} \nabla \log \pi(x/\lambda)$, meaning as $\lambda \uparrow \infty$ the amount of gradient information in the proposal is reduced provided π is suitably regular. The following result makes this intuition precise. To avoid repetitions, we state here the result for both the Langevin and the Barker proposal that we will introduce in the next section.

Proposition 2.2. *Fix $x \in \mathbb{R}$ and let the density π be such that $\nabla \log \pi$ is bounded in some neighbourhood of zero. Then the Langevin and Barker candidate kernels Q_λ^M and Q_λ^B, defined in* (6) *and* (16) *respectively, both satisfy*

$$\| Q_\lambda^{M/B}(x, \cdot) - Q^R(x, \cdot) \|_{TV} \leq \Theta\left(1/\lambda\right),$$

where Q^R is the (Gaussian) random walk candidate kernel.

The same intuition applies to the Hamiltonian case provided L is fixed, since each gradient term in the proposal is also $\Theta(1/\lambda)$. While there are many well-known measures of distance between two distributions, we argue that total variation is an appropriate choice here, since it has an explicit focus on how much the two kernels overlap and is invariant under bijective transformations of the state space (including re-scaling coordinates).

While the above statements provide useful heuristic arguments, in order to obtain more rigorous results one should prove that the spectral gaps decay to 0 at the same rate as $\lambda \uparrow \infty$, which we leave to future work. We note, however, that the conjecture that the algorithms behave similarly for large values of λ is supported by the simulations of Section 5.1.

2.5 Extensions

The lower bound of Theorem 2.1 extends naturally to the $k > 1$ setting, becoming instead $\geq \Theta(\lambda^k)$, and so the rate of decay remains polynomial in λ for any k. Analogously, we expect the corresponding upper bound for gradient-based schemes to remain exponential and become $\leq \Theta\left(e^{-k(\gamma\lambda^{-(1+q)}+q\log(\lambda))}\right)$, although the details of this are left for future work. We explore examples of this nature through simulations in Section 5 and find empirically that the single component case is informative also of more general cases. Further extensions in which a different λ_i is chosen in each of the k directions can also be considered, with each $\lambda_i \downarrow 0$ at a different rate. We conjecture that in this setting the λ_i that decays most rapidly will dictate the behaviour of spectral gaps, though such an analysis is beyond the scope of the present work.

3 Combining robustness and efficiency

The results of Section 2 show that the two gradient-based samplers considered there are much less robust to heterogeneity than the random walk algorithm. In this section, we introduce a novel and simple to implement gradient-based scheme that shares the superior scaling properties of the Langevin and Hamiltonian schemes, but also retains the robustness of the random walk sampler, both in terms of geometric ergodicity and robustness to tuning.

3.1 Locally-balanced Metropolis–Hastings

Consider a continuous-time Markov jump process on $\mathbb{R}^d$ with associated generator

$$\mathcal{L}f(x) = \int [f(y) - f(x)] g\left(\frac{\pi(y)q(y,x)}{\pi(x)q(x,y)}\right) Q(x, dy), \tag{11}$$

for some suitable function $f : \mathbb{R}^d \to \mathbb{R}$, where $\pi(x)$ is a probability density, $Q(x, dy) := q(x, y)dy$ is a transition kernel and the *balancing* function $g : (0, \infty) \to (0, \infty)$ satisfies

$$g(t) = tg\left(1/t\right). \tag{12}$$

A discrete state-space version of this process with symmetric Q was introduced in Power and Goldman [2019]. The dynamics of the process are such that if the current state $X_t = x$, the next jump will be determined by a Poisson process with intensity

$$Z(x) := \int g\left(\frac{\pi(y)q(y,x)}{\pi(x)q(x,y)}\right) Q(x, dy), \tag{13}$$

and the next state is drawn from the kernel

$$Q^{(g)}(x, dy) := Z(x)^{-1} g\left(\frac{\pi(y)q(y,x)}{\pi(x)q(x,y)}\right) Q(x, dy).$$

The $L^2(\mathbb{R}^d)$ adjoint, or *forward operator* $\mathcal{A}$ (e.g. Fearnhead et al. [2018]) is given by

$$\mathcal{A}h(x) = \int h(y) g\left(\frac{\pi(x)q(x,y)}{\pi(y)q(y,x)}\right) q(y, x)dy - h(x)Z(x).$$

Note that in the case $h(x) = \pi(x)$ using (12) the first expression on the right-hand side can be written

$$\int g\left(\frac{\pi(y)q(y,x)}{\pi(x)q(x,y)}\right)\pi(x)q(x,y)dy = \pi(x)Z(x),$$

meaning $\mathcal{A}\pi = 0$, suggesting π is invariant. It can therefore serve as a starting point for designing Markov chain Monte Carlo algorithms.

In the 'locally-balanced' framework for discrete state-space Metropolis–Hastings introduced in Zanella [2019], candidate kernels are of the form

$$\tilde{Q}(x,dy) = \tilde{Z}(x)^{-1}g\left(\frac{\pi(y)}{\pi(x)}\right)\mu_\sigma(y-x)dy, \tag{14}$$

meaning the *embedded Markov chain* of (11) with the choice $Q(x,dy) := \mu_\sigma(y-x)dy$, where $\mu_\sigma(y-x) := \sigma^{-d}\mu((y-x)/\sigma)$ for some symmetric density μ. It is well-known that the invariant density of the embedded chain does not coincide with that of the process when jumps are not of constant intensity, in this case becoming proportional to $Z(x)\pi(x)$, as shown in Zanella [2019]. As a result a Metropolis–Hastings step is employed to correct for the discrepancy. In Power and Goldman [2019] it is suggested that as an alternative the jump process can be simulated exactly.

The challenge with employing either of these strategies on a continuous state space is that the integral (13) will typically be intractable. To overcome this issue we take two steps, and for simplicity we first describe these on $\mathbb{R}$ (there are two options on $\mathbb{R}^d$ for $d > 1$, which are discussed in Section 3.3). The first step is to consider a first-order Taylor series expansion of $\log\pi$ within g (again with a symmetric choice of Q), leading to the family of processes with generator

$$Lf(x) = \int [f(y) - f(x)]g\left(e^{\nabla \log\pi(x)(y-x)}\right)\mu_\sigma(y-x)dy.$$

We refer to candidate kernels in Metropolis–Hastings algorithms that are constructed using the embedded Markov chain of this new process as *first-order* locally-balanced proposals, taking the form

$$Q^{(g)}(x,dy) = Z(x)^{-1}g\left(e^{\nabla \log\pi(x)(y-x)}\right)\mu_\sigma(y-x)dy, \tag{15}$$

where $Z(x) := \int g(e^{\nabla \log\pi(x)(y-x)})\mu_\sigma(y-x)dy$. This second step is to note that, if particular choices of g are made, then $Z(x)$ becomes tractable. In fact, if the balancing function $g(t) = \sqrt{t}$ and a Gaussian kernel μ_σ are chosen, then the result is the Langevin proposal

$$Q^M(x,dy) \propto e^{\nabla \log\pi(x)(y-x)/2}\mu_\sigma(y-x)dy.$$

Thus, MALA can be viewed as a particular instance of this class. Other choices of g are, however, also possible, and give rise to different gradient-based algorithms. In the next section we explore what a sensible choice of g might look like.

Remark 3.1. *One can also think at* (12) *as a requirement to ensure that the proposals in* (15) *are exact (i.e. π-reversible) at the first order. In particular, in the supplement it is shown that a proposal $Q^{(g)}$ defined as in* (15) *is π-reversible with respect to log-linear density functions if and only if* (12) *holds.*

3.2 The Barker proposal on $\mathbb{R}$

The requirement for the balancing function g to satisfy $g(t) = tg(1/t)$ is in fact also imposed on the acceptance rate of a Metropolis–Hastings algorithm to produce a π-reversible Markov chain. Indeed, setting $t := \pi(y)q(y,x)/(\pi(x)q(x,y))$ and assuming $\alpha(x,y) := \alpha(t)$, then the detailed

balance equations can be written $\alpha(t) = t\alpha(1/t)$. Possible choices of g can therefore be found by considering suggestions for α in the literature. One choice proposed in Barker [1965] is

$$g(t) = \frac{t}{1+t}.$$

The work of Peskun [1973] and Tierney [1998] showed that this choice of α is inferior to the more familiar Metropolis–Hasting rule $\alpha(t) = \min(1, t)$ in terms of asymptotic variance. The same conclusion cannot, however, be drawn when considering the choice of balancing function g.

In fact, the choice $g(t) = t/(1+t)$ was shown to minimize asymptotic variances of Markov chain estimators in some discrete settings in Zanella [2019]. In addition, as shown below, this particular choice of g leads to a fully tractable candidate kernel that can be easily sampled from.

Proposition 3.1. *If $g(t) = t/(1+t)$, then the normalising constant $Z(x)$ in (15) is $1/2$.*

The resulting proposal distribution is

$$Q^B(x, dy) = 2\frac{\mu_\sigma(y-x)}{1+e^{-\nabla \log \pi(x)(y-x)}}dy. \tag{16}$$

We refer to Q^B as the *the Barker proposal*. A simple sampling strategy to generate $y \sim Q^B(x, \cdot)$ is given in Algorithm 1.

Algorithm 1 Generating a Barker proposal on $\mathbb{R}$

Require: the current point $x \in \mathbb{R}$.

1. Draw $z \sim \mu_\sigma(\cdot)$
2. Calculate $p(x, z) = 1/(1 + e^{-z\nabla \log \pi(x)})$
3. Set $b(x, z) = 1$ with probability $p(x, z)$, and $b(x, z) = -1$ otherwise
4. Set $y = x + b(x, z) \times z$

Output: the resulting proposal y.

Proposition 3.2. *Algorithm 1 produces a sample from Q^B on $\mathbb{R}$.*

Algorithm 1 shows that the magnitude $|y - x| = |z|$ of the proposed move does not depend on the gradient $\nabla \log \pi(x)$ here, it is instead dictated only by the choice of symmetric kernel μ_σ. The *direction* of the proposed move is, however, informed by both the magnitude and direction of the gradient. Examining the form of $p(x, z)$, it becomes clear that if the signs of z and $\nabla \log \pi(x)$ are in agreement, then $p(x, z) > 1/2$, and indeed as $z\nabla \log \pi(x) \uparrow \infty$ then $e^{-z\nabla \log \pi(x)} \downarrow 0$ and so $p(x, z) \uparrow 1$. Hence, if the indications from $\nabla \log \pi(x)$ are that $\pi(x + z) \gg \pi(x)$, then it is highly likely that $b(x, z)$ will be set to 1 and $y = x + z$ will be the proposed move. Conversely, if $z\nabla \log \pi(x) < 0$, then there is a larger than 50% chance that the proposal will instead be $y = x - z$. As $\nabla \log \pi(x) \uparrow \infty$ the Barker proposal converges to μ_σ truncated on the right, and similarly to μ_σ truncated on the left as $\nabla \log \pi(x) \downarrow -\infty$. See Figure 1 for an illustration.

The multiplicative term $1/(1 + e^{-\nabla \log \pi(x)(y-x)})$ in (16), which incorporates the gradient information, injects skewness into the base kernel μ_σ (as can be clearly seen in the left-hand plot of Figure 1). Indeed, the resulting distribution Q^B is an example of a *skew-symmetric* distribution [Azzalini, 2013, eq.(1.3)]. Skew-symmetric distributions are a tractable family of (skewed) probability density functions that are obtained by multiplying a symmetric base density function with the cumulative distribution function (cdf) of a symmetric random variable. We refer

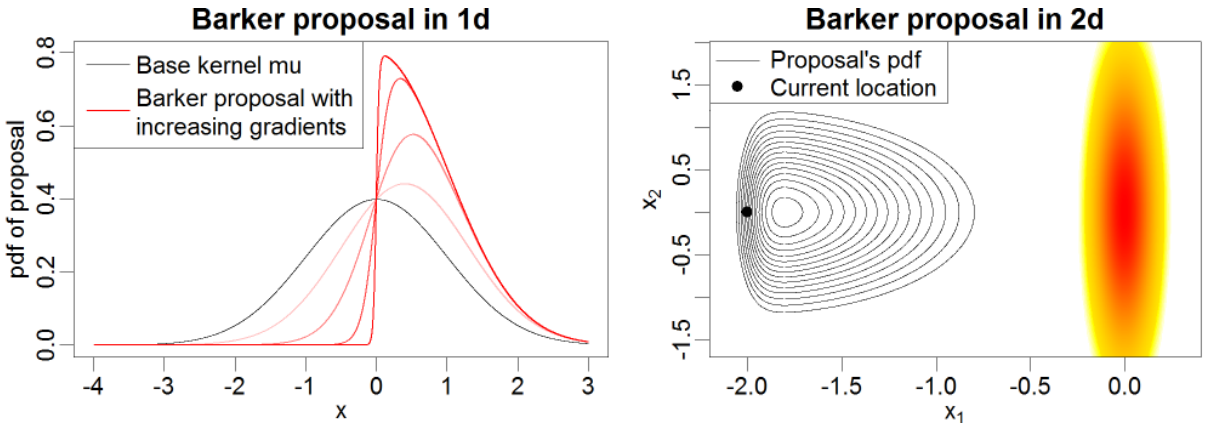

Figure 1: Left: density of the Barker proposal in one dimension. Current location is $x = 0$ and the four lines with increasing red intensity correspond to $\nabla \log \pi(x)$ equal to 1, 3, 10 and 50. Right: density of the Barker proposal in two dimensions. Solid lines display the proposal density contours, heat colours refer to the target density, and the current location is $x = (-2, 0)$.

to Azzalini [2013, Ch.1] for more details, including a more general version of Propositions 3.1 and 3.2. In the context of skewed distributions the Gaussian cdf is often used, leading to the skew-normal distribution introduced in Azzalini [1985]. In our context, however, the Barker proposal (which leads to the logistic cdf $p(x, z)$ in Algorithm 1) is the only skew-symmetric distribution that can be obtained from (15) using a balancing function g satisfying $g(t) = tg(1/t)$. See the supplement for more detail.

3.3 The Barker proposal on $\mathbb{R}^d$

There are two natural ways to extend the Barker proposal to $\mathbb{R}^d$, for $d > 1$. The first is to treat each coordinate separately, and generate the proposal $y = (y_1, ..., y_d)$ by applying Algorithm 1 independently to each coordinate. This corresponds to generating a z_i and $b_i(x, z_i)$ for each $i \in \{1, ..., d\}$, and choosing the sign of each b_i using

$$p_i(x, z_i) = \frac{1}{1 + e^{-z_i \partial_i \log \pi(x)}},$$

where $\partial_i \log \pi(x)$ denotes the partial derivative of $\log \pi(x)$ with respect to x_i. Writing $Q_i^B(x, dy_i)$ to denote the resulting Barker proposal candidate kernel for the ith coordinate, the full candidate kernel Q^B can then be written

$$Q^B(x, dy) = \prod_{i=1}^{d} Q_i^B(x, dy_i). \tag{17}$$

The full Metropolis–Hastings scheme using the Barker proposal mechanism for a target distribution is given in Algorithm 2 (see the supplement for more details and variations of the algorithm, such as a pre-conditioned version). Note that the computational cost of each iteration of the algorithm is essentially equivalent to that of MALA and will be typically dominated by the cost of computing the gradient and density of the target.

The second approach to deriving a multivariate Barker proposal consists of sampling $z \in \mathbb{R}^d$ from a d-dimensional symmetric distribution, and then choosing whether or not to flip the sign of *every* coordinate at the same time, using a single global $\check{b}(x, z) \in \{-1, 1\}$, to produce the global proposal $y = x + \check{b}(x, z) \times z$. In this case the probability that $\check{b}(x, z) = 1$ will be

$$\check{p}(x, z) = \frac{1}{1 + e^{-z^T \nabla \log \pi(x)}}. \tag{19}$$

Algorithm 2 Metropolis–Hastings with the Barker proposal on $\mathbb{R}^d$

Require: starting point for the chain $x^{(0)} \in \mathbb{R}^d$, and scale $\sigma > 0$.
Set $t = 0$ and do the following:

1. Given $x^{(t)} = x$, draw y_i using Algorithm 1 independently for $i \in \{1, ..., d\}$
2. Set $x^{(t+1)} = y$ with probability $\alpha^B(x, y)$, where

$$\alpha^B(x, y) = \min\left(1, \frac{\pi(y)}{\pi(x)} \times \prod_i \frac{1 + e^{(x_i - y_i)\partial_i \log \pi(x)}}{1 + e^{(y_i - x_i)\partial_i \log \pi(y)}}\right). \tag{18}$$

   Otherwise set $x^{(t+1)} = x$
3. If $t + 1 < N$, set $t \leftarrow t + 1$ and return to step 1, otherwise stop.

Output: the Markov chain $\{x^{(0)}, \ldots, x^{(N)}\}$.

This second approach doesn't allow gradient information to feed into the proposal as effectively as in the first case. Specifically, only the global inner product $z^T \nabla \log \pi(x)$ is considered, and the decision to alter the sign of every component of z is taken based solely on this value. In other words, once $z \sim \mu_\sigma$ has been sampled, gradient information is only used to make a single binary decision of choosing between the two possible proposals $x+z$ and $x-z$, while in the first strategy gradient information is used to choose between 2^d possible proposals $\{x + b \cdot z : b \in \{-1, 1\}^d\}$ (where $b \cdot z := (b_1 z_1, ..., b_d z_d)$). Indeed, the following proposition shows that the second strategy cannot improve over the random walk Metropolis by more than a factor of two.

Proposition 3.3. *Let $\check{P}^B$ denote the modified Barker proposal on $\mathbb{R}^d$ using* (19)*. Then Gap$(P^R) \geq$ Gap$(\check{P}^B)/2$.*

One can also make a stronger statement than the above proposition, namely that if this strategy is employed, only a constant factor improvement over the Random Walk Metropolis can be achieved in terms of asymptotic variance, for any $L^2(\pi)$ function of interest. Given Proposition 3.3 we choose to use the first strategy described to produce Barker proposals on $\mathbb{R}^d$, and the multi-dimensional candidate kernel given in (17). In the following sections we will show both theoretically and empirically that this choice does indeed have favourable robustness and efficiency properties.

4 Robustness, scaling and ergodicity results for the Barker proposal

In this section we establish results concerning robustness to tuning, scaling with dimension and geometric ergodicity for the Barker proposal scheme. As we will see, the method enjoys the superior efficiency of gradient-based algorithms in terms of scaling with dimension, but also shares the favourable robustness properties of the random walk Metropolis when considering both robustness to tuning and geometric ergodicity.

4.1 Robustness to tuning

We now examine the robustness to tuning of the Barker proposal using the framework introduced in Section 2. We write Q^B_λ and P^B_λ to denote the candidate and Metropolis–Hastings kernels for the Barker proposal targeting the distribution $\pi^{(\lambda)}$ defined therein, and P^B for the case $\lambda = 1$. The following result characterizes the behaviour of the spectral gap of P^B_λ as $\lambda \downarrow 0$.

Theorem 4.1. *Assume Condition 2.1 and* $\mathrm{Gap}(P^B) > 0$. *Then it holds that*

$$\mathrm{Gap}(P^B_\lambda) = \Theta(\lambda) \,, \quad \text{as } \lambda \downarrow 0 \,.$$

Comparing Theorem 4.1 with Theorems 2.1-2.4 from Section 2.3 we see that the Barker proposal inherits the robustness to tuning of random walk schemes and is significantly more robust than the Langevin and Hamiltonian algorithms. In the next section we establish general conditions under which $\mathrm{Gap}(P^B) > 0$.

4.2 Geometric ergodicity

In this section we study the class of target distributions for which the Barker proposal produces a geometrically ergodic Markov chain. We show that geometric ergodicity can be obtained even when the gradient term in the proposal grows faster than linearly, which is typically not the case for MALA and HMC.

Recall that a Markov chain is called *geometrically ergodic* if

$$\|P^t(x,\cdot) - \pi(\cdot)\|_{TV} \leq CV(x)\rho^t, \qquad t \geq 1 \,, \tag{20}$$

for some $C < \infty$, Lyapunov function $V : \mathbb{R}^d \to [1,\infty)$, and $\rho < 1$, where $\|\mu(\cdot) - \nu(\cdot)\|_{TV} := \sup_{A\in\mathcal{B}} |\mu(A) - \nu(A)|$ for probability measures μ and ν. When such a condition can be established for a reversible Markov chain, then a Central Limit Theorem exists for any square-integrable function [Roberts and Rosenthal, 2004].

We prove geometric ergodicity results for generic proposals as in (15), assuming g to be bounded and monotone, and μ_σ to have lighter than exponential tails. Following the discussion in Section 3.3 we consider proposals that are independent across components, leading to

$$Q^{(g)}(x,dy) = \prod_{i=1}^d Q_i^{(g)}(x,dy_i) = \prod_{i=1}^d \frac{g\left(e^{\partial_i \log\pi(x)(y_i - x_i)}\right)\mu_\sigma(y_i - x_i)dy_i}{Z_i(x)} \,, \tag{21}$$

where $Z_i(x) := \int_{\mathbb{R}} g(e^{\partial_i \log\pi(x)(y_i-x_i)})\mu_\sigma(y_i - x_i)dy_i$. With a slight abuse of notation, we use μ_σ to represent one and d-dimensional densities. The Barker proposal in (17) is the special case obtained by taking $g(t) = t/(1+t)$.

For the results of this section, we make the simplifying assumption that π is spherically symmetric outside a ball of radius $R < \infty$.

