Asymptotically Optimal Exact Minibatch Metropolis-Hastings

Ruqi Zhang[*]
Cornell University
rz297@cornell.edu

A. Feder Cooper[*]
Cornell University
afc78@cornell.edu

Christopher De Sa
Cornell University
cdesa@cs.cornell.edu

Abstract

Metropolis-Hastings (MH) is a commonly-used MCMC algorithm, but it can be intractable on large datasets due to requiring computations over the whole dataset. In this paper, we study *minibatch MH* methods, which instead use subsamples to enable scaling. We observe that most existing minibatch MH methods are inexact (i.e. they may change the target distribution), and show that this inexactness can cause arbitrarily large errors in inference. We propose a new exact minibatch MH method, *TunaMH*, which exposes a tunable trade-off between its batch size and its theoretically guaranteed convergence rate. We prove a lower bound on the batch size that any minibatch MH method *must* use to retain exactness while guaranteeing fast convergence—the first such bound for minibatch MH—and show TunaMH is asymptotically optimal in terms of the batch size. Empirically, we show TunaMH outperforms other exact minibatch MH methods on robust linear regression, truncated Gaussian mixtures, and logistic regression.

1 Introduction

Bayesian inference is widely used for probabilistic modeling of data. Specifically, given a dataset $\mathcal{D} = \{x_i\}_{i=1}^N$ and a θ-parameterized model, it aims to compute the posterior distribution

$$\pi(\theta) \propto \exp\left(-\textstyle\sum_{i=1}^N U_i(\theta)\right), \text{where } U_i(\theta) = -\log p(x_i|\theta) - \tfrac{1}{N}\log p(\theta).$$

Here $p(\theta)$ is the prior and the $p(x_i|\theta)$ give the likelihood of observing x_i given the parameter θ. We assume the data are conditionally independent given θ. The U_i have a natural interpretation as component *energy functions* with π acting as a Gibbs measure. In practice, computing $\pi(\theta)$ is often intractable and thus requires using approximate methods, such as Markov chain Monte Carlo (MCMC). MCMC uses sampling to estimate the posterior and is guaranteed to converge asymptotically to the true distribution, π [9].

The Metropolis-Hastings (MH) algorithm [16, 21] is one of the most commonly used MCMC methods. In each step, MH generates a proposal θ' from a distribution $q(\cdot|\theta)$, and accepts it with probability

$$a(\theta, \theta') = \min\left(1, \tfrac{\pi(\theta')q(\theta|\theta')}{\pi(\theta)q(\theta'|\theta)}\right) = \min\left(1, \exp\left(\textstyle\sum_{i=1}^N (U_i(\theta) - U_i(\theta'))\right) \cdot \tfrac{q(\theta|\theta')}{q(\theta'|\theta)}\right). \quad (1)$$

If accepted, the chain transitions to θ'; otherwise, it remains at the current state θ. This accept/reject step can be quite costly when N is large, since it entails computing a sum over the entire dataset.

Prior work has proposed many approaches to mitigate the cost of this decision step [5]. One popular approach involves introducing stochasticity: instead of computing over the entire dataset, a subsample, or *minibatch*, is used to compute an approximation. These minibatch MH methods can be divided into

[*]Equal contribution.

34th Conference on Neural Information Processing Systems (NeurIPS 2020), Vancouver, Canada.

two classes, *exact* and *inexact*, depending on whether or not the target distribution π is necessarily preserved. Inexact methods introduce asymptotic bias to the target distribution, trading off correctness for speedups [6, 17, 23, 24, 26]. Exact methods either require impractically strong constraints on the target distribution [20, 27], limiting their applicability in practice, or they negatively impact efficiency, counteracting the speedups that minibatching aims to provide in the first place [4, 12]. Moreover, all existing exact methods operate on the belief that there is a trade-off between batch size and convergence rate—between scalability and efficiency. Yet no prior work formally exposes this trade-off, and most prior work gives no convergence rate guarantees. Given these various considerations, it is not entirely clear how to evaluate which minibatch MH method to use.

In this paper we forge a path ahead to untangle this question. While inexact methods have been prominent recently due to their efficiency, they are not reliable: we show that the stationary distribution of any inexact method can be arbitrarily far from the target π. This means they can yield disastrously wrong inference results in practice, and it is difficult to tell just how bad those results can be.

We therefore turn our attention to exact methods and introduce *TunaMH*.[2] Compared to prior work, we make milder assumptions, which enables TunaMH to apply to a wider variety of inference tasks. More specifically, we require local rather than global bounds on the target distribution [20, 27] and do not rely on the Bernstein-von Mises approximation [5, 7, 12]. TunaMH is guaranteed to retain sample efficiency in the presence of minibatching: its convergence rate (measured by the spectral gap) is within a constant factor of standard, non-minibatch MH. More importantly, TunaMH also enables us to rigorously characterize the trade-off between scalability and efficiency. It has a hyperparameter χ, which enables tuning the trade-off between expected batch size and convergence rate.

By exposing this trade-off, our analysis raises the natural question: *is TunaMH optimal for this trade-off?* That is, could another exact algorithm use an asymptotically smaller average batch size while having the same convergence rate guarantees? We explore this in Section 4; under the same mild assumptions we use to derive TunaMH, we prove a lower bound on the expected batch size for *any* exact minibatch MH method that can keep a reasonable convergence rate. To our knowledge, we are the first to prove a lower bound of this nature for minibatch MH. Moreover, TunaMH is *asymptotically optimal* in balancing the expected batch size and convergence rate. It remains exact and efficient while on average using the smallest possible number of samples. In summary:

- We demonstrate that any inexact minibatch MH method can be arbitrarily inaccurate (Section 2.1).
- We introduce a new exact method, TunaMH (Section 3), with a lower bound on its convergence rate (in terms of the spectral gap) and a tunable hyperparameter to balance the trade-off between convergence rate and batch size.
- We prove a lower bound on the batch size for any exact minibatch MH method given a target convergence rate—the first such lower bound in this area. This result indicates that the expected batch size of TunaMH is asymptotically optimal in terms of the problem parameters (Section 4).
- We show empirically that TunaMH outperforms state-of-the-art exact minibatch MH methods on robust linear regression, truncated Gaussian mixture, and logistic regression (Section 5).

2 Preliminaries and Drawbacks of Prior Minibatch MH Methods

We first formally define the class of methods that we study theoretically in this paper: minibatch MH methods of the form of Algorithm 1. This class contains methods that sample a proposal from distribution q (which we always assume results in the chain being ergodic), and choose to accept or reject it by calling some randomized subroutine, `SubsMH`, which outputs 1 or 0 for "accept" or "reject," respectively. Algorithms in this class have several notable properties. First, `SubsMH` is *stateless*: each acceptance decision is made independently, without carrying over local state associated with the MH procedure between steps. Many prior methods are stateless [6, 12, 17, 26]. We do not consider *stateful* methods, in which the decision depends on previous state; they are difficult to analyze due to running on an extended state space [3, 24]. Second, `SubsMH` takes a function that computes energy *differences* $U_i(\theta) - U_i(\theta')$ and outputs an acceptance decision. We evaluate efficiency in terms of how many times `SubsMH` calls this function, which we term the *batch size* the method uses. Third, `SubsMH` takes parameters that bound the maximum magnitude of the energy differences. Specifically, as in Cornish et al. [12], we assume:

[2] TunaMH since it *tunes* the efficiency-scalability trade-off and uses a Poisson (French for "fish") variable.

Algorithm 1 Stateless, Energy-Difference-Based Minibatch Metropolis-Hastings

given: state space Θ, energy functions $U_1, \ldots, U_N : \Theta \to \mathbb{R}$, proposal dist. q, initial state $\theta \in \Theta$
given: parameters $c_1, \ldots, c_N$, C, M from Assumption 1, randomized algorithm `SubsMH`
loop
  sample $\theta' \sim q(\cdot|\theta)$
  define function $\Delta U : \{1, \ldots, N\} \to \mathbb{R}$, such that $\Delta U(i) = U_i(\theta) - U_i(\theta')$
  call subroutine $o \leftarrow \texttt{SubsMH}(\Delta U, N, q(\theta|\theta')/q(\theta'|\theta), c_1, \ldots, c_N, C, M(\theta, \theta'))$
  if $o = 1$, **update** $\theta \leftarrow \theta'$
end loop

Assumption 1. *For some constants $c_1, \ldots, c_N \in \mathbb{R}_+$, with $\sum_i c_i = C$, and symmetric function $M : \Theta \times \Theta \to \mathbb{R}_+$, for any $\theta, \theta' \in \Theta$, the energy difference is bounded by $|U_i(\theta) - U_i(\theta')| \leq c_i M(\theta, \theta')$.*

One can derive such a bound, which can be computed in $O(1)$ time, for many common inference problems: for example, if each energy function U_i is L_i-Lipschitz continuous, then it suffices to set $c_i = L_i$ and $M(\theta, \theta') = \|\theta - \theta'\|$ (See Appendix J for examples of c_i and M on common problems). Note that the `SubsMH` method may choose *not* to use these bounds in its decision. We allow this so the form of Algorithm 1 can include methods that do not require such bounds. Most existing methods can be described in this form [4, 6, 12, 17, 26]. For example, standard MH can be written by setting `SubsMH` to a subroutine that computes the acceptance rate a as in (1) and outputs 1 (i.e., accept) with probability a.

Such minibatch MH methods broadly come in two flavors: *inexact* and *exact*. We next establish the importance of being exact and demonstrate how TunaMH resolves drawbacks in prior work.

2.1 The Importance of Being Exact

Inexact methods are popular due to helping scale MH to new heights [6, 17, 24, 26]. They approximate the MH acceptance ratio to within an error tolerance (> 0), trading off exactness for efficiency gains. Surprisingly, the bias from inexactness can be arbitrarily large even when the error tolerance is small.

Theorem 1. *Consider any minibatch MH method of the form in Algorithm 1 that is inexact (i.e. does not necessarily have π as its stationary distribution for all π satisfying Assump. 1). For any constants $\delta \in (0, 1)$ and $\rho > 0$, there exists a target distribution π and proposal distribution q such that if we let $\tilde{\pi}$ denote a stationary distribution of the inexact minibatch MH method on this target, it satisfies*

$$\mathrm{TV}(\pi, \tilde{\pi}) \geq \delta \text{ and } \mathrm{KL}(\pi, \tilde{\pi}) \geq \rho.$$

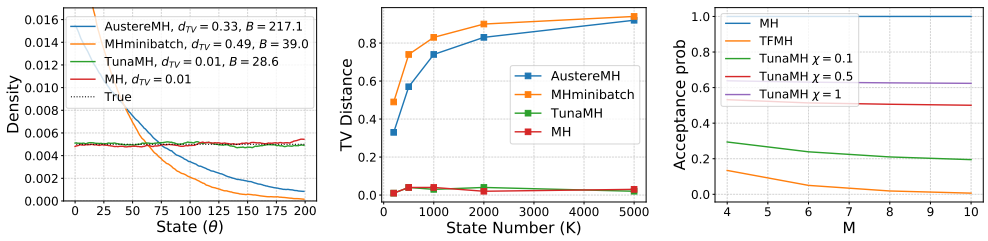
where TV is the total variation distance and KL is the Kullback–Leibler divergence.

Theorem 1 shows that when using any inexact method, there always exists a target distribution π (factored in terms of energy functions U_i) and proposal distribution q such that it will approximate π arbitrarily poorly. This can happen even when individual errors are small; they can still accumulate a very large overall error. We prove Theorem 1 via a simple example—a random walk along a line, in which the inexact method causes the chain to step towards one direction more often than the other, even though its steps should be balanced (Appendix A). Note that it may be possible to avoid a large error by using some specific proposal distribution, but such a proposal is hard to know in general.

We use AustereMH [17] and MHminibatch [26] to empirically validate Theorem 1. For these inexact methods, we plot density estimates with the number of states $K = 200$ in Figure 1a (see Appendix J.1 for using other K); the stationary distribution diverges from the target distribution significantly. Moreover, the TV distance between the density estimate and the true density increases as K increases on this random walk example (Figure 1b). By contrast, our exact method (Section 3) keeps a small TV distance on all K and estimates the density accurately with an even smaller average batch size. We also tested AustereMH on robust linear regression, a common task, to show that the error of inexact methods can be large on standard problems (Appendix J.1).

2.2 Issues with Existing Exact Methods

This observation suggests that we should be using exact methods when doing minibatch MH. However, existing approaches present additional drawbacks, which we discuss below.



(a) (b) (c)

Figure 1: Existing MH method issues. (a)-(b) Inexact methods can diverge a lot from true distribution. "d_{TV}" and "B" denote the TV distance and the batch size respectively. (c) SMH has low and TunaMH with different values of hyperparameter χ has high acceptance rates.

Factorized MH and Scalable MH are stateless, exact minibatch methods. Factorized MH (FMH) decomposes the acceptance rate into a product of factors, which allows for rejecting a proposal based on a minibatch of data [4, 10, 11]. Truncated FMH (TFMH) is a FMH variant that maintains geometric ergodicity; it falls back on standard MH in a step when the bound on the factors reaches a certain threshold [12]. No matter how this threshold is set, we can construct tasks where TFMH is either arbitrarily inefficient (rejecting arbitrarily often, slowing convergence), or degrades entirely to standard MH.

Statement 1. *For any constant $p \in (0, 1)$, there exists a target distribution such that TFMH either has an acceptance rate which is less than p times that of standard MH, or it completely degrades to standard MH (summing over the whole dataset at each step).*

We prove this statement in Appendix C using an example of a uniform distribution along a line, where we let x_i take one of two values, $\{-M/N, M/N\}$ with $M > 0$. We show that the acceptance rate of TFMH can be arbitrarily low by increasing M, which we also empirically verify in Figure 1c.

To improve the acceptance rate of TFMH, Scalable MH (SMH) introduces control variates, which approximate U_i with a Taylor series around the mode [12]. However, it only works with unimodal posteriors and high-quality Bernstein-von Mises approximations—conditions that do not hold for many common inference tasks.

PoissonMH is a stateless minibatch MH method adapted from an algorithm designed for scaling Gibbs sampling on factor graphs [27]. However, unlike our method, it requires strong assumptions—specifically, a global upper bound on the energy. Such an upper bound usually does not exist and, even if it does, can be very large, resulting in an impractically large batch size.

FlyMC is a stateful method, which means it uses auxiliary random variables to persist state across different MH steps [20]. It requires a lower bound on the likelihood function, which is typically more demanding than Assumption 1 and does not have theoretical performance guarantees.

Other exact methods exist based on Piecewise Deterministic Markov Processes [7, 8]. They require regularity conditions only available for some problems, so their practical utility is limited.

3 TunaMH: Asymptotically Optimal Exact MH

In this section, we present our method, TunaMH, which evades the issues of prior exact methods discussed in Section 2.2. Like SMH [12], our method works on distributions for which an *a priori* bound on the energy differences is known (Assumption 1).

Our algorithm, presented in Algorithm 2, takes as parameters $c_1, \ldots, c_N$, C, and M from Assumption 1, along with an additional hyperparameter, $\chi > 0$. It proceeds in four steps. First, like any MH method, it generates a proposal θ' from given distribution q. Second, it samples a batch size B from a Poisson distribution. This makes the expected number of energy functions U_i evaluated by our method at each step $\mathbf{E}[B] = \chi C^2 M^2(\theta, \theta') + CM(\theta, \theta')$[3]. Importantly, this means the batch

[3]Note that $\mathbf{E}[B]$ is typically $<< N$ and can be decreased using small step sizes. If, however, $\mathbf{E}[B] > N$, then we can simply use standard MH in that iteration, similar to TFMH.

Algorithm 2 TunaMH

given: initial state $\theta \in \Theta$; proposal dist. q; hyperparameter χ; Asm. 1 parameters c_i, C, M
loop
  propose $\theta' \sim q(\cdot|\theta)$ and **compute** $M(\theta, \theta')$
  ▷ Form minibatch $\mathcal{I}$
  sample $B \sim \text{Poisson}\left(\chi C^2 M^2(\theta, \theta') + CM(\theta, \theta')\right)$
  initialize minibatch indices $\mathcal{I} \leftarrow \emptyset$ (an initially empty multiset)
  for $b \in \{1, \ldots, B\}$ **do**
    sample i_b such that $\mathbf{P}(i_b = i) = c_i/C$, for $i = 1 \ldots N$
    with probability $\frac{\chi c_{i_b} CM^2(\theta,\theta') + \frac{1}{2}(U_{i_b}(\theta') - U_{i_b}(\theta) + c_{i_b} M(\theta,\theta'))}{\chi c_{i_b} CM^2(\theta,\theta') + c_{i_b} M(\theta,\theta')}$ **add** i_b to $\mathcal{I}$
  end for
  ▷ Accept/reject step based on minibatch $\mathcal{I}$
  compute MH ratio $r \leftarrow \exp\left(2\sum_{i\in\mathcal{I}} \operatorname{artanh}\left(\frac{U_i(\theta) - U_i(\theta')}{c_i M(\theta,\theta')(1+2\chi CM(\theta,\theta'))}\right)\right) \cdot \frac{q(\theta'|\theta)}{q(\theta|\theta')}$
  with probability $\min(1, r)$, set $\theta \leftarrow \theta'$
end loop

size may vary from iteration to iteration, and the expected size depends on θ and θ'. For example, TunaMH may tend to set B larger for larger-distance proposals with a higher $M(\theta, \theta')$. Third, it samples (with replacement) a minibatch of size B, but for each data point it samples, it has some probability of *ejecting* this point from the minibatch. Finally, it accepts the proposed θ' with some probability, computed using a sum over the post-ejection minibatch. Our method can be derived by carefully replacing the auxiliary variables in PoissonMH with *local* Poisson variables whose distributions change each iteration depending on the pair (θ, θ') (Appendix D). By construction TunaMH is exact; it preserves the target distribution π as its stationary distribution. This is because TunaMH is *reversible*, meaning its transition operator T satisfies $\pi(\theta)T(\theta, \theta') = \pi(\theta')T(\theta', \theta)$ for any $\theta, \theta' \in \Theta$. This is a common condition that guarantees that a MCMC method has π as its stationary distribution [9, 18].

