Exploring the generalizability of the optimal 0.234 acceptance rate in random-walk Metropolis and parallel tempering algorithms

Aidan Li[1*], Liyan Wang[1], Tianye Dou[1], Jeffrey S. Rosenthal[1]

[1*]Department of Statistical Sciences, University of Toronto.

*Corresponding author. E-mail: aidan.li@mail.utoronto.ca;
Contributing authors: liyan.wang@mail.utoronto.ca;
tianye.dou@mail.utoronto.ca; jeff@math.toronto.edu;

Abstract

For random-walk Metropolis (RWM) and parallel tempering (PT) algorithms, an asymptotic acceptance rate of around 0.234 is known to be optimal in the high-dimensional limit. Yet, the practical relevance of this value is uncertain due to the restrictive conditions underlying its derivation. We synthesise previous theoretical advances in extending the 0.234 acceptance rate to more general settings, and demonstrate the applicability and generalizability of the 0.234 theory for practitioners with a comprehensive empirical simulation study on a variety of examples examining how acceptance rates affect Expected Squared Jumping Distance (ESJD). Our experiments show the optimality of the 0.234 acceptance rate for RWM is surprisingly robust even in lower dimensions across various proposal and multimodal target distributions which may or may not have an i.i.d. product density. Experiments on parallel tempering also show that the idealized 0.234 spacing of inverse temperatures may be approximately optimal for low dimensions and non i.i.d. product target densities, and that constructing an inverse temperature ladder with spacings given by a swap acceptance of 0.234 is a viable strategy. However, we observe the applicability of the 0.234 acceptance rate heuristic diminishes for both RWM and PT algorithms below a certain dimension which differs based on the target density, and that inhomogeneously scaled components in the target density further reduces its applicability in lower dimensions.

Keywords: Markov chain Monte Carlo, Metropolis algorithm, optimal scaling, acceptance rate, parallel tempering

1 Introduction

Markov chain Monte Carlo (MCMC) methods, such as the Metropolis algorithm, are used to draw samples from complex high-dimensional target probability distributions (Brooks et al., 2011; Robert and Casella, 2004). They enjoy strong theoretical asymptotic guarantees of accuracy, converging to the target distribution in stationarity (Brooks et al., 2011; Robert and Casella, 2004). However, they may also be inefficient and take an intractable amount of time to provide a good approximation of the target distribution they are sampling (Rosenthal, 1995). Therefore, the analysis of their running time is an important practical issue to consider.

Roberts et al. (1997) proved an important result: in the state space high-dimensional limit $d \to \infty$, the asymptotic acceptance rate of a Metropolis algorithm is central to its efficiency, and that for a target distribution that has density π with respect to Lebesgue measure on $\mathbb{R}^d$ where $\pi(\mathbf{x}) = \prod_{i=1}^{d} f(x_i)$ has an i.i.d. product form, and where f is a sufficiently smooth single-dimensional density function such that $\log f$ is continuously differentiable and $\mathbb{E}_f\left[\left(\frac{f'(X)}{f(X)}\right)^2\right] < \infty$, an asymptotic acceptance rate of approximately 0.234 maximises the efficiency of a random-walk Metropolis (RWM) algorithm with a Gaussian increment proposal distribution starting at stationarity. However, these assumptions are very restrictive. Many practical scenarios often involve finite and relatively small dimensions. Additionally, many probability distributions do not have densities of the i.i.d. product form, and for those that do, sampling a one-dimensional target is much simpler than using RWM. A burgeoning active research area is dedicated to showing that this figure still applies beyond these assumptions, and much of this focuses on the i.i.d. product assumption.

Many papers have shown that the optimal average acceptance rate of 0.234 for RWM holds greater generality than the assumptions used in Roberts et al. (1997). These extensions include: non-Gaussian proposals (Neal and Roberts, 2011), discrete hypercube target distributions (Roberts, 1998), Gibbs random fields (Breyer and Roberts, 2000), independent target components with inhomogeneous scaling (Bédard, 2007; Bédard and Rosenthal, 2008; Roberts and Rosenthal, 2001), partial updates where not all components are updated at once (Neal and Roberts, 2006), elliptical symmetric unimodal target densities (Sherlock and Roberts, 2009), and infinite-dimensional target distributions with non-trivial dependence structures (Mattingly et al., 2012). The 0.234 acceptance rate in optimal scaling has also been shown to apply to the spacing and swapping between inverse temperatures in the parallel tempering MCMC method (Atchadé et al., 2011; Roberts and Rosenthal, 2014). That being said, there are some cases in which the 0.234 case may not hold, such as in the case of discontinuous targets (Neal et al., 2012) where the optimal acceptance rate is approximately 0.1353. Additionally, these results of optimality are still theoretical ideals, with similar assumptions to the product i.i.d. form and in the limit of $d \to \infty$ where the Markov chain converges to a diffusion process.

These results are of great interest to practitioners who would like to understand how to tune their MCMC algorithms to maximise their efficiency. Previous simulation studies have been carried out on more general target distributions like products of independent, non-identical components (Bédard, 2008; Roberts and Rosenthal, 2001),

and for the Markov modulated Poisson process (Sherlock et al., 2010), but there is still a lack of thorough experimentation and guidance for how handy the 0.234 rule is on realistic target distributions that may have lower or higher dimensionality and/or multimodality. Our research examines how necessary these assumptions are for the theoretical result to remain relevant: where can we relax some assumptions in lower dimensions and still have the optimal acceptance rate of approximately 0.234, and where can we not do this? And in the end, how applicable is the 0.234 figure for the practitioner, who may want to use MCMC for complicated target distributions beyond the necessary assumptions in finite dimensions? We thoroughly dissect various aspects of the Metropolis algorithm and experiment with them, and in doing so, we show empirically where the theoretical ideal value can still align with more realistic scenarios.

We begin our paper with a description of the RWM and its optimal scaling framework in Section 2. In Section 3, we describe RWM experiments in low dimensions that show where an acceptance rate of 0.234 can be optimal in the sense that it maximises the Expected Squared Jumping Distance (ESJD) for various proposal and target densities which are not necessarily all i.i.d. product forms. Next, we describe the parallel tempering method and its own optimal scaling framework in Section 4, and show using experiments in Section 5 that the 0.234 swap acceptance rate in parallel tempering may also be optimal in lower dimensions for the multivariate Gaussian target density and a Gaussian mixture target density which is not of an i.i.d. product form. Lastly, we discuss the implications of our findings in Section 6.

2 Random-walk Metropolis Background

We first introduce the Metropolis algorithm and the Optimal Scaling framework. We then discuss measures to evaluate the efficiency of MCMC algorithms, such as the expected squared jumping distance, and explain how the asymptotic acceptance rate of a MCMC algorithm is tied to the algorithm's efficiency.

2.1 Metropolis Algorithm

A Metropolis algorithm (Metropolis et al., 1953) constructs a Markov chain $\mathbf{x}^{(0)}, \mathbf{x}^{(1)}, \ldots$ that converges to the target probability distribution π in the long-term after many runs, i.e. the Markov chain has a stationary distribution equal to the target distribution. When taking each step in the Markov chain, we generate a new state $\mathbf{y} = \mathbf{x}^{(t)} + \epsilon$ where ϵ is sampled from an *increment* "proposal" density Q having $Q(\epsilon) = Q(-\epsilon)$ for all ϵ in its domain and accept this new state with probability $\alpha(\mathbf{x}, \mathbf{y})$ given by $\alpha = \min\left\{1, \frac{\pi(\mathbf{y})}{\pi(\mathbf{x})}\right\}$. If the proposed value is accepted, we set the next state $\mathbf{x}^{(t+1)} = \mathbf{y}$. Otherwise, we set $\mathbf{x}^{(t+1)} = \mathbf{x}^{(t)}$.

2.2 Optimal Scaling Framework

Optimal Scaling (Roberts et al., 1997; Roberts and Rosenthal, 2001) is one of the most successful frameworks for performing asymptotic analysis of high-dimensional MCMC

methods, and provides mathematically-grounded guidance on how to best optimise MCMC performance by tuning the "scaling" parameter(s) of the proposal distribution. Typically, a multivariate Gaussian distribution with mean 0 is used as the increment proposal distribution, and the proposal's scaling is its variance σ^2. If the proposal's variance is too small, too many proposed steps are accepted by the algorithm, but each step does not explore the state space much. Vice versa, if the proposal's variance is too large, too few proposed steps are accepted, leading to slow exploration of the state space since the algorithm stays at a state for too long. Therefore, the proposed variance is crucial to the performance of the algorithm; the optimal scaling should balance the exploration of new areas in the state space with exploiting high-density areas of the target distribution. Different Metropolis algorithms have shown to have different optimal choices of the proposal distribution variance σ^2.