Condition 4.1. *There exists* $R < \infty$ *and a differentiable function* $f : (0,\infty) \to (0,\infty)$ *with* $\lim_{r\to\infty} f'(r) = -\infty$ *and* $f'(r)$ *non-increasing for* $r > R$ *such that* $\log\pi(x) = f(\|x\|)$ *for* $r > R$.

Theorem 4.2. *Let* $g : (0,\infty) \to (0,\infty)$ *be a bounded and non-decreasing function,* $\int_{\mathbb{R}} \exp(sw)\mu_\sigma(w)dw < \infty$ *for every* $s > 0$, *and* $\inf_{w\in(-\delta,\delta)} \mu_\sigma(w) > 0$ *for some* $\delta > 0$. *If the target density* π *satisfies Condition 4.1, then the Metropolis–Hastings chain with proposal* $Q^{(g)}$ *is* π*-a.e. geometrically ergodic.*

We note that tail regularity assumptions such as Condition 4.1 are common in this type of analysis (e.g. Jarner and Hansen [2000], Durmus et al. [2017a]). As an intuitive example, the condition is satisfied in the exponential family $\pi(x) \propto \exp(-\alpha\|x\|^\beta)$ for all $\beta > 1$. As a contrast, for MALA and HMC it is known that for $\beta > 2$ the sampler fails to be geometrically ergodic [Roberts and Tweedie, 1996, Livingstone et al., 2019]. We expect the Barker proposal to be geometrically ergodic also for the case $\beta = 1$, although we do not prove it in this work.

4.3 Scaling with dimensionality

In this section we provide preliminary results suggesting that the Barker proposal enjoys scaling behaviour analogous to that of MALA in high-dimensional setings, meaning that under appropriate assumptions it requires the number of iterations per effective sample to grow as $\Theta(d^{1/3})$ with the number of dimensions d as $d \to \infty$. Similarly to Section 4.2, we prove results for general proposals $Q^{(g)}$ as in (21) with balancing functions g satisfying $g(t) = t\, g(1/t)$. The Barker proposal is a special case of the latter family.

We perform an asymptotic analysis for $d \to \infty$ using the framework introduced in Roberts et al. [1997]. The main idea is to study the rate at which the proposal step size σ needs to decrease as $d \to \infty$ to obtain well-behaved limiting behaviour for the MCMC algorithm under consideration (such as a $\Theta(1)$ acceptance rate and convergence to a non-trivial diffusion process after appropriate time re-scaling). Based on the rate of decrease of σ one can infer how the number of MCMC iterations required for each effective sample increases as $d \to \infty$. For example, in the case of the random walk Metropolis σ^2 must be scaled as $\Theta(d^{-1})$ as $d \to \infty$ to have a well-behaved limit [Roberts et al., 1997], which leads to RWM requiring $\Theta(d)$ iterations for each effective sample. By contrast, for MALA it is sufficient to take $\sigma^2 = \Theta(d^{-1/3})$ as $d \to \infty$, which leads to only $\Theta(d^{1/3})$ iterations for each effective sample [Roberts and Rosenthal, 1998]. While these analyses are typically performed under simplifying assumptions, such as having a target distribution with i.i.d. components, the results have been extended in many ways (e.g. removing the product-form assumption, see Mattingly et al. [2012]) obtaining analogous conclusions. See also Beskos et al. [2013] for optimal scaling analysis of HMC and Roberts and Rosenthal [2016] for rigorous connections between optimal scaling results and computational complexity statements.

In this section we focus on the scaling behaviour of Metropolis–Hastings algorithms with proposal $Q^{(g)}$ as in (21), when targeting distributions of the form $\pi(x) = \prod_{i=1}^{d} f(x_i)$, where f is a one-dimensional smooth density function. Given the structure of $Q^{(g)}$ and $\pi(\cdot)$, the acceptance rate takes the form $\alpha(x, y) = \min\left\{1, \prod_{i=1}^{d} \alpha_i(x_i, y_i)\right\}$, where

$$\alpha_i(x_i, y_i) = \frac{f(y_i)}{f(x_i)} \frac{g\left(e^{\phi'(y_i)(x_i - y_i)}\right)}{g\left(e^{\phi'(x_i)(y_i - x_i)}\right)} \frac{Z_i(x_i)}{Z_i(y_i)}\,, \tag{22}$$

and $\phi = \log f$. In such a context, the scaling properties of the MCMC algorithms under consideration are typically governed by the behaviour of $\log(\alpha_i(x_i, y_i))$ as y_i gets close to x_i, or more precisely by degree of the leading term in the Taylor series expansion of $\log(\alpha_i(x_i, x_i + \sigma u_i))$ in powers of σ as $\sigma \to 0$ for fixed x_i and u_i. For example, in the case of the random walk Metropolis one has $\log(\alpha_i(x_i, x_i + \sigma u_i)) = \Theta(\sigma)$ as $\sigma \to 0$, which in fact implies the proposal variance σ^2 must decrease at a rate $\Theta(d^{-1})$ to obtain a non-trivial limit. By contrast, when the MALA proposal is used, one has $\log(\alpha_i(x_i, x_i + \sigma u_i)) = \Theta(\sigma^3)$ as $\sigma \to 0$, which in turn leads to $\sigma^2 = \Theta(d^{-1/3})$. See Sections 2.1-2.2 of Durmus et al. [2017b] for a more detailed and rigorous discussion on the connection between the Taylor series expansion of $\log(\alpha_i(x_i, y_i))$ and MCMC scaling results. The following proposition shows that the condition $g(t) = t\, g(1/t)$, when combined with some smoothness assumptions, is sufficient to ensure that the proposals $Q^{(g)}$ lead to $\log(\alpha_i(x_i, x_i + \sigma u_i)) \leq \Theta(\sigma^3)$ as $\sigma \to 0$.

Proposition 4.1. *Let $g : (0, \infty) \to (0, \infty)$ and $g(t) = t\, g(1/t)$ for all t. If g is three times continuously differentiable and $\int_{\mathbb{R}} g^{(j)}(e^{sw})\mu(w)dw < \infty$ for all $s > 0$ and $j \in \{0, 1, 2, 3\}$, where $g^{(j)} : (0, \infty) \to (0, \infty)$ is the j-th derivative of g, then*

$$\log(\alpha_i(x_i, x_i + \sigma u_i)) \leq \Theta(\sigma^3) \qquad \text{as } \sigma \to 0\,, \tag{23}$$

for any x_i and u_i in $\mathbb{R}$.

Proposition 4.1 suggests that Metropolis–Hastings algorithms with proposals $Q^{(g)}$ such that $g(t) = t\,g(1/t)$ have scaling behaviour analogous to MALA, meaning that $\sigma^2 = \Theta(d^{-1/3})$ is sufficient to ensure a non-trivial limit and thus $\Theta(d^{1/3})$ iterations are required for each effective sample. To make these arguments rigorous one should prove weak convergence results for $d \to \infty$, as in Roberts and Rosenthal [1998]. Proving such a result for a general g would require a significant amount of technical work, thus going beyond the scope of this section. In this paper we rather support the conjecture of $\Theta(d^{1/3})$ scaling for $Q^{(g)}$ by means of simulations (see Section 5.2). While Proposition 4.1 only shows $\log(\alpha_i(x_i, x_i + \sigma u_i)) \leq \Theta(\sigma^3)$, it is possible to show that $\log(\alpha_i(x_i, x_i + \sigma u_i)) = \Theta(\sigma^3)$ with some extra assumptions on ϕ to exclude exceptional cases (see the supplement for more detail).

5 Simulations with fixed tuning parameters

Throughout Sections 5 and 6, we choose the symmetric density μ_σ within the random walk and Barker proposals to be $N(0, \sigma^2 \mathbb{I}_d)$ for simplicity. Note, however, that any symmetric density μ_σ could in principle be used. It would be interesting to explore the impact of different choices of μ_σ to the performances of the Barker algorithm, and we leave such a comparison to future work.

5.1 Illustrations of robustness to tuning

We first provide an illustration of the robustness to tuning of the random walk, Langevin and Barker algorithms in three simple one-dimensional settings. In each case we approximate the expected squared jump distance (ESJD) using 10^4 Monte Carlo samples and standard Rao–Blackwellisation techniques, across of range of different proposal step-sizes between 0.01 and 100. As is clearly shown in Figure 2, all algorithms perform similarly when the step-size is smaller than optimal, as suggested in Section 2.4. As the step-size increases beyond this optimum, however, behaviours begin to differ. In particular the ESJD for MALA rapidly decays to zero, whereas in the random walk and Barker cases the reduction is much less pronounced. In fact, the rate of decay is similar for the two schemes, which is to be expected following the results of Sections 4.1 and 2.3. See the supplement for a similar illustration on a 20-dimensional example.

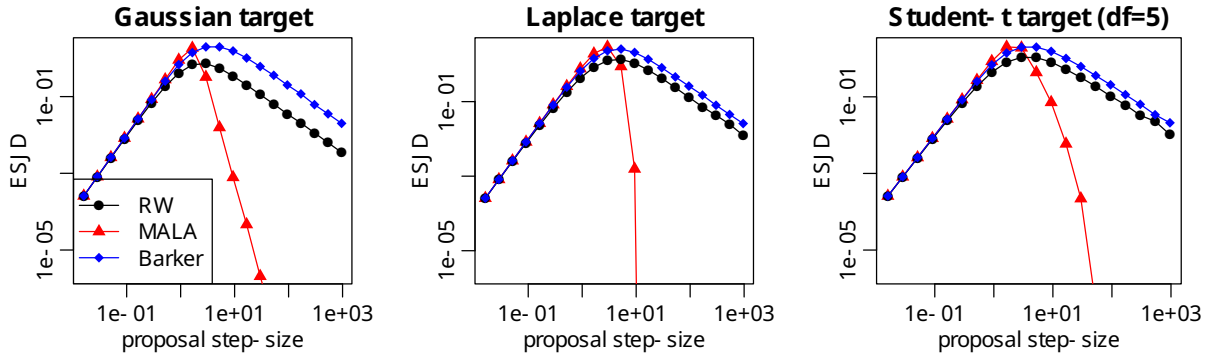

Figure 2: Expected squared jump distance (ESJD) against proposal step-size for RWM, MALA and Barker on different 1-dimensional targets.

5.2 Comparison of efficiency on isotropic targets

Next we compare the expected squared jump distance of the random walk, Langevin and Barker schemes when sampling from isotropic distributions of increasing dimension, with opti-

mised proposal scale (chosen to maximise expected squared jumping distance). This setup is favourable to MALA, which is the least robust scheme among the three, as the target distribution is homogeneous and the proposal step-size optimally-chosen. We consider target distributions with independent and identically distributed (i.i.d.) components, corresponding to the scenario studied theoretically in Section 4.3. We set the distribution of each coordinate to be either a standard normal distribution or a hyperbolic distribution, corresponding to $\log \pi(x) = -\sum_{i=1}^{d} x_i^2/2 + const$ and $\log \pi(x) = -\sum_{i=1}^{d}(0.1 + x_i^2)^{1/2} + const$, respectively. Figure 3 shows how the ESJD per coordinate decays as dimension increases for the three algo-

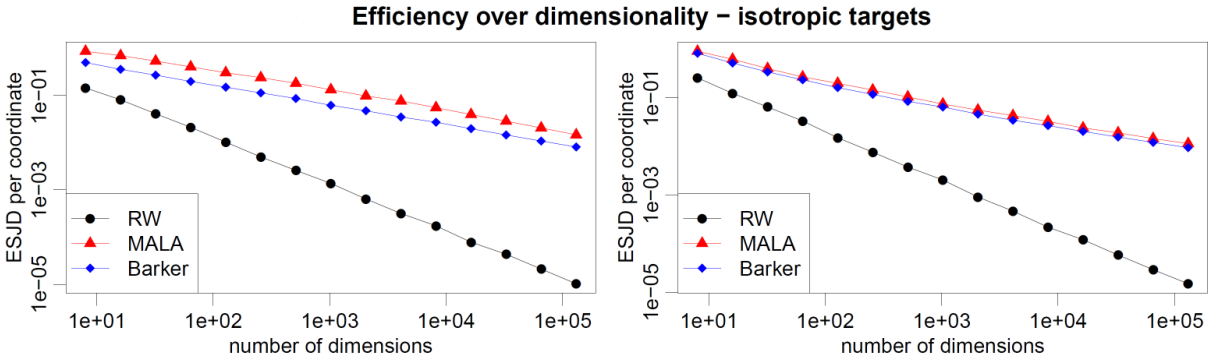

Figure 3: ESJD against dimensionality for RWM, MALA and Barker schemes with optimally-tuned step size. The target distribution has i.i.d. coordinates following either a Gaussian distribution (left plot) or a hyperbolic one (right plot).

rithms. For MALA and Barker the ESJD appears to decrease at the same rate as d increases, which is in accordance with the preliminary results in Section 4.3. In the Gaussian case, MALA outperforms Barker roughly by a factor of 2 regardless of dimension (more precisely, the ESJD ratio lies between 1.7 and 2.5 for all values of d in Figure 3), while in the hyperbolic case the same factor is around 1.2, again independently of dimension (ESJD ratio between 1.1 and 1.25 for all values of d in Figure 3). The rate of decay for the random walk Metropolis is faster, as predicted by the theory.

6 Simulations with Adaptive Markov chain Monte Carlo

In this section we illustrate how robustness to tuning affects the performance of adaptive MCMC methods.

6.1 Adaptation strategy and algorithmic set-up

We use Algorithm 4 in Section 5 of Andrieu and Thoms [2008] to adapt the tuning parameters within each scheme. Specifically, in each case a Markov chain is initialised using a chosen global proposal scale σ_0 and an identity pre-conditioning matrix $\Sigma_0 = \mathbb{I}_d$, and at each iteration the global scale and pre-conditioning matrix are updated using the equations

$$\log(\sigma_t) = \log(\sigma_{t-1}) + \gamma_t \times (\alpha(X^{(t)}, Y^{(t)}) - \bar{\alpha}_*) \tag{24}$$

$$\mu_t = \mu_{t-1} + \gamma_t \times (X^{(t)} - \mu_{t-1}) \tag{25}$$

$$\Sigma_t = \Sigma_{t-1} + \gamma_t \times ((X^{(t)} - \mu_t)(X^{(t)} - \mu_t)^T - \Sigma_{t-1}). \tag{26}$$

Here $X^{(t)}$ denotes the current point in the Markov chain, $Y^{(t)}$ is the proposed move, $\mu_0 = 0$, $\bar{\alpha}_*$ denotes some ideal acceptance rate for the algorithm and the parameter γ_t is known as the learning rate. We set $\bar{\alpha}_*$ to be 0.23 for RWM, 0.57 for MALA and 0.40 for Barker. We tried changing the value of $\bar{\alpha}_*$ for Barker in the range $[0.2, 0.6]$ without observing major differences.

In our simulations we constrain Σ_t to be diagonal (i.e. all off-diagonal terms in (26) are set to 0). This is often done in practice to avoid having to learn a dense pre-conditioning matrix, which has both a high computational cost and would require a large number of MCMC samples. See the supplement for full details on the pre-conditioned Barker schemes obtained with both diagonal and dense matrix Σ_t, including pseudo-code of the resulting algorithms.

We set the learning rate to $\gamma_t := t^{-\kappa}$ with $\kappa \in (0.5, 1)$, as for example suggested in [Shaby and Wells, 2010]. Small values of κ correspond to more aggressive adaptation, and for example Shaby and Wells [2010] suggest using $\kappa = 0.8$. In the simulations of Section 6.2 we use $\kappa = 0.6$ as this turned out to be a good balance between fast adaptation and stability for MALA ($\kappa = 0.8$ resulted in too slow adaptation, while values of κ lower than 0.6 led to instability). The adaptation of RWM and Barker was not very sensitive to the value of κ. Unless specified otherwise, all algorithms are randomly initialized with each coordinate sampled independently from a normal distribution with standard deviation 10. Following the results from the optimal scaling theory [Roberts and Rosenthal, 2001], we set the starting value for the global scale as $\sigma_0^2 = 2.4^2/d$ for RWM and $\sigma_0^2 = 2.4^2/d^{1/3}$ for MALA. For Barker we initialize σ_0 to the same values as MALA.

6.2 Performance on target distributions with heterogeneous scales

In this section we compare the adaptive algorithms described above when sampling from target distributions with significant heterogeneity of scales across their components. We consider 100-dimensional target distributions with different types of heterogeneity, tail behaviour and degree of skewness according to the following four scenarios:

(1) *(One coordinate with small scale; Gaussian target)* In the first scenario, we consider a Gaussian target with zero mean and diagonal covariance matrix. We set the standard deviation of the first coordinate to 0.01 and that of the other coordinates to 1. This scenario mirrors the theoretical framework of Sections 2 and 4.1 in which a single coordinate is the source of heterogeneity.

(2) *(Coordinates with random scales; Gaussian target)* Here we modify scenario 1 by generating the standard deviations of each coordinate randomly, sampling them independently from a log-normal distribution. More precisely, we sample $\log(\eta_i) \sim N(0, 1)$ independently for $i = 1, \ldots, 100$, where η_i is the standard deviation of the i-th component.

(3) *(Coordinates with random scales; Hyperbolic target)* In the third scenario we change the tail behaviour of the target distribution, replacing the Gaussian with a hyperbolic distribution (a smoothed version of the Laplace distribution to ensure $\log \pi \in C^1(\mathbb{R}^d)$). In particular, we set $\log \pi(x) = -\sum_{i=1}^d (\epsilon + (x_i/\eta_i)^2)^{1/2} + c$, with $\epsilon = 0.1$ and c being a normalizing constant. The scale parameters $(\eta_i)_i$ are generated randomly as in scenario 2.

(4) *(Coordinates with random scales; Skew-normal target)* Finally, we consider a non-symmetric target distribution, which represents a more challenging and realistic situation. We assume that the i-th coordinate follows a skew-normal distribution with scale η_i and skewness parameter α, meaning that $\log \pi(x) = -\frac{1}{2}\sum_{i=1}^d (x_i/\eta_i)^2 + \sum_{i=1}^d \log \Phi(\alpha x_i/\eta_i) + c$, with c being a normalizing constant. We set $\alpha = 4$ and generate the η_i's randomly as in scenario 2.

First we provide an illustration of the behaviour of the three algorithms by plotting the trace plots of tuning parameters and MCMC trajectories - see Figure 4 for the results in scenario 1. The adaptation of tuning parameters for the Barker scheme stabilises within a few hundred iterations, after which the algorithm performance appears to be stable and efficient. On the contrary both RWM and MALA struggle to learn the heterogeneous scales and the adaptation process has either just stabilized or not yet stabilized after 10^4 iterations. Looking

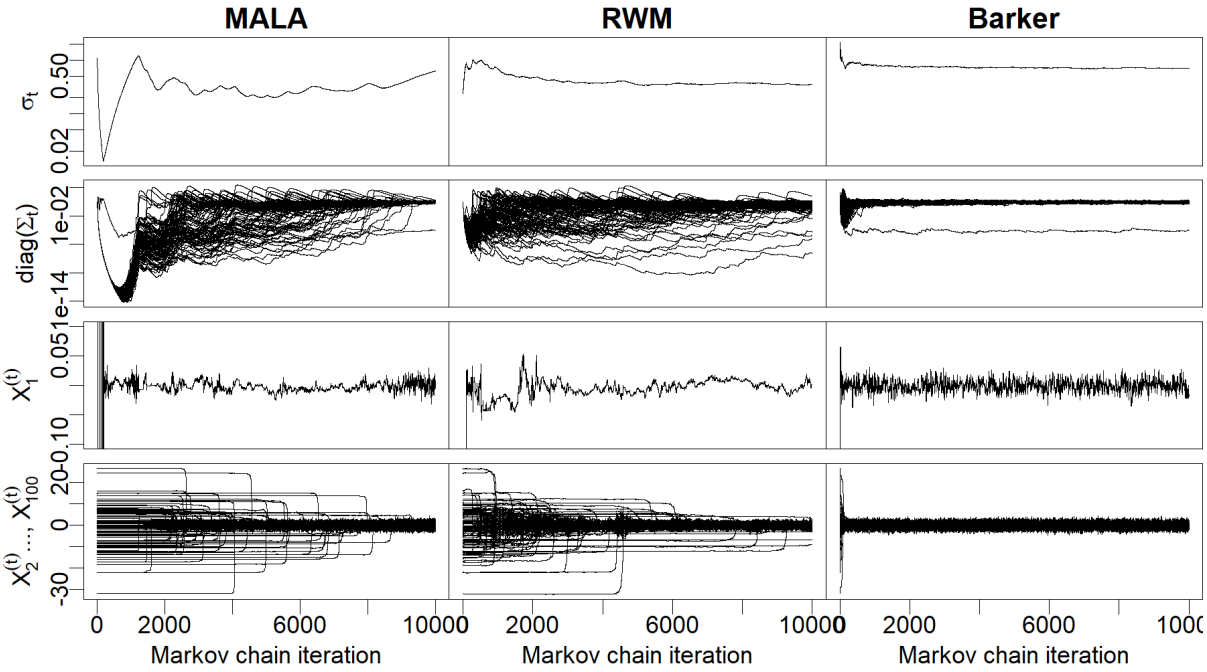

Figure 4: RWM, MALA and Barker schemes with adaptive tuning as in (24)-(26) and learning rate set to $\gamma_t = t^{-\kappa}$ with $\kappa = 0.6$. The target distribution is a 100-dimensional Gaussian in which the first component has standard deviation 0.01 and all others have unit scale. First row: adaptation of the global scale σ_t; second row: adaptation of the local scales $diag(\Sigma_t) = (\Sigma_{t,ii})_{i=1}^{100}$; third row: trace plot of first coordinate; fourth row: trace plots of coordinates from 2 to 100 (superposed).

at the behaviour of MALA in Figure 4 we see that, in order for the algorithm to achieve a non-zero acceptance rate, the global scale parameter σ_t must first be reduced considerably to accommodate the smallest scale of $\pi(\cdot)$. At this point the algorithm can slowly begin to learn the components of the pre-conditioning matrix Σ_t, but this learning occurs very slowly because the comparatively small value for σ_t results in poor mixing across all other dimensions than the first. Analogous plots for Scenarios 2, 3 and 4 are given in the supplement and display comparable behaviour.