Compared to previous exact methods, a significant benefit of TunaMH is that we can prove theoretical guarantees on its efficiency. Specifically, its convergence speed is guaranteed to be close to standard MH and χ allows us to control how close. To show this, we lower bound the convergence rate of TunaMH in terms of the *spectral gap*, which is commonly used to characterize convergence speed in the MCMC literature [15, 18, 25, 27, 28]. The larger the spectral gap, the faster the chain converges.

Definition 1. *The* spectral gap *of a reversible Markov chain is the distance between the largest and second-largest eigenvalues of its transition operator. That is, if the eigenvalues of the transition operator are* $1 = \lambda_1 > \lambda_2 \geq \lambda_3 \cdots$*, then the spectral gap is* $\gamma = 1 - \lambda_2$.

Theorem 2. *TunaMH (Algorithm 2) is reversible with stationary distribution* π*. Let* $\bar{\gamma}$ *denote the spectral gap of TunaMH, and let* γ *denote the spectral gap of standard MH with the same target distribution and proposal distribution. Then,*

$$\bar{\gamma} \geq \exp\left(-\frac{1}{\chi} - 2\sqrt{\frac{\log 2}{\chi}}\right) \cdot \gamma.$$

Intuitively, this theorem (proof in Appendix E) suggests the convergence rate of TunaMH is at most a constant slower than that of standard MH, and can be increased by adjusting the hyperparameter χ. Recall that χ also controls the batch size of TunaMH. Effectively, this means χ is a *dial* that allows us to directly tune the trade-off between convergence rate and batch size. When χ is large, the batch size B is large and the spectral gap ratio, $\bar{\gamma}/\gamma$, is close to 1: the larger batch size is less scalable but keeps a high convergence rate. Conversely, when χ is small, the batch size is small and the spectral gap ratio is close to 0: we trade off slow-downs in convergence rate for scalability. For example, for any $0 < \kappa < 1$, to guarantee the spectral gap ratio $\bar{\gamma}/\gamma \geq \kappa$ it suffices to set (Appendix F)

$$\chi = \frac{4}{(1-\kappa)\log(1/\kappa)}, \quad \text{giving an average batch size of } \mathbf{E}[B] = \frac{4C^2M^2(\theta,\theta')}{(1-\kappa)\log(1/\kappa)} + CM(\theta, \theta'). \quad (2)$$

In practice, we usually want to minimize the wall-clock time to achieve a certain estimate error, which requires tuning χ to optimally balance scalability and efficiency. We attempt to derive a theoretically

optimal value of χ in Appendix G by minimizing the product of the relaxation time—a measure of the number of steps needed—and the expected wall-clock time per step. Note that this product may be loose in bounding the total wall-clock time (we leave tightening this bound to future work), making the derived χ larger than necessary. In Section 5 we give a simple heuristic to tune χ, which works well and is generally better than the derived value.

Theorem 2 only requires the mild constraints of Assumption 1 on the target distribution, so applies in many scenarios and compares well to other exact methods. SMH further requires a Bernstein-von Mises approximation to have guarantees on its batch size and acceptance rate. PoissonMH provides convergence rate guarantees, but demands the strong assumption that the target distribution has a global upper bound on the energy. FlyMC does not have any theoretical guarantees on performance.

4 Towards Optimal Exact Minibatch MH

In Theorem 2, we expose the trade-off between convergence rate and batch size in TunaMH. Here, we take this analysis a step further to investigate the limits of how efficient an exact minibatch MH method can be. To tackle this problem, we derive a lower bound on the batch size for any minibatch MH method that retains exactness and fast convergence. We then show that TunaMH is asymptotically optimal in terms of its dependence on the problem parameters C and M. In other words, it is not possible to outperform TunaMH in this sense with a method in the class described by Algorithm 1.

Theorem 3. *Consider any stateless exact minibatch MH algorithm described by Algorithm 1, any state space Θ (with $|\Theta| \geq 2$), any $C > 0$, and any function $M : \Theta \times \Theta \to \mathbb{R}^+$. Suppose that the algorithm guarantees that, for some constant $\kappa \in (0, 1)$, for any distribution, the ratio between the spectral gap of minibatch MH $\hat{\gamma}$ and the spectral gap of standard MH γ is bounded by $\hat{\gamma} \geq \kappa\gamma$. Then there must exist a distribution π over Θ and proposal q such that the batch size B of that algorithm, when deciding whether to accept any transition $\theta \to \theta'$, is bounded from below by*

$$\mathbf{E}[B] \geq \zeta \cdot \kappa \cdot \left(C^2 M^2(\theta, \theta') + CM(\theta, \theta')\right) \tag{3}$$

for some constant $\zeta > 0$ independent of algorithm and problem parameters.

To prove this theorem, we construct a random walk example over two states, then consider the smallest batch size a method requires to distinguish between two different stationary distributions (Appendix H). The impact of Theorem 3 is three-fold:

First, it provides an upper bound on the performance of algorithms of Algorithm 1's form: in each iteration, the average batch size of any exact minibatch MH method of the form of Algorithm 1 must be set as in (3) in order to maintain a reasonable convergence rate. To the best of our knowledge, this is the first theorem that rigorously proves a ceiling for the possible performance of minibatch MH.

Second, TunaMH achieves this upper bound. In fact, Theorem 3 suggests that TunaMH is *asymptotically optimal* in terms of the problem parameters, C and M. To see this, observe that when we ignore κ, both expressions that bound $\mathbf{E}[B]$ in (2) and (3) are $\Theta(C^2M^2(\theta, \theta') + CM(\theta, \theta'))$. Thus TunaMH reaches the lower bound, achieving asymptotic optimality in terms of C and M. (Of course, this sense of "optimality" does not rule out potential constant-factor improvements over TunaMH or improvements that depend on κ.)

Lastly, this result suggests directions for developing new exact minibatch MH algorithms: to be significantly faster than TunaMH, we either need to introduce additional assumptions to the problem or to develop new stateful algorithms.

In prior work, when assuming a very concentrated posterior, some methods' batch size can scale in $\mathcal{O}(1)$ [5, 7, 12] or $\mathcal{O}(1/\sqrt{N})$ [12] in terms of the dataset size N while maintaining efficiency. Theorem 3 is compatible with these results, further demonstrating this is essentially the *best* dependency on N an exact minibatch MH method can achieve. We show this by explicitly assuming the dependency of C and M on N, as in SMH [12], yielding the following corollary (proof in Appendix I):

Corollary 1. *Suppose that C increases linearly with N ($C = \Theta(N)$) and $M(\theta, \theta')$ scales in $\Theta(N^{-(h+1)/2})$ for some constant $h > 0$. Then the lower bound in Theorem 3 becomes $\Theta(N^{(1-h)/2})$. In particular, it is $\Theta(1)$ when $h = 1$, and $\Theta(1/\sqrt{N})$ when $h = 2$.*

That is, TunaMH matches the state-of-the-art's dependency on N, and this dependency is optimal. Similarly, since C and M are the only problem parameters in the lower bound in Theorem 3, we can

also get the optimal dependency on the other problem parameters by explicitly assuming the relation of them with C and M.

5 Experiments

We compare TunaMH to MH, TFMH, SMH (i.e. TFMH with MAP control variates) and FlyMC. We only include PoissonMH in the Gaussian mixture experiment, as it is not applicable in the other tasks. All of these methods are unbiased, so they have the same stationary distribution. To ensure fair wall-clock time comparisons, we coded each method in Julia; our implementations are at least as fast as, if not faster than, prior implementations. For each trial, we use Gaussian random walk proposals. We tune the proposal stepsize separately for each method to reach a target acceptance rate, and report averaged results and standard error from the mean over three runs. We set χ to be roughly the largest value that keeps $\chi C^2 M^2(\theta, \theta') < 1$ in most steps; we keep χ as high as possible while the average batch size is around its lower bound $CM(\theta, \theta')$. We found this strategy works well in practice. We released the code at `https://github.com/ruqizhang/tunamh`.

5.1 Robust Linear Regression

We first test TunaMH on robust linear regression [12, 20]. We use a Student's t-distribution with degree of freedom $v = 4$ and set data dimension $d = 100$ (Appendix J). We tune each method separately to a 0.25 target acceptance rate. To measure efficiency, we record effective sample size (ESS) per second—a common MCMC metric for quantifying the number of effectively independent samples a method can draw from the posterior each second [9]. Figure 2a shows TunaMH is the most efficient for all dataset sizes N; it has the largest ESS/second. For minibatch MH methods, Figure 2b compares the average batch size. TunaMH's batch size is significantly smaller than FlyMC's—about 35x with $N = 10^5$. TFMH has the smallest batch size, but this is because it uses a very small step size to reach the target acceptance rate (Table 2 in Appendix J.2). This leads to poor efficiency, which we can observe in its low ESS/second.

MAP variants Since TFMH and FlyMC have variants that use the *maximum a posteriori* (MAP) solution to boost performance, we also test TunaMH in this scheme. SMH uses MAP to construct control variates for TFMH to improve low acceptance rates. We consider both first- and second-order approximations (SMH-1 and SMH-2). FlyMC uses MAP to tighten the lower bound (FlyMC-MAP). For our method (TunaMH-MAP) and MH (MH-MAP), we simply initialize the chain with the MAP solution. Figure 2c shows that TunaMH performs the best even when previous methods make use of MAP. With control variates, SMH does increase the acceptance rate of TFMH, but this comes at the cost of a drastically increased batch size (Figure 2d) which we conjecture is due to the control variates scaling poorly in high dimensions ($d = 100$).[4] FlyMC-MAP tightens the bounds, entailing a decrease in the batch size. However, as clear in the difference in ESS/second, it is still less efficient than TunaMH due to its strong dependence between auxiliary variables and the model parameters—an issue that previous work also documents [24].

5.2 Truncated Gaussian Mixture

Next we test on a task with a multimodal posterior, a very common problem in machine learning. This demonstrates the advantage of TunaMH not relying on MAP, because MAP is a single solution and therefore is unable to reflect all possible modes in multimodal distributions. As a result, methods that rely on MAP tuning or MAP-based control variates are unable to perform well on such problems.

We consider a Gaussian mixture. To get bounds on TunaMH, TFMH, SMH, and FlyMC, we truncate the posterior, bounding $\theta_1, \theta_2 \in [-3, 3]$ similar to Zhang and De Sa [27]. We can include PoissonMH because its required bound exists after truncation. As in Seita et al. [26], we use a tempered posterior $\pi(\theta) \propto \exp\left(-\beta \sum_i U_i(\theta)\right)$ with $N = 10^6$ and $\beta = 10^{-4}$. Figure 3a compares performance, showing symmetric KL versus wall-clock time. TunaMH is the fastest, converging after 1 second, whereas the others take much longer. As expected, SMH-1 performs worse than TFMH, verifying the control variate is unhelpful for multimodal distributions. FlyMC and FlyMC-MAP are also inefficient; their performance is on par with standard MH, indicating negligible benefits from minibatching.

[4] Control variates worked well in the SMH paper [12] because all experiments had small dimension ($d = 10$).

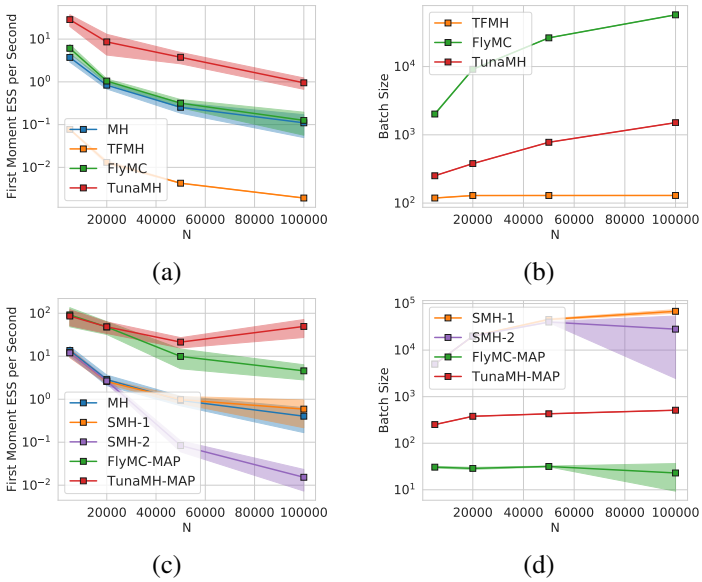


Figure 2: Robust linear regression, $d = 100$. (a) ESS/second without MAP. (b) Average batch size without MAP. (c) ESS/second with MAP. (d) Average batch size with MAP.

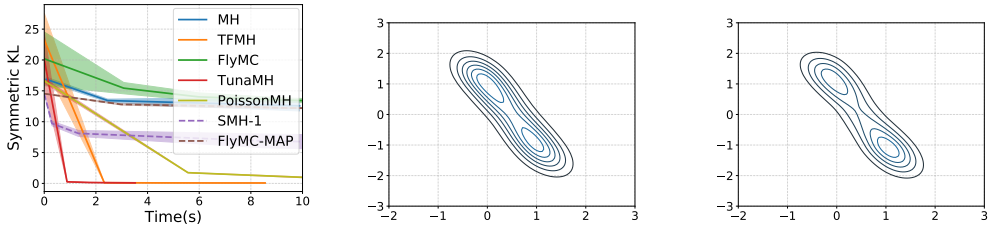


Figure 3: Truncated Gaussian mixture. (a) Symmetric KL comparison. (b) True distribution. (c) Denstity estimate of TunaMH after 1 second.

TunaMH also performs significantly better in terms of batch size, especially in comparison to PoissonMH (Table 1). This is due to TunaMH's local bound on the energy, as opposed to PoissonMH's global bound. This also allows TunaMH to run on more problem types, such as robust linear (Section 5.1) and logistic (Section 5.3) regression. To illustrate the estimate quality, we also visualize the density estimate after 1 second; TunaMH's estimate (Figure 3c) is very close to the true distribution (Figure 3b), while the other methods do not provide on-par estimates within the same time budget (Appendix J.3).

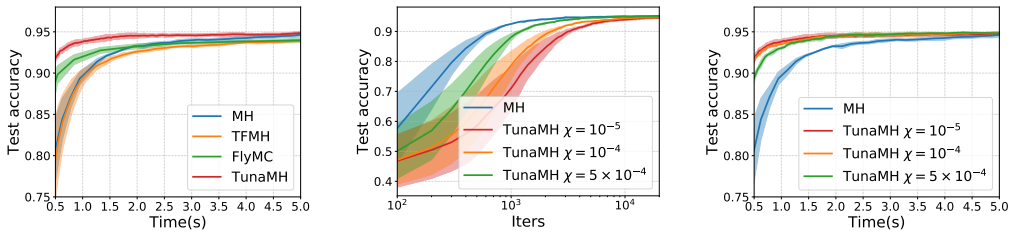
5.3 Logistic Regression on MNIST

Lastly we apply TunaMH to logistic regression on the MNIST image dataset of handwritten number digits. Mirroring the work of FlyMC [20], we aim to classify 7s and 9s using the first 50 principal components as features. We set $\chi = 10^{-5}$ following our heuristic. In Figure 4a we see that TunaMH is the fastest of all methods to converge, as measured by wall-clock time. We also compare average batch size in Table 1. TunaMH's average batch size is 4x smaller than FlyMC's. TFMH again has the smallest batch size, but sacrifices efficiency by using a small step size in order to achieve the target acceptance rate. Thus, overall, TFMH is again inefficient in these experiments.

Effect of Hyperparameter χ To understand the effect of χ in TunaMH, we report results with varying χ. Figure 4b plots test accuracy as a function of the number of iterations. As χ increases, TunaMH's convergence rate approaches standard MH. This verifies our theoretical work: χ acts like a dial to control convergence rate and batch size trade-off—mapping to the efficiency-scalability

Table 1: Avg. batch size $\pm$ SE from the mean on 3 runs. PoissonMH not applicable to logistic reg.

Tasks	TFMH	FlyMC	PoissonMH	TunaMH
Gaussian Mixture	13.91 ± 0.016	811.52 ± 234.16	3969.67 ± 327.26	86.45 ± 0.04
Logistic Regression	39.28 ± 0.12	1960.19 ± 150.96	—	504.07 ± 0.33



(a) (b) (c)

Figure 4: MNIST logistic regression. (a) Test accuracy comparison. (b)-(c) TunaMH's test accuracy for various χ. Batch size for $\chi = 10^{-5}, 10^{-4}, 5 \times 10^{-4}$ is 504.07, 810.35 and 2047.91 respectively.

trade-off. Figure 4c shows TunaMH's wall-clock time performance is not sensitive to χ, as the performance is superior to standard MH regardless of how we set it. However, χ needs to be tuned in order to achieve the best performance. Previous methods do not have such a dial, so they are unable to control this trade-off to improve the sampling efficiency.

6 Conclusion and Future Work

After demonstrating that inexact methods can lead to arbitrarily incorrect inference, we focus our work in this paper on exact minibatch MH methods. We propose a new exact method, TunaMH, which lets users trade off between batch size and guaranteed convergence rate—between scalability and efficiency. We prove a lower bound on the batch size that any minibatch MH method must use to maintain exactness and convergence rate, and show TunaMH is asymptotically optimal. Our experiments validate these results, demonstrating that TunaMH outperforms state-of-the-art exact methods, particularly on high-dimensional and multimodal distributions.