We next discuss how to evaluate the efficiency of an MCMC algorithm for determining optimal scaling. A very popular measure of efficiency is the *Expected Squared Jumping Distance* (ESJD) metric (Gelman and Pasarica, 2007; Yang et al., 2020) which we use for our experiments. The expected squared jumping distance measures how far, in expectation, the MCMC chain moves in a single iteration. For the standard random-walk Metropolis algorithm, we define this as

$$\mathbb{E}\left[\left\|x^{(t+1)} - x^{(t)}\right\|^2\right] \approx \frac{1}{n-1}\sum_{i=1}^{n-1}\left\|x^{(t+1)} - x^{(t)}\right\|^2$$

where n is the total number of iterations of the algorithm. Maximising the ESJD aligns with minimizing the first-order auto-correlation of the Markov chain and subsequently maximises efficiency if the higher-order auto-correlations are monotonically increasing relative to the first-order auto-correlation (Gelman and Pasarica, 2007; Yang et al., 2020).

There are other notions of efficiency of a Markov chain, but in the high-dimensional limit $d \to \infty$ where the chain converges to a diffusion process, all efficiency measures are effectively equivalent (Roberts et al., 1997; Roberts and Rosenthal, 2001). A key result of these referenced papers is that, given a Metropolis algorithm with a Gaussian increment proposal distribution $Q = N(0, \frac{\ell^2}{d} I_d)$ where $\ell > 0$ is a fixed scaling constant and I_d is the identity matrix, maximising the algorithm's speed measure $h(\ell)$, which is a function of the scaling constant ℓ, yields the most efficient asymptotic diffusion. Furthermore, the speed measure has a clear relation to a much simpler quantity to estimate: the *asymptotic acceptance rate* of the proposed new states (moves) of the algorithm, defined as

$$a = \lim_{n\to\infty} \frac{\#\text{ accepted moves}}{n}.$$

Both the speed measure $h(\ell)$ and asymptotic acceptance rate $a(\ell)$ are functions of the scaling constant ℓ. As mentioned in Section 1, when the target distribution that has density π with respect to Lebesgue measure on $\mathbb{R}^d$ has an i.i.d. product form $\pi(\mathbf{x}) = \prod_{i=1}^{d} f(x_i)$, where f is a sufficiently smooth single-dimensional density function

such that $\log f$ is continuously differentiable and $\mathbb{E}_f\left[\left(\frac{f'(X)}{f(X)}\right)^2\right] < \infty$, for a random-walk Metropolis algorithm with a Gaussian increment proposal distribution starting at stationarity, the scaling constant ℓ that maximises the speed measure $h(\ell)$ corresponds to an asymptotic acceptance rate $a(\ell)$ of approximately 0.234 (Roberts et al., 1997; Roberts and Rosenthal, 2001). Hence, an asymptotic acceptance rate of approximately 0.234 should necessarily optimise a measure of efficiency such as the ESJD of the algorithm.

3 Random Walk Metropolis Simulations

In this section, we explore the applicability of the 0.234 acceptance rate figure for RWM algorithms using several practical RWM examples in lower dimensions. We begin with simple examples where the target densities meet all the key assumptions required for the 0.234 acceptance rate to be optimal, as established in the literature. Next, we investigate examples that do not use Gaussian proposal distributions, and then we explore other densities that may lack key assumptions, such as smoothness or the i.i.d. product form. We estimate the Expected Squared Jumping Distance (ESJD) of the algorithm as a metric of the algorithm's efficiency as mentioned in Section 2.2 and aim to see under what conditions an acceptance rate of approximately 0.234 maximises the ESJD.

Our experiments show which acceptance rate value (which is implicitly determined by the proposal variance) maximises the ESJD for these various target distributions in lower dimensions. It is important to note that functions of acceptance rates are somewhat flat around the optimal values and that the values and figures reported are still prone to random error inherent in a Monte Carlo simulation. To reduce the random error, we run each experiment over 5 different seeds and average the results. In each experiment, we run RWM simulations with 40 different variance values using a Gaussian proposal (unless otherwise mentioned) for 100,000 runs each and recorded the ESJD and acceptance rate estimates for the simulations. For each example, we conduct this experiment over multiple dimension sizes. Overall, each example has 200 simulations total over 5 seeds, and each simulation runs for 100,000 iterations.

3.1 Simpler Examples: Gamma and Beta i.i.d. Targets

We begin with simpler examples demonstrating that the 0.234 acceptance rate is still roughly optimal for lower dimensions with any i.i.d. product target distribution $\pi(\mathbf{x}) = \prod_{i=1}^{d} f(x_i)$ and a Gaussian proposal. As mentioned in Section 2, Roberts et al. (1997) proved the 0.234 optimal acceptance rate for i.i.d. product target distributions with a standard Gaussian proposal distribution in the infinite-dimensional limit, and Roberts and Rosenthal (2001) and Bédard (2008) extended this with MCMC simulations showing that with a i.i.d. product target distribution and Gaussian proposal, the 0.234 acceptance rate seems approximately optimal for dimensions as low as 10. Our experiment provides further evidence for this being the case, and we extend this with simulations for $d = 2, 5$.

We present two examples: the first example sets the singular dimensional component density f to be $Gamma(3, 2)$ with shape parameter 3 and scale parameter 2, and the second example uses $Beta(3, 2)$ as the component density f with shape parameters 3 and 2. We use a multivariate Gaussian with a mean of 0 as the increment proposal distribution in both cases. We run this experiment in dimensions 2, 5, 10, 30, and 50.

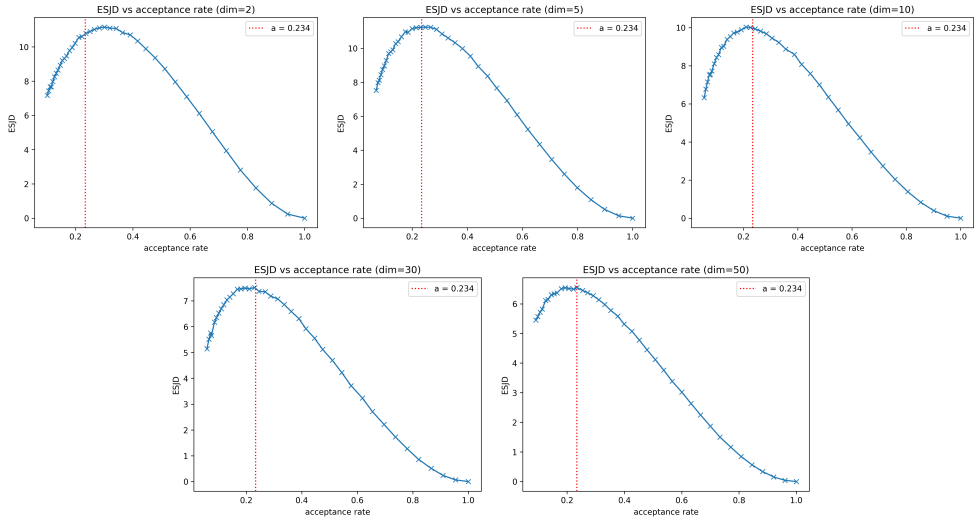

Fig. 1: ESJD for various RWM acceptance rates for the i.i.d. gamma target distribution in dimensions $d = 2$, 5, 10, 30, 50 (from top-left to bottom-right).

Table 1: Experimental optimal RWM acceptance rates for i.i.d. gamma and beta target distributions in different dimensions. Numerical values are the observed experimental acceptance rates maximising the ESJD.

	Dimension d				
Target Density	2	5	10	30	50
i.i.d. $Gamma(3, 2)$	0.3009	0.2483	0.2095	0.2239	0.2338
i.i.d. $Beta(3, 2)$	0.3780	0.2868	0.2384	0.2364	0.2353

As we can see in Figures 1, 2, and Table 1, a 0.234 acceptance rate is still quite efficient in dimensions as low as 5. For an i.i.d. gamma target distribution, there is a significant decrease in the optimal acceptance rate when the dimension d increases from 5 to 10. Subsequently, the optimal acceptance rate approaches 0.234 as the dimension increases. In the case of an i.i.d. beta target distribution, the acceptance rate is already close to 0.234 at $d = 10$, and as the dimension increases further, the optimal acceptance rate observed in experiments gets closer to 0.234.

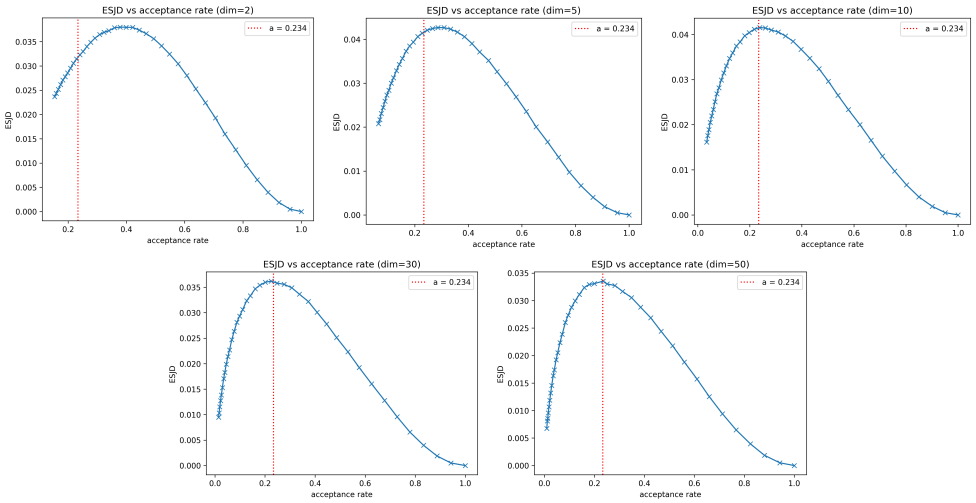

Fig. 2: ESJD for various RWM acceptance rates for the i.i.d. beta target distribution in dimensions $d = 2, 5, 10, 30, 50$ (from top-left to bottom-right).