We then compare algorithms in a more quantitative way, by looking at the average mean squared error (MSE) of MCMC estimators of the first moment of each coordinate, which is a standard metric in MCMC. For any $h : \mathbb{R}^d \to \mathbb{R}$, define the corresponding MSE as $\mathbb{E}\big[(\hat{h}^{(t)} - \mathbb{E}_\pi[h])^2\big]$ where $\hat{h}^{(t)} = (t - t_{burn})^{-1} \sum_{i=t_{burn}+1}^{t} h(X^{(i)})$ is the MCMC estimator of $\mathbb{E}_\pi[h]$ after t iterations of the algorithm. Here t_{burn} is a burn-in period, which we set to $t_{burn} = \lfloor t/2 \rfloor$, where $\lfloor \cdot \rfloor$ denotes the floor function. Below, we report the average MSE for the collection of test functions given by $h(x) = x_i/\eta_i$ for $i = 1, \ldots, d$ after t MCMC iterations (rescaling by η_i is done to give equal importance to each coordinate).

In addition, we also monitor the rate at which the pre-conditioning matrix Σ_t converges to the covariance of π, denoted as Σ, in order to measure how quickly the adaptation mechanism learns suitable local tuning parameters. We consider the l^2-distance between the diagonal elements of Σ_t and Σ on the log scale. This leads to the following measure of convergence of the tuning parameters after t MCMC iterations:

$$d_t = \mathbb{E}\left[\frac{1}{\sqrt{d}}\left(\sum_{i=1}^{d}(\log(\Sigma_{t,ii}) - \log(\Sigma_{ii}))^2\right)^{1/2}\right], \tag{27}$$

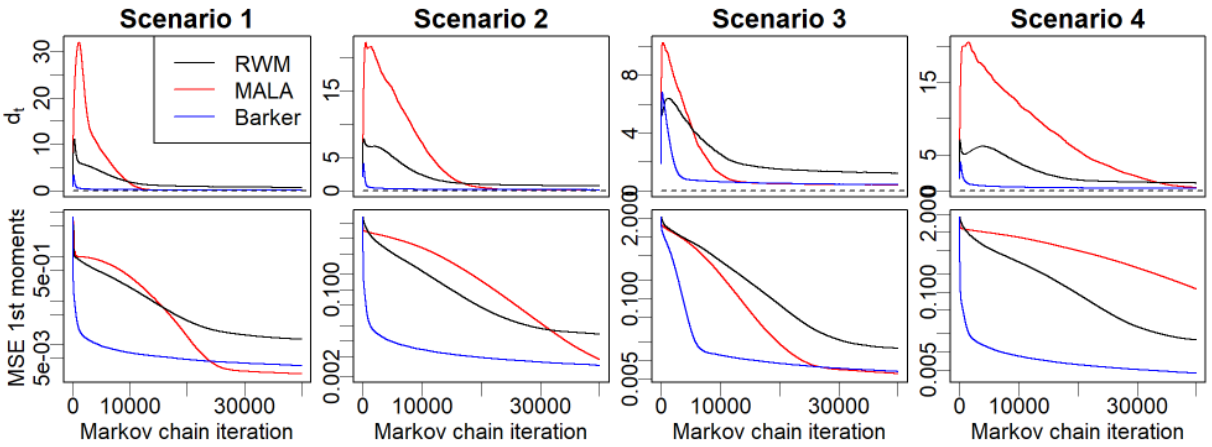

Figure 5: Comparison of RWM, MALA and Barker on the four target distributions (Scenarios 1 to 4) described in Section 6.2, averaging over ten repetitions of each algorithm. First row: convergence of tuning parameters, measured by d_t defined in (27). Second row: Mean Square Error (MSE) of MCMC estimators of first moments averaged over all coordinates.

where the expectation is with respect the Markov chain $(X^{(t)})_{t\geq 1}$. We use the log scale as it is arguably more appropriate than the natural one when comparing step-size parameters, and we focus on diagonal terms as both Σ_t and Σ are diagonal here. Monitoring the convergence of d_t to 0 we can compare the speed at which good tuning parameters are found during the adaptation process for different schemes.

Figure 5 displays the evolution of d_t and the MSE defined above over 4×10^4 iterations of each algorithms, where d_t and the MSE are estimated by averaging over 100 independent runs of each algorithm. The results are in accordance with the illustration in Figure 4, and suggest that the Barker scheme is robust to different types of targets and heterogeneity and results in very fast adaptation, while both MALA and RWM require significantly more iterations to find good tuning parameters. The tuning parameters of MALA appear to exhibit more unstable behaviour than RWM in the first few thousands iterations (larger d_t), while after that they converge more quickly, which again is in accordance with the behaviour observed in Figure 5 and with the theoretical considerations of Sections 2 and 4.1. To further quantify the tuning period, we define the time to reach a stable level of tuning as $\tau_{adapt}(\epsilon) = \inf\{t \geq 1 : d_t \leq \epsilon\}$ for some $\epsilon > 0$. We take $\epsilon = 1$ and report the resulting values in Table 6.2, denoting $\tau_{adapt}(1)$ simply as τ_{adapt}. The results show that in these examples Barker always has the smallest adaptation time, with a speed-up compared to RWM of at least 34x in all four scenarios, and a speed-up compared to MALA ranging between 3x (scenario 3) and 30x (scenario 2). The adaptation times τ_{adapt} tend to increase from scenario 1 to scenario 4, suggesting that the target distribution becomes more challenging as we move from scenario 1 to 4. The hardest case for Barker seems to be the hyperbolic target, although even there the tuning stabilized in roughly 3,000 iterations, while the hardest case for MALA is the skew-normal, in which tuning stabilized in roughly 30,000 iterations.

The differences in the adaptation times have a direct implication on the resulting MSE of MCMC estimators, which is intuitive because the Markov chain will typically start sampling efficiently from π only once good tuning parameters are found. As we see from the second row of Figure 5 and the second part of Table 6.2, the MSE of Barker is already quite low (between 0.007 and 0.012) after 10^4 iterations in all scenarios, while RWM and MALA need significantly more iterations to achieve the same MSE. After finding good tuning parameters and having sampled enough, MALA is slightly more efficient than Barker for the Gaussian target in Scenario 1 and equally efficient in the hyperbolic target of Scenario 3, which is consistent with the simulations

Table 1: Adaptation times (τ_{adapt}) and mean squared errors (MSE) from 10k, 20k and 40k iterations of the RWM, MALA and Barker algorithms under each of the four heterogeneous scenarios described in Section 6.2.

	Method	τ_{adapt}	MSE_{10k}	MSE_{20k}	MSE_{40k}
	RWM	18,757	0.200	0.036	0.013
1	MALA	10,785	0.348	0.016	0.002
	Barker	524	0.007	0.005	0.003
	RWM	19,163	0.228	0.045	0.013
2	MALA	17,298	0.644	0.147	0.004
	Barker	542	0.007	0.005	0.003
	RWM	>40k	0.409	0.080	0.016
3	MALA	10,630	0.248	0.019	0.006
	Barker	3,294	0.012	0.009	0.007
	RWM	>40k	0.315	0.092	0.016
4	MALA	34,340	0.813	0.488	0.112
	Barker	1,427	0.008	0.006	0.004

of Section 5.2 under optimal tuning.

6.3 Comparison on a Poisson random effects model

In this section we consider a Poisson hierarchical model of the form

$$\begin{aligned} y_{ij}|\eta_i &\overset{ind}{\sim} \text{Poisson}(\exp(\eta_i)) && j = 1, \dots, n_i\,, \\ \eta_i|\mu &\overset{ind}{\sim} \text{N}(\mu, \sigma_\eta^2) && i = 1, \dots, I\,, \\ \mu &\sim \text{N}(0, 10^2)\,, \end{aligned} \tag{28}$$

and test the algorithms on the task of sampling from the resulting posterior distribution $p(\mu, \eta_1, \dots, \eta_I|\mathbf{y})$, where $\mathbf{y} = (y_{ij})_{ij}$ denotes the observed data. In our simulations we set $I = 50$ and $n_i = 5$ for all i, leading to 51 unkown parameters and 250 observations.

The model in (28) is an example of a generalized linear model that induces a posterior distribution with light tails and potentially large gradients of $\log \pi$, which creates a challenge for gradient-based algorithms. In particular, the task of sampling from the posterior becomes harder when either the observations $(y_{ij})_{ij}$ contain large values or they are heterogeneous across values of $i \in \{1, \dots, I\}$. The former case results in a more peaked posterior distribution with larger gradients, while the latter induces heterogeneity across the posterior distributions of the parameters η_i.

In our simulations we consider three scenarios, corresponding to increasingly challenging target distributions:

(1) In the first scenario we take $\sigma_\eta = 1$ and generate the data $\mathbf{y}$ from the model in (28) assuming the data-generating value of μ to be $\mu^* = 5$ and sampling the data-generating values of $\eta_1, \dots, \eta_I$ from their prior distribution.

(2) In the second scenario we increase the value of σ_η to 3, which induces more heterogeneity across the parameters $\eta_1, \dots, \eta_I$.

(3) In the third scenario we keep $\sigma_\eta = 3$ and increase the values of μ^* to 10, thus inducing larger gradients.

Similarly to Section 6.2, we first provide an illustration of the behaviour of the tuning parameters and MCMC trace plots for RWM, MALA and Barker in Figure 6. Here all algorithms

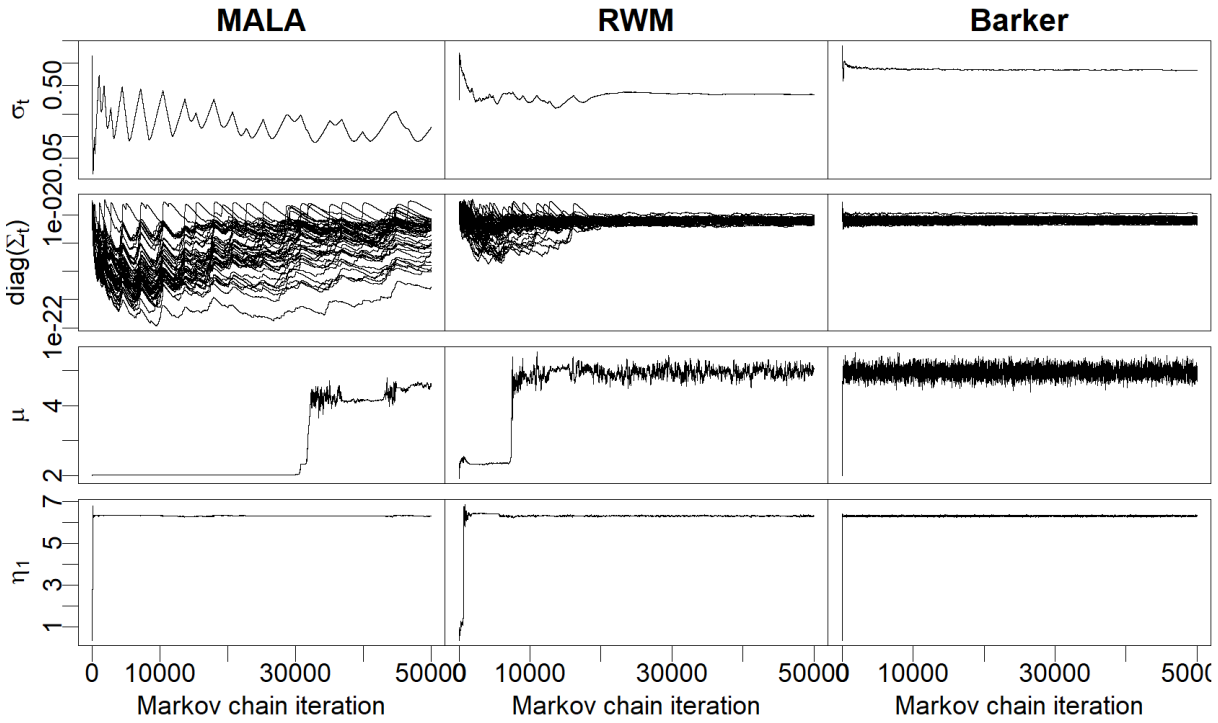

Figure 6: Behavior of RWM, MALA and Barker on the posterior distribution from the Poisson hierarchical model in (28). Data are generated as in the first scenario of Section 6.3. First row: adaptation of the global scale σ_t; second row: adaptation of the local scales $diag(\Sigma_t) = (\Sigma_{t,ii})_{i=1}^{100}$; third row: trace plot of the parameter μ; fourth row: trace plots of the parameter η_1.

are run for 5×10^4 iterations, with the target defined in the first scenario. We use the adaptation strategy of Section 6.2 for tuning, following (24)-(26) with $\kappa = 0.6$ and Σ_t constrained to be diagonal, and initialize the chains from a random configuration sampled from the prior distribution of the model. In this example, the random walk converges to stationarity in roughly 10,000 iterations while the Barker scheme takes a few hundreds. By contrast MALA struggles to converge and exhibits unstable behaviour even after 5×10^4 iterations. Note that the first 3×10^4 iterations of MALA, in which the parameter μ appears to be constant, do not correspond to rejections but rather to moves with very small increments in the μ component.

We then provide a more systematic comparison between the algorithms under consideration in Table 6.3. In addition to RWM, MALA and Barker, we also consider a state-of-the-art implementation of adaptive Hamiltonian Monte Carlo (HMC), namely the Stan [Stan Development Team, 2020] implementation of the No-U-Turn Sampler (NUTS) [Hoffman and Gelman, 2014] as well as of standard HMC (referred to as "static HMC" in the Stan package). The NUTS algorithm is a variant of standard HMC in which the number of leapfrog iterations, i.e. the parameter L in (34), is allowed to depend on the current state (using a "No-U-Turn" criterion). The resulting number of leapfrog steps (and thus log-posterior gradient evaluations) per iteration is not fixed in advance but rather tuned adaptively depending on the hardness of the problem. This is also the case for the static HMC algorithm implementation in Stan, as in that case the total integration time in (34) is fixed and the step-size and mass matrix are adapted. For both algorithms we use the default Stan version that learns a diagonal covariance/mass matrix during the adaptation process. This is analogous to constraining the preconditioning matrix Σ_t for RWM, MALA and Barker to be diagonal, as we are doing here.

Table 6.3 reports the results of the simulations for the five algorithms in each of the three scenarios. For each algorithm, we report the number of log-posterior gradient evaluations and the minimum and median effective sample size (ESS) across the 51 unknown parameters. The

Table 2: Comparison of sampling schemes on the posterior distribution arising from the Poisson hierarchical model in (28). Blocks of rows from 1 to 3 refer to the three data-generating scenarios described in Section 6.3. All numbers are averaged across ten repetitions of each algorithm. For each algorithm we report: number of iterations; number of leapfrog steps per iteration and total number of gradient evaluations (when applicable); estimated ESS (minimum and median across parameters); minimum ESS per hundred gradient evaluations (with standard deviation across the ten repetitions).

	Method	*Iterations* (n)	*Leapfrog steps* /n	*Gradient calls* (g)	*ESS*	*ESS*/g $\times$ 100
	RWM	5×10^4	-	-	(49,66)	-
	MALA	5×10^4	-	5×10^4	(648,727)	1.30 ± 2.73
1	Barker	5×10^4	-	5×10^4	(1445,1587)	2.89 ± 0.07
	HMC	2×10^3	89.5	1.8×10^5	(285,1954)	0.25 ± 0.78
	NUTS	2×10^3	8.5	1.7×10^4	(1175,1822)	6.95 ± 1.68
	RWM	5×10^4	-	-	(0.4,10.6)	-
	MALA	5×10^4	-	5×10^4	(0.0,8.0)	<0.01
2	Barker	5×10^4	-	5×10^4	(1365,1563)	2.73 ± 0.13
	HMC	2×10^3	797	1.6×10^6	(25,1949)	<0.01
	NUTS	2×10^3	57.7	1.2×10^5	(942,1826)	1.19 ± 1.14
	RWM	5×10^4	-	-	(0.0,5.3)	-
	MALA	5×10^4	-	5×10^4	(0.0,0.2)	<0.01
3	Barker	5×10^4	-	5×10^4	(1301,1594)	2.60 ± 0.92
	HMC	2×10^3	8103	1.6×10^7	(3.3,899)	<0.01
	NUTS	2×10^3	179	3.5×10^5	(137,348)	0.012 ± 0.14

ESS values are computed with the `effectiveSize` function from the `coda` R package [Plummer et al., 2006], discarding the first half of the samples as burn-in. The RWM, MALA and Barker schemes are run for 5×10^4 iterations, and the HMC and NUTS schemes for 2×10^3 iterations. The latter is the default value in the Stan package and in this example corresponds to a number of gradient evaluations between 1.7×10^4 and 1.6×10^7. All numbers in Table 6.3 are averaged over ten independent replications of each algorithm. We use the minimum ESS per gradient evaluation as an efficiency metric, of which we report the mean and standard deviation across the ten replicates (multiplied by 100 to facilitate readability).

According to Table 6.3, NUTS is the most efficient scheme in scenario 1, while Barker is the most efficient one in scenarios 2 and 3. This is in accordance with the intuition of Barker being a more robust scheme, as the target distribution becomes more challenging as we move from scenario 1 to 3. MALA struggles to converge to stationarity in scenarios 2 and 3 (with an estimated ESS around zero), while it performs better in scenario 1, although with a high variability across different runs (shown by the large standard deviation in the last column). The RWM displays low ESS values for all three scenarios, although with a less dramatic deterioration going from scenario 1 to 3. Interestingly, the performances of Barker are remarkably stable across scenarios (with an ESS of around 1400), as well as across parameters for which ESS is computed (in all cases the minimum and median ESS are close to each other) and across repetitions (shown by the relatively small standard deviation in the last column). We note that NUTS is also remarkably effective taking into consideration that it is not an algorithm designed with a major emphasis on robustness, but that performance does degrade when moving from scenario 1 to scenario 3. As in the MALA case, static HMC struggles to converge in scenarios 2 and 3 and is not very efficient in scenario 1. Note that NUTS, and in particular HMC, compensate for the increasing difficulty of the target by increasing the number of leapfrog steps per iteration. For example, the drop in efficiency of NUTS between scenarios 1 and 2

is mostly due to the increase in average number of leapfrog iterations from 8.5 to 57.7 rather than in a decrease in ESS. Somewhat surprisingly, in static HMC the number of leapfrog steps per iteration is increased significantly more than NUTS, which could either be due to genuine algorithmic differences or to variations in the details of the adaptation strategy implemented in Stan. Overall, Barker and NUTS are the two most efficient algorithms in these simulation, with a relative efficiency that depends on the scenario under consideration: NUTS being roughly 2.4 times more efficient in scenario 1, Barker 2.3 times more efficient in scenario 2 and Barker 40 times more efficient in scenario 3.

6.4 Additional simulations reported in the supplement

In the supplement we report additional simulations for some of the above experiments. As a sensitivity check, we also performed simulations using the tamed Metropolis-adjusted Langevin algorithm [Brosse et al., 2018] and the truncated Metropolis-adjusted Langevin algorithm [Roberts and Tweedie, 1996, Atchade, 2006], two more robust modifications to MALA in which large gradients are controlled by monitoring the size of $\|\nabla \log \pi(x)\|$. The schemes do offer some added stability compared to MALA in terms of controlling large gradients, but ultimately are still very sensitive to heterogeneity of the target distribution and to the choice of the truncation level, and do not exhibit the same robustness observed in the case of the Barker scheme. See the supplement for implementation details, results and further discussion.

7 Discussion

We have introduced a new gradient-based MCMC method, *the Barker proposal*, and have demonstrated both analytically and numerically that it shares the favourable scaling properties of other gradient-based approaches, along with an increased level of robustness, both in terms of geometric ergodicity and robustness to tuning (as defined in the present paper). The most striking benefit of the method appears to be in the context of adaptive Markov chain Monte Carlo. Evidence suggests that combining the efficiency of a gradient-based proposal mechanism with a method that exhibits robustness to tuning gives a combination of stability and speed that is very desirable in this setting, and can lead to efficient sampling that requires minimal practitioner input.

The theoretical results in this paper could be extended by studying in greater depth the large λ regime (Section 2.4) and the high-dimensional scaling of the Barker proposal (Section 4.3). Of course, there are many other algorithms that could be considered under the robustness to tuning framework, and it is worthwhile future work to explore which features of a scheme result in either robustness to tuning or a lack of it. Extensions to the Barker proposal that incorporate momentum and exhibit the $d^{-1/4}$ decay in efficiency with dimension enjoyed by Hamiltonian Monte Carlo may be possible, as well as the development of other methods within the first-order locally-balanced proposal framework introduced in Section 3, or indeed schemes that are exact at higher orders.