To guide our analysis, we formalized a class of stateless, energy-difference-based minibatch MH methods, to which most prior methods belong. While TunaMH is asymptotically optimal for this class, future work could develop new exact methods that are better by a constant factor or on some restricted class of distributions. It would also be interesting to develop effective theoretical tools for analyzing stateful methods, since these methods could potentially bypass our lower bound.

Broader Impact

Our work shines a light on how to scale MCMC methods responsibly. We make the case that inexact minibatch MH methods can lead to egregious errors in inference, which suggests that—particularly for high-impact applications [14, 22]—we should avoid their use. We provide an alternative: a minibatch MH method that guarantees correctness, while also maintaining an optimal balance between efficiency and scalability, enabling its safe use on large-scale applications.

Acknowledgements

This work was supported by a gift from SambaNova Systems, Inc. and funding from Adrian Sampson. We thank Jerry Chee, Yingzhen Li, and Wing Wong for helpful feedback on the manuscript.

References

[1] José A Adell and Pedro Jodrá. Exact Kolmogorov and total variation distances between some familiar discrete distributions. *Journal of Inequalities and Applications*, 2006(1):64307, 2006.

[2] Shigeki Aida. Uniform positivity improving property, sobolev inequalities, and spectral gaps. *Journal of functional analysis*, 158(1):152–185, 1998.

[3] Christophe Andrieu and Gareth O. Roberts. The pseudo-marginal approach for efficient Monte Carlo computations. *The Annals of Statistics*, 37(2):697–725, 2009.

[4] Marco Banterle, Clara Grazian, Anthony Lee, and Christian P Robert. Accelerating Metropolis-Hastings algorithms by delayed acceptance. *Foundations of Data Science*, 1:103, 2019.

[5] Rémi Bardenet, Arnaud Doucet, and Chris Holmes. On Markov chain Monte Carlo methods for tall data. *The Journal of Machine Learning Research*, 18(1):1515–1557, 2017.

[6] Rémi Bardenet, Arnaud Doucet, and Chris Holmes. Towards scaling up Markov chain Monte Carlo: an adaptive subsampling approach. In *International Conference on Machine Learning*, 2014.

[7] Joris Bierkens, Paul Fearnhead, Gareth Roberts, et al. The zig-zag process and super-efficient sampling for Bayesian analysis of big data. *The Annals of Statistics*, 47(3):1288–1320, 2019.

[8] Alexandre Bouchard-Côté, Sebastian J Vollmer, and Arnaud Doucet. The bouncy particle sampler: A nonreversible rejection-free Markov chain Monte Carlo method. *Journal of the American Statistical Association*, 113(522):855–867, 2018.

[9] Steve Brooks, Andrew Gelman, Galin Jones, and Xiao-Li Meng. *Handbook of Markov Chain Monte Carlo*. CRC Press, 2011.

[10] David M Ceperley. Path integrals in the theory of condensed helium. *Reviews of Modern Physics*, 67(2):279, 1995.

[11] J Andrés Christen and Colin Fox. Markov chain monte carlo using an approximation. *Journal of Computational and Graphical statistics*, 14(4):795–810, 2005.

[12] Robert Cornish, Paul Vanetti, Alexandre Bouchard-Côté, George Deligiannidis, and Arnaud Doucet. Scalable Metropolis-Hastings for exact Bayesian inference with large datasets. *International Conference on Machine Learning*, 2019.

[13] Masatoshi Fukushima, Yoichi Oshima, and Masayoshi Takeda. *Dirichlet forms and symmetric Markov processes*, volume 19. Walter de Gruyter, 2010.

[14] Andrew Gelman, Alex Kiss, and Jeffrey Fagan. An Analysis of the New York City Police Department's Stop-and-Frisk Policy in the Context of Claims of Racial Bias. *Journal of the American Statistical Association*, 102(479):813–823, 2007.

[15] Martin Hairer, Andrew M Stuart, Sebastian J Vollmer, et al. Spectral gaps for a Metropolis–Hastings algorithm in infinite dimensions. *The Annals of Applied Probability*, 24(6):2455–2490, 2014.

[16] W. Keith Hastings. Monte Carlo sampling methods using Markov chains and their applications. 1970.

[17] Anoop Korattikara, Yutian Chen, and Max Welling. Austerity in MCMC land: Cutting the Metropolis-Hastings budget. In *International Conference on Machine Learning*, pages 181–189, 2014.

[18] David A Levin and Yuval Peres. *Markov chains and mixing times*, volume 107. American Mathematical Society, 2017.

[19] PA W Lewis and Gerald S Shedler. Simulation of nonhomogeneous Poisson processes by thinning. *Naval research logistics quarterly*, 26(3):403–413, 1979.

[20] Dougal Maclaurin and Ryan Prescott Adams. Firefly Monte Carlo: Exact MCMC with subsets of data. In *Twenty-Fourth International Joint Conference on Artificial Intelligence*, 2015.

[21] 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.

[22] Emma Pierson, Sam Corbett-Davies, and Sharad Goel. Fast threshold tests for detecting discrimination. volume 84 of *Proceedings of Machine Learning Research*, pages 96–105, Playa Blanca, Lanzarote, Canary Islands, 09–11 Apr 2018. PMLR. URL `http://proceedings.mlr.press/v84/pierson18a.html`.

[23] Matias Quiroz, Minh-Ngoc Tran, Mattias Villani, Robert Kohn, and Khue-Dung Dang. The block-poisson estimator for optimally tuned exact subsampling mcmc. *arXiv preprint arXiv:1603.08232*, 2016.

[24] Matias Quiroz, Robert Kohn, Mattias Villani, and Minh-Ngoc Tran. Speeding up MCMC by efficient data subsampling. *Journal of the American Statistical Association*, 114(526):831–843, 2019.

[25] Daniel Rudolf. Explicit error bounds for markov chain monte carlo. *arXiv preprint arXiv:1108.3201*, 2011.

[26] Daniel Seita, Xinlei Pan, Haoyu Chen, and John Canny. An efficient minibatch acceptance test for Metropolis-Hastings. *Uncertainty in Artificial Intelligence*, 2017.

[27] Ruqi Zhang and Christopher M De Sa. Poisson-Minibatching for Gibbs Sampling with Convergence Rate Guarantees. In *Advances in Neural Information Processing Systems*, pages 4923–4932, 2019.

[28] Ruqi Zhang, A Feder Cooper, and Christopher De Sa. AMAGOLD: Amortized Metropolis adjustment for efficient stochastic gradient MCMC. *International Conference on Artificial Intelligence and Statistics*, 2020.

A Proof of Theorem 1

In this section, we prove Theorem 1, which asserts that any inexact stateless MH algorithm can produce arbitrarily large bias between its target distribution (the distribution we are trying to sample from) and its stationary distribution (the distribution that the chain actually produces samples from asymptotically).

Proof. Let $\mathcal{A}$ denote the `SubsMH` in Algorithm 1 of the minibatch MH method in question. Since $\mathcal{A}$ is inexact, there must exist a state space Θ, proposal distribution q, and target distribution μ, satisfying Assumption 1 with parameters $c_1, \ldots, c_N, C, M$, where

$$\mu(\theta) \propto \exp\left(-\sum_{i=1}^{N} V_i(\theta)\right)$$

for some N and energy functions $V_1, \ldots, V_N$, such that $\mathcal{A}$ run on μ with proposal distribution q does not have stationary distribution μ.

Next, let $a_\mu(\theta, \theta')$ denote the acceptance probability of algorithm $\mathcal{A}$ on the above task for a proposed transition from θ to θ'. Assume by way of contradiction that on this problem, it is always true that

$$\frac{a_\mu(\theta, \theta')}{a_\mu(\theta', \theta)} = \frac{\mu(\theta')q(\theta|\theta')}{\mu(\theta)q(\theta'|\theta)}.$$

If this were true, then the overall transition probability of this chain, for $\theta \neq \theta'$, would be

$$T_\mu(\theta, \theta') = q(\theta'|\theta) \cdot a_\mu(\theta, \theta')$$

and it would hold that

$$\mu(\theta)T_\mu(\theta, \theta') = \mu(\theta')T_\mu(\theta', \theta).$$

That is, the chain would be reversible, also known as satisfying detailed balance. But it is a standard result that for any reversible chain, μ must be a stationary distribution of that chain. We have now derived a contradiction, which establishes that our assumption is false. That is, there exists a $\theta, \theta' \in \Theta$ such that

$$\frac{a_\mu(\theta, \theta')}{a_\mu(\theta', \theta)} \neq \frac{\mu(\theta') \cdot q(\theta|\theta')}{\mu(\theta) \cdot q(\theta'|\theta)}.$$

Explicitly, this means that if we define the function ΔV such that

$$\Delta V(i) = V_i(\theta) - V_i(\theta'),$$

then for this subsampling problem,

$$\frac{\mathbf{E}\left[\mathcal{A}(\Delta V, N, q(\theta|\theta')/q(\theta'|\theta), c_1, \ldots, c_N, C, M(\theta, \theta'))\right]}{\mathbf{E}\left[\mathcal{A}(-\Delta V, N, q(\theta'|\theta)/q(\theta|\theta'), c_1, \ldots, c_N, C, M(\theta, \theta'))\right]} \neq \frac{\mu(\theta') \cdot q(\theta|\theta')}{\mu(\theta) \cdot q(\theta'|\theta)}. \tag{4}$$

Without loss of generality, assume that

$$q(\theta|\theta')/q(\theta'|\theta) \leq 1.$$

(This is without loss of generality since we can ensure it is the case by swapping θ and θ'.) We fixed θ and θ' to be the pair satisfying Equation 4 throughout this section.

Constructing an example. We use this to prove the theorem by a constructive example. Let $x_1, \ldots, x_N$ be defined by

$$x_i = \Delta V(i) = V_i(\theta) - V_i(\theta').$$

Define X as the sum

$$X = \sum_{i=1}^{N} x_i.$$

For some parameter $K \in \mathbb{N}$ (to be defined later), consider the state space Ω defined as

$$\Omega = \{(k, z) \mid k \in \{0, \ldots, K-1\},\ 0 \leq z \leq \exp(kX)\},$$

using the natural measure for a finite disjoint union of measure spaces. Define a target distribution over Ω given by the density

$$\pi(k, z) \propto \exp\left(-\sum_{i=1}^{N} k \cdot x_i\right),$$

or equivalently

$$\pi(k, z) \propto \exp\left(-\sum_{i=1}^{N} U_i(k, z)\right) \text{ where } U_i(k, z) = kx_i.$$

Define a proposal distribution $\hat{q}$, such that, starting from (k, z):

- With probability $1/4$, we sample z' uniformly from $[0, \exp(kX)]$ and propose a transition to (k, z').
- With probability $1/4$, we propose a transition to $(k-1, z)$, if it is in Ω.
- With probability $\frac{1}{4} \cdot \frac{q(\theta|\theta')}{q(\theta'|\theta)}$, we propose a transition to $(k+1, z)$, if it is in Ω.
- With the remaining probability, we just propose to stay at (k, z).

This is effectively acting as a random walk over k, and our goal will be to show that while the true target distribution π has a marginal in k that is the uniform distribution, the minibatch MH method causes the chain's transition to be biased to step more in one direction than another, resulting in a highly biased stationary distribution (where we can make the bias arbitrarily large by setting K).

We use the same c_i and C as before, and define a new function $\hat{M}$ such that

$$\hat{M}((k, z), (k+1, z)) = \hat{M}((k, z), (k-1, z)) = M(\theta, \theta')$$

and $\hat{M}(\cdots) = 0$ for other proposed transitions (we can set $\hat{M}$ however we want for pairs of states that are never proposed in a transition, since this will not affect the algorithm). Clearly, this setup satisfies Assumption 1, since the original distribution did.

Now, consider what our minibatch MH method will do when run on this task. There are three cases to consider.

Proposed changes in z. When a proposed change in z is made, the resulting ΔU will be uniformly 0, and the probability of the reverse transition will be equal (1/4 in both directions), so the algorithm will be passed the arguments

$$\mathcal{A}(0, N, 1, c_1, \ldots, c_N, C, 0).$$

Since this does not depend at all on z or k, this means that the acceptance probability of these transitions will be the same regardless of the state. Call this probability α_0.

A proposal to decrease k. When a proposal is made to decrease k, the probability of the forward and reverse transitions will be

$$\hat{q}((k-1, z)|(k, z)) = \frac{1}{4} \text{ and } \hat{q}((k, z)|(k-1, z)) = \frac{1}{4} \cdot \frac{q(\theta|\theta')}{q(\theta'|\theta)}.$$

It follows that

$$\frac{\hat{q}((k, z)|(k-1, z))}{\hat{q}((k-1, z)|(k, z))} = \frac{q(\theta|\theta')}{q(\theta'|\theta)}.$$

The energy function difference for this proposal will be

$$\Delta U(i) = U_i((k, z)) - U_i((k-1, z)) = kx_i - (k-1)x_i = x_i,$$

so in particular $\Delta U = \Delta V$. And, of course for this transition $\hat{M}$ will take on the value $M(\theta, \theta')$. So, the minibatch MH algorithm will be passed the arguments

$$\mathcal{A}(\Delta V, N, q(\theta|\theta')/q(\theta'|\theta), c_1, \ldots, c_N, C, M(\theta, \theta')),$$

and so it will accept with probability

$$\mathbf{E}\left[\mathcal{A}(\Delta V, N, q(\theta|\theta')/q(\theta'|\theta), c_1, \ldots, c_N, C, M(\theta, \theta'))\right].$$

Call this probability α_-.

A proposal to increase k. When a proposal is made to increase k, the probability of the forward and reverse transitions will be

$$\hat{q}((k+1,z)|(k,z)) = \frac{1}{4} \cdot \frac{q(\theta|\theta')}{q(\theta'|\theta)}. \text{ and } \hat{q}((k,z)|(k+1,z)) = \frac{1}{4}.$$

It follows that

$$\frac{\hat{q}((k,z)|(k+1,z))}{\hat{q}((k+1,z)|(k,z))} = \frac{q(\theta'|\theta)}{q(\theta|\theta')}.$$

The energy function difference for this proposal will be

$$\Delta U(i) = U_i((k,z)) - U_i((k+1,z)) = kx_i - (k+1)x_i = -x_i,$$

so in particular $\Delta U = -\Delta V$. And, as before for this transition $\hat{M}$ will take on the value $M(\theta,\theta')$. So, the minibatch MH algorithm will be passed the arguments

$$\mathcal{A}(-\Delta V, N, q(\theta'|\theta)/q(\theta|\theta'), c_1, \ldots, c_N, C, M(\theta,\theta')),$$

and so it will accept with probability

$$\mathbf{E}\left[\mathcal{A}(-\Delta V, N, q(\theta'|\theta)/q(\theta|\theta'), c_1, \ldots, c_N, C, M(\theta,\theta'))\right].$$

Define the probability α_+ as

$$\alpha_+ = \mathbf{E}\left[\mathcal{A}(-\Delta V, N, q(\theta'|\theta)/q(\theta|\theta'), c_1, \ldots, c_N, C, M(\theta,\theta'))\right] \cdot \frac{q(\theta|\theta')}{q(\theta'|\theta)}.$$

The resulting Markov chain. From the above analysis, we can conclude that the Markov chain that results from subsampling algorithm $\mathcal{A}$ applied to this method is as follows. Starting from (k,z), if we let $\hat{T}$ denote the transition operator of this Markov chain,

- With probability $\frac{1}{4} \cdot \alpha_0$, we sample z' uniformly from $[0, \exp(kX)]$ and transition to (k,z').
- With probability $\frac{1}{4} \cdot \alpha_-$, we transition to $(k-1,z)$, if it is in Ω.
- With probability $\frac{1}{4} \cdot \alpha_+$, we transition to $(k+1,z)$, if it is in Ω.
- With the remaining probability, we just stay at (k,z).

Consider the distribution

$$\nu(k,z) \propto \left(\frac{\alpha_+}{\alpha_-}\right)^k.$$

It is easy to see that this Markov chain satisfies detailed balance with ν as its stationary distribution. In particular,

$$\begin{aligned}
\nu(k,z) \cdot T((k-1,z)|(k,z)) &= \left(\frac{\alpha_+}{\alpha_-}\right)^k \cdot \frac{1}{4} \cdot \alpha_- \\
&= \left(\frac{\alpha_+}{\alpha_-}\right)^{k-1} \cdot \frac{1}{4} \cdot \alpha_+ \\
&= \nu(k-1,z) \cdot T((k,z)|(k-1,z)).
\end{aligned}$$

So ν will be a stationary distribution of the minibatch MH chain $\hat{T}$.

Observe that the marginal distribution of k in π is

$$\pi(k) = \int_0^{\exp(kX)} \pi(k,z)\, dz \propto \exp\left(-\sum_{i=1}^N k \cdot x_i\right) \cdot \exp(kX) = 1,$$

so the marginal distribution of k in the target distribution is actually the uniform distribution. On the other hand, using the same derivation, the marginal distribution of k in ν is

$$\nu(k) \propto \left(\frac{\alpha_+}{\alpha_-}\right)^k \cdot \exp(kX) = \left(\frac{\alpha_+}{\alpha_-} \cdot \exp(X)\right)^k.$$

We know immediately by substituting our definitions of α_+ and α_- into (4) that

$$\frac{\alpha_-}{\alpha_+} \neq \frac{\mu(\theta')}{\mu(\theta)} = \exp\left(\sum_{i=1}^{N}(V_i(\theta) - V_i(\theta')\right) = \exp\left(\sum_{i=1}^{N} x_i\right) = \exp(X).$$

As a consequence, we know that

$$\frac{\alpha_+}{\alpha_-} \cdot \exp(X) \neq 1.$$

Call this constant

$$A = \frac{\alpha_+}{\alpha_-} \cdot \exp(X),$$

and observe that $A \neq 1$ and that A is independent of our choice of K (which still remains unset). This gives

$$\nu(k) \propto A^k.$$

Explicitly, this distribution will be

$$\nu(k) = \frac{1}{\sum_{k=0}^{K-1} A^k} \cdot A^k = \frac{1-A}{1-A^K} \cdot A^k.$$

Since the total variation distance between two probability measures is lower bounded by the TV-distance between their marginal distributions in any one variable, and similarly the KL divergence is *also* lower bounded by the KL divergence between its marginal distributions in any one variable (both these facts follow directly from the monotonicity property of the f-divergence, of which the KL-divergence and TV-distance are both instances), to prove this theorem it suffices to show both TV-distance and KL-divergence bounds on the marginal distributions in k. We do this now.