Our simulation study provides further evidence that for i.i.d. product target distributions, the optimal acceptance rate for maximizing ESJD is approximately 0.234 even in low dimensions, and that the optimal acceptance rate gets closer to 0.234 as the dimension increases. This result holds for both i.i.d. gamma and beta target distributions, as evidenced by the trace plots and histograms demonstrating the efficacy of the Metropolis algorithm across different dimensions.

3.2 Non-Gaussian Proposal Densities

Next, we investigate the efficacy of different proposal distributions for the Metropolis algorithm. One of the conditions assumed by the 0.234 theorem of Roberts et al. (1997) is the use of a multivariate Gaussian increment proposal distribution $Q(x) = N(0, \sigma^2 I_d)$. The additional results in Roberts and Rosenthal (2001) depend on the assumptions of light-tailed proposals. In contrast, Jarner and Roberts (2002) extend this analysis by examining the behaviour of the algorithms with heavy-tailed proposals. They find that, unlike the diffusion behaviour seen with light-tailed proposals, heavy-tailed proposals lead to a different dynamic characterized by abrupt movements followed by periods of inactivity. Furthermore, Neal and Roberts (2011) investigated optimal scaling results for non-Gaussian proposals; specifically, they investigated heavy-tailed proposal distributions such as the Cauchy distribution, and spherical proposal distributions of fixed radius. In their RWM simulation study with three different continuous i.i.d. product target densities, both the Gaussian and Cauchy proposals had an estimated asymptotically optimal acceptance rate of approximately 0.234 in $d = 100$. They also found that a spherical proposal with a fixed radius of 2.38 achieved an optimal acceptance rate of about 0.234 in $d = 2$ and $d \geq 10$.

We extend this work with two different proposal distributions in $d = 2, 5, 10, 20, 50$. The first proposal distribution we consider is the multivariate Laplace distribution, which may also be referred to as the double exponential distribution. We use location parameter $\mu = 0$ and covariance parameter $\Sigma = \sigma^2 I_d$, with σ to be chosen. We also consider an example with the uniform proposal distribution within the closed interval $\left[x_i - \frac{b}{2}, x_i + \frac{b}{2}\right]$ for each component i, where $b > 0$ is the interval length. We use the standard multivariate Gaussian as the target density π.

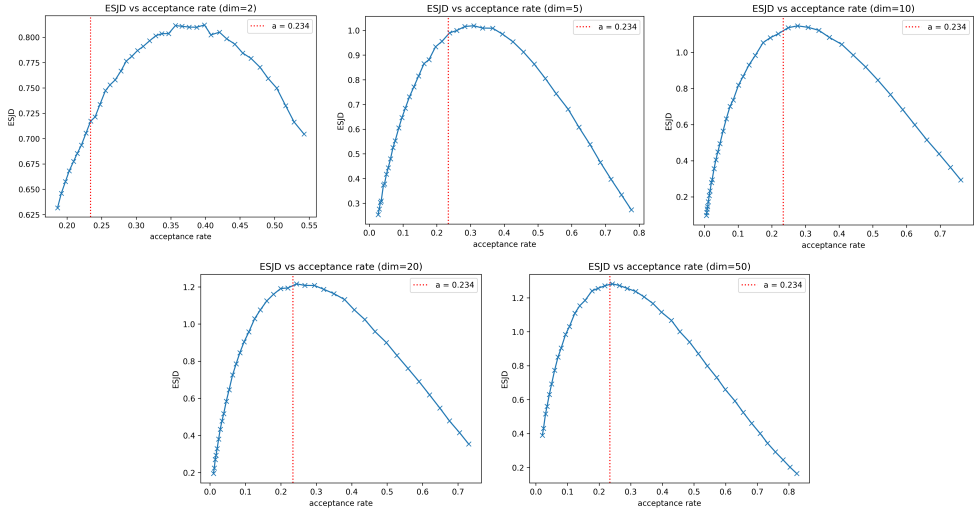

Fig. 3: ESJD as a function of acceptance rate for the Laplace proposal distribution, with dimensions $d = 2, 5, 10, 20, 50$ (from top-left to bottom-right).

Table 2: Experimental optimal RWM acceptance rates for Laplace and Uniform proposal distributions in different dimensions. Numerical values are the observed experimental acceptance rates maximising the ESJD.

	Dimension d				
Proposal Distribution[1]	2	5	10	20	50
Laplace	0.3986	0.3097	0.2701	0.2449	0.2311
Uniform	0.3204	0.2791	0.2623	0.2435	0.2392

[1]Parameters for each proposal distribution are given in the experiment description above.

As observed in Figures 3, 4 and Table 2, an acceptance rate of approximately 0.234 remains efficient and maximizes ESJD even in lower dimensions ($d = 10$) under both Laplace and uniform proposal distributions, and that the optimal acceptance rate tends to converge to 0.234 as the dimensionality of the problem increases. Specifically, for a multivariate i.i.d. Gaussian target distribution, an acceptance rate close to

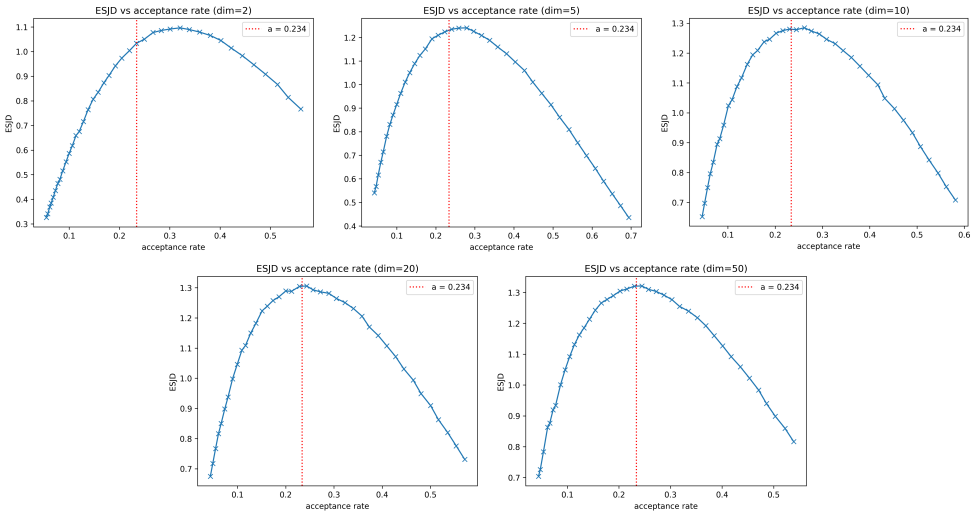

Fig. 4: ESJD as a function of acceptance rate for the Uniform proposal distribution, with dimensions $d = 2, 5, 10, 20, 50$ (from top-left to bottom-right).

0.234 maximizes ESJD. This finding is consistent across various proposal distributions, including the Laplace, Uniform, and Gaussian.

3.3 Non-smooth Target Densities: Continuous Hypercube

Consider a RWM using a Gaussian proposal and a target density still of the product form $\pi(\mathbf{x}) = \prod_{i=1}^{d} f(x_i)$, but where the individual component density f is Uniform(a, b). Unlike the previous examples, the component density f and its logarithm are not continuously differentiable on $\mathbb{R}$ due to the discontinuities at the boundaries $x_i = a$ and $x_i = b$. Since this is one of the assumptions in the proof, the asymptotic results where the optimal acceptance rate is 0.234 may not apply. This discontinuous target density case has been explored by Neal et al. (2012), which found that for product densities with a discontinuity at the boundary, such as the unit hypercube $[0, 1]^d$, the efficiency of the RWM is maximised with an asymptotic optimal acceptance rate of 0.1353 as $d \to \infty$. We provide simulation studies in various dimensions to investigate the applicability of this particular 0.1353 acceptance rate figure in lower dimensions on the hypercube $[0, 1]^d$.

Table 3: Hypercube target experimental optimal RWM acceptance rates for different dimensions.