Acknowledgements

The authors thank Marco Frangi for preliminary work in his Master's thesis related to the ergodicity arguments in the paper. SL acknowledges support from the engineering and physical sciences research council through grant number EP/K014463/1 (i-like). GZ acknowledges support from the European research council starting grant 306406 (N-BNP), and by the Italian ministry of education, universities and research PRIN Project 2015SNS29B.

References

Horst Alzer. On some inequalities for the incomplete gamma function. *Mathematics of Computation of the American Mathematical Society*, 66(218):771–778, 1997.

Christophe Andrieu and Johannes Thoms. A tutorial on adaptive mcmc. *Statistics and computing*, 18(4):343–373, 2008.

Yves F Atchade. An adaptive version for the metropolis adjusted langevin algorithm with a truncated drift. *Methodology and Computing in applied Probability*, 8(2):235–254, 2006.

Adelchi Azzalini. A class of distributions which includes the normal ones. *Scandinavian journal of statistics*, pages 171–178, 1985.

Adelchi Azzalini. *The skew-normal and related families*. Institute of Mathematical Statistics Monographs. Cambridge University Press, 2013.

Av A Barker. Monte carlo calculations of the radial distribution functions for a proton-electron plasma. *Australian Journal of Physics*, 18(2):119–134, 1965.

Alexandros Beskos, Natesh Pillai, Gareth Roberts, Jesus-Maria Sanz-Serna, and Andrew Stuart. Optimal tuning of the hybrid monte carlo algorithm. *Bernoulli*, 19(5A):1501–1534, 2013.

Alexandros Beskos, Gareth Roberts, Alexandre Thiery, and Natesh Pillai. Asymptotic analysis of the random walk metropolis algorithm on ridged densities. *The Annals of Applied Probability*, 28(5):2966–3001, 2018.

Steve Brooks, Andrew Gelman, Galin Jones, and Xiao-Li Meng. *Handbook of markov chain monte carlo*. CRC press, 2011.

Nicolas Brosse, Alain Durmus, Éric Moulines, and Sotirios Sabanis. The tamed unadjusted langevin algorithm. *Stochastic Processes and their Applications*, 2018.

Arnak S Dalalyan. Theoretical guarantees for approximate sampling from smooth and log-concave densities. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 79(3):651–676, 2017.

Simon Duane, Anthony D Kennedy, Brian J Pendleton, and Duncan Roweth. Hybrid monte carlo. *Physics letters B*, 195(2):216–222, 1987.

Alain Durmus and Eric Moulines. Nonasymptotic convergence analysis for the unadjusted langevin algorithm. *The Annals of Applied Probability*, 27(3):1551–1587, 2017.

Alain Durmus, Eric Moulines, and Eero Saksman. On the convergence of hamiltonian monte carlo. *arXiv preprint arXiv:1705.00166*, 2017a.

Alain Durmus, Gareth O Roberts, Gilles Vilmart, Konstantinos C Zygalakis, et al. Fast Langevin based algorithm for MCMC in high dimensions. *The Annals of Applied Probability*, 27(4):2195–2237, 2017b.

Paul Fearnhead, Joris Bierkens, Murray Pollock, Gareth O Roberts, et al. Piecewise deterministic markov processes for continuous-time monte carlo. *Statistical Science*, 33(3):386–412, 2018.

Walter Gautschi. Some elementary inequalities relating to the gamma and incomplete gamma function. *Journal of Mathematics and Physics*, 38(1-4):77–81, 1959.

W Keith Hastings. Monte carlo sampling methods using markov chains and their applications. 1970.

Matthew D Hoffman and Andrew Gelman. The no-u-turn sampler: adaptively setting path lengths in hamiltonian monte carlo. *Journal of Machine Learning Research*, 15(1):1593–1623, 2014.

Søren Fiig Jarner and Ernst Hansen. Geometric ergodicity of Metropolis algorithms. *Stochastic processes and their applications*, 85(2):341–361, 2000.

Werner Krauth. *Statistical mechanics: algorithms and computations*, volume 13. OUP Oxford, 2006.

Samuel Livingstone, Michael Betancourt, Simon Byrne, and Mark Girolami. On the geometric ergodicity of hamiltonian monte carlo. *Bernoulli*, page To appear, 2019.

Jonathan C Mattingly, Natesh S Pillai, and Andrew M Stuart. Diffusion limits of the random walk Metropolis algorithm in high dimensions. *The Annals of Applied Probability*, 22(3): 881–930, 2012.

Nicholas Metropolis, Arianna W Rosenbluth, Marshall N Rosenbluth, Augusta H Teller, and Edward Teller. Equation of state calculations by fast computing machines. *The journal of chemical physics*, 21(6):1087–1092, 1953.

Radford M Neal. Slice sampling. *The annals of statistics*, 31(3):705–767, 2003.

Radford M Neal. Mcmc using hamiltonian dynamics. *Handbook of markov chain monte carlo*, 2(11):2, 2011.

Peter H Peskun. Optimum monte-carlo sampling using markov chains. *Biometrika*, 60(3): 607–612, 1973.

Martyn Plummer, Nicky Best, Kate Cowles, and Karen Vines. Coda: Convergence diagnosis and output analysis for mcmc. *R News*, 6(1):7–11, 2006. URL `https://journal.r-project.org/archive/`.

Samuel Power and Jacob Vorstrup Goldman. Accelerated sampling on discrete spaces with non-reversible markov processes. *arXiv preprint arXiv:1912.04681*, 2019.

Gareth O Roberts and Jeffrey S Rosenthal. Optimal scaling of discrete approximations to Langevin diffusions. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 60(1):255–268, 1998.

Gareth O Roberts and Jeffrey S Rosenthal. Optimal scaling for various Metropolis-Hastings algorithms. *Statistical science*, 16(4):351–367, 2001.

Gareth O Roberts and Jeffrey S Rosenthal. General state space markov chains and mcmc algorithms. *Probability surveys*, 1:20–71, 2004.

Gareth O Roberts and Jeffrey S Rosenthal. Complexity bounds for Markov chain Monte Carlo algorithms via diffusion limits. *Journal of Applied Probability*, 53(2):410–420, 2016.

Gareth O Roberts and Richard L Tweedie. Exponential convergence of langevin distributions and their discrete approximations. *Bernoulli*, 2(4):341–363, 1996.

Gareth O Roberts, Andrew Gelman, and Walter R Gilks. Weak convergence and optimal scaling of random walk Metropolis algorithms. *The annals of applied probability*, 7(1):110–120, 1997.

Jeffrey S Rosenthal. Asymptotic Variance and Convergence Rates of Nearly-Periodic Markov Chain Monte Carlo Algorithms. *Journal of the American Statistical Association*, 98(461): 169–177, 2003.

Benjamin Shaby and Martin T Wells. Exploring an adaptive metropolis algorithm. *Technical report*, 2010.

Stan Development Team. RStan: the R interface to Stan, 2020. URL `http://mc-stan.org/`. R package version 2.19.3.

Andrew M Stuart. Inverse problems: a bayesian perspective. *Acta numerica*, 19:451–559, 2010.

Luke Tierney. A note on Metropolis-Hastings kernels for general state spaces. *The Annals of Applied Probability*, 8(1):1–9, 1998.

Dawn Woodard, Scott Schmidler, and Mark Huber. Sufficient conditions for torpid mixing of parallel and simulated tempering. *Electronic Journal of Probability*, 14:780–804, 2009.

Giacomo Zanella. Informed proposals for local mcmc in discrete spaces. *Journal of the American Statistical Association*, pages 1–27, 2019.

Supplement to 'On the robustness of gradient-based MCMC algorithms.'

The ary material contains proofs of the theoretical results and additional figures related to the simulations. It also includes some background on the key techniques used for the proofs of Section 2, a proof of Condition 2.3 for the exponential family class and details related to skew-symmetry and pre-conditioning of the Barker proposal. In this supplement, we number equations, figures and lemmas differently from the main paper, e.g. (1) rather than (1.1), to avoid confusion between the two documents.

A Tools to bound spectral gaps

To establish lower bounds on spectral gaps we use the following Lemma.

Lemma A.1. *Consider two Metropolis–Hastings kernels P_1 and P_2 with associated candidate kernels $Q_1(x, dy) = q_1(x, y)dy$ and $Q_2(x, dy) = q_2(x, y)dy$ and common target distribution π. If there is a $\gamma > 0$ such that $q_1(x, y) \geq \gamma q_2(x, y)$ for all fixed x, y with $x \neq y$, then*

$$\mathrm{Gap}(P_1) \geq \gamma \mathrm{Gap}(P_2). \tag{29}$$

Proof. For any $f \in L^2_{0,1}(\pi)$, it holds that

$$\begin{aligned}
&\int \{f(y) - f(x)\}^2 \pi(dx) P_1(x, dy) \\
&= \int \{f(y) - f(x)\}^2 \min\left\{1, \frac{\pi(y)q_1(y, x)}{\pi(x)q_1(x, y)}\right\} \pi(x)q_1(x, y)dx\, dy \\
&= \int \{f(y) - f(x)\}^2 \min\left\{\pi(x)q_1(x, y), \pi(y)q_1(y, x)\right\} dx\, dy \\
&\geq \gamma \int \{f(y) - f(x)\}^2 \min\left\{\pi(x)q_2(x, y), \pi(y)q_2(y, x)\right\} \\
&= \gamma \int \{f(y) - f(x)\}^2 \min\left\{1, \frac{\pi(y)q_2(y, x)}{\pi(x)q_2(x, y)}\right\} \pi(x)q_2(x, y)dx\, dy \\
&= \gamma \int \{f(y) - f(x)\}^2 \pi(dx) P_2(x, dy).
\end{aligned}$$

The result follows from the Dirichlet forms characterization of spectral gaps in (3). □

To find upper bounds we use the notion of *conductance* for a Markov chain. Define the conductance of a set $K \in \mathcal{B}$ with $0 < \pi(K) \leq 1/2$ for a π-reversible Markov chain with transition kernel P as

$$\Phi(K) := \frac{\int_K \pi(dx)P(x, K^c)}{\pi(K)},$$

which is the conditional probability $\mathbb{P}(X^{(t+1)} \in K^c | X^{(t)} \in K)$ provided $X^{(t)} \sim \pi(\cdot)$. Recall the spectral gap bound for P that for any such K

$$\mathrm{Gap}(P) \leq 2\Phi(K). \tag{30}$$

This can be seen directly by setting $g(x) = \pi(K^C)\mathbb{I}(x \in K) - \pi(K)\mathbb{I}(x \in K^c)$, letting $f(x) := g(x)/\int g(x)^2\pi(dx)$ and computing the Dirichlet form of f using (3). Here $\mathbb{I}(\cdot)$ denotes the indicator function.

B Change of variables and isomorphic Markov chains

In this section we provide two lemmas showing that bijective mappings do not change the spectral gaps of Markov chains, nor the Metropolis-Hastings dynamics. These lemmas will allow us to prove the results of Section 2 working with the equivalent formulation where the target is fixed and the proposal distribution is changing, rather than having a target that changes with λ. This will in turn allow us to exploit results such as Lemma A.1, thus significantly simplifying the proofs.

We follow the terminology of Johnson and Geyer (2013), and introduce the notion of *isomorphic* Markov chains. Intuitively, two Markov chains $(X^{(t)})_{t\geq 1}$ and $(Y^{(t)})_{t\geq 1}$ are isomorphic if $(\phi(X^{(t)}))_{t\geq 1}$ is equal in distribution to $(Y^{(t)})_{t\geq 1}$ for some bijective map ϕ. More formally, let $(X^{(t)})_{t\geq 1}$ and $(Y^{(t)})_{t\geq 1}$ be Markov chains with transition kernels P and K and state spaces $(S, \mathcal{A})$ and $(T, \mathcal{B})$, respectively. We say that $(X^{(t)})_{t\geq 1}$ and $(Y^{(t)})_{t\geq 1}$ are isomorphic if there exists a bijective function ϕ from S to T such that

$$P(x, A) = K(\phi(x), \phi(A)) \qquad x \in S,\ A \in \mathcal{A}. \tag{31}$$

Equation (31) means that $K(\phi(x), \cdot)$ is the push-forward of $P(x, \cdot)$ under ϕ for every $x \in \mathbb{R}^d$, which we write as $K = \phi \circ P$. We will use $\circ$ to denote the push-forward operator for both probability distributions and transition kernels, so that $(\phi \circ \pi)(B) = \pi(\phi^{-1}(B))$ and $(\phi \circ P)(y, B) = P(\phi^{-1}(y), \phi^{-1}(B))$.

Isomorphic Markov chains share the same convergence behaviour and, in particular, they have the same L^2-spectral gap, as stated in the following lemma (see Lemma 1 of Papaspiliopoulus et al. (2019) for a proof of analogous results).

Lemma B.1. *Isomorphic Markov chains have the same L^2-spectral gap.*

In the following we will exploit the fact that the Metropolis-Hastings (MH) algorithm preserves isomorphisms of Markov chains under transformations of both the target and candidate distributions, as shown by the following lemma.

Lemma B.2. *Let $\phi : S \to T$ be a bijective function and $(X^{(t)})_{t\geq 1}$ and $(Y^{(t)})_{t\geq 1}$ be Metropolis-Hastings Markov chains defined on $(S, \mathcal{A})$ and $(T, \mathcal{B})$ with target distributions π and $\phi \circ \pi$, respectively, and proposal kernels Q and $\phi \circ Q$, respectively. Then $(X^{(t)})_{t\geq 1}$ and $(Y^{(t)})_{t\geq 1}$ are isomorphic Markov chains.*

Proof. Let $\mu^{\phi}(dy, dy') := (\phi \circ \pi)(dy')(\phi \circ Q)(y', dy)$ and $\mu^{\phi}_T(dy, dy') := \mu^{\phi}(dy', dy)$. Then using Proposition 1 of Tierney [1998] there exists a set $R^{\phi} \in \mathcal{B} \times \mathcal{B}$ such that μ^{ϕ} and μ^{ϕ}_T are mutually absolutely continuous on R^{ϕ} and mutually singular on its complement. The Radon-Nikodym derivative $d\mu^{\phi}/d\mu^{\phi}_T(y, y')^T$ is therefore finite and positive when restricted to R^{ϕ}. Let $r^{\phi}(y, y') := d\mu^{\phi}/d\mu^{\phi}_T(y, y')$ if $(y, y') \in R^{\phi}$ and $r^{\phi}(y, y') := 0$ otherwise. Then the Metropolis–Hastings acceptance probability for the chain $(Y_t)_{t\geq 1}$ can be written $\alpha^{\phi}(y, y') := \min(1, r^{\phi}(y, y'))$. Similarly, letting $\mu(dx, dx') := \pi(dx')Q(x', dx)$ and $\mu_T(dx, dx') := \mu(dx', dx)$ the acceptance probability for $(X_t)_{t\geq 1}$ can be written $\alpha(x, x') := \min(1, r(x, x'))$ where $r(x, x') := d\mu/d\mu_T(x, x')$ when $(x, x') \in R \in S \times S$ and 0 otherwise, with R defined analogously to R^{ϕ} for the measures μ and μ_T.

Note first that from the definitions of push-forward measure and transition kernel given above that $\mu^{\phi}(A, B) = \mu(\phi^{-1}(A), \phi^{-1}(B))$ and $\mu^{\phi}_T(A, B) = \mu_T(\phi^{-1}(A), \phi^{-1}(B))$ for any $(A, B) \in \mathcal{B} \times \mathcal{B}$. From this it follows that $R \in \mathcal{A} \times \mathcal{A}$ is the pre-image under ϕ of $R^{\phi} \in \mathcal{B} \times \mathcal{B}$, and further that

$$\alpha^{\phi}(y, y') = \min(1, r^{\phi}(y, y')) = \min(1, r(\phi^{-1}(y), \phi^{-1}(y')) = \alpha(\phi^{-1}(y), \phi^{-1}(y')).$$

Denoting the transition kernels of $(X^{(t)})_{t\geq 1}$ and $(Y^{(t)})_{t\geq 1}$ as P and K respectively, it therefore holds that

$$\begin{aligned}K(y,B) =&\delta_B(y)\int_T(1-\alpha^\phi(y,y'))\phi\circ Q(y,dy')+\int_B\alpha^\phi(y,y')\phi\circ Q(y,dy')\\ =&\delta_{\phi^{-1}(B)}(\phi^{-1}(y))\int_S(1-\alpha(\phi^{-1}(y),x'))Q(\phi^{-1}(y),dx')\\ &+\int_{\phi^{-1}(B)}\alpha^\phi(\phi^{-1}(y),x')Q(\phi^{-1}(y),dx')\\ =&P(\phi^{-1}(y),\phi^{-1}(B))\end{aligned}$$

meaning that $(X^{(t)})_{t\geq 1}$ and $(Y^{(t)})_{t\geq 1}$ are isomorphic. □

C Proofs

Throughout the proofs we often use $\|\cdot\|$ to denote the standard euclidean norm $\|\cdot\|_2$.

C.1 Proofs for Section 2

Proof of Proposition 2.1. We first establish that $\mu(\delta_\lambda)\geq\mu(\delta)$ whenever $\lambda\leq\lambda_0$ for some $\lambda_0>0$. In cases (i) and (iii) $\mu(z)$ is monotonically decreasing in $\|z\|_2^2$. So when $\lambda<1$ it holds that

$$\|\delta_\lambda\|_2^2=\sum_{i=1}^d(y_i-x_i)^2+\lambda^2(y_1-x_1)^2\leq\|\delta\|_2^2,$$

which proves the condition for $\lambda_0=1$. In case (ii) $\mu(z)$ is monotonically decreasing in $\|z\|_1$, so again $\|\delta_\lambda\|_1=\lambda|y_1-x_1|+\sum_{i=2}^d|y_i-x_i|\leq\|\delta\|$ when $\lambda<1$. The statement that $\sup_{z_1\in\mathbb{R}}\mu_1(z_1)<\infty$ follows by noting that in all three cases the marginal μ_1 is known in closed form and is, respectively, a Gaussian, Laplace and Student's t distribution, all of which have bounded density. □

Proof of Theorem 2.1. Instead of studying directly P_λ^R, we will study the transition kernel $\tilde{P}_\lambda^R$ corresponding to a Metropolis-Hastings (MH) algorithm with proposal $\phi\circ Q^R$ and target $\phi\circ\pi^{(\lambda)}$, for some bijective $\phi:\mathbb{R}^d\to\mathbb{R}^d$. By Lemma B.2, $\tilde{P}_\lambda^R$ and P_λ^R induce isomorphic Markov chains and thus by Lemma B.1 we have $\mathrm{Gap}(\tilde{P}_\lambda^R)=\mathrm{Gap}(P_\lambda^R)$. We consider ϕ given by $\phi(x_1,\ldots,x_d)=(\lambda^{-1}x_1,x_2,\ldots,x_d)$. It follows that $\phi\circ\pi^{(\lambda)}=\pi$ and that $\tilde{Q}_\lambda^R=\phi\circ Q$ satisfies $\tilde{Q}_\lambda^R(x,dy)=\tilde{q}_\lambda^R(x,y)dy$ with

$$\tilde{q}_\lambda^R(x,y):=\frac{\lambda}{\sigma^d}\mu\left(\frac{\delta_\lambda}{\sigma}\right),\tag{32}$$

and δ_λ defined as in equation (5).