Bounding the total variation distance. Now, we compute the total variation distance between π and ν. For this bit of the proof, we will just consider the marginal distribution in k, as this provides a lower bound on the TV distance between the joint distribution. For simplicity, for the rest of the proof, we let $\tilde{\pi}$ denote this marginal distribution of k in ν, and also let π denote the marginal distribution of k in π. By the definition of total variation distance,

$$\begin{aligned}\mathrm{TV}(\pi, \tilde{\pi}) &= \frac{1}{2}\sum_{k=0}^{K-1} |\tilde{\pi}(k) - \pi(k)| \\ &= \frac{1}{2}\sum_{k=0}^{K-1} \left|\frac{1-A}{1-A^K} \cdot A^k - \frac{1}{K}\right|.\end{aligned}$$

If $A < 1$,

$$\begin{aligned}\mathrm{TV}(\pi, \tilde{\pi}) &= \sum_{k=0}^{K_0} \left(\frac{1-A}{1-A^K} \cdot A^k - \frac{1}{K}\right) \\ &= \frac{1-A^{K_0}}{1-A^K} - \frac{K_0}{K} \end{aligned} \tag{5}$$

where K_0 is the largest k such that

$$\frac{1-A}{1-A^K} \cdot A^k > \frac{1}{K}.$$

By solving the above equation, we have

$$K_0 = \left\lfloor \frac{\log(1-A^K) - \log(1-A) - \log(K)}{\log(A)} \right\rfloor.$$

We can lower bound K_0 by

$$\begin{aligned}K_0 &\geq \frac{\log(1-A^K) - \log(1-A) - \log(K)}{\log(A)} - 1 \\ &\geq \frac{-\log(1-A) - \log(K)}{\log(A)} - 1.\end{aligned}$$

It follows that the first term in (5) becomes

$$\frac{1-A^{K_0}}{1-A^K} \geq \frac{1-\frac{1}{KA(1-A)}}{1-A^K} \geq 1-\frac{1}{KA(1-A)}.$$

We can also upper bound K_0 and then the second term can be bounded as the following

$$\frac{K_0}{K} \leq \frac{\log(1-A^K)-\log(K)}{K\log(A)}.$$

When $K \geq \frac{\log\left(1-\exp\left(-\frac{1}{2}\right)\right)}{\log(A)}$, we have $\log(1-A^K) \geq -\frac{1}{2}$. Since $\log(K) \leq K^{\frac{1}{2}}$ and $K^{-1} \leq K^{-\frac{1}{2}}$, we have

$$\frac{K_0}{K} \leq \frac{-\frac{1}{2}K^{-1}-K^{-\frac{1}{2}}}{\log(A)} \leq -\left(\frac{3}{2\log(A)}\right)K^{-\frac{1}{2}}.$$

Therefore, the TV distance is bounded by

$$\begin{aligned}\mathrm{TV}(\pi,\tilde{\pi}) &\geq 1-\frac{1}{KA(1-A)}+\left(\frac{3}{2\log(A)}\right)K^{-\frac{1}{2}} \\ &\geq 1+\left(\frac{3}{2\log(A)}-\frac{1}{A(1-A)}\right)K^{-\frac{1}{2}}.\end{aligned}$$

To make $\mathrm{TV}(\pi,\tilde{\pi}) \geq \delta$, we just need to set

$$K \geq \frac{\left(\frac{3}{2\log(A)}-\frac{1}{A(1-A)}\right)^2}{(1-\delta)^2}.$$

Similarly, if $A > 1$,

$$\begin{aligned}\mathrm{TV}(\pi,\tilde{\pi}) &= \sum_{k=K_0}^{K-1}\left(\frac{1-A}{1-A^K}\cdot A^k-\frac{1}{K}\right) \\ &= \frac{A^K-A^{K_0}}{A^K-1}-\frac{K-K_0}{K} \\ &= \frac{K_0}{K}-\frac{A^{K_0}-1}{A^K-1}\end{aligned}$$

where

$$K_0 = \left\lceil \frac{\log(A^K-1)-\log(A-1)-\log(K)}{\log(A)} \right\rceil$$

which is the smallest k such that

$$\frac{1-A}{1-A^K}\cdot A^k > \frac{1}{K}.$$

We can get an upper bound of K_0 by

$$\begin{aligned}K_0 &\leq \frac{\log(A^K-1)-\log(A-1)-\log(K)}{\log(A)}+1 \\ &= \log_A\left(\frac{A^K-1}{K(A-1)}\right)+1.\end{aligned}$$

Therefore,

$$\begin{aligned}\frac{A^{K_0}-1}{A^K-1} &\leq \frac{A\cdot\left(\frac{A^K-1}{K(A-1)}\right)-1}{A^K-1} \\ &= \frac{A}{K(A-1)}-\frac{1}{A^K-1}.\end{aligned}$$

We can lower bound K_0 by

$$K_0 \geq \log_A \left(A^K - 1\right) - \log_A(A-1) - \log_A(K).$$

When $K \geq 1 - \log_A(A-1)$, $A^K - 1 \geq A^{K-1}$. Then we have

$$\begin{aligned} K_0 &\geq \log_A \left(A^{K-1}\right) - \log_A(A-1) - \log_A(K) \\ &= K - 1 - \log_A(A-1) - \log_A(K). \end{aligned}$$

It follows that

$$\frac{K_0}{K} \geq 1 - \frac{1}{K} - \frac{\log_A(A-1)}{K} - \frac{\log_A(K)}{K}.$$

Since $\log(K) \leq K^{\frac{1}{2}}$ and $K^{-1} \leq K^{-\frac{1}{2}}$, the TV distance can be bounded by

$$\begin{aligned} \text{TV}(\pi, \tilde{\pi}) &\geq 1 - \frac{1}{K} - \frac{\log_A(A-1)}{K} - \frac{\log_A(K)}{K} - \frac{A}{K(A-1)} + \frac{1}{A^K - 1} \\ &\geq 1 - \left(1 + \log_A(A-1) + \frac{1}{\log(A)} + \frac{A}{A-1}\right) K^{-\frac{1}{2}}. \end{aligned}$$

To make $\text{TV}(\pi, \tilde{\pi}) \geq \delta$, we just need

$$K \geq \left(\frac{1 + \log_A(A-1) + \frac{1}{\log(A)} + \frac{A}{A-1}}{1-\delta}\right)^2.$$

Since we could set K arbitrarily, it is clear that we can do this.

Bounding the KL divergence. We can compute KL divergence between π and $\tilde{\pi}$ as follows

$$\begin{aligned} \text{KL}(\pi, \tilde{\pi}) &= \sum_{k=0}^{K-1} \frac{1}{K} \cdot \log\left(\frac{1}{K} \cdot \frac{1 - A^K}{(1-A)A^k}\right) \\ &= \frac{1}{K} \cdot \sum_{k=0}^{K-1} \left[\log\left(\frac{1}{K} \cdot \frac{1 - A^K}{(1-A)}\right) - k\log(A)\right] \\ &= \log\left(\frac{1 - A^K}{K(1-A)}\right) - \frac{\log(A)}{K} \sum_{k=0}^{K-1} k \\ &= \log\left(\frac{1 - A^K}{K(1-A)}\right) - \frac{(K-1)\log(A)}{2} \end{aligned}$$

If $A < 1$, we have

$$\begin{aligned} \text{KL}(\pi, \tilde{\pi}) &= \log\left(1 - A^K\right) - \log((1-A)K) - \frac{K\log(A)}{2} + \frac{\log(A)}{2} \\ &\geq \log\left(1 - A^K\right) - \left(\frac{1 - A + \log(A)}{2}\right) K + \frac{\log(A)}{2}. \end{aligned}$$

The last equation is because $\log(x) \leq \frac{x}{2}$.

To further simplify the above equation, we first note that $1 - A + \log(A) < 0$ when $A \neq 1$. And then when $K \geq \log_A\left(1 - A^{\frac{1}{2}}\right)$, we have $1 - A^K \geq A^{\frac{1}{2}}$. It follows that we can simplify it to be

$$\text{KL}(\pi, \tilde{\pi}) \geq \log(A) - \left(\frac{1 - A + \log(A)}{2}\right) K.$$

To make $\text{KL}(\pi, \tilde{\pi}) \geq \rho$, it is clear that we just need to set

$$K \geq \frac{2(\rho - \log(A))}{A - 1 - \log(A)}.$$

Consider when $A > 1$,

$$\mathrm{KL}(\pi, \tilde{\pi}) = \log\left(\frac{A^K - 1}{K(A-1)}\right) - \frac{(K-1)\log(A)}{2}.$$

If $K \geq \frac{\log(2)}{\log(A)}$, we have that $A^K - 1 \geq \frac{A^K}{2}$. It follows that

$$\begin{aligned}\mathrm{KL}(\pi, \tilde{\pi}) &\geq K\log(A) - \log(K) - \log(2A-2) - \frac{K\log(A)}{2} \\ &= \frac{K\log(A)}{2} - \log(K) - \log(2A-2).\end{aligned}$$

To make $\mathrm{KL}(\pi, \tilde{\pi}) \geq \rho$, we need

$$\frac{K\log(A)}{2} - \log(K) \geq \rho + \log(2A-2).$$

Let $K = \exp(y)$. By Taylor series, we know $\exp(y) \geq \frac{y^2}{2}$. Then it follows that

$$\frac{y^2\log(A)}{4} - y \geq \rho + \log(2A-2).$$

Solve the above inequality, we can get

$$y \geq \frac{1 + 2\cdot\frac{\log(A)}{4}\cdot\left(\rho + \log(2A-2)\right)}{2\cdot\frac{\log(A)}{4}} = \frac{2 + \log(A)\left(\rho + \log(2A-2)\right)}{\log(A)}.$$

It follows that it suffices to set

$$K \geq \exp\left(\frac{2 + \log(A)\left(\rho + \log(2A-2)\right)}{\log(A)}\right).$$

Concluding the proof. The theorem now follows from choosing a K large enough that both the TV distance inequality we derived and the KL divergence inequality we derived are satisfied. □

B Connection between Theorem 1 and TV Bound of Inexact MH Methods

Some inexact methods such as MHSubLhd [6] have bounded TV distance between the target distribution and the approximate distribution (see Proposition 3.2 in Bardenet et al. [6]). We would like to emphasize that Theorem 1 is compatible with these results. Specifically, Proposition 3.2 assumes P_{MH} has a bounded mixing time. It is well known that this produces a TV bound for any kernel by coupling [18]. Our theorem does not have this assumption; it suggests that for MHSubLhd, with a given user-specified error, there exists a target distribution and proposal satisfying Theorem 1, on which P_{MH} either does not have bounded mixing time or the mixing time is large enough such that the TV bound is greater than δ.

C Proof of Statement 1

Proof. We prove this by construction. Consider a dataset $\{x_i\}_{i=1}^N$. The data instances can take two values $\{-\frac{M}{N}, \frac{M}{N}\}$ where M is a positive constant. Assume that half of the data instances take value $\frac{M}{N}$ and the remaining take $-\frac{M}{N}$. Let the target distribution be $\pi(\theta) = \frac{1}{Z}\exp\left(\theta\cdot\sum_{i=1}^N x_i\right)$ and the domain for θ be $\{0, 1, \ldots, K-1\}$. We define the proposal distribution to be the following

$$p(\theta,\theta) = \frac{1}{2}, \quad \text{for all } \theta; \quad p(\theta,\theta-1) = \frac{1}{4}, \quad p(\theta,\theta+1) = \frac{1}{4} \quad \text{for } \theta \in \{1,\ldots,K-2\};$$

and $p(0,1) = p(K-1,K-2) = \frac{1}{2}$.

Recall that FMH factorizes the target distribution $\pi(\theta)$ and the proposal distribution $p(\theta)$ as follows

$$\pi(\theta) \propto \prod_{i=1}^{m} \pi_i(\theta), \qquad p(\theta) \propto \prod_{i=1}^{m} p_i(\theta)$$

where $m \geq 1$ and π_i and p_i are some non-negative functions. Then the acceptance rate is given by

$$a_{\text{FMH}}(\theta, \theta') = \prod_{i=1}^{m} \min\left(1, \frac{\pi(\theta')p_i(\theta', \theta)}{\pi(\theta)p_i(\theta, \theta')}\right).$$

A common choice is to set $m = N$. On this example, we can write the acceptance rate of transitioning from θ to $\theta' = \theta + 1$ in FMH as follows

$$a_{\text{FMH}}(\theta, \theta') = \prod_{i=1}^{N} \min\left(1, \exp(x_i)\right) = \left(\exp\left(-\frac{M}{N}\right)\right)^{\frac{N}{2}} = \exp\left(-\frac{M}{2}\right).$$

It is easy to show that the acceptance rate of transitioning from θ to $\theta' = \theta - 1$ in FMH is the same.

When $M > -2\log(p)$, it is clear that the acceptance rate of FMH is less than p. By contrast, the acceptance rate of standard MH is

$$a_{\text{MH}}(\theta, \theta') = \min\left(1, \exp\left(\pm \sum_{i=1}^{N} x_i\right)\right) = 1.$$

In order to preserve geometric ergodicity, Cornish et al. [12] introduces *truncated FMH* (TFMH) which forces FMH degrade to standard MH when the energy exceeds a threshold R. If we set hyperparameter $R > M/2$, then in each step, the value of a_{TFMH} will be the same as a_{FMH}. Therefore, if setting $M > -2\log(p)$, we have

$$\frac{a_{\text{TFMH}}}{a_{\text{MH}}} \leq \frac{p}{1} = p.$$

If we set $R \leq M/2$, TFMH falls back to standard, full-batch MH — using the whole dataset at each step. This proves the statement. □

D Construction of Algorithm 2

Algorithm 2 can be derived by carefully replacing the global bounds on the energy in PoissonMH [27] with local bounds on the energy differences (Assumption 1). PoissonMH is a variant of Poisson Gibbs and therefore inherits the same assumptions for Gibbs sampling on graphical models, which are often violated in the applications of MH. In particular, PoissonMH works on *factor graphs* which define a distribution $\pi(\theta)$ over a set of factors $\{\phi_i(\theta)\}_{i=1}^{N}$ as follows

$$\pi(\theta) \propto \exp\left(\sum_{i=1}^{N} \phi_i(\theta)\right).$$

PoissonMH assumes that each factor ϕ_i is non-negative without the loss of generality (we can add a positive constant to ϕ_i to make it non-negative without changing the distribution) and is bounded globally by a constant M_i. That is

$$0 \leq \phi_i(\theta) \leq M_i \text{ for all } \theta.$$

This assumption does not hold for most applications of MH, such as the linear and logistic regression experiments in Section 5.

Let $L = \sum_i M_i$ and define Poisson auxiliary variable s_i as the following

$$s_i|\theta \sim \text{Poisson}\left(\frac{\lambda M_i}{L} + \phi_i(\theta)\right),$$

Algorithm 3 PoissonMH

given: initial state $\theta \in \Theta$; proposal dist. q; hyperparameter λ; Global bounds M_i, L
loop
  propose $\theta' \sim q(\cdot|\theta)$
  for $i \in \{1, \ldots, N\}$ **do**
    sample $s_i \sim \text{Poisson}\left(\frac{\lambda M_i}{L} + \phi_i(\theta)\right)$
  end for
  form minibatch $\mathcal{S} \leftarrow \{i | s_i > 0\}$
  compute MH ratio $r \leftarrow \frac{\exp\left(\sum_{i\in\mathcal{S}} s_i \log\left(1+\frac{L}{\lambda M_i}\phi_i(\theta')\right)\right) q(\theta'|\theta)}{\exp\left(\sum_{i\in\mathcal{S}} s_i \log\left(1+\frac{L}{\lambda M_i}\phi_i(\theta)\right)\right) q(\theta|\theta')}$
  with probability $\min(1, r)$, set $\theta \leftarrow \theta'$
end loop

where $\lambda > 0$ is a hyperparameter. Running standard MH on the joint distribution of θ and s_i results in the following acceptance ratio

$$r_{\text{PoissonMH}}(\theta, \theta') = \frac{\exp\left(\sum_i s_i \log\left(1 + \frac{L}{\lambda M_i}\phi_i(\theta')\right)\right) q(\theta'|\theta)}{\exp\left(\sum_i s_i \log\left(1 + \frac{L}{\lambda M_i}\phi_i(\theta)\right)\right) q(\theta|\theta')}.$$

Here, the sum is essentially performed over the set of index i whose s_i is greater than zero. When $s_i = 0$, it is clear that the factor ϕ_i will not appear in the acceptance ratio $r_{\text{PoissonMH}}$. Thus PoissonMH enables using a subset of factors for the MH decision step (Algorithm 3).

To construct our method from this, we can define the factor ϕ_i in the factor graph to be

$$\phi_i(x) = \frac{U_i(\theta) + U_i(\theta')}{2} - U_i(x) + \frac{c_i}{2} M(\theta, \theta') \tag{6}$$

where $x \in \{\theta, \theta'\}$. It is easy to see that ϕ_i satisfy $0 \le \phi_i(x) \le c_i M(\theta, \theta')$. And then we define the Poisson variables s_i as the follows

$$s_i|(\theta, \theta') \sim \text{Poisson}\left(\frac{\lambda c_i}{C} + \phi_i(\theta)\right) = \text{Poisson}\left(\frac{\lambda c_i}{C} + \frac{U_i(\theta') - U_i(\theta) + c_i M(\theta, \theta')}{2}\right).$$

These Poisson auxiliary variables $\{s_i\}_{i=1}^N$ are called *local*, because their distributions change each iteration depending on the current pair (θ, θ') and only rely on local bounds in Assumption 1. This is in contrast to the *global* auxiliary variables used in PoissonMH and FlyMC which are used to form a joint distribution with θ and both require global bounds in their conditional distributions.