Dimension d	2	5	10	20	30	50	100
Optimal Acceptance Rate	0.4359	0.2578	0.2095	0.1561	0.1507	0.1398	0.1331

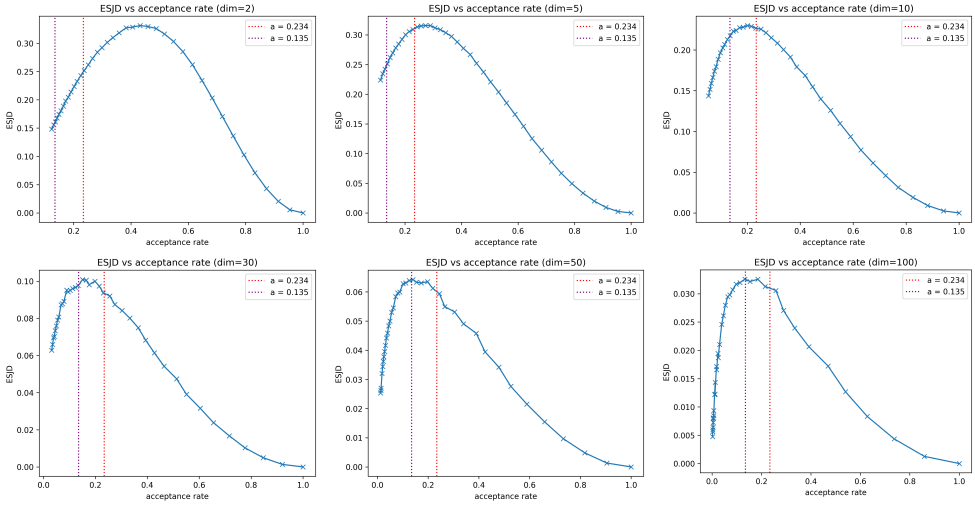

Fig. 5: ESJD for various RWM acceptance rates for the continuous hypercube target distribution in dimensions $d = 2, 5, 10, 30, 50, 100$ (from top-left to bottom-right).

We have a fairly similar ESJD vs acceptance rate curve in all dimensions as shown by Figure 5. Still, unlike the previous examples, which seem to be fairly consistently optimal with an acceptance rate of 0.234, we find (refer to Table 3) that the optimal acceptance rate for the hypercube target changes significantly as the dimension increases. This is because discontinuities introduce additional complexity in the target distribution, affecting the behaviour and performance of the RWM. As the dimension d increases, the algorithm's performance converges to the theoretical 0.1353 value. In this specific case, the shrinking optimal acceptance rate might be because of the unique zero or non-zero form of the target density beyond the hypercube boundary. As discussed by Neal et al. (2012), whether a proposed move is accepted depends completely on whether the proposed state is inside the hypercube. Subsequently, when using a Gaussian proposal, the probability that at least one proposed component will lie outside the boundary increases, leading to a higher probability of instant rejection.

3.4 Multimodal Target Densities

In this subsection, we investigate target distributions with multimodal densities. Many real-world probability and data distributions are multimodal, and standard RWM algorithms may struggle to explore all the modes of a multimodal distribution in a reasonable amount of time particularly when the modes are far apart. Additionally, multimodal densities may not necessarily fall under the i.i.d. product umbrella of target densities, and even if they do, it is debatable how relevant the 0.234 rule is in these cases. For example, a RWM algorithm may achieve a 0.234 acceptance rate in its run over many iterations and yet completely fail to escape a mode to explore other modes. We illustrate this with Figure 6, where we attempt to draw 10 million samples from a multimodal distribution with three modes but fail to escape the central mode.

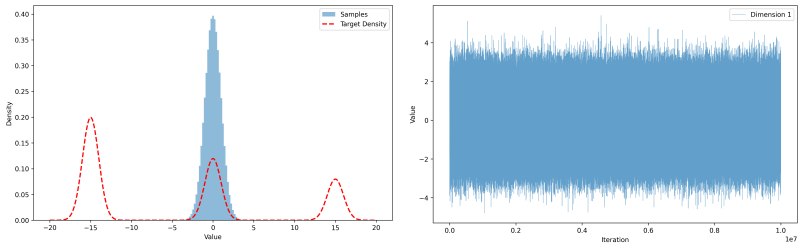

Fig. 6: Histogram and traceplot of 10 million samples drawn from a multimodal single-dimensional target density. The true density values are traced with the dotted lines.

When we have widely-separated modes, a RWM with the Gaussian proposal distribution struggles to reach the other modes in finite time regardless of the proposal scaling and the dimension of the target distribution. Therefore, for these multimodal target density experiments, we deliberately set the modes to be close enough to each other so that a RWM still mixes between the modes reasonably often.

3.4.1 The "Rough Carpet" Target Density

We explore two multimodal target densities in this section: first, we examine a simpler i.i.d. product density, and then one with inhomogeneous scaling factors. Our first multimodal distribution has the form $\pi = \prod_{i=1}^{n} f(x_i)$, where the single component density f is a one-dimensional density with three modes. We define our single-component density:

$$f(x) = 0.5N(x|\mu_1, 1) + 0.3N(x|\mu_2, 1) + 0.2N(x|\mu_3, 1) \tag{1}$$

where $\mu_1, \mu_2, \mu_3 \in \mathbb{R}$ and $N(x|\mu_i, 1)$ indicates the density of the univariate Gaussian distribution at point x with mean μ_i and variance 1. This forms a "rough carpet"-like appearance for the target distribution when we take the product of this single-component density over many dimensions (see the visualization of the first component in Figure 7). Since we have 3^d modes, the density values at each mode may not be very high. Yet, since this is a i.i.d. product form, the theoretical result of 0.234 being the optimal acceptance rate should also apply.

We conduct various experiments in dimensions $d = 20, 30, 50$ where we use the single-component density $f(x) = 0.5N(x \mid -5, 1) + 0.3N(x \mid 0, 1) + 0.2N(x \mid 5, 1)$. As mentioned (refer to Figure 6), we set the modes to be reasonably close so that the standard RWM can jump between modes somewhat frequently.

Table 4: Rough carpet product target distribution experimental optimal RWM acceptance rates for different dimensions.

Dimension d	20	30	50
Optimal Acceptance Rate	0.2133	0.2202	0.2291

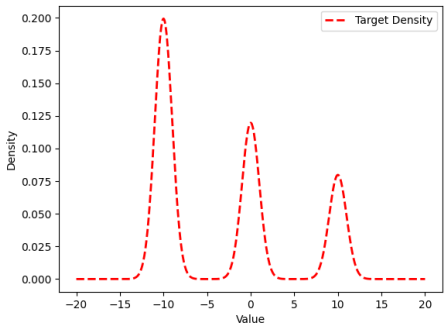

Fig. 7: Single-dimensional component density for the "rough carpet" target distribution.

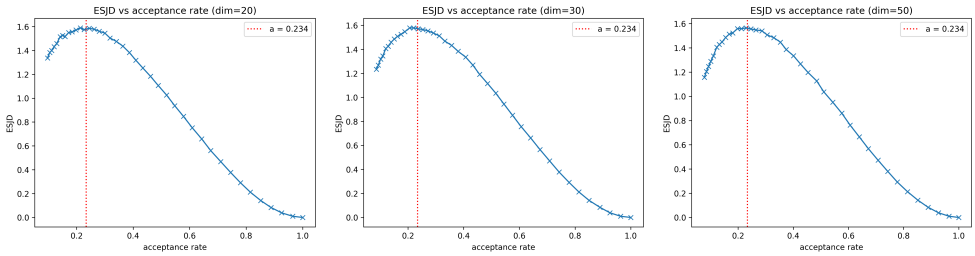

Fig. 8: ESJD for various RWM acceptance rates for the rough carpet product target distribution in dimensions $d = 20, 30, 50$ (from left to right).

We report our results for the non-scaled target distribution (Section 3.4.1 point 1) in Figure 8 and Table 4. We find that the exact optimal acceptance rate that maximises ESJD is slightly below 0.234 in low dimensions, but that 0.234 is still quite applicable, and that the empirically optimal acceptance rate gets closer to 0.234 with increasing dimensionality. This seems consistent with Roberts et al. (1997) since our target density, despite being multimodal, is a product of a single-dimensional density in the d components.

3.4.2 The "Rough Carpet" Target Density: Inhomogeneously Scaled Components

Next, we examine a multimodal distribution with a density of the form $\pi = \prod_{i=1}^{n} C_i f(C_i x_i)$. This density is a more general form of the previous one by including inhomogeneous component-wise scaling factors C_i, where $C_i > 0$ and $\mathbb{E}(C_i) = 1$ and $Var(C_i) < \infty$. This inhomogeneously-scaled i.i.d. product target density was discussed by Roberts and Rosenthal (2001), which proved the extension of the 0.234 asymptotically optimal acceptance rate to this case. We use the same single-component density f as the previous experiment in Section 3.4.1, independently sample the component

scaling factors $C_i \sim \text{Uniform}[0, 2]$ and report our results for the inhomogeneously scaled target distribution in Figure 9 and Table 5.