First we show that for all $\lambda\leq\lambda_0$ and all $x,y\in\mathbb{R}^d$ it holds that $\tilde{q}_\lambda^R(x,y)\geq\lambda\tilde{q}_1^R(x,y)$, where $\lambda_0>0$ is the value defined in Condition 2.1. From (32), we have

$$\frac{\tilde{q}_\lambda^R(x,y)}{\tilde{q}_1^R(x,y)}=\lambda\frac{\mu(\delta_\lambda/\sigma)}{\mu(\delta/\sigma)}.\tag{33}$$

Condition 2.1 guarantees $\mu(\delta_\lambda/\sigma)\geq\mu(\delta/\sigma)$ for all $\lambda\leq\lambda_0$, which together with (33) gives $\tilde{q}_\lambda^R(x,y)\geq\lambda\tilde{q}_1^R(x,y)$. Combining the latter inequality with Lemma A.1 gives

$$\mathrm{Gap}(\tilde{P}_\lambda^R)\geq\lambda\mathrm{Gap}(\tilde{P}_1^R)=\Theta(\lambda)\qquad\text{as }\lambda\downarrow 0\,.$$

To show that $\mathrm{Gap}(\tilde{P}_\lambda^R)\leq\Theta(\lambda)$, take $X^{(t)}\sim\pi(\cdot)$ and $X^{(t+1)}|X^{(t)}\sim\tilde{P}_\lambda^R(X^{(t)},\cdot)$. We consider the set $K:=\{y\in\mathbb{R}^d:|y_1|>k\}$, with k chosen such that $0<\pi(K)<1/2$ (since $\pi(\cdot)$

is defined on a Polish space, it is tight, meaning this is always possible). Recall from (30) that $\mathrm{Gap}(\tilde{P}_\lambda^R) \le 2\mathbb{P}(X^{(t+1)} \in K^c \,|\, X^{(t)} \in K)$. We have

$$\begin{aligned}\mathbb{P}(X^{(t+1)} \in K^c \,\big|\, X^{(t)} \in K) &\le \mathbb{P}(|X_1^{(t)} + \sigma\lambda^{-1}\xi_1| \le k \,\big|\, X^{(t)} \in K) \\ &= \mathbb{P}(-X_1^{(t)} - k \le \sigma\lambda^{-1}\xi_1 \le -X_1^{(t)} + k \,\big|\, X^{(t)} \in K) \\ &\le \sigma^{-1}\lambda 2k \sup_{z_1 \in \mathbb{R}} \mu_1(z_1)\,,\end{aligned}$$

where ξ_1 is the first component of $\xi \sim \mu$. Conditon 2.1 implies $\sup_{z_1\in\mathbb{R}} \mu_1(z_1) < \infty$, giving that $\mathrm{Gap}(\tilde{P}_\lambda^R) \le \Theta(\lambda)$ for $\lambda \downarrow 0$, as desired. □

Proof of Theorem 2.2. This follows directly from the proof of Theorem 2.4 below, by noting that setting $L = 1$ in Hamiltonian Monte Carlo gives the Langevin algorithm. □

Proof of Theorem 2.3. Similarly to the proof of Theorem 2.1, instead of studying directly P_λ^M, we will study the MH transition kernel $\tilde{P}_\lambda^M$ with proposal $\phi \circ Q_\lambda^M$ and target $\phi \circ \pi^{(\lambda)}$. Lemma B.2 and Lemma B.1 imply $\mathrm{Gap}(\tilde{P}_\lambda^M) = \mathrm{Gap}(P_\lambda^M)$. We consider the same ϕ as in the proof of Theorem 2.1, which we write as $\phi(x) = \Sigma_\lambda^{1/2} x$ with

$$\Sigma_\lambda = \begin{pmatrix} \lambda^{-2} & (0,\dots,0) \\ (0,\dots,0)^T & \mathbb{I}_{d-1} \end{pmatrix}.$$

We have $\pi = \phi \circ \pi^{(\lambda)}$ as stated above. Also, $\tilde{Q}_\lambda^M = \phi \circ Q_\lambda^M$ satisfies $\tilde{Q}_\lambda^M(y,\cdot) = N(y + \frac{\sigma^2}{2}\Sigma_\lambda \nabla \log\pi(y), \sigma^2\Sigma_\lambda)$ where Σ_λ is as above. This is a fairly standard calculation, analogous to the derivation of the preconditioned MALA algorithm with preconditioning matrix $\Sigma_\lambda^{1/2}$, which we report here for completeness. By definition of ϕ and $Q_\lambda^M(x,\cdot) = N(x + \frac{\sigma^2}{2}\nabla \log\pi^{(\lambda)}(x), \sigma^2\mathbb{I}_d)$, we have $\phi \circ Q_\lambda^M(x,\cdot) = N\left(\phi\big(x + \frac{\sigma^2}{2}\nabla\log\pi^{(\lambda)}(x)\big), \sigma^2\Sigma_\lambda\right)$ for each $x \in \mathbb{R}^d$. Also, since $\log\pi^{(\lambda)}(x) = \log\pi(\phi(x)) + const$ and $\phi(x) = \Sigma_\lambda^{1/2}x$, we have $\nabla\log\pi^{(\lambda)}(x) = \Sigma_\lambda^{1/2}\nabla\log\pi(\phi(x))$. Therefore

$$\phi\left(x + \frac{\sigma^2}{2}\nabla\log\pi^{(\lambda)}(x)\right) = \Sigma_\lambda^{1/2}\left(x + \frac{\sigma^2}{2}\Sigma_\lambda^{1/2}\nabla\log\pi(\phi(x))\right) = \phi(x) + \frac{\sigma^2}{2}\Sigma_\lambda\nabla\log\pi(\phi(x))$$

meaning that $\tilde{Q}_\lambda^M(\phi(x),\cdot)$ is the push-forward of $Q_\lambda^M(x,\cdot)$ under ϕ for every $x \in \mathbb{R}^d$, as desired.

We now prove $\mathrm{Gap}(\tilde{P}_\lambda^M) \le \Theta(e^{-\lambda^{-\alpha}})$ as $\lambda \downarrow 0$ for some $\alpha > 0$. We take $\sigma = 1$ for simplicity of notation (or otherwise replace λ by $\sigma^{-1}\lambda$) and we assume $\lambda < 1$ without loss of generality (we are studying a limit $\lambda \downarrow 0$). Let $(X^{(t)})_{t=1}^\infty$ be a Markov chain with transition kernel $\tilde{P}_\lambda^M$ started in stationarity. We consider the sets $A_\lambda := \{y \in \mathbb{R}^d : |y_1| \le \lambda^{-1/(2\tilde{\gamma})}\}$, where $\tilde{\gamma} = \max\{1,\gamma\}$, and $K := \{y \in \mathbb{R}^d : |y_1| > k\}$, with k chosen such that $0 < \pi(K) < 1/2$. Given

$$\epsilon \in \left(0, \liminf_{|x_1|\to\infty}\left(\inf_{(x_2,\dots,x_d)\in\mathbb{R}^{d-1}}\left|\frac{\partial\log\pi(x)}{\partial x_1}\right|\|x\|^\gamma\right)\right),$$

Condition 2.3(i) implies that we can choose k large enough such that

$$\left|\frac{\partial\log\pi(x)}{\partial x_1}\right|\|x\|^\gamma \ge \epsilon \qquad \text{for all } x \in K\,.$$

We will now show $\Pr(X^{(t+1)} \in K^c \,|\, X^{(t)} \in K) \le \Theta(e^{-\lambda^{-\alpha}})$ for some $\alpha > 0$ as $\lambda \downarrow 0$. Note that

$$\begin{aligned}\pi(K)\mathbb{P}(X^{(t+1)} \in K^c | X^{(t)} \in K) = \mathbb{P}(X^{(t+1)} \in K^c | X^{(t)} \in K \cap A_\lambda)\mathbb{P}(X^{(t)} \in K \cap A_\lambda) \\ + \mathbb{P}(X^{(t+1)} \in K^c | X^{(t)} \in K \cap A_\lambda^c)\mathbb{P}(X^{(t)} \in K \cap A_\lambda^c),\end{aligned}$$

meaning that

$$\pi(K)\mathbb{P}(X^{(t+1)} \in K^c | X^{(t)} \in K) \leq \mathbb{P}(X^{(t+1)} \in K^c | X^{(t)} \in K \cap A_\lambda) + \mathbb{P}(X^{(t)} \in K \cap A_\lambda^c).$$

Condition 2.3(ii) implies $\mathbb{P}(X^{(t)} \in K \cap A_\lambda^c) \leq \mathbb{P}(X^{(t)} \in A_\lambda^c) \leq \Theta(e^{-\lambda^{-\beta/(2\tilde{\gamma})}})$.

Also, given $Y = (Y_1, \ldots, Y_d)$ with $Y | X^{(t)} \sim \tilde{Q}_h^M(X^{(t)}, \cdot)$, we have

$$\Pr(X^{(t+1)} \in K^c \,|\, X^{(t)} \in K \cap A_\lambda) \leq \Pr(|Y_1| \leq k \,|\, X^{(t)} \in K \cap A_\lambda)\,.$$

Denote $\frac{\partial}{\partial x_1} \log \pi(x)$ by $\partial_1(x)$ for brevity. If $X^{(t)} \in K \cap A_\lambda$ we have $|X_1^{(t)}| \leq \lambda^{-1/(2\tilde{\gamma})} \leq \lambda^{-1/2}$ and

$$|\partial_1(X^{(t)})| \geq \epsilon \|X^{(t)}\|^{-\gamma} \geq \epsilon \lambda^{\gamma/(2\tilde{\gamma})} \geq \epsilon \lambda^{1/2}\,,$$

which imply

$$\begin{aligned} |Y_1| &= |X_1^{(t)} + \lambda^{-2}\partial_1(X^{(t)}) + \lambda^{-1}\xi_1| \\ &\geq \lambda^{-2}|\partial_1(X^{(t)})| - \lambda^{-1}|\xi_1| - |X_1^{(t)}| \\ &\geq \lambda^{-3/2}\epsilon - \lambda^{-1}|\xi_1| - \lambda^{-1/2}\,, \end{aligned}$$

where $\xi_1 \sim N(0,1)$. It follows that

$$\begin{aligned} \Pr(|Y_1| \leq k \,|\, X^{(t)} \in K \cap A_\lambda) &\leq \Pr(\lambda^{-3/2}\epsilon - \lambda^{-1}|\xi_1| - \lambda^{-1/2} \leq k \,|\, X^{(t)} \in K \cap A_\lambda) \\ &= \Pr(|\xi_1| \geq \epsilon\lambda^{-1/2} - \lambda^{1/2} - k\lambda)\,. \end{aligned}$$

Since $\Pr(|\xi_1| \geq t) \leq \exp(-t^2/2)$ for every $t > 0$ (which follows from standard bounds on Gaussian tails and $\xi_1 \sim N(0,1)$) and $\epsilon\lambda^{-1/2} - \lambda^{1/2} - k\lambda \geq 2\lambda^{-1/3}$ eventually as $\lambda \downarrow 0$, it follows

$$\Pr(|Y_1| \leq k \,|\, X^{(t)} \in K \cap A_\lambda) \leq \Theta(e^{-\lambda^{-2/3}}) \qquad \text{as } \lambda \downarrow 0\,.$$

Combining the inequalities above and noting that $\pi(K)$ does not depend on λ, it follows that $\mathbb{P}(X^{(t+1)} \in K^c | X^{(t)} \in K) \leq \Theta(e^{-\lambda^{-\alpha}})$ for $\alpha = \min\{\beta/(2\tilde{\gamma}), 2/3\} > 0$ as $\lambda \downarrow 0$. Finally, the conductance bound in (30) imply $\mathrm{Gap}(\tilde{P}_\lambda^M) \leq \Theta(e^{-\lambda^{-\alpha}})$ as $\lambda \downarrow 0$. □

Proof of Theorem 2.4. Similarly to the case of the random walk and Langevin schemes, we will study the MH transition kernel $\tilde{P}_\lambda^H$ with proposal $\phi \circ Q_\lambda^H$ and target $\phi \circ \pi^{(\lambda)}$, and exploit the fact that $\mathrm{Gap}(\tilde{P}_\lambda^H) = \mathrm{Gap}(P_\lambda^H)$ by Lemma B.2 and Lemma B.1. Considering $\phi(x) = \Sigma_\lambda^{1/2} x$ as above we have $\pi = \phi \circ \pi^{(\lambda)}$ and $\tilde{Q}_\lambda^H = \phi \circ Q_\lambda^H$ evolving according to a preconditioned HMC algorithm as follows. Writing the current point $x \in \mathbb{R}^d$ as $x(0)$, as in Section 2.3.3 of the the paper, the proposal $y := x(L) \sim \tilde{Q}_\lambda^H(x, \cdot)$ is obtained using the update

$$x(L) = x(0) + \sigma^2 \left(\frac{L}{2}\Sigma_\lambda \nabla \log \pi(x(0)) + \sum_{j=1}^{L-1} (L-j)\Sigma_\lambda \nabla \log \pi\left(x(j)\right) \right) + L\sigma\Sigma_\lambda^{1/2}\xi(0), \qquad (34)$$

where each $x(j)$ is defined recursively in the same manner, and $\xi(0) \sim N(0, \mathbb{I}_d)$. It is easy to check that $\tilde{Q}_\lambda^H = \phi \circ Q_\lambda^H$ using the same calculations as in the proof of Theorem 2.3.

We now prove $\mathrm{Gap}(\tilde{P}_\lambda^H) \leq \Theta(e^{-\lambda^{-\alpha}})$ as $\lambda \downarrow 0$ for some $\alpha > 0$. To simplify the notation in the following we prove the equivalent statement that $\mathrm{Gap}(\tilde{P}_{h^{-1}}^H) \leq \Theta(e^{-h^\alpha})$ as $h \to \infty$. Fix $\delta \in (0, (1-q)/2)$ with q defined in Condition 2.2 and consider the sets $A_h := \{y \in \mathbb{R}^d : |y_1| < k + h\}$ and $K := \{y \in \mathbb{R}^d : |y_1| > k\}$. Here k is chosen large enough that Lemma C.1 below is satisfied and that $0 < \pi(K) < 1/2$, which can always be done thanks to the tightness and positiveness of π (see above). Lemma C.1 implies that if $X^{(t)} \in K \cap A_h$, $|\xi_1| \leq h^{1-\delta}$ and $h \geq h_0$,

where $h_0 = \lambda_0^{-1}$ with λ_0 defined as in Lemma C.1, then $X^{(t+1)} \in K$. We now upper bound the probability $\mathbb{P}(X^{(t+1)} \in K^c | X^{(t)} \in K)$. First note that

$$\begin{aligned}\mathbb{P}(X^{(t+1)} \in K^c, X^{(t)} \in K) = \mathbb{P}(X^{(t+1)} \in K^c | X^{(t)} \in K \cap A_h)\mathbb{P}(X^{(t)} \in K \cap A_h) \\ + \mathbb{P}(X^{(t+1)} \in K^c | X^{(t)} \in K \cap A_h^c)\mathbb{P}(X^{(t)} \in K \cap A_h^c),\end{aligned}$$

which implies

$$\mathbb{P}(X^{(t+1)} \in K^c, X^{(t)} \in K) \leq \mathbb{P}(X^{(t+1)} \in K^c | X^{(t)} \in K \cap A_h) + \mathbb{P}(X^{(t)} \in K \cap A_h^c).$$

Breaking out the first term on the right-hand side gives

$$\begin{aligned}&\mathbb{P}(X^{(t+1)} \in K^c | X^{(t)} \in K \cap A_h) \leq \\ &\mathbb{P}(X^{(t+1)} \in K^c | X^{(t)} \in K \cap A_h, |\xi_1| \leq h^{1-\delta})\mathbb{P}(|\xi_1| \leq h^{1-\delta}) + \mathbb{P}(|\xi_1| > h^{1-\delta}),\end{aligned}$$

which, using the result of Lemma C.1, reduces to

$$\mathbb{P}(X^{(t+1)} \in K^c | X^{(t)} \in K \cap A_h) \leq \mathbb{P}(|\xi_1| > h^{1-\delta}) .$$

Hence we obtain the overall bound

$$\mathbb{P}(X^{(t+1)} \in K^c, X^{(t)} \in K) \leq \mathbb{P}(|\xi_1| > h^{1-\delta}) + \mathbb{P}(X^{(t)} \in K \cap A_h^c).$$

Using standard bounds on Gaussian tails and $\xi_1 \sim N(0,1)$, we have $\mathbb{P}(|\xi_1| > h^{1-\delta}) \leq \exp(-h^{2(1-\delta)}/2)$. Also, from Lemma C.3, we have $\mathbb{P}(X^{(t)} \in K \cap A_h^c) \leq \Theta(e^{-\gamma h^{1+q} - q\log(h)})$ as $h \to \infty$ for some $\gamma \in (0, \infty)$. Hence, since $\delta < (1-q)/2$ and $\mathbb{P}(X^{(t)} \in K) = \pi(K)$ is constant with respect to h, we obtain

$$\mathbb{P}(X^{(t+1)} \in K^c | X^{(t)} \in K) \leq \Theta\left(e^{-\gamma h^{1+q} - q\log(h)}\right) \qquad \text{as } h \to \infty .$$

Finally, the conductance bound in (30) gives

$$\mathrm{Gap}(\tilde{P}_{h^{-1}}^H) \leq \Theta\left(e^{-\gamma h^{1+q} - q\log(h)}\right) \qquad \text{as } h \to \infty .$$

□

Proof of Proposition 2.2. For the Langevin case, standard results on the total variation distance between two Gaussian measures with differing means reveals that

$$\|Q_\lambda^M(x,\cdot) - Q^R(x,\cdot)\|_{TV} = 1 + \frac{1}{\sqrt{2\pi}} \int_0^t e^{-u^2/2} du.$$

where $t = \sigma|\nabla \log \pi(x/\lambda)|/(4\lambda)$. Because $\nabla \log \pi$ is bounded in a neighbourhood of zero, for large enough λ we can write $t \leq C/\lambda$ for some $C < \infty$. Then note that as $\lambda \uparrow \infty$

$$\int_0^{C/\lambda} e^{-u^2/2} du \leq \frac{C}{\lambda}.$$

For the Barker case, note that the total variation distance here can be written

$$\frac{1}{2} \int \mu_\sigma(z) |2g(e^{\nabla \log \pi(x/\lambda) z/\lambda}) - 1| dz,$$

where $g(t) = 1/(1+t^{-1})$. Setting $u := \nabla \log \pi(x) z$, a Taylor series expansion about $u = 0$ of $2g(u)$ is

$$2g(u) = 1 + \frac{u}{2} + g''(\xi) u^2,$$

for some ξ satisfying $|\xi| \leq |\nabla \log \pi(x) z|$, using the Lagrange form of the remainder. Substituting this into the integral and simplifying gives

$$\frac{1}{2}\int \mu_\sigma(z)\left|\frac{u}{2} + g''(\xi)u^2\right| dz \leq \frac{1}{4\lambda}|\nabla \log \pi(x/\lambda)| \int |z|\mu_\sigma(z)dz + \frac{1}{2\lambda^2}\nabla \log \pi(x/\lambda)^2 \int |g''(\xi)|z^2\mu_\sigma(z)dz.$$

For large enough λ the boundedness assumption allows us to write $|\nabla \log \pi(x/\lambda)| \leq c$ for some $c < \infty$. In addition note that $g''(\xi) = 2e^{-2\xi}/(1+e^{-\xi})^3 - e^{-\xi}/(1+e^{-\xi})^2$, and so $\sup_{\xi\in\mathbb{R}} |g''(\xi)| = (6\sqrt{3})^{-1}$. Substituting into the bound and evaluating the two integrals gives an upper bound to the total variation distance of

$$\left(\frac{c}{4}\sqrt{\frac{2\sigma^2}{\pi}}\right)\lambda^{-1} + \left(\frac{c^2\sigma^2}{12\sqrt{3}}\right)\lambda^{-2},$$

which is $\Theta(1/\lambda)$ as $\lambda \uparrow \infty$, as desired.

□

C.1.1 Lemmas used to prove Theorem 2.4

Lemma C.1. *Assume Condition 2.2 and let $\delta \in (0,1)$. For every $L \geq 1$ there exist $\lambda_0 > 0$ and a large enough k such that for every $\lambda \leq \lambda_0$, $|\xi_1| \leq \lambda^{-(1-\delta)}$ and $|x_1(0)| \in [k, k+\lambda^{-1})$ it holds that $|x_1(L)| \geq k$, where $x(L)$ is defined in (34).*

Proof. Recall that, for each $i \geq 1$, $x_1(i)$ is implicitly a function of the starting location $x_1(0)$, the parameter λ and the noise ξ_1. For notational convenience, in the following we set $h = \lambda^{-1}$ and study the limit $h \uparrow \infty$. In order to prove the thesis it is sufficient to show that for fixed, sufficiently large $k > 0$ we have

$$\inf_{\xi_1, x_1(0)} |x_1(L)| \to \infty \qquad \text{as } h \uparrow \infty\,, \tag{35}$$

where ξ_1 and $x_1(0)$ in the infimum are restricted as in the lemma's statement, i.e. $\xi_1 \in (-h^{1-\delta}, h^{1-\delta})$ and $x_1(0) \in (-(k+h), -k] \cup [k, k+h)$. In order to prove (35) we will show that for all $i \geq 1$, as $h \uparrow \infty$ we have

$$\Theta(h^{2\sum_{j=0}^{i-1} q^j}) \leq \inf|x_1(i)| \leq \sup|x_1(i)| \leq \Theta(h^{q^i + 2\sum_{j=0}^{i-1} q^j}), \tag{36}$$

$$\Theta(h^{2\sum_{j=1}^{i} q^j}) \leq \inf|\partial_1(i)| \leq \sup|\partial_1(i)| \leq \Theta(h^{q^{i+1} + 2\sum_{j=1}^{i} q^j})\,, \tag{37}$$

where infima and suprema run over $\xi_1 \in (-h^{1-\delta}, h^{1-\delta})$ and $x_1(0) \in (-(k+h), -k] \cup [k, k+h)$ as in (35), and $\partial_1(i)$ stands for $\partial \log \pi_1/\partial x_1(x_1(i))$. Note that, for any $i \geq 1$, (37) is implied by (36) thanks to $\inf|x_1(1)| \to \infty$ as $h \to \infty$ and (7). Thus it suffices to prove that (36) holds for all $i \geq 1$, which we will do by induction over i.

In the following, k is chosen large enough that $c|x_1|^q \leq |\partial \log \pi_1/\partial x_1(x_1)| \leq C|x_1|^q$ for some $0 < c \leq C < \infty$ and all $|x_1| > k$, which can be done by (7). Also, unless otherwise stated, we assume $\xi_1 \in (-h^{1-\delta}, h^{1-\delta})$ and $x_1(0) \in (-(k+h), -k] \cup [k, k+h)$, and all infima and suprema are taken over those sets.

Considering $i = 1$, we have $x_1(1) = x_1(0) + h\xi_1 + (h^2/2)\partial_1(0)$, which implies

$$\frac{h^2}{2}|\partial_1(0)| - |x_1(0)| - h|\xi_1| \leq |x_1(1)| \leq \frac{h^2}{2}|\partial_1(0)| + |x_1(0)| + h|\xi_1|\,.$$

Then, since $|\xi_1| \in (0, h^{1-q})$, $|x_1(0)| \in [k, k+h)$ and $ck^q \leq c|x_1(0)|^q \leq |\partial_1(0)| \leq C|x_1(0)|^q \leq C(h+k)^q$, we have

$$\Theta(h^2) = \frac{h^2}{2}ck^q - (k+h) - h^{2-\delta} \leq |x_1(1)| \leq \frac{h^2}{2}C(h+k)^q + (k+h) + h^{2-q} = \Theta(h^{2+q})$$

meaning that (36) is satisfied for $i = 1$.