The acceptance ratio r_{TunaMH} is the same as $r_{\text{PoissonMH}}$ but with the new definitions of s_i and ϕ_i. We outline TunaMH using the notation of ϕ_i and s_i in Algorithm 4.

We now show that Algorithm 4 is statistically equivalent to Algorithm 2. To see this, we first use *thinning*, a commonly used technique [7, 8, 12, 19, 27], to quickly resample all s_i from their new distributions in each iteration in Algorithm 4. This is achieved by replacing the global bounds with the local bounds in Algorithm 4 in the Appendix of Zhang and De Sa [27]. Specifically, we first sample B from a Poisson distribution

$$B \sim \text{Poisson}(\lambda + CM(\theta, \theta')).$$

Here $\lambda + CM(\theta, \theta')$ is an upper bound on $\mathbf{E}[\sum_i s_i]$. We then form the minibatch by running

for $b \in \{1, \ldots, B\}$ **do**
  sample i_b such that $\mathbf{P}(i_b = i) = c_i/C$, for $i = 1 \ldots N$
  with probability $\frac{\lambda c_{i_b} + C\phi_{i_b}(\theta)}{\lambda c_{i_b} + C c_{i_b} M(\theta,\theta')}$ **add** i_b to $\mathcal{I}$
end for

By substituting $\lambda = \chi C^2 M^2(\theta, \theta')$ and the expression of ϕ_i, we can get the part of "form minibatch $\mathcal{I}$" in Algorithm 2.

Algorithm 4 TunaMH

given: initial state $\theta \in \Theta$; proposal dist. q; λ; Asm. 1 parameters c_i, C, M; function ϕ_i defined in (6)
loop
  propose $\theta' \sim q(\cdot|\theta)$ and **compute** $M(\theta, \theta')$
  for $i \in \{1, \dots, N\}$ **do**
    sample $s_i \sim \text{Poisson}\left(\frac{\lambda c_i}{C} + \phi_i(\theta)\right)$
  end for
  form minibatch $\mathcal{S} \leftarrow \{i | s_i > 0\}$
  compute MH ratio $r \leftarrow \frac{\exp\left(\sum_{i\in\mathcal{S}} s_i \log\left(1+\frac{C}{\lambda c_i}\phi_i(\theta')\right)\right) q(\theta'|\theta)}{\exp\left(\sum_{i\in\mathcal{S}} s_i \log\left(1+\frac{C}{\lambda c_i}\phi_i(\theta)\right)\right) q(\theta|\theta')}$
  with probability $\min(1, r)$, set $\theta \leftarrow \theta'$
end loop

To see that the MH ratio in Algorithm 2 and 4 are equivalent, we can write out r in Algorithm 4 using the above fast way of resampling s_i

$$r_{\text{TunaMH}} = \frac{\exp\left(\sum_{i\in\mathcal{I}} \log\left(1 + \frac{C}{\lambda c_i}\phi_i(\theta')\right)\right) q(\theta'|\theta)}{\exp\left(\sum_{i\in\mathcal{I}} \log\left(1 + \frac{C}{\lambda c_i}\phi_i(\theta)\right)\right) q(\theta|\theta')}.$$

We then substitute the definition of ϕ_i in (6) and it follows that

$$r_{\text{TunaMH}} = \exp\left(\sum_{i\in\mathcal{I}} \left(\log\left(\frac{2\lambda c_i + C\left(U_i(\theta) - U_i(\theta') + c_i M(\theta,\theta')\right)}{2\lambda c_i + C\left(U_i(\theta') - U_i(\theta) + c_i M(\theta,\theta')\right)}\right)\right)\right) \cdot \frac{q(\theta'|\theta)}{q(\theta|\theta')}.$$

We can rearrange the $\log$ term inside r_{TunaMH} as

$$\begin{aligned}
&\log\left(\frac{2\lambda c_i + C\left(U_i(\theta) - U_i(\theta') + c_i M(\theta,\theta')\right)}{2\lambda c_i + C\left(U_i(\theta') - U_i(\theta) + c_i M(\theta,\theta')\right)}\right) \\
&\quad = \log\left(\frac{2\lambda c_i + C\left(U_i(\theta) - U_i(\theta')\right) + c_i C M(\theta,\theta')}{2\lambda c_i + C\left(U_i(\theta') - U_i(\theta)\right) + c_i C M(\theta,\theta')}\right) \\
&\quad = \log\left(\frac{1 + \frac{C}{2\lambda c_i + c_i C M(\theta,\theta')}\left(U_i(\theta) - U_i(\theta')\right)}{1 + \frac{C}{2\lambda c_i + c_i C M(\theta,\theta')}\left(U_i(\theta') - U_i(\theta)\right)}\right) \\
&\quad = 2\,\text{artanh}\left(\frac{C\left(U_i(\theta) - U_i(\theta')\right)}{c_i(2\lambda + C M(\theta,\theta'))}\right).
\end{aligned}$$

So r_{TunaMH} can be written as

$$r_{\text{TunaMH}} = \exp\left(2\sum_{i\in\mathcal{I}} \text{artanh}\left(\frac{C\left(U_i(\theta) - U_i(\theta')\right)}{c_i(2\lambda + C M(\theta,\theta'))}\right)\right) \cdot \frac{q(\theta'|\theta)}{q(\theta|\theta')}.$$

Finally setting λ to be $\chi C^2 M^2(\theta, \theta')$ produces the MH ratio in Algorithm 2.

By proving the equivalence of the minibatch and the MH ratio, we show that Algorithm 2 and 4 are statistically equivalent.

E Proof of Theorem 2

In this section, we prove Theorem 2, which asserts that TunaMH is reversible and has stationary distribution π, and gives bounds on its spectral gap relative to the spectral gap of the original Metropolis-Hastings algorithm.

Proof. For convenience, we prove Theorem 2 using Algorithm 4 statement which is statistically equivalent to Algorithm 2. The transition operator can be written as the following

$$
\begin{aligned}
&T(\theta, \theta') \\
&= \mathbf{E}\left\{ q(\theta'|\theta) \min\left(1, \frac{q(\theta|\theta') \exp\left(\sum_i \left[s_i \log\left(\frac{\lambda c_i}{C} + \phi_i(\theta')\right) - \log s_i!\right]\right)}{q(\theta'|\theta) \exp\left(\sum_i \left[s_i \log\left(\frac{\lambda c_i}{C} + \phi_i(\theta)\right) - \log s_i!\right]\right)}\right)\right\} \\
&= \mathbf{E}\left\{ q(\theta'|\theta) \min\left(1, \frac{q(\theta|\theta') \exp\left(\sum_i \left[s_i \log\left(\frac{\lambda c_i}{C} + \phi_i(\theta')\right)\right]\right)}{q(\theta'|\theta) \exp\left(\sum_i \left[s_i \log\left(\frac{\lambda c_i}{C} + \phi_i(\theta)\right)\right]\right)}\right)\right\} \\
&= \sum_s \left\{ q(\theta'|\theta) \min\left(1, \frac{q(\theta|\theta') \exp\left(\sum_i \left[s_i \log\left(\frac{\lambda c_i}{C} + \phi_i(\theta')\right)\right]\right)}{q(\theta'|\theta) \exp\left(\sum_i \left[s_i \log\left(\frac{\lambda c_i}{C} + \phi_i(\theta)\right)\right]\right)}\right)\right\} \prod_i p(s_i|\theta, \theta') \\
&= \sum_s \left\{ q(\theta'|\theta) \min\left(\exp\left(\sum_i \left[s_i \log\left(\frac{\lambda c_i}{C} + \phi_i(\theta)\right) - \phi_i(\theta) - \frac{\lambda c_i}{C} - \log s_i!\right]\right),\right.\right. \\
&\qquad \left.\left. \frac{q(\theta|\theta') \exp\left(\sum_i \left[s_i \log\left(\frac{\lambda c_i}{C} + \phi_i(\theta')\right)\right]\right)}{q(\theta'|\theta) \exp\left(\sum_i \phi_i(\theta) + \frac{\lambda c_i}{C} + \log s_i!\right)}\right)\right\} \\
&= \sum_s \left\{ q(\theta'|\theta) \min\left(\exp\left(\sum_i \left[s_i \log\left(\frac{\lambda c_i}{C} + \phi_i(\theta)\right) - \phi_i(\theta) - \frac{\lambda c_i}{C} - \log s_i!\right]\right),\right.\right. \\
&\qquad \left.\left. \frac{q(\theta|\theta')}{q(\theta'|\theta)} \exp\left(\sum_i \left[s_i \log\left(\frac{\lambda c_i}{C} + \phi_i(\theta')\right) - \phi_i(\theta) - \frac{\lambda c_i}{C} - \log s_i!\right]\right)\right)\right\}
\end{aligned}
$$

Multiplying $\pi(\theta)$ to both sides produces

$$
\begin{aligned}
&\pi(\theta) T(\theta, \theta') \\
&\quad = \frac{1}{Z} \exp\left(-\sum_i U_i(\theta)\right) T(\theta, \theta') \\
&\quad = \frac{1}{Z} \sum_s \min\left(q(\theta'|\theta) \exp\left(\sum_i \left[s_i \log\left(\frac{\lambda c_i}{C} + \phi_i(\theta)\right)\right.\right.\right. \\
&\qquad \left.\left.\left. - \frac{U_i(\theta) + U_i(\theta')}{2} - \frac{c_i}{2} M(\theta, \theta') - \frac{\lambda c_i}{C} - \log s_i!\right]\right), \\
&\qquad q(\theta|\theta') \exp\left(\sum_i \left[s_i \log\left(\frac{\lambda c_i}{C} + \phi_i(\theta')\right)\right.\right. \\
&\qquad \left.\left.\left. - \frac{U_i(\theta) + U_i(\theta')}{2} - \frac{c_i}{2} M(\theta, \theta') - \frac{\lambda c_i}{C} - \log s_i!\right]\right)\right)\Bigg).
\end{aligned}
$$

It is clear that the expression is symmetric in θ and θ'. Therefore the chain is reversible and its stationary distribution is $\pi(\theta)$. This proves the first part of the theorem.

To prove the second part of the theorem, the bound on the spectral gap, we continue to reduce the transition probability in the previous proof to

$$
\begin{aligned}
&\pi(\theta)T(\theta,\theta') \\
&\quad= \frac{1}{Z}\sum_s \min\left(q(\theta'|\theta)\exp\left(\sum_i\left[s_i\log\left(\frac{\lambda c_i}{C}+\phi_i(\theta)\right)\right.\right.\right. \\
&\qquad\left.\left.\left.-\frac{U_i(\theta)+U_i(\theta')}{2}-\frac{c_i}{2}M(\theta,\theta')-s_i\log\frac{\lambda c_i}{C}\right]\right), \\
&\qquad q(\theta|\theta')\exp\left(\sum_i\left[s_i\log\left(\frac{\lambda c_i}{C}+\phi_i(\theta')\right)\right.\right. \\
&\qquad\left.\left.\left.-\frac{U_i(\theta)+U_i(\theta')}{2}-\frac{c_i}{2}M(\theta,\theta')-s_i\log\frac{\lambda c_i}{C}\right]\right)\right) \\
&\qquad\cdot\prod_i\frac{1}{s_i!}\exp\left(-\frac{\lambda c_i}{C}\right)\left(\frac{\lambda c_i}{C}\right)^{s_i} \\
&\quad= \frac{1}{Z}\sum_s \min\left(q(\theta'|\theta)\exp\left(\sum_i\left[s_i\log\left(1+\frac{C}{\lambda c_i}\phi_i(\theta)\right)\right.\right.\right. \\
&\qquad\left.\left.\left.-\frac{U_i(\theta)+U_i(\theta')}{2}-\frac{c_i}{2}M(\theta,\theta')\right]\right), \\
&\qquad q(\theta|\theta')\exp\left(\sum_i\left[s_i\log\left(1+\frac{C}{\lambda c_i}\phi_i(\theta')\right)-\frac{U_i(\theta)+U_i(\theta')}{2}-\frac{c_i}{2}M(\theta,\theta')\right]\right)\Bigg) \\
&\qquad\cdot\prod_i\frac{1}{s_i!}\exp\left(-\frac{\lambda c_i}{C}\right)\left(\frac{\lambda c_i}{C}\right)^{s_i}.
\end{aligned}
$$

Note that s_i here are non-negative integers that a Poisson variable can take, not variables. So if we let $r_i \sim \text{Poisson}\left(\frac{\lambda c_i}{C}\right)$ and r_i to be all independent, we can write this as

$$
\begin{aligned}
\pi(\theta)T(\theta,\theta') = \frac{1}{Z}\mathbf{E}\min&\left(q(\theta'|\theta)\exp\left(\sum_i r_i\log\left(1+\frac{C}{\lambda c_i}\phi_i(\theta)\right)\right),\right. \\
&\left.q(\theta|\theta')\exp\left(\sum_i r_i\log\left(1+\frac{C}{\lambda c_i}\phi_i(\theta')\right)\right)\right) \\
&\cdot\exp\left[-\frac{1}{2}\left(\sum_i U_i(\theta)+\sum_i U_i(\theta')+CM(\theta,\theta')\right)\right].
\end{aligned}
$$

Assume $G(\theta,\theta')$ is the transition operator of standard MH. Consider the ratio

$$
\begin{aligned}
&\frac{\pi(\theta)T(\theta,\theta')}{\pi(\theta)G(\theta,\theta')} \\
&\quad= \frac{1}{Z}\mathbf{E}\min\left(q(\theta'|\theta)\exp\left(\sum_i r_i\log\left(1+\frac{C}{\lambda c_i}\phi_i(\theta)\right)\right),\right. \\
&\qquad\left.q(\theta|\theta')\exp\left(\sum_i r_i\log\left(1+\frac{C}{\lambda c_i}\phi_i(\theta')\right)\right)\right) \\
&\qquad\cdot\exp\left[-\frac{1}{2}\left(\sum_i U_i(\theta)+\sum_i U_i(\theta')+CM(\theta,\theta')\right)\right] \\
&\qquad\cdot\left[1\Big/\left(\frac{1}{Z}\min\left(q(\theta'|\theta)\exp\left(-\sum_i U_i(\theta)\right),q(\theta|\theta')\exp\left(-\sum_i U_i(\theta')\right)\right)\right)\right].
\end{aligned}
$$

We know that $\frac{\min(A,B)}{\min(C,D)} = \min\left(\frac{A}{\min(C,D)}, \frac{B}{\min(C,D)}\right) \geq \min\left(\frac{A}{C}, \frac{B}{D}\right)$. The last inequality is due to the fact that $\frac{1}{\min(C,D)} \geq \frac{1}{C}$ and $\frac{1}{\min(C,D)} \geq \frac{1}{D}$.

With this inequality, we can continue simplifying the ratio,

$$
\begin{aligned}
&\frac{\pi(\theta)T(\theta,\theta')}{\pi(\theta)G(\theta,\theta')} \\
&\quad \geq \mathbf{E}\left[\min\left(\frac{\exp\left(\sum_i r_i \log\left(1+\frac{C}{\lambda c_i}\phi_i(\theta)\right)\right)}{\exp\left(-\sum_i U_i(\theta)\right)}, \frac{\exp\left(\sum_i r_i \log\left(1+\frac{C}{\lambda c_i}\phi_i(\theta')\right)\right)}{\exp\left(-\sum_i U_i(\theta')\right)}\right)\right] \\
&\qquad \cdot \exp\left[-\frac{1}{2}\left(\sum_i U_i(\theta) + \sum_i U_i(\theta') + CM(\theta,\theta')\right)\right] \\
&\quad = \mathbf{E}\Bigg[\min\Bigg(\exp\left(\sum_i\left(r_i \log\left(1+\frac{C}{\lambda c_i}\phi_i(\theta)\right) - \phi_i(\theta)\right)\right), \\
&\qquad \exp\left(\sum_i\left(r_i \log\left(1+\frac{C}{\lambda c_i}\phi_i(\theta')\right) - \phi_i(\theta')\right)\right)\Bigg)\Bigg] \\
&\quad = \mathbf{E}\Bigg[\max\Bigg(\exp\left(\sum_i\left(\phi_i(\theta) - r_i \log\left(1+\frac{C}{\lambda c_i}\phi_i(\theta)\right)\right)\right), \\
&\qquad \exp\left(\sum_i\left(\phi_i(\theta') - r_i \log\left(1+\frac{C}{\lambda c_i}\phi_i(\theta')\right)\right)\right)\Bigg)^{-1}\Bigg].
\end{aligned}
$$

Because $f(x) = \frac{1}{x}$ is a convex function, by Jensen's inequality it follows

$$
\begin{aligned}
\frac{\pi(\theta)T(\theta,\theta')}{\pi(\theta)G(\theta,\theta')} \geq \mathbf{E}\Bigg[&\max\Bigg(\exp\left(\sum_i\left(\phi_i(\theta) - r_i \log\left(1+\frac{C}{\lambda c_i}\phi_i(\theta)\right)\right)\right), \\
&\exp\left(\sum_i\left(\phi_i(\theta') - r_i \log\left(1+\frac{C}{\lambda c_i}\phi_i(\theta')\right)\right)\right)\Bigg)\Bigg]^{-1}.
\end{aligned}
$$

We use $\max(A,B) \leq (A^p + B^p)^{\frac{1}{p}}$ to remove the $\max$ function.