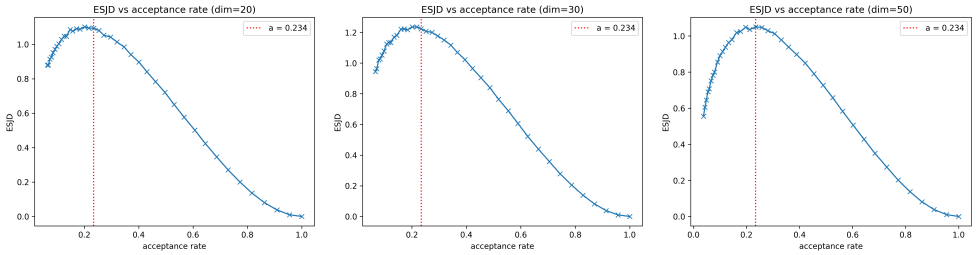

Fig. 9: ESJD for various RWM acceptance rates for the inhomogenously scaled rough carpet target distribution in dimensions $d = 20$, 30, 50 (from left to right).

Table 5: Inhomogenously scaled rough carpet product target distribution experimental optimal RWM acceptance rates for different dimensions.

Dimension d	20	30	50
Optimal Acceptance Rate	0.2009	0.2174	0.2351

Yet again, we find that the optimal acceptance rate converges to about 0.234 with increasing dimensionality. While the lower-dimensional cases such as $d = 20, 30$ have a larger degree of deviation from the 0.234 figure than the previous homogeneously scaled case, in $d = 50$ we find that the figure is very close to 0.234. A possible explanation for this is that the dimensions $d = 20, 30$ might be too low for the law of large numbers to apply for the inhomogeneous scaling factors; i.e. the scaling factors C_i sampled from Uniform$[0, 2]$ may not have a sample mean close enough to 1 in low dimensions, but in $d = 50$ or higher the sample mean is much closer to 1.

3.4.3 A Tale of Three Mixtures

The above "rough carpet" distributions are multimodal, but notice that they still follow the general product form $\pi = \prod_{i=1}^{n} f(x_i)$. Here, we examine a distribution that does not have this product assumption at all; instead, we have a weighted sum of densities as our target density. In this section, we examine a mixture of three Gaussians; a multimodal distribution with just three modes regardless of the number of dimensions. The general form of this density is

$$\pi(x) = w_1 N(x|\mu_1, \Sigma_1) + w_2 N(x|\mu_2, \Sigma_2) + w_3 N(x|\mu_3, \Sigma_3) \tag{2}$$

where we have non-negative scalar weights $w_i \geq 0$ such that $w_1 + w_2 + w_3 = 1$, and N is a multivariate Gaussian with mean $\mu_i \in \mathbb{R}^d$ and covariance matrices Σ_i.

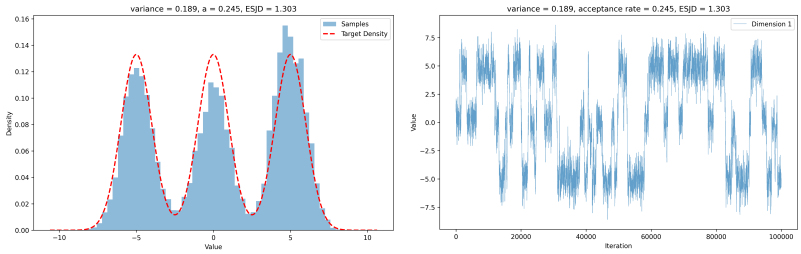

Fig. 10: A histogram and traceplot of the first component from one of the simulations in the three mixture target distribution experiment after 100,000 iterations.

For these experiments in $d = 20, 30, 50$, we set the weights $w_1 = w_2 = w_3 = \frac{1}{3}$, the covariances $\Sigma_1 = \Sigma_2 = \Sigma_3 = I_d$, and set the means $\mu_1 = (\epsilon, 0, 0, \ldots, 0), \mu_2 = (0, 0, \ldots, 0), \mu_3 = (-\epsilon, 0, 0, \ldots, 0)$ for some constant $\epsilon \in \mathbb{R}$, i.e. just varying the first component of the means and setting the other components of the means to 0. As mentioned (refer to Figure 6), we choose the constant ϵ in the first dimension to be some value such that jumping between modes still happens frequently enough; for the remainder of this section we use $\epsilon = 5.0$. We refer the reader to Figure 10 for an example histogram and traceplot to illustrate how the Markov chain explores the three modes.

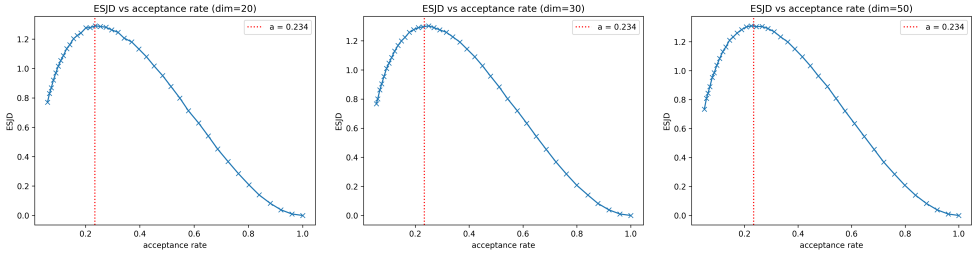

Fig. 11: ESJD for various RWM acceptance rates for the three mixture distribution in dimensions $d = 20$, 30, 50 (from left to right).

Table 6: Three mixture target distribution experimental optimal RWM acceptance rates for different dimensions.

Dimension d	20	30	50
Optimal Acceptance Rate	0.2358	0.2497	0.2250

We find the ESJD vs acceptance rate trends look almost identical in dimensions $d = 20, 30, 50$, and ESJD continues to be maximised at an acceptance rate of around

0.234 (see Figure 11 and Table 6). Clearly, even though the i.i.d. assumption does not hold, the 0.234 figure still seems to be quite applicable here.

3.4.4 A Tale of Three Mixtures: Inhomogenous Scaling Factors

We repeat this experiment with inhomogeneous scaling factors inside the covariance matrices Σ_i for each mode. Concretely, each mode has covariance matrix $\Sigma_i = diag(C_1, C_2, \ldots, C_d)$ where C_i are i.i.d. samples from the Uniform$[0, 2]$ distribution such that $\mathbb{E}[C_i] = 1$ and $\mathbb{E}[C_i^2] < \infty$. For this experiment, we use the same diagonal covariance matrix for each of the three modes. We report our results in Figure 12 and Table 7. We observe a very similar phenomenon as the inhomogeneously-scaled rough carpet density: in lower dimensions, the optimal acceptance rate is significantly lower than 0.234, but as d increases, we get much closer to 0.234.

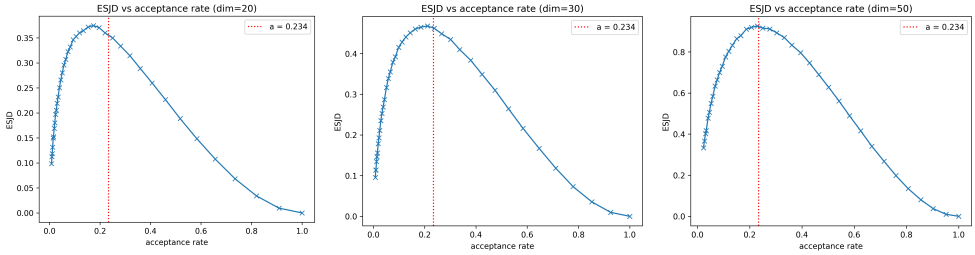

Fig. 12: ESJD for various RWM acceptance rates for the three mixture distribution with inhomogeneous scaling in each component in dimensions $d = 20, 30, 50$ (from left to right).

Table 7: Inhomogenously scaled three mixture target distribution experimental optimal RWM acceptance rates for different dimensions.

Dimension d	20	30	50
Optimal Acceptance Rate	0.1740	0.2107	0.2290

Overall, the three-mixture examples and their results yield promising avenues for extension of the theory, going beyond the i.i.d. product assumption to a target density that is a weighted sum of individual Gaussian densities. Furthermore, the 0.234 figure seems to be relevant even for inhomogeneous scaling in each Gaussian in a Gaussian mixture.

4 Parallel Tempering Background

This section offers a brief introduction to the parallel tempering method and its own optimal scaling framework that we conduct experiments on in Section 5. Many of the

terms in Section 2 are re-used but have slightly different definitions for the parallel tempering context.