We then show that if (36) and (37) hold for $i = 1, \dots, \ell - 1$, where $\ell \geq 2$, then they also hold for $i = \ell$. First note that when $\ell \geq 2$, (34) implies

$$x_1(\ell) = x_1(\ell - 1) + h\xi_1 + \frac{h^2}{2}\partial_1(0) + h^2 \sum_{j=1}^{\ell-1} \partial_1(j)\,. \tag{38}$$

From (38) and $|\xi_1| \in (0, h^{1-q})$, we can deduce that

$$h^2|\partial_1(\ell-1)| - |x_1(\ell-1)| - h^2 \sum_{j=0}^{\ell-2} |\partial_1(j)| \leq |x_1(\ell)| \leq |x_1(\ell-1)| + h^{2-q} + h^2 \sum_{j=0}^{\ell-1} |\partial_1(j)|. \tag{39}$$

Combining the lower bound in (39) with (36) and (37) for $i = 1, \dots, \ell - 1$ we obtain

$$\begin{aligned}\inf |x_1(\ell)| &\geq \inf h^2|\partial_1(\ell-1)| - \sup \left(|x_1(\ell-1)| + h^2 \sum_{j=0}^{\ell-2} |\partial_1(j)| \right) \\ &\geq \Theta(h^{2\sum_{j=0}^{\ell-1} q^j}) - \Theta(h^{q^{\ell-1} + 2\sum_{j=0}^{\ell-2} q^j}) = \Theta(h^{2\sum_{j=0}^{\ell-1} q^j})\,,\end{aligned}$$

where the last equality follows from $q^{\ell-1} + 2\sum_{j=0}^{\ell-2} q^j \leq 2\sum_{j=0}^{\ell-1} q^j$. Thus the lower bound in (36) holds also for $i = \ell$. Similarly, combining the upper bound in (39) with (36) and (37) for $i = 1, \dots, \ell - 1$ we obtain

$$\begin{aligned}\sup |x_1(\ell)| &\leq \sup \left(|x_1(\ell-1)| + h^{2-q} + h^2 \sum_{j=0}^{\ell-1} |\partial_1(j)| \right) \\ &\leq \Theta(h^{q^{\ell-1} + 2\sum_{j=0}^{\ell-2} q^j} + h^{2-q} + h^{q^i + 2\sum_{j=0}^{\ell-1} q^j}) = \Theta(h^{q^i + 2\sum_{j=0}^{\ell-1} q^j})\,,\end{aligned}$$

where the last equality follows from $2 - q \leq q^{\ell-1} + 2\sum_{j=0}^{\ell-2} q^j \leq q^i + 2\sum_{j=0}^{\ell-1} q^j$. Thus the upper bound in (36) holds also for $i = \ell$ and the proof is complete. □

Lemma C.2. *Condition 2.2 (ii) implies that there exist t, c and C in $(0, \infty)$ such that*

$$\pi_1(x_1) \leq Ce^{-c|x_1|^{1+q}}\,, \qquad \text{for all } |x_1| \geq t\,. \tag{40}$$

Proof. Condition 2.2 implies that there exists $t, c \in (0, \infty)$ such that

$$\left| \frac{d}{dx_1} \log \pi_1(x_1) \right| \geq c|x_1|^{1+q}\,, \qquad \text{for all } |x_1| \geq t\,.$$

Since $\log \pi_1 \in C_1(\mathbb{R})$, the above implies that either

$$\frac{d}{dx_1} \log \pi_1(x_1) > cx_1^{1+q} \quad \text{or} \quad \frac{d}{dx_1} \log \pi_1(x_1) < -cx_1^{1+q}\,,$$

holds for all $|x_1| \geq t$. Since $\int \pi_1(x_1)dx_1 = 1$ the latter option must be true. Computing the anti-derivative gives

$$\log \pi_1(x_1) \leq -cx_1^{1+q} + \log C,$$

for some constant $\log C$. An analogous argument can be used in the case $x_1 \downarrow -\infty$, and the two combined give the result. □

Lemma C.3. *If Condition 2.2 holds and $X_1 \sim \pi_1(\cdot)$, then there exists $\gamma \in (0, \infty)$ such that*

$$\mathbb{P}(|X_1| > k + h) \leq \Theta\left(e^{-\gamma h^{1+q} - q\log(h)}\right) \qquad \text{as } h \to \infty.$$

Proof. Using Lemma C.2, provided $k+h > t$ we have

$$\begin{aligned}\mathbb{P}(|X_1| > k+h) =& \int_{k+h}^{\infty} \pi_1(x_1)dx_1 + \int_{-\infty}^{-(k+h)} \pi_1(x_1)dx_1 \\ \leq& 2C\int_{k+h}^{\infty} e^{-cx_1^{1+q}}dx_1 \\ =& 2C\frac{c^{-1/(q+1)}}{q+1}\Gamma\left(\frac{1}{1+q}, c(k+h)^{1+q}\right),\end{aligned}$$

where $\Gamma(a,b) := \int_b^{\infty} u^{a-1}e^{-u}du$ is the incomplete Gamma function. For the case $q > 0$ the upper bound of Gautschi [1959], which is described on pages 771-772 of Alzer [1997], states that for fixed $a \in (0,1)$ and $x > 0$ we have

$$\Gamma(a, x^{-a}) \leq e^{-x^{-a}}\frac{c_a}{a}\left((x^{-a} + c_a^{-1})^a - x\right),$$

where $c_a := \Gamma(1+a)^{1/(1-a)}$. Setting $C_2 := 2Cc^{-1/(q+1)}/(q+1)$, $a := 1/(1+q)$ and using this upper bound gives

$$\mathbb{P}(|X_1| > k+h) \leq e^{-c(k+h)^{1+q}}C_2\frac{c_a}{a}\left[\left(c(k+h)^{\frac{1}{a}} + c_a^{-1}\right)^a - c^a(k+h)\right].$$

We use a Taylor series expansion of $f(c(k+h)^{1/a} + c_a^{-1})$ about $f(c(k+h)^{1/a})$, where $f(x) = x^a$. The terms each have a different power of h. This gives

$$(c(k+h)^{\frac{1}{a}} + c_a^{-1})^a = c^a(k+h) + c_a^{-1}a(c(k+h)^{\frac{1}{a}})^{a-1} + O(h^{(a-2)/a})$$

Since $a = 1/(1+q) < 1$ then $(a-1)/a = -q$ and $(a-2)/a = -(1+2q)$, and therefore

$$(c(k+h)^{\frac{1}{a}} + c_a^{-1})^a - c^a(k+h) = \Theta((c(k+h)^{\frac{1}{a}})^{a-1}) = \Theta(h^{-q}).$$

Combining with the above, we can write that for any fixed k and fixed $q > 0$, there exists $\gamma \in (0,\infty)$ such that as $h \uparrow \infty$

$$\mathbb{P}(|X_1| > k+h) \leq \Theta\left(e^{-\gamma h^{1+q} - q\log(h)}\right).$$

In the case $q = 0$ the integral $\int_{k+h}^{\infty} e^{-cx_1}dx_1 = e^{-c(k+h)}/c$ and the result is immediate. □

C.2 Proofs for Section 3

Proof of Proposition 3.1. Setting $y - x = z$, then $t(z) = e^{z\nabla \log \pi(x)}$ and $1/t(z) = e^{-z\nabla \log \pi(x)} = t(-z)$, meaning

$$\begin{aligned}Z(x) =& \int_{\mathbb{R}} \frac{t(z)}{1+t(z)}\mu_\sigma(z)dz \\ =& \int_0^{\infty}\left(\frac{t(z)}{1+t(z)}\mu_\sigma(z) + \frac{t(-z)}{1+t(-z)}\mu_\sigma(-z)\right)dz.\end{aligned}$$

Noting that $\mu_\sigma(z) = \mu_\sigma(-z)$ and $t(-z) = 1/t(z)$ then gives

$$Z(x) = \int_0^{\infty}\mu_\sigma(z)dz = \frac{1}{2}$$

which completes the proof. □

Proof of Proposition 3.2. Assume $y = x + b(x,z) \times z$ is generated using Algorithm 1. Then for any $A \in \mathcal{B}(\mathbb{R})$

$$\mathbb{P}[y \in A] = \mathbb{P}\left[\{z \in A - x\} \cap \{b(x,z) = 1\}\right] + \mathbb{P}\left[\{-z \in A - x\} \cap \{b(x,z) = -1\}\right].$$

Note that the second term on the right-hand side can be re-written

$$\mathbb{P}\left[\{z \in A - x\} \cap \{b(x,-z) = -1\}\right],$$

owing to the symmetry of μ_σ. Because of this, we can write

$$\begin{aligned}
\mathbb{P}[y \in A] &= \int_{A-x} \frac{e^{z\nabla \log \pi(x)}}{1 + e^{z\nabla \log \pi(x)}} \mu_\sigma(z)dz + \int_{A-x} \frac{1}{1 + e^{-z\nabla \log \pi(x)}} \mu_\sigma(z)dz \\
&= 2\int_{A-x} \frac{e^{z\nabla \log \pi(x)}}{1 + e^{z\nabla \log \pi(x)}} \mu_\sigma(z)dz \\
&= Q^B(x, A)
\end{aligned}$$

which completes the proof. □

Proof of Proposition 3.3. We establish a point-wise bound on the candidate transition densities of the two algorithms. Combining this with Lemma A.1 gives an equivalent bound on the spectral gaps. To reach this point-wise bound, first note that the candidate transition density associated with the Random Walk algorithm is $q^R(x, x+z) = \mu_\sigma(z)$ for any $x, z \in \mathbb{R}^d$. Now, for the modified Barker proposal, the candidate density can be written

$$\begin{aligned}
\check{q}^B(x, x+z) &= \mu_\sigma(z)\check{p}(x,z) + \mu_\sigma(-z)(1 - \check{p}(x,-z)) \\
&= \mu_\sigma(z)\left(\check{p}(x,z) - \check{p}(x,-z) + 1\right) \\
&= 2\check{p}(x,z)\mu_\sigma(z),
\end{aligned}$$

where on the last line we have used that $\check{p}(x,-z) = 1 - \check{p}(x,z)$. Noting that $\tilde{p}(x,z) \leq 1$ establishes that $q^R(x, x+z) \geq \check{q}^B(x, x+z)/2$ for any $x, z \in \mathbb{R}^d$, and upon combining this with Lemma A.1 the result follows. □

C.3 Proofs for Section 4

Interestingly, the proof of the lower bound of Theorem 4.1 is analogous to the one of Theorem 2.1, providing further insight into the similarity between the Barker scheme and random walk in terms of robustness to scales.

Proof of Theorem 4.1. As in the proof of Theorem 2.1, we write Q^B_λ to denote the Barker candidate kernel targeting $\pi^{(\lambda)}$, and $\tilde{Q}^B_\lambda(x, dy) := \tilde{q}^B_\lambda(x,y)dy$ to denote the isomorphic kernel defined as $\tilde{Q}^B_\lambda = \phi \circ Q^B_\lambda$, where ϕ is the same function used in the proof of Theorem 2.1. Also, we denote by P^B_λ and $\tilde{P}^B_\lambda$ the Metropolis-Hastings kernels with candidate kernels Q^B_λ and $\tilde{Q}^B_\lambda$, respectively, and target distributions $\pi^{(\lambda)}$ and π, respectively.

From (16) and (17) it follows that

$$\tilde{q}^B_\lambda(x,y) = 2^d \, \frac{\lambda}{\sigma^d} \mu\left(\frac{\delta_\lambda}{\sigma}\right) \prod_{i=1}^{d} \left(1 + e^{-\partial_i \log \pi(x)(y_i - x_i)}\right)^{-1}. \tag{41}$$

Here we are using μ to denote the d-dimensional distribution obtained by proposing each coordinate independently as in Section 3.3. We therefore have

$$\frac{\tilde{q}^B_\lambda(x,y)}{\tilde{q}^B_1(x,y)} = \lambda \frac{\mu(\delta_\lambda/\sigma)}{\mu(\delta/\sigma)}, \tag{42}$$

which holds after noting that $(1 + e^{-\partial_i \log \pi(x)(y_i - x_i)})$ does not depend on λ, and hence cancels in the ratio. Note that the expression above coincides with the expression for the random walk proposals in (33). Thus, arguing as in the proof of Theorem 2.1, we have that $\tilde{q}^B_\lambda(x, y) \geq \lambda \tilde{q}^B(x, y)$ for all $\lambda \leq \lambda_0$ and all $x, y \in \mathbb{R}^d$, where $\lambda_0 \leq 1$ is the value defined in Condition 2.1. Combining the latter inequality with Lemma A.1 and using the isomorphism property between $\tilde{P}^B_\lambda$ and P^B_λ given in Lemmas B.1 and B.2, we obtain

$$\mathrm{Gap}(P^B_\lambda) \geq \lambda \mathrm{Gap}(P^B) = \Theta(\lambda) \qquad \text{as } \lambda \downarrow 0\,.$$

To show that $\mathrm{Gap}(P^B_\lambda) \leq \Theta(\lambda)$, note that $\tilde{q}^B_\lambda(x, y) \leq 2^d \tilde{q}^R_\lambda(x, y)$ for all $x, y \in \mathbb{R}^d$ by (41) and (4). Thus, Lemma A.1 and Theorem 2.1 give $\mathrm{Gap}(P^B_\lambda) \leq 2^d \mathrm{Gap}(P^R_\lambda) = \Theta(\lambda)$ as $\lambda \downarrow 0$. □

C.3.1 Proof of Theorem 4.2

The following lemma, which is an extension of Theorem 4.1 of [Roberts and Tweedie, 1996], provides generic sufficient conditions for the geometric ergodicity of Metropolis–Hastings algorithms.

Lemma C.4. *Let P be a ϕ-irreducible and aperiodic Metropolis–Hastings kernel on $\mathbb{R}^d$ with proposal Q such that compact sets are small under P. If there exist a function $V : \mathbb{R}^d \to (0, \infty)$ such that $\sup_{x \in \mathbb{R}^d} \frac{QV(x)}{V(x)} < \infty$ and*

$$\liminf_{\|x\| \to +\infty} \int_{R^d} q(x, y)\alpha(x, y)\, dy > \limsup_{\|x\| \to \infty} \frac{QV(x)}{V(x)}\,, \tag{43}$$

then P is π-a.e. geometrically ergodic.

Proof. We show that (43) implies the following Foster-Lyapunov drift conditions:

$$\sup_{x \in \mathbb{R}^d} \frac{PV(x)}{V(x)} < \infty \text{ and } \limsup_{\|x\| \to \infty} \frac{PV(x)}{V(x)} < 1\,,$$

which imply π-a.e. geometric ergodicity (see e.g. Theorem 3.1 and Lemma 3.5 of Jarner and Hansen [2000]). First note that

$$\begin{aligned}\frac{PV(x)}{V(x)} &= \int_{\mathbb{R}^d} \left(\frac{V(y)}{V(x)} \alpha(x, y) + 1 - \alpha(x, y) \right) q(x, y) dy \\ &\leq \int_{\mathbb{R}^d} \frac{V(y)}{V(x)} q(x, y) dy + \int_{\mathbb{R}^d} (1 - \alpha(x, y))\, q(x, y) dy \leq \frac{QV(x)}{V(x)} + 1\,,\end{aligned}$$

which implies $\sup_{x \in \mathbb{R}^d} \frac{PV(x)}{V(x)} \leq \sup_{x \in \mathbb{R}^d} \frac{QV(x)}{V(x)} + 1 < \infty$. Also, the inequalities above imply

$$\frac{PV(x)}{V(x)} \leq 1 - \left(\int_{\mathbb{R}^d} \alpha(x, y) q(x, y) dy - \frac{QV(x)}{V(x)} \right)\,. \tag{44}$$

From (43) we have

$$\begin{aligned}0 &< \liminf_{\|x\| \to +\infty} \int_{\mathbb{R}^d} q(x, y)\alpha(x, y)\, dy - \limsup_{\|x\| \to \infty} \frac{QV(x)}{V(x)} \\ &\leq \liminf_{\|x\| \to +\infty} \left(\int_{\mathbb{R}^d} q(x, y)\alpha(x, y)\, dy - \frac{QV(x)}{V(x)} \right)\,.\end{aligned} \tag{45}$$

Combining (44) and (45) we obtain $\limsup_{\|x\| \to \infty} \frac{PV(x)}{V(x)} < 1$, as desired. □

We will show that the conditions of Lemma C.4 are satisfied when considering a Lyapunov function $V_s(x) = \exp(s\|x\|_\infty)$ based on the sup norm, $\|x\|_\infty = \sup_i |x_i|$.

In the following results we denote $\sup_{t>0} g(t)$ by M. We denote the log-target and its derivatives as $U(x) = \log \pi(x)$ and $U_i(x) = \frac{\partial}{\partial x_i} U(x)$, respectively. Condition 4.1 implies that $\nabla U(x) = f'(\|x\|)\frac{x}{\|x\|}$ and $U_i(x) = f'(\|x\|)\frac{x_i}{\|x\|}$ for $\|x\| > R$. Also, we denote the kernel $Q^{(g)}$ in (21) as Q for brevity and its density function as

$$q(x,y) = \prod_{i=1}^{d} \frac{g(e^{w_i U_i(x)})\mu_\sigma(w)}{Z_i(x)} = \prod_{i=1}^{d} q_i(w_i; x)\,, \tag{46}$$

where $w_i = y_i - x_i$ and $q_i(w_i; x) = g(e^{w_i U_i(x)})\mu_\sigma(w)/Z_i(x)$.

First, we provide some simple results on the behaviour of g, Z_i and q_i that will be useful afterwards.

Lemma C.5. *Let $g : (0,\infty) \to (0,\infty)$ be bounded, non-decreasing and such that $g(t) = tg(1/t)$ for all $t > 0$. Then $g(t) \geq g(1)\min\{1,t\}$ and $\frac{g(1)}{2} \leq Z_i(x) \leq M$, where $M = \sup_{t>0} g(t)$.*

Proof. If $t \geq 1$ then $g(t) \geq g(1) = g(1)\min\{1,t\}$ by the monotonicity of g. If $t < 1$ then $g(t) = tg(1/t) \geq tg(1) = g(1)\min\{1,t\}$ by $g(t) = tg(1/t)$ and the monotonicity of g. From $Z_i(x) = \int_{\mathbb{R}} g(e^{w_i U_i(x)})\mu_\sigma(w)dw$ and $g(t) \leq M$ it follows $Z_i(x) \leq M$. If $U_i(x) \leq 0$, then $g(e^{w_i U_i(x)}) \geq g(1)$ for all $w \leq 0$ and thus $Z_i(x) \geq \int_{-\infty}^{0} g(1)\mu_\sigma(w)dw = \frac{g(1)}{2}$. The case $U_i(x) \geq 0$ is analogous. □

Lemma C.6. *If g is bounded and non-decreasing, then $Z_i(x) \to \frac{M}{2}$ as $U_i(x) \to -\infty$ or $U_i(x) \to +\infty$ and for all $w_i \in \mathbb{R}$ it holds*

$$\begin{aligned} q_i(w_i; x) &\to 2\mu_\sigma(w_i)\,\mathbb{I}_{(-\infty,0]}(w_i) \qquad \text{as } U_i(x) \to -\infty \text{ and} \\ q_i(w_i; x) &\to 2\mu_\sigma(w_i)\,\mathbb{I}_{[0,+\infty)}(w_i) \qquad \text{as } U_i(x) \to +\infty\,. \end{aligned}$$

Proof. Consider the case $U_i(x) \to -\infty$. From $g(t) = t\,g(1/t) \leq tM$ it follows $g(t) \to 0$ as $t \to 0$. Also, from the boundedness and monotonicity of g it holds $g(t) \to M$ as $t \to \infty$. Therefore, for all $w_i \in \mathbb{R}$,

$$g(\exp(w_i U_i(x))) \to M\,\mathbb{I}_{(-\infty,0]}(w_i) \qquad \text{as } U_i(x) \to -\infty\,. \tag{47}$$

Thus, from the bounded convergence theorem $Z_i(x) \to \int_{-\infty}^{0} M\,\mu_\sigma(w_i)dw_i = \frac{M}{2}$ as $U_i(x) \to -\infty$ and, consequently, $q_i(w_i; x) \to 2\mu_\sigma(w_i)\,\mathbb{I}_{(-\infty,0]}(w_i)$ as $U_i(x) \to -\infty$. The case $U_i(x) \to +\infty$ is analogous. □

We now provide two lemmas that will be used to prove the inequality in (43).

Lemma C.7. *Suppose Condition 4.1 holds. Let $V_s(x) = \exp(s\|x\|_\infty)$ and Q the kernel with density q as in* (46)*. Then*

$$\inf_{s>0} \limsup_{\|x\|\to\infty} \frac{QV_s(x)}{V_s(x)} = 0\,.$$

Proof. Let $x \in \mathbb{R}^d$ and $Y \sim Q(x,\cdot)$. Since $V_s(y) \leq \sum_{i=1}^{d} \exp(s|y_i|)$ we have

$$\mathbb{E}\left[\frac{V_s(Y)}{V_s(x)}\right] \leq \sum_{i=1}^{d} \mathbb{E}\left[\frac{e^{s|Y_i|}}{e^{s\|x\|_\infty}}\right].$$

We now bound $\mathbb{E}\left[e^{s(|Y_i| - \|x\|_\infty)}\right]$ differently depending on whether $|x_i| \leq \frac{1}{2}\|x\|_\infty$ or $\frac{1}{2}\|x\|_\infty < |x_i| \leq \|x\|_\infty$.