$$
\begin{aligned}
\frac{\pi(\theta)T(\theta,\theta')}{\pi(\theta)G(\theta,\theta')} \geq \mathbf{E}\Bigg[&\left(\exp\left(p\sum_i\left(\phi_i(\theta) - r_i \log\left(1+\frac{C}{\lambda c_i}\phi_i(\theta)\right)\right)\right)\right) + \\
&\exp\left(p\sum_i\left(\phi_i(\theta') - r_i \log\left(1+\frac{C}{\lambda c_i}\phi_i(\theta')\right)\right)\right)\Bigg]^{\frac{1}{p}}\Bigg]^{-1}.
\end{aligned}
$$

Since $x^{\frac{1}{p}}$ is concave, by Jensen's inequality

$$
\begin{aligned}
\frac{\pi(\theta)T(\theta,\theta')}{\pi(\theta)G(\theta,\theta')} &\geq \mathbf{E}\Bigg[\exp\left(p\sum_i\left(\phi_i(\theta) - r_i \log\left(1+\frac{C}{\lambda c_i}\phi_i(\theta)\right)\right)\right) + \\
&\qquad \exp\left(p\sum_i\left(\phi_i(\theta') - r_i \log\left(1+\frac{C}{\lambda c_i}\phi_i(\theta')\right)\right)\right)\Bigg]^{-\frac{1}{p}} \\
&= \Bigg[\prod_i \mathbf{E}\exp\left(p\phi_i(\theta) - pr_i \log\left(1+\frac{C}{\lambda c_i}\phi_i(\theta)\right)\right) + \\
&\qquad \prod_i \mathbf{E}\exp\left(p\phi_i(\theta') - pr_i \log\left(1+\frac{C}{\lambda c_i}\phi_i(\theta')\right)\right)\Bigg]^{-\frac{1}{p}}.
\end{aligned}
$$

$\mathbf{E}\left[\exp\left(-pr_i\log\left(1+\frac{C}{\lambda c_i}\phi_i(\theta)\right)\right)\right]$ is the moment generating function of the Poisson random variable r_i evaluated at

$$t=-p\log\left(1+\frac{C}{\lambda c_i}\phi_i(\theta)\right).$$

We know that

$$\mathbf{E}\exp(r_it)=\exp\left(\frac{\lambda c_i}{C}\left(\exp(t)-1\right)\right),$$

therefore,

$$\mathbf{E}\left[\exp\left(-pr_i\log\left(1+\frac{C}{\lambda c_i}\phi_i(\theta)\right)\right)\right]=\exp\left(\frac{\lambda c_i}{C}\left(1+\frac{C}{\lambda c_i}\phi_i(\theta)\right)^{-p}-\frac{\lambda c_i}{C}\right).$$

Substituting this into the original expression produces

$$\begin{aligned}\frac{\pi(\theta)T(\theta,\theta')}{\pi(\theta)G(\theta,\theta')}\geq\Bigg[&\prod_i\exp\left(\frac{\lambda c_i}{C}\left(1+\frac{C}{\lambda c_i}\phi_i(\theta)\right)^{-p}-\frac{\lambda c_i}{C}+p\phi_i(\theta)\right)\\&+\prod_i\exp\left(\frac{\lambda c_i}{C}\left(1+\frac{C}{\lambda c_i}\phi_i(\theta')\right)^{-p}-\frac{\lambda c_i}{C}+p\phi_i(\theta')\right)\Bigg]^{-\frac{1}{p}}.\end{aligned}$$

Considering the term inside exp. Define a function $f(y)=\frac{\lambda c_i}{C}\left(1+\frac{C}{\lambda c_i}y\right)^{-p}-\frac{\lambda c_i}{C}+py$ for $y\geq 0$. It is clear that $f(0)=0$. The first derivative is

$$f'(y)=p+(-p)\left(1+\frac{C}{\lambda c_i}y\right)^{-p-1}$$

which is also 0 at $y=0$. The second and third derivatives are

$$f''(y)=(-p)(-p-1)\frac{C}{\lambda c_i}\left(1+\frac{C}{\lambda c_i}y\right)^{-p-2},\tag{7}$$

$$f'''(y)=(-p)(-p-1)(-p-2)\left(\frac{C}{\lambda c_i}\right)^2\left(1+\frac{C}{\lambda c_i}y\right)^{-p-3}.\tag{8}$$

By Taylor series, we have

$$f(y)=f(0)+f'(0)y+\frac{f''(0)}{2!}y^2+\frac{f'''(v)}{3!}y^3$$

where v is between 0 and y. By (8), we know that $f'''(v)\leq 0$, therefore since $y\geq 0$, we have

$$\begin{aligned}f(y)&\leq f(0)+f'(0)y+\frac{f''(0)}{2!}y^2\\&=\frac{f''(0)}{2!}y^2.\end{aligned}$$

Substituting $y=\phi_i(\theta)$ produces

$$\begin{aligned}f(\phi_i(\theta))&\leq(-p)(-p-1)\frac{C}{\lambda c_i}\phi_i^2(\theta)\\&\leq(-p)(-p-1)\frac{C}{\lambda c_i}c_i^2M^2(\theta,\theta').\end{aligned}$$

Similarly, we can get

$$f(\phi_i(\theta'))\leq p(p+1)\frac{C}{\lambda c_i}c_i^2M^2(\theta,\theta').$$

Substituting these to the spectral ratio, we get

$$\begin{aligned}\frac{\pi(\theta)T(\theta,\theta')}{\pi(\theta)G(\theta,\theta')} &\geq \left[2\prod_i \exp\left(p(p+1)\frac{C}{\lambda c_i}c_i^2 M^2(\theta,\theta')\right)\right]^{-\frac{1}{p}} \\ &= \left[2\exp\left(\sum_i p(p+1)\frac{C}{\lambda}c_i M^2(\theta,\theta')\right)\right]^{-\frac{1}{p}} \\ &= \left[2\exp\left(p(p+1)\frac{C^2}{\lambda}M^2(\theta,\theta')\right)\right]^{-\frac{1}{p}} \\ &= 2^{-\frac{1}{p}}\exp\left(-(p+1)\frac{C^2}{\lambda}M^2(\theta,\theta')\right).\end{aligned}$$

Now, we maximize the R.H.S. with respect to p. Let $E = \frac{C^2}{\lambda}M^2(\theta,\theta')$, then it becomes

$$2^{-\frac{1}{p}}\exp\left(-(p+1)E\right) = \exp\left(-E - pE - \frac{1}{p}\log 2\right).$$

The maximum is attained at $p = \sqrt{\frac{\log 2}{E}}$ and the value is

$$\exp\left(-E - 2\sqrt{E\log 2}\right).$$

It follows that

$$\frac{\pi(\theta)T(\theta,\theta')}{\pi(\theta)G(\theta,\theta')} \geq \exp\left(-\frac{C^2}{\lambda}M^2(\theta,\theta') - 2\sqrt{\frac{C^2}{\lambda}M^2(\theta,\theta')\log 2}\right).$$

We set $\lambda = \chi C^2 M^2(\theta,\theta')$, it becomes

$$\frac{\pi(\theta)T(\theta,\theta')}{\pi(\theta)G(\theta,\theta')} \geq \exp\left(-\frac{1}{\chi} - 2\sqrt{\frac{\log 2}{\chi}}\right).$$

We complete the theorem by a Dirichlet form argument. We can write the Dirichlet form $\mathcal{E}(f)$ of a Markov chain with transition operator G as [13]:

$$\mathcal{E}(f) = \frac{1}{2}\int\int\left[\left(f(\theta) - f(\theta')\right)^2\right]G(\theta,\theta')\pi(\theta)d\theta d\theta'.$$

If we let $L_0^2(\pi)$ to be the Hilbert space of functions f such that f has mean zero and is square integrable with respect to probability measure π. It follows that the spectral gap γ of a Markov chain is [2]

$$\gamma = \inf_{f\in L_0^2(\pi):Var_\pi[f]=1}\mathcal{E}(f).$$

From this, it is easy to get that

$$\begin{aligned}\bar{\gamma} &= \inf_{f\in L_0^2(\pi):Var_\pi[f]=1}\left[\frac{1}{2}\int\int\left[\left(f(\theta) - f(\theta')\right)^2\right]T(\theta,\theta')\pi(\theta)d\theta d\theta'\right] \\ &\geq \exp\left(-\frac{1}{\chi} - 2\sqrt{\frac{\log 2}{\chi}}\right)\cdot\inf_{f\in L_0^2(\pi):Var_\pi[f]=1}\left[\frac{1}{2}\int\int\left[\left(f(\theta) - f(\theta')\right)^2\right]G(\theta,\theta')\pi(\theta)d\theta d\theta'\right] \\ &= \exp\left(-\frac{1}{\chi} - 2\sqrt{\frac{\log 2}{\chi}}\right)\cdot\gamma.\end{aligned}$$

□

F Derivation of Equation (2)

Based on the bound in Theorem 2, to make sure that the spectral ratio $\bar{\gamma}/\gamma \geq \kappa$, we can set χ such that

$$\exp\left(-\frac{1}{\chi} - 2\sqrt{\frac{\log 2}{\chi}}\right) = \kappa.$$

Solving the above equation gives us

$$\chi = \frac{(2\log 2 - \log\kappa + 2\sqrt{\log 2(\log 2 - \log\kappa)})}{\log^2\kappa} \leq \frac{4}{(1-\kappa)\log(1/\kappa)}.$$

Since the spectral gap ratio is monotonically increasing w.r.t. χ, we can instead set χ to the upper bound

$$\chi = \frac{4}{(1-\kappa)\log(1/\kappa)}$$

which guarantees that $\bar{\gamma}/\gamma \geq \kappa$.

G Theoretically Optimal Value of χ

The overall wall-clock time L for a chain to converge can be represented as the number of steps times the wall-clock time l of each step. We then minimize an upper bound of this overall wall-clock time to get the optimal value of χ.

Consider a lazy Markov chain on a finite state Θ. The *relaxation time* t_{rel} of a Markov chain is defined to be the inverse of the spectral gap γ: $t_{\text{rel}} = 1/\gamma$. The *mixing time* t_{mix}, i.e. the number of steps required for a chain to converge to within TV distance δ to the target distribution π, is bounded by Levin and Peres [18]

$$t_{\text{mix}} \leq t_{\text{rel}} \log\left(\frac{1}{\delta \cdot \min_{\theta\in\Theta}\pi(\theta)}\right).$$

It follows that the overall wall-clock time L is upper bouned by

$$L = l \cdot t_{\text{mix}} \leq l \cdot t_{\text{rel}} \log\left(\frac{1}{\delta \cdot \min_{\theta\in\Theta}\pi(\theta)}\right).$$

We assume that the expected wall clock time to run a step is proportional to the batch size plus some constant, which measures the cost of computing the proposal. Specifically, We use η and ξ to denote the time to get a proposal θ' and compute a U_i in a step. Then we can write the time of a step l as

$$l = B\xi + \eta.$$

In order to minimize L, we can instead minimize its upper bound, which is equivalent to minimize

$$l \cdot t_{\text{rel}} = (B\xi + \eta) \cdot \frac{1}{\gamma}. \tag{9}$$

Recall that for TunaMH, the average batch size over all steps is

$$\mathbf{E}_{(\theta,\theta')\sim\pi(\theta)q(\theta'|\theta)}[\chi C^2 M^2(\theta,\theta') + CM(\theta,\theta')],$$

and the spectral gap $\bar{\gamma}$ is lower bounded by the spectral gap of standar MH γ such that

$$\bar{\gamma} \geq \exp\left(-\frac{1}{\chi} - 2\sqrt{\frac{\log 2}{\chi}}\right) \cdot \gamma.$$

Substituting the expression of batch size and spectral gap to (9) gives

$$l \cdot t_{\text{rel}} \leq \left(\mathbf{E}_{(\theta,\theta')\sim\pi(\theta)q(\theta'|\theta)}[\chi C^2 M^2(\theta,\theta') + CM(\theta,\theta')]\xi + \eta\right) \cdot \exp\left(\frac{1}{\chi} + 2\sqrt{\frac{\log 2}{\chi}}\right) \cdot \frac{1}{\gamma}.$$

To minimize the RHS of the above equation over χ, we let the derivative w.r.t. χ to be zero and get,

$$\begin{aligned}
&\xi C^2 \mathbf{E}_{(\theta,\theta')\sim\pi(\theta)q(\theta'|\theta)}[M^2(\theta,\theta')]\chi^{-1} + (\xi C \mathbf{E}_{(\theta,\theta')\sim\pi(\theta)q(\theta'|\theta)}[M(\theta,\theta')] + \eta)\chi^{-2} \\
&\quad + \sqrt{\log 2}\xi C^2 \mathbf{E}_{(\theta,\theta')\sim\pi(\theta)q(\theta'|\theta)}[M^2(\theta,\theta')]\chi^{-\frac{1}{2}} \\
&\quad + \sqrt{\log 2}(\xi C \mathbf{E}_{(\theta,\theta')\sim\pi(\theta)q(\theta'|\theta)}[M(\theta,\theta')] + \eta)\chi^{-\frac{3}{2}} \\
&\quad = \xi C^2 \mathbf{E}_{(\theta,\theta')\sim\pi(\theta)q(\theta'|\theta)}[M^2(\theta,\theta')].
\end{aligned}$$

When χ is small, the LHS is approximately $(\xi C \mathbf{E}_{(\theta,\theta')\sim\pi(\theta)q(\theta'|\theta)}[M(\theta,\theta')] + \eta)\chi^{-2}$ which gives us

$$\chi = \sqrt{\frac{\xi C \mathbf{E}_{(\theta,\theta')\sim\pi(\theta)q(\theta'|\theta)}[M(\theta,\theta')] + \eta}{\xi C^2 \mathbf{E}_{(\theta,\theta')\sim\pi(\theta)q(\theta'|\theta)}[M^2(\theta,\theta')]}}.$$

When it is quick to get a proposal ($\eta \approx 0$) and the variance of M is small, we can further simplify it to

$$\chi = \frac{1}{\sqrt{C \mathbf{E}_{(\theta,\theta')\sim\pi(\theta)q(\theta'|\theta)}[M(\theta,\theta')]}}.$$

In practice, we can get the above theoretically optimal value of χ by empirically estimating the mean and variance of $M(\theta,\theta')$. Note that even if these empirical estimates are accurate, there may exist better χ, since the upper bounds (the mixing time bound and the spectral gap bound) we use to get the optimal value may be loose. We give a simpler heuristic to tune χ in practice in Section 5.

H Proof of Theorem 3

First, we will show the following lemma, which gives half of what we want to have in the theorem.

Lemma 1. *Considering the same setting as the theorem, the average batch size B of any exact, stateless minibatch MH algorithm at any iteration follows*

$$\mathbf{E}[B] \geq 2^{-18} \cdot \kappa C^2 M^2(\theta,\theta') - 2^{-4} \cdot \kappa.$$

Proof. We prove the lemma by construction. First, observe that since the state space Θ has at least two states, we can restrict our attention to just two of those states, by choosing a π that has zero mass on any other state in the space and a q that never proposes transitioning out to any of those other states (at which π has zero mass). Such a proposal will still be ergodic, so it still satisfies our general assumption that we consider only ergodic chains in this paper. Without loss of generality, suppose that those two states are $\{-\frac{M}{2}, \frac{M}{2}\}$ (this is without loss of generality because we can always just rename the states), and let C denote the constant in the theorem statement and define (with a bit of abuse of notation) the constant $M := M(-\frac{M}{2}, \frac{M}{2})$. By doing this, we can (again without loss of generality) restrict our attention to the case where $\Theta = \{-\frac{M}{2}, \frac{M}{2}\}$.

Next, we construct our counterexample. Let the dataset be $\{x_i\}_{i=1}^N$ where $x_i \in \{-1, 1\}$. We let the domain for parameter θ to be $\{-\frac{M}{2}, \frac{M}{2}\}$, and the target distribution to be

$$\pi(\theta) = \frac{1}{Z}\exp\left(-\sum_{i=1}^N U_i(\theta)\right) = \frac{1}{Z}\exp\left(-\frac{C\theta}{N}\sum_{i=1}^N x_i\right)$$

where $U_i(\theta) = \frac{C}{N} \cdot \theta x_i$. Note that by letting N become large, any minibatch MH algorithm that queries the energy difference oracle some number of times will observe a distribution of energy differences that is arbitrarily close to a sequence of independent identically distributed random variables supported on $\{\pm\frac{CM}{N}\}$.