4.1 Parallel Tempering Method

The parallel tempering method (Geyer, 1991), also known as the Metropolis-coupled Markov chain Monte Carlo algorithm or the replica exchange method, is a version of the Metropolis algorithm that specialises in sampling from multimodal distributions. In this case, we sample from a target distribution $\pi(\cdot)$ with density $f_d(\mathbf{x})$ on a state space $\mathcal{X}$. We define a sequence of tempered target densities $f_d^{\beta_j}(\mathbf{x})$ where $0 \leq \beta_n < \beta_{n-1} < \cdots < \beta_1 < \beta_0 = 1$ are the inverse temperature values. We require that $f_d^{\beta_0}(\mathbf{x}) = f_d(\mathbf{x})$ (that is, that $\beta_0 = 1$). Typically, the tempered target densities are simply powers of the original density, and we assume this for all parallel tempering experiments. Then, we run a Markov chain for each of the $n+1$ values of β, each with its own tempered target density. Eventually, the $\beta_0 = 1$ chain, also called the "cold" chain, with its stationary density $f_d^{\beta_0}(\mathbf{x}_0)$ should correspond to the original target density of interest $f_d(\mathbf{x})$, and the idea is that using information from other chains with higher temperatures (lower β) will speed up the mixing in the cold chain.

In each iteration of the algorithm, we alternate between *within temperature* and *temperature swap* moves. The former indicates that each chain takes a standard RWM update within its respective $f_d^{\beta_j}$. The latter indicates that for two chains with β_j, β_k inverse temperature values that are adjacent in the sequence of β values, switch the chain values of $\mathbf{x}_j$ and $\mathbf{x}_k$ with probability

$$\min\left\{1, \frac{f_d^{\beta_j}(\mathbf{x}_k) f_d^{\beta_k}(\mathbf{x}_j)}{f_d^{\beta_j}(\mathbf{x}_j) f_d^{\beta_k}(\mathbf{x}_k)}\right\} \tag{3}$$

4.1.1 Optimal Scaling for Parallel Tempering

Since the hot chains can explore the state space more quickly, we would like to maximise how frequently we can swap values from the hottest chain to the coldest chain so that the cold chain can mix faster and escape local modes. To do this, we maximise the effective speed with which the chain values move along in the inverse temperature domain. The spacings of the inverse temperatures β_i's are crucial to this efficiency. If β_j and β_k are too far apart, we usually reject these swaps described by (3), but if they are too near, the swaps will not improve mixing. We would like to swap just the right amount to maximise mixing in the temperature domain.

The *expected squared jumping distance* (ESJD) for parallel tempering thus refers to the expected squared jump in inverse temperatures. Formalising this, when we attempt to swap the chain values between the inverse temperatures β and $\gamma := \beta + \epsilon$ where $\beta, \epsilon > 0$ and $\beta, \gamma \leq 1$, the swap is either accepted, in which case the values move a squared distance of $(\gamma - \beta)^2 = \epsilon$, or the swap is rejected, in which case the distance moved is 0. This leads to a very natural definition for ESJD, which is

$$ESJD = \mathbb{E}[(\gamma - \beta)^2] = \epsilon^2 \times \mathbb{E}[\Pr(\text{swap accepted})]. \tag{4}$$

Maximising the asymptotic ESJD effectively maximises the efficiency of the attempted swap moves in providing mixing in the temperature space, or in other words, maximises the speed with which the chain values move in the inverse temperature space.

Under the restrictions of an i.i.d. product target density $f_d(\mathbf{x}) = \prod_{i=1}^{d} f(x_i)$, where the tempered distributions are powers of the original density, and the inverse-temperature spread $\epsilon = d^{-1/2}\ell$ for some positive constant ℓ, choosing ℓ such that the inverse temperature spacing yields a swap acceptance probability of approximately 0.234 is also optimal (Atchadé et al., 2011) in the sense that it maximises the ESJD of a parallel tempering algorithm.

5 Parallel Tempering Simulations

In this section, we provide parallel tempering experiments that investigate how well the 0.234 optimal swap acceptance rate holds for the various multimodal target distributions in Section 3.4 in finite dimensions. The parallel tempering method is highly effective for exploring multimodal distributions (Atchadé et al., 2011; Hastings, 1970) and thus we aim our experiments at these distributions to learn more about their applicability in practice.

5.1 Methodology

We conduct experiments that examine the trend of ESJD with swap acceptance rate for each of the "rough carpet" distribution from Section 3.4.1, three-mixture distribution from Section 3.4.3, and standard multivariate Gaussian distribution as a baseline for comparison. For each distribution, we run many simulations where each simulation is a parallel tempering algorithm with 40 different average swap acceptance rate values. We explain how we set up these algorithms to match these swap rates in Section 5.1.1. Each parallel tempering algorithm runs for 20,000 iterations, so every individual chain in the algorithm takes 20,000 steps. We attempt a swap every 20 steps. Furthermore, we ran each algorithm instance over 5 different seeds and average the results to reduce the effects of randomness caused by a particular seed. Overall, each experiment on a single distribution has 200 simulations which runs for 20,000 iterations per chain, and the number of chains for each parallel tempering algorithm simulation is determined by the inverse temperature spacing given by the intended average swap acceptance rate. We refer the reader to Section 5.1.1 for details on how these inverse temperatures were set up.

For each target distribution, we set the tempered target distribution to simply be the original distribution raised to the power of β. We set the modes to be far away enough such that the other modes were unreachable by the standard random-walk Metropolis algorithm after 100,000 iterations, but still reasonably close so that the parallel tempering method would show the cold chain values would swap between modes somewhat frequently.

5.1.1 Constructing an Inverse Temperature Ladder Iteratively

The most common method to construct an inverse temperature ladder selects the inverse temperatures using a geometric series spacing. However, since we are examining

the 0.234 swap acceptance rule, we need to have a way of constructing the inverse temperature spacings such that the probability of a swap between adjacent chains is approximately 0.234 (or any other value). To construct an inverse temperature ladder with a desired swap acceptance rate s, we use an iterative procedure adapted from (Atchadé et al., 2011, Section 2.2). Starting with $\beta_0 = 1$ and a minimum value $\beta_{min} = 0.01$, we iteratively add β's to the ladder. Let β_{curr} denote the most recent β in the ladder. We initialize $\rho_n = 0.5$ ($n = 1$ initially) and set $\beta^* = \beta_{curr}(1+e^{\rho_n})^{-1}$. To determine if β^* should be added to the ladder, we draw N samples (our implementation uses $N = 3000$) from the target distribution tempered by β^* and β_{curr}, and calculate the average swap probability a. If a is within $s \pm 0.005$, we add β^* to the ladder and set $\beta_{curr} = \beta^*$. Otherwise, we update $\rho_{n+1} = \rho_n + n^{-0.25}(a - s)$ and recalculate β^*. This process continues until $\beta^* \leq \beta_{min}$, at which point we add β_{min} to the ladder and terminate.

5.2 Multivariate Gaussian Target Distribution

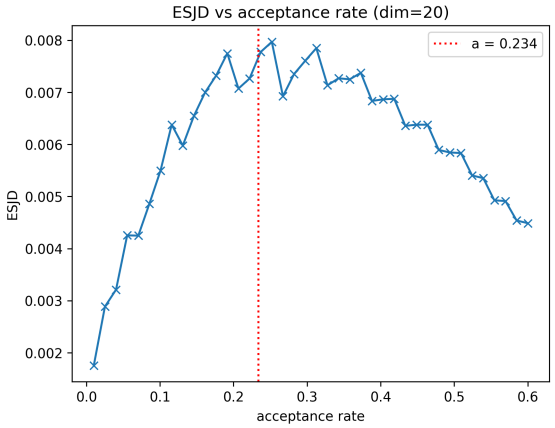

Fig. 13: ESJD vs swap acceptance rate for parallel tempering multivariate Gaussian target in dimension 20.

We first give an example of the standard multivariate Gaussian distribution to verify the correctness of our implementation. Note that the standard multivariate Gaussian as a target distribution satisfies the necessary assumptions of Theorem 1 in Atchadé et al. (2011): our target is a product of individual component densities, and we temper the distributions by raising them to the power of the original density, and $\epsilon \searrow 0$ such that $\epsilon = d^{-1/2}\ell$ for some positive constant ℓ. We find that the results are consistent with the theoretical figure, and that the optimal swap acceptance rate is around 0.234. Please refer to Figure 13 for an illustration. Although the iterative temperature construction algorithm described in Section 5.1.1 may be costly in general, this example illustrates that if the practitioner is willing and able to construct

the inverse temperatures with a spacing given by a swap acceptance probability, constructing a ladder with a swap acceptance probability of about 0.234 may be a viable and optimal strategy in the presence of certain restrictions on the target density.

5.3 Multimodal Distributions

5.3.1 Rough Carpet Distribution

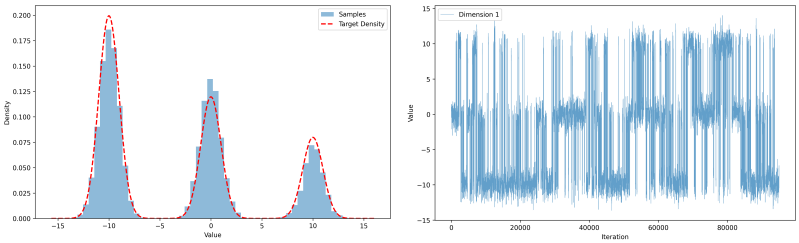

Fig. 14: Parallel tempering histogram and traceplot of the cold chain first component in the carpet distribution after 100,000 iterations.