If $|x_i| \leq \frac{1}{2}\|x\|_\infty$ it follows from the triangle inequality that $|x_i + w| - \|x\|_\infty \leq |x_i| + |w| - \|x\|_\infty \leq |w| - \|x\|_\infty/2$ for any $w \in \mathbb{R}$. Also, from (46) and Lemma C.5 we have $q_i(w_i; x) \leq \frac{2M}{g(1)}\mu_\sigma(w_i)$. It follows

$$\mathbb{E}\left[e^{s(|Y_i|-\|x\|_\infty)}\right]\mathbb{I}\left(|x_i| \leq \frac{\|x\|_\infty}{2}\right) \leq \frac{2M}{g(1)}e^{-s\|x\|_\infty/2}\int_{\mathbb{R}} e^{s|w|}\mu_\sigma(w)dw\,,$$

and thus

$$\limsup_{\|x\|\to\infty} \mathbb{E}\left[e^{s(|Y_i|-\|x\|_\infty)}\right]\mathbb{I}\left(|x_i| \leq \frac{\|x\|_\infty}{2}\right) = 0\,. \tag{48}$$

If $\frac{1}{2}\|x\|_\infty < |x_i| \leq \|x\|_\infty$ we have

$$\mathbb{E}\left[e^{s(|Y_i|-\|x\|_\infty)}\right]\mathbb{I}\left(|x_i| > \frac{\|x\|_\infty}{2}\right) \leq \mathbb{I}\left(|x_i| > \frac{\|x\|_\infty}{2}\right)\int_{\mathbb{R}} e^{s(|x_i+w|-|x_i|)}q_i(w;x)dw\,.$$

If $\|x\| \to \infty$ and $|x_i| > \frac{\|x\|_\infty}{2}$ it follows $|x_i| \to \infty$. Moreover, by Condition 4.1 and $|x_i| > \frac{\|x\|_\infty}{2}$, we have $U_i(x) \leq \frac{f(\|x\|)}{2} \to -\infty$ as $x_i \to +\infty$ and $U_i(x) \geq -\frac{f(\|x\|)}{2} \to +\infty$ as $x_i \to -\infty$. Therefore, by Lemma C.6

$$\limsup_{\|x\|\to\infty} \mathbb{I}\left(|x_i| > \frac{\|x\|_\infty}{2}\right)\int_{\mathbb{R}} e^{s(|x_i+w|-|x_i|)}q_i(w;x)dw \leq 2\int_{-\infty}^{0} e^{sw}\mu_\sigma(w)dw\,.$$

Combining the last two displayed equations we get

$$\limsup_{\|x\|\to\infty} \mathbb{E}\left[e^{s(|Y_i|-\|x\|_\infty)}\right]\mathbb{I}\left(|x_i| > \frac{\|x\|_\infty}{2}\right) \leq 2\int_{-\infty}^{0} e^{sw}\mu_\sigma(w)dw\,. \tag{49}$$

From (48), (49) and basic properties of the lim sup we get

$$\limsup_{\|x\|\to\infty} \mathbb{E}\left[e^{s(|Y_i|-\|x\|_\infty)}\right] \leq 2\int_{-\infty}^{0} e^{sw}\mu_\sigma(w)dw\,.$$

Thus

$$\limsup_{\|x\|\to\infty} \mathbb{E}\left[\frac{V_s(Y)}{V_s(x)}\right] \leq d\left(\int_{-\infty}^{0} e^{sw}2\mu_\sigma(w)dw\right)$$

which goes to 0 as $s \to \infty$. □

Lemma C.8. *Assume that* $\inf_{w\in(-\delta,\delta)}\mu_\sigma(w) > 0$ *for some* $\delta > 0$. *Under Condition 4.1 it holds*

$$\liminf_{\|x\|\to\infty}\int_{\mathbb{R}^d} q(x,y)\alpha(x,y)dy > 0\,. \tag{50}$$

Proof. Let $w = y - x$ and $\mu_\sigma(w) = \prod_{i=1}^d \mu_\sigma(w_i)$. Also, denote by $\alpha(w;x) = \alpha(x,y)$ the MH acceptance rate when moving from x to y. We write $f(w;x) \gtrsim g(w;x)$ if the function $f(w;x)$ is greater or equal than $g(w;x)$ up to positive constants independent of x and w. From Lemma C.5 we have $\frac{g(1)}{2} \leq Z_i(x) \leq M$ and thus

$$\begin{aligned}
&q(w;x)\alpha(w;x)\\
&= \frac{\mu_\sigma(w)}{\prod_{i=1}^d Z_i(x)}\min\left\{\prod_{i=1}^d g(e^{w_iU_i(x)}), e^{U(x+w)-U(x)}\prod_{i=1}^d \frac{g(e^{-w_iU_i(x+w)})Z_i(x)}{Z_i(x+w)}\right\}\\
&\gtrsim \mu_\sigma(w)\min\left\{\prod_{i=1}^d g(e^{w_iU_i(x)}), e^{U(x+w)-U(x)}\prod_{i=1}^d g(e^{-w_iU_i(x+w)})\right\}.
\end{aligned}$$

Then, using $g(t) \geq g(1)\min\{1,t\}$ from Lemma C.5 we obtain

$$\begin{aligned}
&q(w;x)\alpha(w;x) \\
&\gtrsim \mu_\sigma(w)\min\left\{\prod_{i=1}^d g(e^{w_iU_i(x)}), g(1)^d e^{U(x+w)-U(x)+\sum_{i=1}^d \min\{-w_iU_i(x+w),0\}}\right\} \\
&\gtrsim \mu_\sigma(w)\min\left\{\prod_{i=1}^d g(e^{w_iU_i(x)}), e^{U(x+w)-U(x)+\sum_{i=1}^d \min\{-w_iU_i(x+w),0\}}\right\}.
\end{aligned}$$

Assume $\|x\|$ large and $w \in A(x)$, where $A(x) = \{w \in \mathbb{R}^d : \|x+w\| \leq \|x\| - \epsilon,\ \|w\| \leq 2\epsilon$ and $x_iw_i \leq 0$ for all $i\}$ for some fixed $\epsilon > 0$. From $x_iw_i \leq 0$ it follows $w_iU_i(x) \geq 0$ and thus, from the monotonicity of g, $g(e^{w_iU_i(x)}) \geq g(1)$. Combining the latter with the last displayed equation we have

$$q(w;x)\alpha(w;x) \gtrsim \mu_\sigma(w)\min\left\{g(1)^d, e^{U(x+w)-U(x)+\sum_{i=1}^d \min\{-w_iU_i(x+w),0\}}\right\}. \tag{51}$$

We now lower bound $U(x+w) - U(x) + \sum_{i=1}^d \min\{-w_iU_i(x+w), 0\}$. For $\|x\| > R$, from Condition 4.1

$$\begin{aligned}
&U(x+w) - U(x) + \sum_{i=1}^d \min\{-w_iU_i(x+w), 0\} \\
&= f(\|x+w\|) - f(\|x\|) + \frac{f'(\|x+w\|)}{\|x+w\|}\sum_{i=1}^d \min\{-w_i(x_i+w_i), 0\}.
\end{aligned}$$

Using the non-increasingness of f' and $w \in A(x)$ we have $f(\|x+w\|) - f(\|x\|) \geq -f'(\|x+w\|)(\|x\| - \|x+w\|) \geq -f'(\|x+w\|)\epsilon$. Thus

$$\begin{aligned}
&U(x+w) - U(x) + \sum_{i=1}^d \min\{-w_iU_i(x+w), 0\} \\
&\geq -f'(\|x+w\|)\left(\epsilon + \frac{\sum_{i=1}^d \min\{-w_i(x_i+w_i), 0\}}{\|x+w\|}\right).
\end{aligned}$$

Since $w \in A(x)$ it follows $x_iw_i \leq 0$ and $\min\{-w_i(x_i+w_i), 0\} \geq -w_i^2 \geq -(2\epsilon)^2$. Thus

$$\inf_{w\in A(x)} \frac{\sum_{i=1}^d \min\{-w_i(x_i+w_i), 0\}}{\|x+w\|} \geq -\frac{(2\epsilon)^2}{\|x\| - 2\epsilon},$$

which goes to 0 as $\|x\| \to \infty$. It follows that

$$\begin{aligned}
&\liminf_{\|x\|\to\infty} \inf_{w\in A(x)} U(x+w) - U(x) + \sum_{i=1}^d \min\{-w_iU_i(x+w), 0\} \geq \\
&\liminf_{\|x\|\to\infty} -f'(\|x\| - 2\epsilon)\epsilon = \infty.
\end{aligned}$$

Combining the last displayed equation with (51) we have

$$\liminf_{\|x\|\to\infty} \inf_{w\in A(x)} q(w;x)\alpha(w;x) \gtrsim \liminf_{\|x\|\to\infty} \inf_{w\in A(x)} \mu_\sigma(w) > 0,$$

where the last inequality holds for sufficiently small ϵ because of the assumption $\inf_{w\in(-\delta,\delta)} \mu_\sigma(w) > 0$. Therefore

$$\begin{aligned}
\liminf_{\|x\|\to\infty} \int_{\mathbb{R}^d} q(x,y)\alpha(x,y)dy &\geq \liminf_{\|x\|\to\infty} \int_{A(x)} q(w;x)\alpha(w;x)dw \\
&\gtrsim \liminf_{\|x\|\to\infty} \int_{A(x)} 1\, dw.
\end{aligned}$$

The proof is completed noting that $\liminf_{\|x\|\to\infty} \int_{A(x)} 1\, dw > 0$ by the construction of $A(x)$. $\square$

Proof of Theorem 4.2. Lemmas C.7 and C.8 imply that there exist an $s > 0$ such that V_s satisfy (43). The thesis then follows from Lemma C.4, noting that compact sets are small for P (which can be deduced from the fact that $\inf \pi(x) > 0$ on compact sets) and that that $\sup_x QV_s(x)/V_s(x) < \infty$ because

$$\frac{QV_s(x)}{V_s(x)} \leq \sum_{i=1}^{d} \int_{\mathbb{R}} e^{s|w_i|} q_i(w_i; x) dw_i \leq 2d \int_{\mathbb{R}} e^{s|w_i|} \mu_\sigma(w_i) dw_i < \infty$$

where we used $e^{\|y\|_\infty - \|x\|_\infty} \leq e^{\|y-x\|_\infty} \leq \sum_i e^{|y_i - x_i|}$, $q_i(w_i; x) \leq 2\mu_\sigma(w_i)$ and $\int_{\mathbb{R}} \exp(s|w|)\mu_\sigma(w)dw \leq 2\int_{\mathbb{R}} \exp(sw)\mu_\sigma(w)dw < \infty$ for every $s > 0$. □

C.4 Proof of Proposition 4.1

Proposition 4.1 follows directly from Lemmas C.9 and C.10 below.

Lemma C.9. *Under the assumptions of Proposition 4.1 we have*

$$\log\left(\frac{f(x_i + \sigma u_i)}{f(x_i)} \frac{g\left(e^{-\phi'(x_i+\sigma u_i)\sigma u_i}\right)}{g\left(e^{\phi'(x_i)\sigma u_i}\right)}\right) = \mathcal{O}\left(\sigma^3\right) \qquad \text{as } \sigma \to 0\,, \tag{52}$$

for all $x_i, w_i \in \mathbb{R}$.

Proof. Define the function b as $b(s) = \log(g(\exp(s)))$ for all $s \in \mathbb{R}$. For any x_i, u_i in $\mathbb{R}$, we have

$$\begin{aligned}
&\log\left(\frac{f(x_i + \sigma u_i)}{f(x_i)} \frac{g\left(e^{-\phi'(x_i+\sigma u_i)\sigma u_i}\right)}{g\left(e^{\phi'(x_i)\sigma u_i}\right)}\right) \\
&= \phi(x_i + \sigma u_i) - \phi(x_i) + b(-\phi'(x_i + \sigma u_i)\sigma u_i) - b(\phi'(x_i)\sigma u_i) \\
&= c_1(x_i)\phi'(x_i)u_i\sigma + c_2(x_i)\frac{u_i^2\sigma^2}{2} + c_3(x_i)\frac{u_i^3\sigma^3}{6} + \mathcal{O}\left(\sigma^4\right) \quad \text{as } \sigma \to 0\,,
\end{aligned} \tag{53}$$

where $c_1(x_i)$ and $c_2(x_i)$ are the coefficients of the second order Taylor expansion about $\sigma = 0$, and are given by $c_1(x_i) = (1 - 2b'(0))\,\phi'(x_i)$ and $c_2(x_i) = (1 - 2b'(0))\,\phi''(x_i)$. To conclude, we now show that the assumptions on g imply $b'(0) = 1/2$ and $c_1(x_i) = c_2(x_i) = 0$. By definition of b it holds that $b'(0) = g'(1)/g(1)$. From $g(t) = t\,g(1/t)$ it follows $g(1+\epsilon) = (1+\epsilon)\,g((1+\epsilon)^{-1})$ and thus $\frac{g(1+\epsilon) - g((1+\epsilon)^{-1})}{2\epsilon} = \frac{g((1+\epsilon)^{-1})}{2}$. Taking the limit $\epsilon \downarrow 0$ and using $(1+\epsilon)^{-1} = 1 - \epsilon + \mathcal{O}(\epsilon^2)$ it follows that $g'(1) = \frac{g(1)}{2}$ and thus $b'(0) = 1/2$ and $c_1(x_i) = c_2(x_i) = 0$. Combining the latter with (53) we obtain (52). □

Remark C.1. *For general ϕ, x_i and u_i, we have $\log(\alpha_i(x_i, x_i + \sigma u_i)) = \Theta(\sigma^3)$ because the third coefficient in the Taylor expansion in* (53)*, which is given by*

$$c_3(x_i) = 6b''(0)\phi'(x_i)\phi''(x_i) - 2b'''(0)\phi'(x_i)^3 + (1 - 3b'(0))\phi'''(x_i)\,,$$

is non-zero in general.

Lemma C.10. *Under the assumptions of Proposition 4.1 we have*

$$\log\left(\frac{Z_i(x_i)}{Z_i(x_i + \sigma u_i)}\right) = \mathcal{O}\left(\sigma^3\right) \qquad \text{as } \sigma \to 0\,,$$

for all $x_i, w_i \in \mathbb{R}$.

Proof. Without loss of generality, assume $g(1) = 1$ throughout the proof. First consider $\log(Z_i(x_i))$, which can be written as

$$Z_i(x_i) = \int_{\mathbb{R}} g\left(e^{\phi'(x_i)(y_i - x_i)}\right) \sigma^{-1} \mu\left(\frac{y_i - x_i}{\sigma}\right) dy_i = \int_{\mathbb{R}} g\left(e^{\phi'(x_i)\sigma s}\right) \mu(s) ds\,. \tag{54}$$

For every non-negative integer j, denote by κ_j the j-th moment of the distribution $\mu(\cdot)$. Note that, since μ is a symmetric pdf, $\kappa_0 = 1$, $\kappa_j = 0$ if j is odd and $\kappa_j > 0$ if j is even. For $j \in \{1, 2, 3\}$, we have

$$\frac{\partial^j}{\partial \sigma^j} Z_i(x_i)\Big|_{\sigma=0} = \int_{\mathbb{R}} \frac{\partial^j}{\partial \sigma^j} g\left(e^{\phi'(x_i)\sigma s}\right)\Big|_{\sigma=0} \mu(s) ds = \int_{\mathbb{R}} \frac{\partial^j}{\partial \sigma^j} g\left(e^{\phi'(x_i)\sigma}\right)\Big|_{\sigma=0} s^j \mu(s) ds = \frac{\partial^j}{\partial \sigma^j} g\left(e^{\phi'(x_i)\sigma}\right)\Big|_{\sigma=0} \kappa_j\,, \tag{55}$$

where the exchange of integration and derivation is justified by the assumptions on g and μ. Using the Taylor expansion of the function $\sigma \mapsto \log(h(\sigma))$ for general h about $\sigma = 0$, and the fact that $Z_i(x_i)\Big|_{\sigma=0} = 1$ and $\frac{\partial^j}{\partial \sigma^j} Z_i(x_i)\Big|_{\sigma=0} = 0$ if j is odd, we have

$$\begin{aligned}\log(Z_i(x_i)) &= \kappa_2 \frac{\partial^2}{\partial \sigma^2} g\left(e^{\phi'(x_i)\sigma}\right)\Big|_{\sigma=0} \frac{\sigma^2}{2} + \mathcal{O}(\sigma^4) \\ &= \kappa_2 (g'(1) + g''(1)) \phi'(x_i)^2 \frac{\sigma^2}{2} + \mathcal{O}(\sigma^4) \quad \text{as } \sigma \to 0\,.\end{aligned} \tag{56}$$

Set $y_i = x_i + \sigma u_i$, then from (54) and (55)

$$\frac{\partial^j}{\partial \sigma^j} Z_i(x_i + \sigma u_i)\Big|_{\sigma=0} = \int_{\mathbb{R}} \frac{\partial^j}{\partial \sigma^j} g\left(e^{\phi'(x_i + \sigma u_i)\sigma s}\right)\Big|_{\sigma=0} \mu(s) ds$$

Reordering the Taylor expansion of $g\left(e^{\phi'(x_i+\sigma u_i)\sigma s}\right)$ about $\sigma = 0$ as a polynomial of s and keeping only even powers in s we get

$$Z_i(x_i + \sigma u_i) = 1 + \kappa_2 (g'(1) + g''(1)) \phi'(x_i)^2 \frac{\sigma^2}{2} + \mathcal{O}(\sigma^3)\,.$$

Using the expansion of $\log(h(\sigma))$ for general h about $\sigma = 0$, and the fact that $Z_i(x_i+\sigma u_i)\Big|_{\sigma=0} = 1$ and $\frac{\partial}{\partial \sigma} Z_i(x_i + \sigma u_i)\Big|_{\sigma=0} = 0$, we have

$$\log(Z_i(x_i + \sigma u_i)) = \kappa_2 (g'(1) + g''(1)) \phi'(x_i)^2 \frac{\sigma^2}{2} + \mathcal{O}(\sigma^3)\,.$$

Combining the latter equation with (56) we have

$$\log\left(\frac{Z_i(x_i)}{Z_i(x_i + \sigma u_i)}\right) = \log(Z_i(x_i)) - \log(Z_i(x_i + \sigma u_i)) = \mathcal{O}(\sigma^3)$$

□

Remark C.2. *For the Barker proposal, the normalization term $Z_i(x_i)$ is constant over x_i and thus Lemma C.10 is trivially satisfied.*

D Condition 2.3 for the exponential family class

Proposition D.1. *Condition 2.3 holds in the case in which there are $\alpha, \beta > 0$ such that*

$$\pi(x) \propto \exp\{-\alpha \|x\|^{\beta}\}$$

Proof. Condition (ii) is immediate. For (i), first note that here

$$\left| \frac{\partial \log \pi(x)}{\partial x_1} \right| \|x\|^{\gamma} = -\alpha\beta x_1 \|x\|^{\gamma+\beta-2}. \tag{57}$$

Note that $\|x\| = \sqrt{(\sum_i x_i^2)}$ is a monotonically increasing function in each $|x_i|$, so the infimum over $(x_2, ..., x_d)$ of (57) is realised at $x_2 = ... = x_d = 0$. Choosing $\gamma = 2$ condition (i) is satisfied because

$$\liminf_{|x_1| \to \infty} \alpha\beta |x_1|^{1+\beta} = \infty.$$

□

E First-order exact Metropolis-Hastings proposals

Intuitively, we would like any method that uses gradient information to be exact at the first order. In a Metropolis-Hastings context, this means a proposal that incorporates gradient information should be reversible with respect to measures that possess a log-linear density function, i.e. $\pi(x) = \exp(ax + b)$ for some $a, b \in \mathbb{R}$. In such cases the gradient at any location encompasses full information and this would therefore seem to be a sensible minimal goal for well-designed gradient-based methods. The Langevin and Hamiltonian schemes both satisfy this stipulation. As the following proposition shows, for any instance of the class defined in (15), the condition $g(t) = tg(1/t)$ is both sufficient and necessary for the proposal distribution to satisfy such a requirement.