We define $c_i = \frac{C}{N}$, and the proposal distribution to be

$$p(\theta,\theta) = \frac{1}{2}, \qquad p(\theta,-\theta) = \frac{1}{2} \qquad \text{for } \theta \in \left\{-\frac{M}{2}, \frac{M}{2}\right\}.$$

Now, let $0 < q < 1$ be some constant, and consider two cases: (1) $\frac{1}{N}\sum_i x_i = q$ and (2) $\frac{1}{N}\sum_i x_i = -q < 0$. Suppose that in both cases the x_i are shuffled at random. These two cases will have different stationary distributions,

$$\pi_1(\theta) = \frac{1}{Z}\exp\left(-Cq\theta\right) \qquad \text{and} \qquad \pi_2(\theta) = \frac{1}{Z}\exp\left(Cq\theta\right),$$

and an exact algorithm must be able to distinguish between them. Therefore by using these cases, we can get a bound on the required batch size needed for the exact MH algorithm to distinguish between them. First, we observe that the two cases are symmetric, such that if T_1 is the transition matrix of the chain in case (1) and T_2 is the transition matrix of the chain in case (2), then $T_1(\theta, \theta') = T_2(\theta', \theta)$. Let $0 < \psi < \frac{1}{2}$ denote the probability that T_1 transitions from $\frac{M}{2}$ to $-\frac{M}{2}$. Then because the MH method is exact and the chain is reversible, the probability of the reverse transition is $\psi\exp(-CMq)$. So, explicitly, the transition operators will look like

$$T_1 = \begin{bmatrix} 1-\psi & \psi e^{-CMq} \\ \psi & 1-\psi e^{-CMq} \end{bmatrix} \qquad \text{and} \qquad T_2 = \begin{bmatrix} 1-\psi e^{-CMq} & \psi \\ \psi e^{-CMq} & 1-\psi \end{bmatrix}.$$

The eigenvectors and eigenvalues of this are

$$T_1\pi_1 = \pi_1 \qquad \text{and} \qquad T_1\begin{bmatrix}-1\\1\end{bmatrix} = \left(1-\psi-\psi\exp(-CMq)\right)\begin{bmatrix}-1\\1\end{bmatrix}.$$

Suppose that we initialize both chains uniformly on $\{-\frac{M}{2}, \frac{M}{2}\}$. Observe that

$$\begin{bmatrix}1/2\\1/2\end{bmatrix} = \begin{bmatrix}\frac{\exp(-CMq)}{1+\exp(-CMq)}\\ \frac{1}{1+\exp(-CMq)}\end{bmatrix} + \frac{1-\exp(-CMq)}{2(1+\exp(-CMq))}\cdot\begin{bmatrix}1\\-1\end{bmatrix},$$

the first vector being π_1 and the second being a multiple of the other eigenvector. Equivalently,

$$\begin{bmatrix}1/2\\1/2\end{bmatrix} = \pi_1 + \frac{1}{2}\tanh\left(\frac{CMq}{2}\right)\cdot\begin{bmatrix}1\\-1\end{bmatrix},$$

and so for any t, after t steps of the Markov chain, the distribution will be

$$T_1^t\begin{bmatrix}1/2\\1/2\end{bmatrix} = \pi_1 + \frac{1}{2}\tanh\left(\frac{CMq}{2}\right)\cdot\left(1-\psi-\psi\exp(-CMq)\right)^t\cdot\begin{bmatrix}1\\-1\end{bmatrix}.$$

Similarly,

$$T_2^t\begin{bmatrix}1/2\\1/2\end{bmatrix} = \pi_2 + \frac{1}{2}\tanh\left(\frac{CMq}{2}\right)\cdot\left(1-\psi-\psi\exp(-CMq)\right)^t\cdot\begin{bmatrix}-1\\1\end{bmatrix}.$$

So, the total variation distance between the state of the chains at time t will be bounded by

$$\text{TV}\left(T_1^t\begin{bmatrix}1/2\\1/2\end{bmatrix}, T_2^t\begin{bmatrix}1/2\\1/2\end{bmatrix}\right) \geq \text{TV}\left(\pi_1,\pi_2\right) - \tanh\left(\frac{CMq}{2}\right)\cdot\left(1-\psi-\psi\exp(-CMq)\right)^t.$$

Also observe that

$$\text{TV}\left(\pi_1,\pi_2\right) = \frac{1}{2}\left\|\begin{bmatrix}\frac{\exp(-CMq)}{1+\exp(-CMq)}\\ \frac{1}{1+\exp(-CMq)}\end{bmatrix} - \begin{bmatrix}\frac{1}{1+\exp(-CMq)}\\ \frac{\exp(-CMq)}{1+\exp(-CMq)}\end{bmatrix}\right\|_1 = \frac{1-\exp(-CMq)}{1+\exp(-CMq)} = \tanh\left(\frac{CMq}{2}\right),$$

so

$$\text{TV}\left(T_1^t\begin{bmatrix}1/2\\1/2\end{bmatrix}, T_2^t\begin{bmatrix}1/2\\1/2\end{bmatrix}\right) \geq \tanh\left(\frac{CMq}{2}\right)\cdot\left(1-\left(1-\psi-\psi\exp(-CMq)\right)^t\right).$$

Also, since we know that our algorithm is guaranteed to have spectral gap ratio at least κ with the original chain, it follows that $\psi \geq \kappa/2$, and so

$$\text{TV}\left(T_1^t\begin{bmatrix}1/2\\1/2\end{bmatrix}, T_2^t\begin{bmatrix}1/2\\1/2\end{bmatrix}\right) \geq \tanh\left(\frac{CMq}{2}\right)\cdot\left(1-\left(1-\frac{\kappa}{2}-\frac{\kappa}{2}\exp(-CMq)\right)^t\right).$$

Now, denote the exact minibatch algorithm to be $\mathcal{A}$. As it runs, the algorithm $\mathcal{A}$ will request data examples by querying the energy difference oracle. Under case (1), we let y_i denote the ith sample

that $\mathcal{A}$ *would have observed* if it requested i or more samples, and similarly we let z_i denote the analogous sample in case (2). Fix some constant $t \in \mathbf{N}$ (which we will set later). We let K_1 denote the total number of samples observed by $\mathcal{A}$ across the first t iterations in case (1), and set

$$\mu = \{y_1, y_2, \ldots, y_{K_1}\}.$$

Similarly, we let K_2 denote the number of samples observed by $\mathcal{A}$ across the first t iterations in case (2), and set

$$\nu = \{z_1, z_2, \ldots, z_{K_2}\}.$$

Now, we fix some constant K (to be set later), and consider the following coupling between the behavior of $\mathcal{A}$ across its first t iterations in case (1) and in case (2). First, let all internal randomness of $\mathcal{A}$ and the proposal process under case (1) and (2) be the same, which means that for a given observation of data examples, the algorithm $\mathcal{A}$ will make the same decision, such as whether to require more data examples or not and whether to accept or not. Second, choose a coupling that minimizes the probability that

$$(y_1, y_2, \ldots, y_{K1}) \neq (z_1, z_2, \ldots, z_{K2}).$$

Such a coupling is guaranteed to exist by the Coupling Lemma, and the probability that these two are not equal will be equal to the total variation distance between their distributions. Third, assign all the other y_i and z_i, for $i > K$, independently according to their distribution.

We are interested in the quantity $p(\mu \neq \nu)$, which bounds the probability that the algorithm may make a different decision in cases (1) and (2). We can decompose this probability into two terms,

$$p(\mu \neq \nu) = p(\mu \neq \nu \text{ and } y_j = z_j \text{ for all } j \leq K) + p(\mu \neq \nu \text{ and } y_j \neq z_j \text{ for some } j \leq K).$$

If $\mu \neq \nu$ but $y_j = z_j$ for all $j \leq K$, the only way that this is possible is for $K_1 > K$ (and, symmetrically, also $K_2 > K$), since otherwise the algorithms would behave identically. So,

$$p(\mu \neq \nu) \leq p(K_1 > K) + p(y_j \neq z_j \text{ for some } j \leq K). \tag{10}$$

By Markov's inequality,

$$p(\mu \neq \nu) \leq \frac{\mathbf{E}[K_1]}{K} + p(y_j \neq z_j \text{ for some } j \leq K).$$

For the second term of (10), we can reduce the case to only considering K samples. Let S_y be the total number of samples y_i that are -1 and let S_z be the total number of samples z_i that are -1. Since $\mathcal{A}$ is effectively sampling a shuffled dataset at some arbitrary indices without replacement, both of these random variables S_y and S_z are—properly speaking—hypergeometric random variables. However, since our dataset size N is arbitrary here, we can by setting N very large work in the limit (as $N \to \infty$) in which these variables become binomial (since sampling with replacement and without replacement can be made to have arbitrarily close to the same distribution by making the dataset large). Observe that (in this limit) S_y follows a binomial distribution $B(K, \frac{1-q}{2})$ and S_z follows a binomial distribution $B(K, \frac{1+q}{2})$. Clearly, if $S_y = S_z$, then we can arrange the coupling so that $(y_1, \ldots, y_K) = (z_1, \ldots, z_K)$. So, by the Coupling Lemma,

$$p(y_j \neq z_j \text{ for some } j \leq K) = p(S_y \neq S_z) = \text{TV}(S_y, S_z).$$

From the analysis in Adell and Jodrá [1], we can bound the total variance distance between these two binomial variables with

$$\text{TV}(S_y, S_z) \leq \sqrt{e} \cdot \frac{\tau}{(1-\tau)^2}$$

where $\tau = \sqrt{\frac{K+2}{2}} \cdot q < 1$. Substituting these bounds, we get

$$p(\mu \neq \nu) \leq \frac{\mathbf{E}[K_1]}{K} + \sqrt{e} \cdot \frac{\tau}{(1-\tau)^2}.$$

But the probability that $\mu \neq \nu$ must be an upper bound on the probability that the distributions of the chains in case (1) and (2) after t steps are not equal, since if $\mu = \nu$ in the coupling then the two chains are in the same state. So, using our bound from earlier, we get

$$\tanh\left(\frac{CMq}{2}\right) \cdot \left(1 - \left(1 - \frac{1}{2}\kappa - \frac{1}{2}\kappa \exp(-CMq)\right)^t\right) \leq \frac{\mathbf{E}[K_1]}{K} + \sqrt{e} \cdot \frac{\tau}{(1-\tau)^2}.$$

Now isolating $\mathbf{E}[K_1]$ gives

$$K \cdot \tanh\left(\frac{CMq}{2}\right) \cdot \left(1 - \left(1 - \frac{1}{2}\kappa - \frac{1}{2}\kappa\exp(-CMq)\right)^t\right) - K \cdot \sqrt{e} \cdot \frac{\tau}{(1-\tau)^2} \leq \mathbf{E}[K_1].$$

Also, observe that

$$\left(1 - \frac{1}{2}\kappa - \frac{1}{2}\kappa\exp(-CMq)\right)^t \leq \left(1 - \frac{1}{2}\kappa\right)^t \leq \exp\left(-\frac{\kappa t}{2}\right),$$

so

$$K \cdot \tanh\left(\frac{CMq}{2}\right) \cdot \left(1 - \exp\left(-\frac{\kappa t}{2}\right)\right) - K \cdot \sqrt{e} \cdot \frac{\tau}{(1-\tau)^2} \leq \mathbf{E}[K_1].$$

This gives us the lower bound on $\mathbf{E}[K_1]$ that we are interested in. Now, it remains to assign q, K, and t. We start by assigning t such that

$$t = \left\lceil 2\kappa^{-1}\log(2)\right\rceil,$$

in which case

$$\exp\left(-\frac{\kappa t}{2}\right) \leq \frac{1}{2}$$

and so

$$K \cdot \frac{1}{2} \cdot \tanh\left(\frac{CMq}{2}\right) - K \cdot \sqrt{e} \cdot \frac{\tau}{(1-\tau)^2} \leq \mathbf{E}[K_1].$$

Now, we add some simplifying assumptions, which we will validate are true later. We assume that

$$\tau = \sqrt{\frac{K+2}{2}} \cdot q \leq \frac{1}{2};$$

in this case

$$\sqrt{e} \cdot \frac{\tau}{(1-\tau)^2} \cdot K \leq 4\sqrt{e} \cdot \tau \leq 5\sqrt{K+2} \cdot q.$$

We set q such that

$$CMq = 1,$$

and we assume that CM is large enough that this assignment of q is within range (i.e. $0 < q < 1$). This gives us

$$K \cdot \frac{1}{2} \cdot \tanh\left(\frac{1}{2}\right) - 5K\sqrt{K+2} \cdot \frac{1}{CM} \leq \mathbf{E}[K_1].$$

Since $\tanh(1/2) > 5/16$, we can simplify this to

$$K \cdot \frac{5}{32} - 5K\sqrt{K+2} \cdot \frac{1}{CM} \leq \mathbf{E}[K_1].$$

All that remains is to assign K. We assign K such that

$$\sqrt{K+2} \cdot \frac{1}{CM} = \frac{1}{64}.$$

In this case, we get

$$K = \frac{C^2M^2}{4096} - 2,$$

and our bound reduces to

$$\left(\frac{C^2M^2}{4096} - 2\right) \cdot \frac{5}{64} \leq \mathbf{E}[K_1].$$

We can simplify this further to

$$2^{-16} \cdot C^2M^2 - \frac{5}{32} \leq \mathbf{E}[K_1].$$

Now, this is a bound on the expected number of samples taken across t iterations. This means that the number of samples taken in any given iteration will be bounded by

$$\frac{\mathbf{E}[K_1]}{t} \geq \frac{2^{-16} \cdot C^2M^2 - \frac{5}{32}}{2\kappa^{-1}\log(2) + 1} = \frac{2^{-16} \cdot \kappa C^2M^2 - \frac{5\kappa}{32}}{2\log(2) + \kappa}.$$

A few more loose bounds, leveraging $\kappa < 1$, gives us

$$\frac{\mathbf{E}[K_1]}{t} \geq 2^{-18} \cdot \kappa C^2 M^2 - \frac{\kappa}{16}.$$

This proves the lemma. □

Next, we will show the following lemma, which characterizes what happens when CM is small.

Lemma 2. *Considering minibatch MH algorithms in the same setting as the theorem, the expected batch size at any iteration must be lower bounded by*

$$\mathbf{E}[B] \geq \frac{\kappa}{2} \min\left(CM(\theta, \theta'), 1\right).$$

Proof. Here, we will prove a lower bound that characterizes the limits of exact stateless minibatch MH algorithms when they use very few examples. Again, without loss of generality we consider a reduction to the two-state case as we did in the proof of the previous lemma. Suppose that a exact stateless minibatch MH algorithm with the same forward and backward proposal probabilities (given some $c_1, \ldots, c_N$, C, and M) requests any energy function examples at all only with probability p. Consider two cases, which have the same $c_1, \ldots, c_N$, C and M. In the first case,

$$\sum_{i=1}^{n} (U_i(\theta) - U_i(\theta')) = CM(\theta, \theta'),$$

while in the second case,

$$\sum_{i=1}^{n} (U_i(\theta) - U_i(\theta')) = -CM(\theta, \theta').$$

These are clearly possible by setting U_i to the limits of what is covered by the bounds. In the first case, the baseline MH method would accept with probability 1. In the second case, it will accept with probability $\exp(-CM(\theta, \theta'))$. Since the stateless MH algorithm is reversible, it must accept in the first case with some probability a and in the second case with probability $a \cdot \exp(-CM(\theta, \theta'))$. But, the algorithm can only distinguish the two cases if it requests samples, which only happens with probability at most p. So,

$$a - a \cdot \exp(-CM(\theta, \theta') \leq p.$$

Since we know that it must be the case that $a \geq \kappa$ (from a straightforward analysis of a two-state case), it follows that

$$\frac{p}{\kappa} \geq \frac{p}{a} \geq 1 - \exp(-CM(\theta, \theta')) \geq \frac{1}{2} \min\left(CM(\theta, \theta'), 1\right).$$

Since p is an obvious lower bound on the expected value of the batch size, it follows that

$$\mathbf{E}[B] \geq \frac{\kappa}{2} \min\left(CM(\theta, \theta'), 1\right).$$

□

To prove Theorem 3 we now combine the results of these two lemmas. We have

$$\mathbf{E}[B] \geq 2^{-18} \cdot \kappa C^2 M^2(\theta, \theta') - 2^{-4} \cdot \kappa.$$

and

$$\mathbf{E}[B] \geq \frac{\kappa}{2} \min\left(CM(\theta, \theta'), 1\right).$$

Since these are both lower bounds, we can combine them to get

$$\begin{aligned}\mathbf{E}[B] &\geq \max\left(2^{-18} \cdot \kappa C^2 M^2(\theta, \theta') - 2^{-4} \cdot \kappa, \frac{\kappa}{2} \min\left(CM(\theta, \theta'), 1\right)\right) \\ &= \kappa \cdot \max\left(2^{-18} \cdot C^2 M^2(\theta, \theta') - 2^{-4}, \frac{1}{2} \min\left(CM(\theta, \theta'), 1\right)\right).\end{aligned}$$

It is obvious from a simple big-$\mathcal{O}$ analysis here that there exists a global constant $\zeta > 0$ such that

$$\mathbf{E}[B] \geq \zeta \cdot \kappa \left(C^2 M^2(\theta, \theta') + CM(\theta, \theta')\right).$$

This proves the theorem.

I Proof of Corollary 1

Proof. Recall that the lower bound on the batch size in each iteration is

$$\mathbf{E}[B] \geq \zeta \cdot \kappa \left(C^2 M^2(\theta, \theta') + CM(\theta, \theta') \right).$$

Since $C = \mathcal{O}(N)$ and $M(\theta, \theta') = \mathcal{O}(N^{-(h+1)/2})$, the expectation of the batch size follows

$$\mathbf{E}[B] = \mathcal{O}(C^2 M^2(\theta, \theta') + CM(\theta, \theta')) = \mathcal{O}(CM(\theta, \theta')) = \mathcal{O}(N^{1-h}/2).$$

When $h = 1$, $\mathbf{E}[B] = \mathcal{O}(1)$ and when $h = 2$, $\mathbf{E}[B] = \mathcal{O}(1/\sqrt{N})$. □

J Experimental Details and Additional Results

J.1 Experiment in Section 2.1

To verify Theorem 1, we empirically construct a distribution in the form of Section A such that AustereMH and MHminibatch are biased on. Note that the proof in Section A shows there must exist such a distribution for any inexact minibatch method but does not tell us how to find one for a specific method. Therefore, in order to find such a distribution, we construct an example and empirically test whether AustereMH and MHminibatch are biased on it.

We let data x_i take one of two values $\{-1, 5\}$. Consider a dataset of size 6000. We let 5000 data take value -1 and the remaining 1000 data take value 5. Define the target distribution $\pi(\theta)$ to be

$$\pi(\theta) \propto \exp\left(-\frac{1}{N} \sum_{i=1}^{N} \theta \cdot x_i \right)$$

where the domain of θ is $\{0, 1, \ldots, K-1\}$. Therefore the number of state is K. Since $\sum_i x_i = 0$, it is clear to see that the stationary distribution of θ is a uniform distribution. We define the proposal distribution to be the following

$$p(\theta, \theta) = \frac{1}{2}, \quad \text{for all } \theta; \quad p(\theta, \theta - 1) = \frac{1}{4}, \quad p(\theta, \theta + 1) = \frac{1}{4} \quad \text{for } \theta \in \{1, \ldots, K-2\};$$

and $p(0, 1) = p(K-1, K-2) = \frac{1}{2}$.