In this section, we use the same target distribution as described in Section 3.4.1. Here, we examine the simpler case with homogeneous scaling factors $f_d(\mathbf{x}) = \prod_{i=1}^{n} f(x_i)$. Our individual component density $f(x)$ is the same as Equation (1), and in the $f(x)$ definition we set $\mu_1 = -10$, $\mu_2 = 0$, $\mu_3 = 10$. We provide an example histogram and traceplot after 100,000 iterations with a constructed ladder to demonstrate what mixing looks like. Unlike the standard random-walk Metropolis which cannot escape from the central mode in this setting of the target distribution, the values in the cold chain swap between modes considerably often due to the swapping of values between chains. Refer to Figure 14 to see the cold chain's mixing in the first component (all components are i.i.d.).

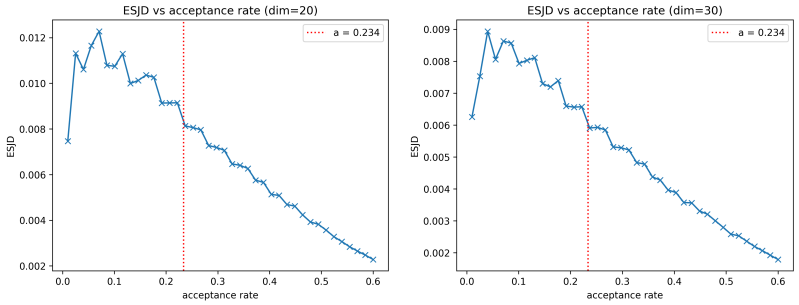

Fig. 15: ESJD vs swap acceptance rate for the parallel tempering rough carpet target density in $d = 20, 30$.

Table 8: Constructed swap acceptance rates that maximise ESJD for the rough carpet product target density.

Dimension d	20	30
Optimal Acceptance Rate	0.07051	0.04025

We conduct this experiment in $d = 20, 30$ and report the results in Figure 15 and Table 8. In both experiments, the optimal swap acceptance rate is less than 0.1, which is a substantial discrepancy from the theoretical figure of 0.234. In fact, the observed optimal swap acceptance rate decreases with increasing dimensionality. This might be due to the awkward shape of the distribution. Since it has 3^d modes, the density at each mode cannot be that high, which might mean the target distribution of a hot chain would look almost uniform and thus hot chains might not be sampling the right values that need to be swapped to colder chains.

5.3.2 Three Mixture Distribution

In this section, we use the same target distribution as described in Section 3.4.3 Equation 2. We set weights $w_1 = w_2 = w_3 = \frac{1}{3}$, means $\mu_1 = (-15, 0, 0, \ldots, 0)$, $\mu_2 = (0, 0, \ldots, 0)$, $\mu_3 = (15, 0, 0, \ldots, 0)$ and covariances $\Sigma_1 = \Sigma_2 = \Sigma_3 = I_d$. We provide an example histogram and traceplot after 100,000 iterations with a constructed ladder to demonstrate what the cold chain mixing looks like in the first component in Figure 16.

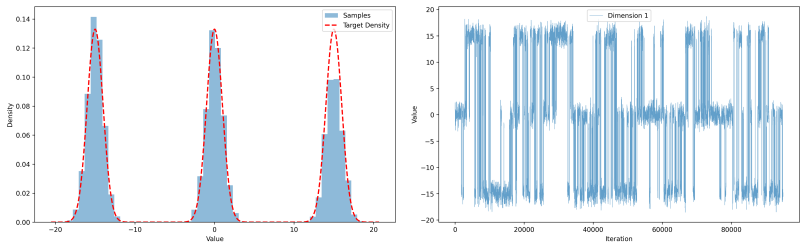

Fig. 16: Parallel tempering histogram and traceplot of the cold chain first component in the Three-Mixture distribution after 100,000 iterations. Note that the rightmost mode is slightly undersampled.

We refer the reader to Figure 17 and Table 9 for the results. Unlike the previous rough carpet density, we find that for the three-mixture density, constructing a ladder with a spacing of 0.234 seems approximately optimal even in low dimensions. However, the target density here is not in the i.i.d. product form. As noted in the Section 3.4.3 experiments, taking the sum of three Gaussians as a target density does seem to allow the RWM optimal scaling results to hold despite being outside of the theoretical assumptions, and this may extend to the parallel tempering optimal scaling framework too.

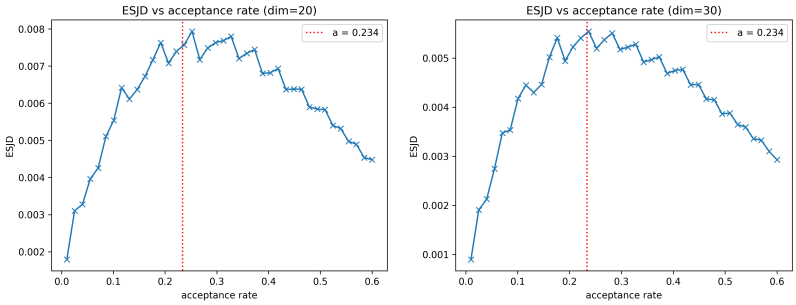

Fig. 17: ESJD vs swap acceptance rate for the parallel tempering three-mixture target density in $d = 20, 30$.

Table 9: Constructed swap acceptance rates that maximise ESJD for the three-mixture target density.

Dimension d	20	30
Optimal Acceptance Rate	0.2520	0.2369

6 Discussion

We have provided an extensive list of experiments examining optimal scaling and acceptance rates for the Random Walk Metropolis (RWM) algorithm with a variety of target and proposal distributions. We have also examined the optimal scaling and acceptance rates for parallel tempering. Overall, the 0.234 acceptance rate for RWM is surprisingly robust and generalizable. Even in lower dimensions, it appears roughly optimal for non-Gaussian proposals and various multimodal target distributions where the target density is a product of i.i.d. single-dimensional component densities such as the "rough carpet", or even a weighted sum of Gaussian densities (a mixture of three Gaussians) which is not included in the theorem assumptions. This result for the Gaussian mixture distribution even appears to extend to a inhomogeneous scaling of each component in each Gaussian, highlighting the robustness of the 0.234 acceptance rate figure and offering a suggestion for further relaxation of the necessary assumptions.

That being said, there are cases where 0.234 does not seem as relevant. The main case is in very low dimensions; in all our experiments with varying target densities, the optimal acceptance rate strays from 0.234 below a certain threshold which varies for each target density. Furthermore, although it is known that the 0.234 acceptance rate is not optimal for the continuous hypercube, this is somewhat surprising at first glance since this target density is continuously differentiable everywhere except on its boundary, which has zero measure. Lastly, the *inhomogeneous scaling* of each component in multimodal distributions is a crucial consideration affecting this optimal acceptance rate in practice; although in the infinite-dimensional limit and for large enough dimensions at about $d = 50$, the heterogeneous scaling factors should not and do not appear

to affect the optimal acceptance rate, we find that in smaller dimensions, including inhomogeneous scaling causes the observed optimal acceptance rate to be strictly less than 0.234. Open questions remain about to what extent or why the inhomogeneous scaling factors may reduce the optimal acceptance rate in smaller dimensions. A possible argument to answer the latter question is that the law of large numbers for the expected value of the sampled scaling factors may not be applicable in smaller dimensions. To avoid this issue altogether, we suggest to practitioners that variables should have transformations applied to ensure they have similar scaling, i.e. we emphasize normalizing variables to have the 0.234 acceptance rate be approximately optimal.

Next, we discuss the optimal swap acceptance rate figure for parallel tempering and compare it with the optimal acceptance rate for random-walk Metropolis. On one hand, the results for the multivariate Gaussian and three-mixture density examples show that the 0.234 swap acceptance rate figure may be optimal even in lower dimensions. Although costly, if the practitioner is willing and able to construct the inverse temperature ladder iteratively with a spacing given by a swap acceptance probability, the spacing dictated by a 0.234 swap acceptance probability may be optimal for the practitioner. However, the theoretical 0.234 figure may not always be applicable: the "rough carpet" product density which satisfies the restrictive i.i.d. product form for the target density shows that the practical optimal swap acceptance rate might be on a case-by-case basis. It is difficult to be sure why this is the case: we hypothesize that perhaps the reason why a swap acceptance rate of 0.234 is more optimal in this case is due to the greater concentration of density in the state space for the multivariate Normal and the three-mixture example compared to the rough carpet density, where density is spread out in 3^d modes.