Proposition E.1. *Let μ_σ be a symmetric probability density function on $\mathbb{R}$ and $\pi(x) = \exp(ax + b)$ for some $a, b \in \mathbb{R}$, with $a \neq 0$. Then a transition kernel of the form in (15) is π-reversible if and only if $g(t) = tg(1/t)$ for every $t > 0$.*

Proof. Since $\nabla \log \pi(x) = a$ for every $x \in \mathbb{R}$, it follows that the normalizing constant, Z, of $q(x, y)$ in (15) is independent of x. First we show that $g(t) = tg(1/t)$ implies reversibility. From the symmetry of μ_σ and $g(t) = tg(1/t)$ it follows

$$\begin{aligned}
\pi(x)q(x, y) &= \exp(ax + b) Z^{-1} g\left(\exp\left(a(y - x)\right)\right) \mu_\sigma(x, y) \\
&= \exp(ax + b) Z^{-1} \exp\left(a(y - x)\right) g\left(\exp\left(-a(y - x)\right)\right) \mu_\sigma(y, x) \\
&= \pi(y)q(y, x)\,,
\end{aligned}$$

which implies that q is π-reversible. Conversely, if q is π-reversible, then

$$1 = \frac{\pi(x)q(x, y)}{\pi(y)q(y, x)} = \frac{\exp(a(x - y)) g\left(1/\exp\left(a(x - y)\right)\right)}{g\left(\exp\left(a(x - y)\right)\right)} = \frac{tg(1/t)}{g(t)}\,,$$

for $t = \exp(a(x - y))$. For $a \neq 0$, $\exp(a(x - y))$ takes any positive value as $x, y \in \mathbb{R}^d$ and thus we have $g(t) = tg(1/t)$ for every $t > 0$. □

Remark E.1. *Note that $\pi(x) = \exp(ax+b)$ is an improper density function because $\int_{\mathbb{R}} \exp(ax+b)dx = \infty$ for any choice of a and b. This, however, does not pose any issue in defining π-reversible kernels as usual.*

F Locally balanced proposals and skew-symmetric distributions

In this section we show that the only balancing function g leading to a skew-symmetric distribution is $g(t) = t/(1+t)$. Following [Azzalini, 2013], a skew-symmetric distribution on $\mathbb{R}$ is distribution for which the probability density can be written

$$f(z) = 2f_0(z)G(z),$$

for any $z \in \mathbb{R}$, where $f_0(z) = f_0(-z)$, $G(z) \geq 0$ and

$$G(z) + G(-z) = 1. \tag{58}$$

In the first-order locally-balanced framework, if the current point is x then the proposal has density

$$f_x(z) = Z(x)^{-1}\mu_\sigma(z)g(e^{\nabla \log \pi(x)z}),$$

where, setting $t = e^{\nabla \log \pi(x)z}$ the balancing function g satisfies

$$g(t) = tg(1/t). \tag{59}$$

Equating (58) and (59) gives $G(z) = g(e^{\nabla \log \pi(x)z}) = g(t)$, implying that in this case

$$G(-z) = g(1/t).$$

Therefore, dividing (58) by $G(1/z)$, using the above and combining with (59) gives

$$t + 1 = \frac{1}{g(1/t)},$$

and combining with (59) gives

$$g(t) = \frac{t}{1+t}.$$

as required.

G Pre-conditioning the Barker proposal

The diagonal non-isotropic version of the Barker scheme (corresponding to using a diagonal preconditioning matrix) is a simple variation of Algorithm 2 from the paper and is described in Algorithm 3. The acceptance probability related to Algorithm 3 is exactly the same $\alpha^B(x, y)$ defined in (18).

Algorithm 3 Diagonal Barker proposal on $\mathbb{R}^d$

Require: current point $x \in \mathbb{R}^d$ and local scales $(\sigma_1, \ldots, \sigma_d) \in (0, \infty)^d$
Independently for each $i \in \{1, \ldots, d\}$ do:

1. Draw $z_i \sim \mu_{\sigma_i}$
2. Calculate $p_i(x, z_i) = 1/(1 + e^{-z_i \partial_i \log \pi(x)})$
3. Set $b_i(x, z_i) = 1$ with probability $p_i(x, z_i)$, and $b_i(x, z_i) = -1$ otherwise
4. Set $y_i = x_i + b_i(x, z_i) \times z_i$

Output: the resulting proposal y.

The general pre-conditioned version of the Barker algorithm is obtained by defining an appropriate linear transformation to the target variables x and then applying the standard Barker

algorithm (Algorithm 2 from the paper) in the transformed space. More precisely, given a target π and a covariance matrix Σ with Cholesky factor C, define the transformed variables $\tilde{x} = (C^T)^{-1}x$ with distribution $\tilde{\pi}(\tilde{x}) \propto \pi(C^T\tilde{x})$ and log-gradient $\nabla \log \tilde{\pi}(\tilde{x}) = \nabla \log \pi(C^T\tilde{x})C^T$. One then applies the standard (isotropic) Barker scheme described in Algorithm 2 to the pre-conditioned target $\tilde{\pi}$. As typically done with pre-conditioned MALA, the resulting preconditioned Barker scheme can be implemented without explicitly defining the auxiliary variables $\tilde{x}$ and transformed target $\tilde{\pi}$, but rather keeping the original target π and modifying the proposal distribution. The resulting pre-conditioned Barker proposal distribution and corresponding Metropolis-Hastings scheme are described in Algorithms 4 and 5, respectively.

Algorithm 4 Preconditioned Barker proposal on $\mathbb{R}^d$

Require: current point $x \in \mathbb{R}^d$ and preconditioning matrix $C = chol(\Sigma)$.

1. Draw $z_i \sim \mu$ independently for each $i \in \{1, \dots, d\}$
2. Calculate $p_i(x, z) = 1/(1 + e^{-z_i c_i(x)})$ where $c_i(x) = (\nabla \log \pi(x) \cdot C^T)_i$
3. For each i, set $\tilde{z}_i = z_i$ with probability $p_i(x, z)$, and $\tilde{z}_i = -z_i$ otherwise
4. Set $y = x + C^T\tilde{z}$ where $\tilde{z} = (\tilde{z}_1, \dots, \tilde{z}_d)$

Output: the resulting proposal y.

Algorithm 5 Metropolis–Hastings with preconditioned Barker proposal

Require: starting point for the chain $x^{(0)} \in \mathbb{R}^d$, and preconditioning matrix $C = chol(\Sigma)$. Set $t = 0$ and do the following:

1. Given $x^{(t)} = x$, draw y using Algorithm 4 and compute

$$\alpha^B(x, y) = \min\left(1, \frac{\pi(y)}{\pi(x)} \times \prod_{i=1}^{d} \frac{1 + e^{-z_i c_i(x)}}{1 + e^{z_i c_i(y)}}\right).$$

   where $z_i = ((C^T)^{-1}(y - x))_i$ and $c_i(x) = (\nabla \log \pi(x) \cdot C^T)_i$
2. Set $x^{(t+1)} = y$ with probability $\alpha^B(x, y)$, and $x^{(t+1)} = x$ otherwise
3. If $t + 1 < N$, set $t \leftarrow t + 1$ and return to step 1, otherwise stop.

Output: the Markov chain $\{x^{(0)}, \dots, x^{(N)}\}$.

H Additional simulation studies

In this section we provide various additional details on the simulation studies presented in the paper.

H.1 Additional example for Section 5.1

In Figure 7 we display a phenomenon analogous to Figure 2 on a 20-dimensional example in which each component of $\pi(\cdot)$ is an independent $N(0, \eta_i^2)$ random variable, with $\eta_1 = 0.01$ and $\eta_i = 1$ for $i = 2, ..., 20$. Here the performance of MALA starts deteriorating drastically as soon as the step-size exceeds the scale of the first component as we would expect from the theory developed in Section 2. On the other hand both the random walk and Barker schemes can

function adequately with larger than optimal step-sizes, and as a result achieve a much higher expected squared jump distance on all the other coordinates.

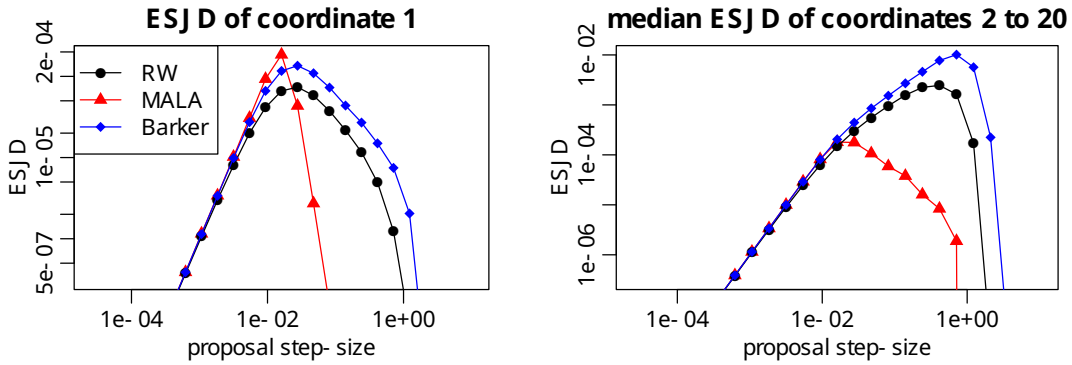

Figure 7: Expected squared jump distance (ESJD) against proposal step-size for RW, MALA and Barker on a 20-dimensional target in which one component has a smaller scale than all others.

H.2 Traceplots for Scenarios 2-4 from Section 6.2

Figure 4 displays the evolution of tuning parameters and MCMC trajectories when targeting the distribution described in Scenario 1 of that section. Here we provide analogous illustrations for Scenarios 2, 3 and 4. Figure 8 displays, for each scenario and each algorithm, the traceplot of the global scale $(\sigma_t)_{t\geq 1}$, the ones of the normalized local scales $(\Sigma_{t,ii}/\Sigma_{ii})_{t\geq 1}$ for $i \in \{1,\dots,100\}$ and the ones of the normalized Markov chains coordinates $(X_i^{(t)}/\Sigma_{ii}^{1/2})_{t\geq 1}$ for $i \in \{1,\dots,100\}$. Here Σ is the covariance of the target distribution and normalization is used to facilitate readability, so that all normalized local scales converge to 1 as $t \to \infty$ and all normalized coordinates have a $N(0,1)$ limiting distribution as $t \to \infty$. Overall, the traceplots for Scenarios 2-4 display a qualitatively similar behaviour to the ones of Scenario 1 in Figure 4. See Section 6.2 for more discussion.

H.3 Comparison to truncated or tamed gradients

Consider Metropolis–Hastings proposals of the form

$$y = x + \frac{\sigma^2}{2} G(x) + \sigma\xi\,,$$

for some $\sigma > 0$, $G : \mathbb{R}^d \to \mathbb{R}^d$ and $\xi \sim N(0, I)$. Setting $G(x) = \nabla \log \pi(x)$ leads to the MALA proposal. A common way to improve the stability of MALA in the literature is to truncate or tame the gradient $\nabla \log \pi(x)$. For example, in the truncated MALA algorithm (MALTA) [Atchade, 2006] we have

$$G(x) = \frac{\delta}{\max\{\delta, \|\nabla \log \pi(x)\|\}} \nabla \log \pi(x)\,,$$

for some $\delta > 0$, while in the component-wise tamed MALA (MALTAc) [Brosse et al., 2018, eq.(4)] the function $G(x) = (G_1(x), \dots, G_d(x))$ is defined component-wise as

$$G_i(x) = \frac{\partial_i(x)}{1 + \sigma^2 |\partial_i(x)|}\,.$$

The above taming is defined in such a way that $|G_i(x)|$ converges to σ^{-2} as $|\partial_i(x)| \to \infty$, meaning that in this case the upper bound for tamed gradients is authomatically chosen in a way that depends on σ.

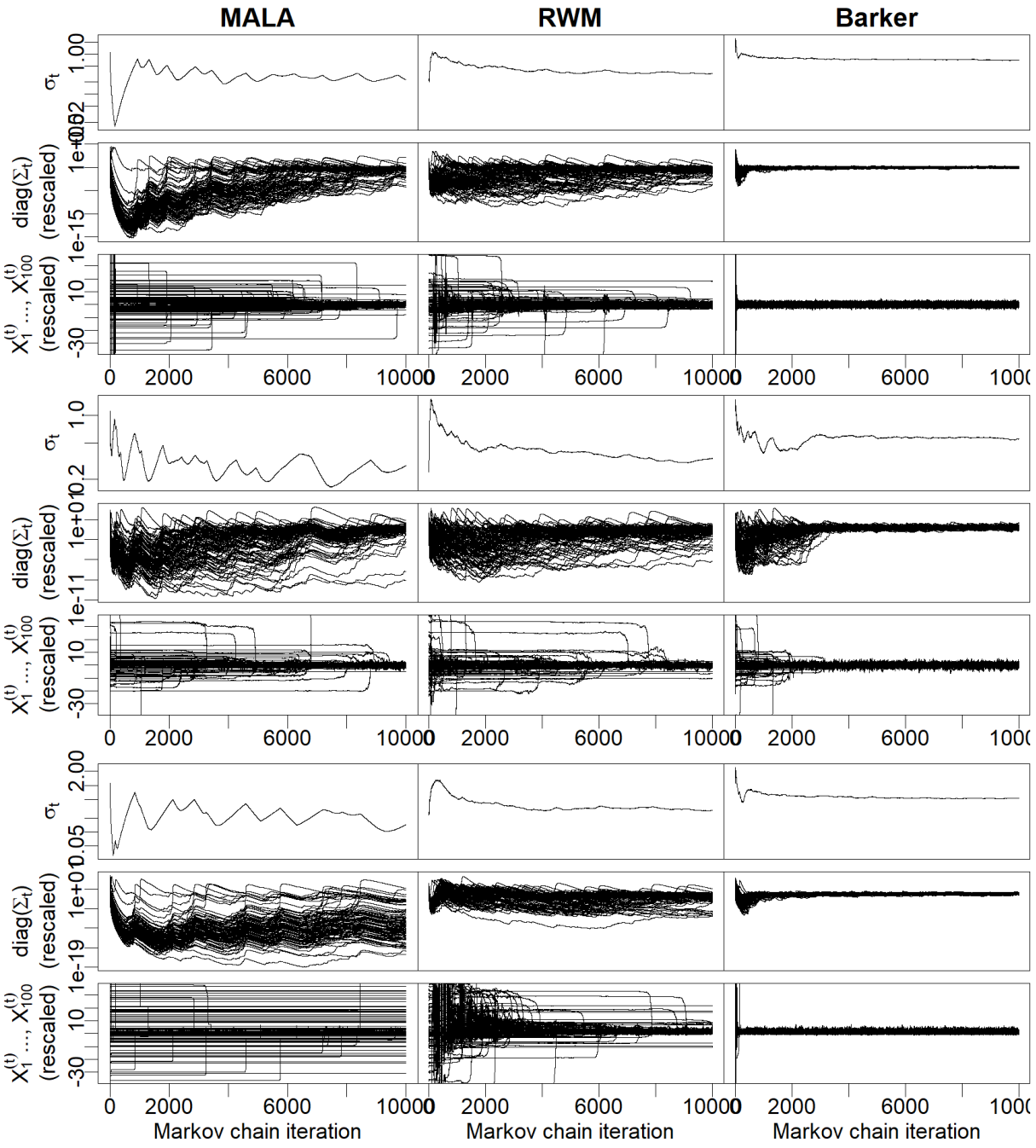

Figure 8: Same as Figure 4 for the target distributions of Scenario 2 (rows 1 to 3), Scenario 3 (rows 4 to 6) and Scenario 4 (rows 7 to 9). For each scenario, the first row displays the traceplot of the global scale σ_t; the second row the ones of the *normalized* local scales $\Sigma_{t,ii}/\Sigma_{ii}$ for $i = 1, \ldots, 100$; and the third row the ones of the *normalized* coordinates $X_i^{(t)}/\Sigma_{ii}^{1/2}$ for $i = 1, \ldots, 100$. See Section 6.2 for more details.

These schemes are effective in achieving geometric ergodicity also for light tails [Atchade, 2006]. They are less effective, however, in terms of being robust to tuning and they are very sensitive to the choice of truncation parameter (respectively δ and σ^{-2}). We illustrate this point in Figure 9. There we compare MALTA, MALTAc and Barker on targets being 100-dimensional Gaussian distributions with one component much smaller than the others, analogously to the first scenario of Section 6.2. For MALTA we set $\delta = 1000$, as is done for example, in Atchade [2006]. We also tried setting $\delta = 100$ without observing major differences. Rows 1-3 of Figure 9

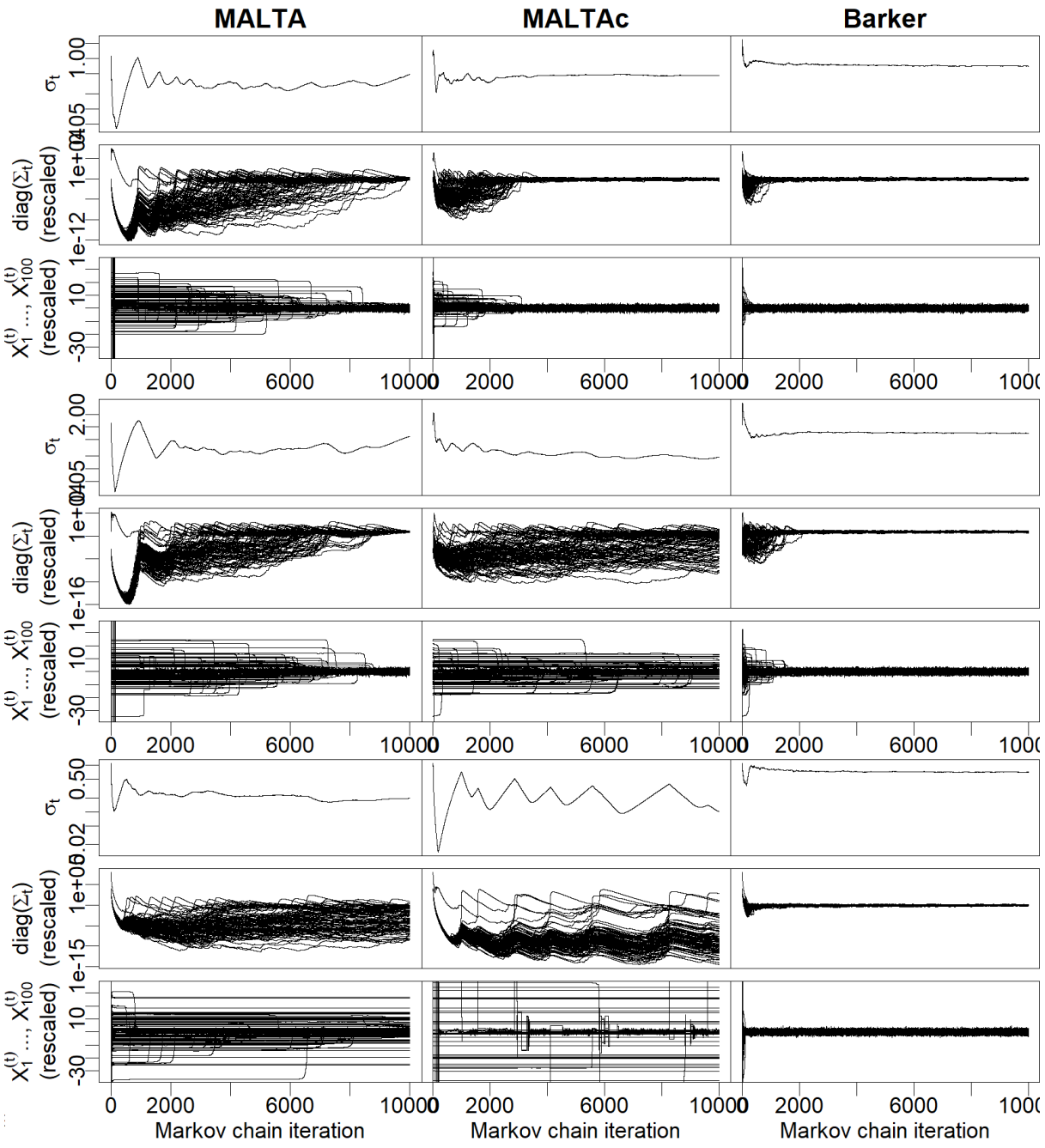

Figure 9: Comparison of MALTA, MALTAc and Barker on target distributions with one small component. Rows 1-3: same target considered in Figure 4 (100-dimensional Gaussian with first component standard deviation equal to 0.01 and all others standard deviations equal to 1). Rows 4-6 and rows 7-9: re-scaled versions where the scales of all coordinates are either multiplied or divided by 100. See Figure 8 for an explanation of each parameter plotted.

consider exactly the same target of Figure 4, which is a 100-dimensional Gaussian where the first component standard deviation is equal to 0.01 and all others standard deviations are equal to 1. In this case both MALTA and MALTAc improve over MALA, and in particular that MALTAc manages to converge to stationarity in around 4000 iterations, although this is still significantly slower than Barker (which only requires few hundred iterations). We then consider modifying the target distribution by either multiplying the scales of all coordinates by 100, resulting in the first component standard deviation equal to 1 and all others standard deviations equal to 100,

or dividing all scales by 100, resulting in the first component standard deviation equal to 10^{-4} and all others standard deviations equal to 10^{-2}. The results are reported in rows 4-6 and rows 7-9 of Figure 9, respectively. We observe a dramatic deterioration in the performances of both MALTA and MALTAc, while the performance of the Barker scheme are much less affected. The underlying reason is that MALTA and MALTAc are highly sensitive to the choice of truncation parameter (respectively δ and σ^{-2}), which needs to be tuned appropriately depending on the scales of the target distribution.

These illustrative simulations suggest that ad-hoc strategies to improve the robustness of gradient-based MCMC, such as truncating or taming gradients, are intrinsically more fragile and sensitive to heterogeneity and scales compared to a more principled solution such as the Barker algorithm, in which robustness arises naturally from the proposal mechanism. In addition to this, truncating and taming can be thought of as introducing a 'bias' into the proposal mechanism, in the sense that the resulting proposal is no longer first-order exact. Depending on how the truncation level δ is scaled with the dimensionality d, this can compromise the $d^{-1/3}$ scaling behaviour discussed in the paper.