We set the hyperparameter error ϵ in AustereMH to be 0.01 and δ in MHminibatch to be 5, following the setting in their original papers [17, 26]. We set batch size m in both methods to be 30. We find that AustereMH and MHminibatch are both inexact on this example and the error increases as we increase K. Thus we empirically verify the statement in Theorem 1.

Besides the density estimate comparison on $K = 200$ shown in Figure 1b, we additionally report the estimate results on other values of K in Figure 5. We see that the results are similar, all showing that TunaMH and standard MH can give accurate estimate whereas inexact methods are seriously wrong.

On Robust Linear Regression We further tested AustereMH on robust linear regression in Section 5.1 with $N = 5000$. We computed the MSE between estimated and true parameters. MH, TunaMH and AustereMH obtained MSE 0.149, 0.15 and 1.19 respectively, indicating inexact method error can be large on typical problems.

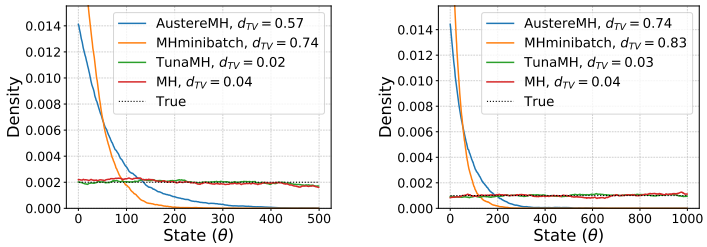
J.2 Robust Linear Regression

We follow the experimental setup of robust linear regression (RLR) in Cornish et al. [12]. Specifically, we have data $x_i \in \mathbb{R}^d$ and $y_i \in \mathbb{R}$. The likelihood is modeled by a student's t-distribution with degrees of freedom v:

$$p(y_i|\theta, x_i) = \text{Student}(y_i - \theta^\intercal x_i | v).$$

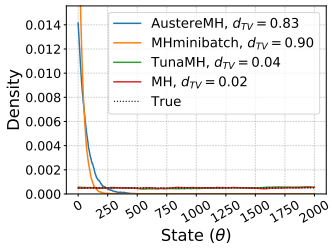
It follows that

$$U_i(\theta) = \frac{v+1}{2} \log\left(1 + \frac{(y_i - \theta^\intercal x_i)^2}{v} \right),$$

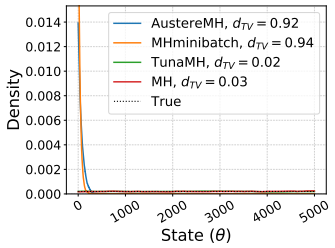


(a) $K = 500$

(b) $K = 1000$



(e) $K = 2000$



(e) $K = 5000$

Figure 5: Density estimate comparison on $K = 500, 1000, 2000, 5000$.

and the first derivative

$$\partial_j U_i(\theta) = -(v+1)\frac{x_{ij}(y_i - \theta^\intercal x_i)}{v + (y_i - \theta^\intercal x_i)^2}.$$

Since the function U_i is Lipschitz continuous, we can easily get the bound used in TunaMH, TFMH and SMH. We set $M(\theta, \theta') = \|\theta - \theta'\|_2$ and then it follows

$$c_i = \sup_{\theta \in \mathbb{R}} \|\nabla U_i(\theta)\|_2 = \frac{v+1}{2\sqrt{v}} \|x_i\|_2 .$$

The data x_i and y_i is generated as follows

$$y_i = \sum_j x_{ij} + \epsilon_i$$

where $\epsilon_i \sim \mathcal{N}(0, 1)$.

In Section 5.1, we set $v = 4$, $d = 100$ and use a flat prior $p(\theta) = 1$. Note that our problem dimension d is much larger than that in the SMH paper [12] ($d = 10$). This makes the control variates in SMH problematic since the bounds they require appear to scale badly in high dimensions.

To reach the target acceptance rate, we set the stepsize in each method as in Table 2 and 3. For TunaMH and TunaMH-MAP, we set $\chi = 1e-5$ for $N = 5000, 20000$ and $\chi = 1e-4$ for $N = 50000, 100000$. For FlyMC and FlyMC-MAP, we set the probability for a data going from dark to bright $q_{d\to b}$ to be 0.01. Without the MAP, we collect 80000 samples after 200000 step burnin. With the MAP, we collect 80000 samples without burnin.

Table 2: Stepsize of methods without the MAP.

	MH	TFMH	FlyMC	TunaMH
RLR $N = 5000$	4e-3	1e-4	2.7e-3	8e-4, $\chi = 1e-5$
RLR $N = 20000$	2e-3	3e-5	1.5e-3	3e-4, $\chi = 1e-5$
RLR $N = 50000$	1.3e-3	1.2e-5	9e-4	2e-4, $\chi = 1e-4$
RLR $N = 100000$	9e-4	6e-6	7e-4	1.7e-4, $\chi = 1e-4$
TGM	3e-1	2.2e-2	1e-2	1e-1
LR	5e-3	1e-4	2e-3	1e-3

Table 3: Stepsize of methods with the MAP.

	MH-MAP	SMH-1	SMH-2	FlyMC-MAP	TunaMH-MAP
RLR $N = 5000$	4e-3	4e-3	4e-3	6e-3	8e-4, $\chi = 1e-5$
RLR $N = 20000$	2e-3	2e-3	2e-3	3.5e-3	3e-4, $\chi = 1e-5$
RLR $N = 50000$	1.2e-3	1.2e-3	1.2e-3	2.5e-3	1.2e-4, $\chi = 1e-4$
RLR $N = 100000$	9e-4	5.9e-4	8e-4	1.7e-3	7e-5. $\chi = 1e-4$
TGM	-	1e-1	-	1e-2	-

J.2.1 Additional Experimental Results with $d = 10$

We ran RLR experiment with $d = 10$ and $N = 10^5$ to compare the performance in low dimensions. The ESS/S for TFMH, FlyMC, TunaMH are 0.02, 0.75, & 1.7, respectively; SMH-1, SMH-2, FlyMC-MAP and TunaMH-MAP are 174.7, 5969.5, 730.8, & 730.1 respectively. This suggests TunaMH is significantly better without MAP/control variates. With MAP/control variates, TunaMH is better than SMH-1, similar to FlyMC and worse than SMH-2.

J.3 Truncated Gaussian Mixture

The data in this truncated Gaussian mixture (TGM) task is generated as follows

$$x_i \sim \frac{1}{2}\mathcal{N}(\theta_1, \sigma_x^2) + \frac{1}{2}\mathcal{N}(\theta_1 + \theta_2, \sigma_x^2)$$

where $\theta_1 = 0, \theta_2 = 1$ and $\sigma^2 = 2$. The posterior θ has two modes at $(\theta_1, \theta_2) = (0, 1)$ and $(\theta_1, \theta_2) = (1, -1)$. In order to get the bounds required by all methods, we truncate the Gaussian by setting $\theta_1, \theta_2 \in [-3, 3]$.

For simplicity we assume a flat prior $p(\theta) = 1$. Then the energy is given by

$$U_i(\theta) = -\log p(x_i|\theta) = \log(2\sqrt{2\pi}\sigma_x) - \log\left[\exp\left(-\frac{(x_i-\theta_1)^2}{2\sigma_x^2}\right) + \exp\left(-\frac{(x_i-\theta_1-\theta_2)^2}{2\sigma_x^2}\right)\right].$$

Denote $E_1 = \exp\left(-\frac{(x_i-\theta_1)^2}{2\sigma_x^2}\right)$ and $E_2 = \exp\left(-\frac{(x_i-\theta_1-\theta_2)^2}{2\sigma_x^2}\right)$. To get the upper bound in TunaMH, TFMH and SMH, we compute the gradient

$$\frac{\partial U_i(\theta)}{\partial \theta_1} = -\frac{1}{E_1 + E_2}\left(E_1 \cdot \frac{x_i - \theta_1}{\sigma_x^2} + E_2 \cdot \frac{x_i - \theta_1 - \theta_2}{\sigma_x^2}\right),$$
$$\frac{\partial U_i(\theta)}{\partial \theta_2} = -\frac{1}{E_1 + E_2}\left(E_2 \cdot \frac{x_i - \theta_1 - \theta_2}{\sigma_x^2}\right).$$

Since $\theta_i \in [-3, 3]$, it follows that

$$\left|\frac{\partial U_i(\theta)}{\partial \theta_1}\right| \le \frac{|x_i| + 3}{\sigma_x^2} + \frac{|x_i| + 3 + 3}{\sigma_x^2} \le \frac{2\,|x_i| + 9}{\sigma_x^2},$$
$$\left|\frac{\partial U_i(\theta)}{\partial \theta_2}\right| \le \frac{|x_i| + 3 + 3}{\sigma_x^2} \le \frac{|x_i| + 6}{\sigma_x^2}.$$

Therefore we can set $M(\theta, \theta') = \|\theta - \theta'\|_2$ and

$$c_i = \sqrt{\left(\frac{2\,|x_i| + 9}{\sigma_x^2}\right)^2 + \left(\frac{|x_i| + 6}{\sigma_x^2}\right)^2}.$$

To use the control variate in SMH, we need to compute the second derivatives

$$\begin{aligned}
\frac{\partial^2 U_i(\theta)}{\partial^2 \theta_1} &= \frac{1}{(E_1+E_2)^2}\cdot\left(E_1\cdot\frac{x_i-\theta_1}{\sigma_x^2}+E_2\cdot\frac{x_i-\theta_1-\theta_2}{\sigma_x^2}\right)^2 \\
&\quad-\left[E_1\cdot\left(\left(\frac{x_i-\theta_1}{\sigma_x^2}\right)^2-\frac{1}{\sigma_x^2}\right)+E_2\cdot\left(\left(\frac{x_i-\theta_1-\theta_2}{\sigma_x^2}\right)^2-\frac{1}{\sigma_x^2}\right)\right]\cdot\frac{1}{E_1+E_2} \\
\frac{\partial^2 U_i(\theta)}{\partial\theta_1\partial\theta_2} &= \frac{1}{(E_1+E_2)^2}\cdot\left(E_2\cdot\left(\frac{x_i-\theta_1-\theta_2}{\sigma_x^2}\right)\right)\cdot\left(E_1\cdot\frac{x_i-\theta_1}{\sigma_x^2}+E_2\cdot\frac{x_i-\theta_1-\theta_2}{\sigma_x^2}\right) \\
&\quad-\left[E_2\left(\left(\frac{x_i-\theta_1-\theta_2}{\sigma_x^2}\right)^2-\frac{1}{\sigma_x^2}\right)\right]\cdot\frac{1}{E_1+E_2} \\
\frac{\partial^2 U_i(\theta)}{\partial^2 \theta_2} &= \frac{1}{(E_1+E_2)^2}\cdot\left(E_1\cdot\frac{x_i-\theta_1}{\sigma_x^2}+E_2\cdot\frac{x_i-\theta_1-\theta_2}{\sigma_x^2}\right)^2 \\
&\quad-\left[E_2\cdot\left(\left(\frac{x_i-\theta_1-\theta_2}{\sigma_x^2}\right)^2-\frac{1}{\sigma_x^2}\right)\right]\cdot\frac{1}{E_1+E_2}.
\end{aligned}$$

Given the parameter space, we have the upper bounds

$$\begin{aligned}
\left|\frac{\partial^2 U_i(\theta)}{\partial^2\theta_1}\right| &\le \left(\frac{2\,|x_i|+9}{\sigma_x^2}\right)^2+\left(\frac{|x_i|+3}{\sigma_x^2}\right)^2+\left(\frac{|x_i|+6}{\sigma_x^2}\right)^2+\frac{2}{\sigma_x^2} \\
\left|\frac{\partial^2 U_i(\theta)}{\partial\theta_1\partial\theta_2}\right| &\le \frac{2\,|x_i|+9}{\sigma_x^2}\cdot\frac{|x_i|+6}{\sigma_x^2}+\left(\frac{|x_i|+6}{\sigma_x^2}\right)^2+\frac{1}{\sigma_x^2} \\
\left|\frac{\partial^2 U_i(\theta)}{\partial^2\theta_2}\right| &\le \left(\frac{2\,|x_i|+9}{\sigma_x^2}\right)^2+\left(\frac{|x_i|+6}{\sigma_x^2}\right)^2+\frac{1}{\sigma_x^2}.
\end{aligned}$$

It follows

$$\bar{U}_{2,i}=\left(\frac{2\,|x_i|+9}{\sigma_x^2}\right)^2+\left(\frac{|x_i|+3}{\sigma_x^2}\right)^2+\left(\frac{|x_i|+6}{\sigma_x^2}\right)^2+\frac{2}{\sigma_x^2}.$$

which is required in SMH-1.

To get the lower bounds in FlyMC, we use the first-order Taylor expansion for $U_i(\theta)$. Higher order approximation is possible but would require heavier computation. By Taylor expansion,

$$U_i(\theta)=U_i(\theta^0)+\nabla U_i(\theta^0)^\intercal(\theta-\theta^0)+\frac{1}{2}(\theta-\theta^0)^\intercal\nabla^2 U_i(c)(\theta-\theta^0)$$

where c is between θ and θ^0.

Then we can define $\log B_i(\theta)$ in FlyMC as the follows

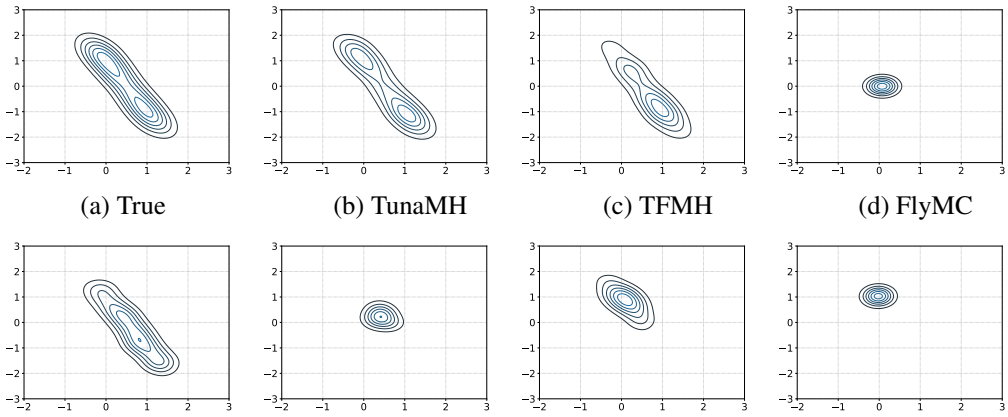
$$\begin{aligned}
\log B_i(\theta) &= -U_i(\theta^0)-\nabla U_i(\theta^0)^\intercal(\theta-\theta^0)-\frac{1}{2}\cdot\max_c\left\|\nabla^2 U_i(c)\right\|_1\cdot\left\|\theta-\theta^0\right\|_1^2 \\
&= -U_i(\theta^0)-\nabla U_i(\theta^0)^\intercal(\theta-\theta^0)-\frac{1}{2}\cdot\bar{U}_{2,i}\cdot\left\|\theta-\theta^0\right\|_1^2.
\end{aligned}$$

The sum of $\log B_i$ is

$$\sum_{i=1}^N\log B_i(\theta)=-N\cdot U_i(\theta^0)-\left(\sum_{i=1}^N\nabla U_i(\theta^0)\right)^\intercal(\theta-\theta^0)-\frac{1}{2}\cdot\sum_{i=1}^N\bar{U}_{2,i}\cdot\left\|\theta-\theta^0\right\|_1^2.$$

We set θ^0 to be 0 and the MAP solution in standard and MAP-tuned FlyMC respectively.

We tune the stepsize of each method to reach the acceptance rate 60% and the value of stepsize is summarized in Table 2 and 3. We set $\chi=10^{-4}$ in TunaMH and $q_{d\to b}=0.01$ in FlyMC and FlyMC-MAP. We compute the symmetric KL between the run-average density estimate and the true distribution. Since this is a two-dimensional problem, we are able to visualize the density estimate. As shown in Figure 6, we plot the density estimate after running the method for 1 second. It is clear to see that the density estimate of TunaMH is close to the truth whereas all other methods are unable to provide accurate density estimate given the time budget.



(a) True (b) TunaMH (c) TFMH (d) FlyMC

(e) PoissonMH (f) MH (g) SMH-1 (h) FlyMC-MAP

Figure 6: Visualization of the density estimate after 1 second.

J.4 Logistic Regression on MNIST

MNIST with only 7s and 9s images contains 12214 training data and 2037 test data. Let h be the sigmoid function. Let the label $y_i \in \{0, 1\}$, then the model in logistic regression (LR) is

$$p(y_i = 1) = h(\theta^\intercal x_i) = \frac{1}{1 + \exp\left(-\theta^\intercal x_i\right)}.$$

It follows that

$$U_i(\theta) = -y_i \log h\left(\theta^\intercal x_i\right) - (1 - y_i) \log h\left(-\theta^\intercal x_i\right).$$

It is easy to see that

$$\left|\partial_j U_i\right| = \left|(h(\theta^\intercal x_i) - y_i) x_{ij}\right| \leq 1 \cdot \left|x_{ij}\right|.$$

Thus we can set $M(\theta, \theta')$ to be $\|\theta - \theta'\|_2$ and c_i to be $\|x_i\|_2$. We use this bound for TunaMH, TFMH and SMH. For FlyMC, we use the same bound on logistic regression as in the FlyMC paper [20].

We set the target acceptance rate to be 60% and the resulted stepsize is reported in Table 2. We set $q_{d \to b}$ to be 0.1 following [20].