Not only that, constructing a temperature ladder iteratively with the "optimal spacing" is challenging and time-consuming. It requires the ability to draw samples from the target distribution, which is itself the main goal of the parallel tempering algorithm. Without a direct way to sample from the target, a practitioner could run an MCMC algorithm with the β value being proposed for the new addition to the ladder, but this may be time-consuming with many attempts required until a suitable β is found for each step of the ladder. Since this optimal spacing may not yield the optimal ESJD in practice for finite, lower-dimensional targets and with a finite number of runs, an alternative approach is to adaptively adjust the spacings of the temperature ladder during the algorithm's runs based on its recent ESJD performance over a recent window as guidance to practitioners. This is known as adaptive parallel tempering (Miasojedow et al., 2013). Previous literature shows that adaptive parallel tempering algorithms can achieve substantial efficiency gains over a standard geometric temperature spacing (Vousden et al., 2015) while still converging to the target distribution (Miasojedow et al., 2013) under the assumptions that the target distribution is sufficiently regular and has tails decaying faster than exponentially. Adaptive parallel tempering may suffice to be "good enough" for most practitioners whilst avoiding the costly temperature construction algorithm.

Lastly, while we have established that the 0.234 acceptance rate for RWM is still fairly good even in lower dimensions and on some target distributions that may not have the i.i.d. product form, we provide some precautions to the practitioner. The

0.234 acceptance rate is a good heuristic to tune the algorithm to, yet there are still many cases where it is not optimal in terms of efficiency, such as the continuous hypercube example in Section 3.3. Even if the 0.234 acceptance rate is in fact optimal, the acceptance rate alone does not necessarily guarantee good samples, as illustrated by Figure 6. What literature (Roberts and Rosenthal, 2001; Sherlock et al., 2010) indicates is that another very important thing in practice beyond just the acceptance rate is defining a good proposal covariance matrix Σ_n that is approximately proportional to the target covariance matrix Σ_π; doing so can significantly increase the asymptotic relative efficiency. Realistically, it may be impossible to know Σ_π in advance or even provide a good estimate for it. In fact, understanding how we can extend this result to correlated targets is an ongoing challenge (Yang et al., 2020) and immediately relevant to the practitioner since real-world data tends to have correlations between variables. Practitioners may consider using the adaptive Metropolis algorithm (Haario et al., 2001) to dynamically update the algorithm's proposal covariance matrix. This adaptive algorithm is very useful and yields significant speed benefits in low and high dimensions (Craiu et al., 2009; Roberts and Rosenthal, 2009), but an important potential pitfall to note is that adaptive methods are not guaranteed to converge to the correct target density; in many cases, they may fail to converge or converge to something completely different (Roberts and Rosenthal, 2007, 2009).

Declarations

The authors did not receive any funding for this submitted work and have no relevant financial or non-financial competing interests. The code implementing the experiments and generating the figures used in this work, and the experiment results stored in JSON format are publicly available on GitHub at https://github.com/aidanmrli/montecarlo.

References

Atchadé, Y.F., Roberts, G.O., Rosenthal, J.S.: Towards optimal scaling of Metropolis-coupled Markov chain Monte Carlo. Statistics and Computing **21**(4), 555–568 (2011) https://doi.org/10.1007/s11222-010-9192-1

Brooks, S., Gelman, A., Jones, G., Meng, X.-L. (eds.): Handbook of Markov Chain Monte Carlo. Chapman and Hall/CRC, New York (2011). https://doi.org/10.1201/b10905

Breyer, L.A., Roberts, G.O.: From Metropolis to diffusions: Gibbs states and optimal scaling. Stochastic Processes and their Applications **90**(2), 181–206 (2000) https://doi.org/10.1016/S0304-4149(00)00041-7

Bédard, M., Rosenthal, J.S.: Optimal scaling of Metropolis algorithms: Heading toward general target distributions. Canadian Journal of Statistics **36**(4), 483–503 (2008) https://doi.org/10.1002/cjs.5550360401

Bédard, M.: Weak convergence of Metropolis algorithms for non-i.i.d. target distributions. The Annals of Applied Probability **17**(4), 1222–1244 (2007) https://doi.org/10.1214/105051607000000096

Bédard, M.: Efficient Sampling Using Metropolis Algorithms: Applications of Optimal Scaling Results. Journal of Computational and Graphical Statistics **17**(2), 312–332 (2008)

Craiu, R.V., Rosenthal, J., Yang, C.: Learn From Thy Neighbor: Parallel-Chain and Regional Adaptive MCMC. Journal of the American Statistical Association **104**(488), 1454–1466 (2009) https://doi.org/10.1198/jasa.2009.tm08393

Geyer, C.J.: Markov chain Monte Carlo maximum likelihood. Computing Science and Statistics: Proceedings of the 23rd Symposium on the Interface, 156–163 (1991)

Gelman, A., Pasarica, C.: Adaptively Scaling the Metropolis Algorithm Using Expected Squared Jumped Distance. SSRN Electronic Journal (2007) https://doi.org/10.2139/ssrn.1010403

Hastings, W.K.: Monte Carlo Sampling Methods Using Markov Chains and Their Applications. Biometrika **57**(1), 97–109 (1970) https://doi.org/10.2307/2334940

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

Jarner, S., Roberts, G.: Polynomial convergence rates of Markov chains. The Annals of Applied Probability **12**, 224–247 (2002)

Miasojedow, B., Moulines, E., Vihola, M.: An adaptive parallel tempering algorithm. Journal of Computational and Graphical Statistics **22**(3), 649–664 (2013)

Mattingly, J.C., Pillai, N.S., Stuart, A.M.: Diffusion limits of the random walk Metropolis algorithm in high dimensions. The Annals of Applied Probability **22**(3), 881–930 (2012) https://doi.org/10.1214/10-AAP754

Metropolis, N., Rosenbluth, A.W., Rosenbluth, M.N., Teller, A.H., Teller, E.: Equation of State Calculations by Fast Computing Machines. The Journal of Chemical Physics **21**(6), 1087–1092 (1953) https://doi.org/10.1063/1.1699114

Neal, P., Roberts, G.: Optimal scaling for partially updating MCMC algorithms. The Annals of Applied Probability **16**(2), 475–515 (2006) https://doi.org/10.1214/105051605000000791

Neal, P., Roberts, G.: Optimal Scaling of Random Walk Metropolis Algorithms with Non-Gaussian Proposals. Methodology and Computing in Applied Probability **13**(3), 583–601 (2011) https://doi.org/10.1007/s11009-010-9176-9

Neal, P., Roberts, G.O., Yuen, W.K.: Optimal scaling of random walk Metropolis

algorithms with discontinuous target densities. Annals of Applied Probability **22**, 1880–1927 (2012)

Robert, C.P., Casella, G.: Monte Carlo Statistical Methods. Springer Texts in Statistics. Springer, New York, NY (2004). https://doi.org/10.1007/978-1-4757-4145-2 . http://link.springer.com/10.1007/978-1-4757-4145-2

Roberts, G.O., Gelman, A., Gilks, W.R.: Weak convergence and optimal scaling of random walk Metropolis algorithms. The Annals of Applied Probability **7**(1), 110–120 (1997)

Roberts, G.O.: Optimal Metropolis algorithms for product measures on the vertices of a hypercube. Stochastics and Stochastics Reports **62**, 275–283 (1998)

Rosenthal, J.S.: Minorization Conditions and Convergence Rates for Markov Chain Monte Carlo. Journal of the American Statistical Association **90**(430), 558–566 (1995) https://doi.org/10.2307/2291067

Roberts, G.O., Rosenthal, J.S.: Optimal scaling for various Metropolis-Hastings Algorithms. Statistical Science **16**(4), 351–367 (2001)

Roberts, G.O., Rosenthal, J.S.: Coupling and ergodicity of adaptive Markov Chain Monte Carlo algorithms. Journal of Applied Probability **44**(2), 458–475 (2007)

Roberts, G.O., Rosenthal, J.S.: Examples of Adaptive MCMC. Journal of Computational and Graphical Statistics **18**(2), 349–367 (2009) https://doi.org/10.1198/jcgs.2009.06134 https://doi.org/10.1198/jcgs.2009.06134

Roberts, G., Rosenthal, J.: Minimising MCMC variance via diffusion limits, with an application to simulated tempering. The Annals of Applied Probability **24** (2014) https://doi.org/10.1214/12-AAP918

Sherlock, C., Fearnhead, P., Roberts, G.O.: The Random Walk Metropolis: Linking Theory and Practice Through a Case Study. Statistical Science **25**(2), 172–190 (2010) https://doi.org/10.1214/10-STS327

Sherlock, C., Roberts, G.: Optimal scaling of the random walk Metropolis on elliptically symmetric unimodal targets. Bernoulli **15**(3), 774–798 (2009) https://doi.org/10.3150/08-BEJ176

Vousden, W.D., Farr, W.M., Mandel, I.: Dynamic temperature selection for parallel tempering in Markov chain Monte Carlo simulations. Monthly Notices of the Royal Astronomical Society **455**(2), 1919–1937 (2015) https://doi.org/10.1093/mnras/stv2422

Yang, J., Roberts, G.O., Rosenthal, J.S.: Optimal Scaling of Random-Walk Metropolis Algorithms on General Target Distributions (2020)