Mixed Hamiltonian Monte Carlo for Mixed Discrete and Continuous Variables

Guangyao Zhou
Vicarious AI
Union City, CA 94587
stannis@vicarious.com

Abstract

Hamiltonian Monte Carlo (HMC) has emerged as a powerful Markov Chain Monte Carlo (MCMC) method to sample from complex continuous distributions. However, a fundamental limitation of HMC is that it can't be applied to distributions with mixed discrete and continuous variables. In this paper, we propose mixed HMC (M-HMC) as a general framework to address this limitation. M-HMC is a novel family of MCMC algorithms that evolves the discrete and continuous variables in tandem, allowing more frequent updates of discrete variables while maintaining HMC's ability to suppress random-walk behavior. We establish M-HMC's theoretical properties, and present an efficient implementation with Laplace momentum that introduces minimal overhead compared to existing HMC methods. The superior performances of M-HMC over existing methods are demonstrated with numerical experiments on Gaussian mixture models (GMMs), variable selection in Bayesian logistic regression (BLR), and correlated topic models (CTMs).

1 Introduction

Markov chain Monte Carlo (MCMC) is one of the most powerful methods for sampling from probability distributions. The Metropolis-Hastings (MH) algorithm is a commonly used general-purpose MCMC method, yet is inefficient for complex, high-dimensional distributions because of the random walk nature of its movements. Recently, Hamiltonian Monte Carlo (HMC) [13, 22, 2] has emerged as a powerful alternative to MH for complex continuous distributions due to its ability to follow the curvature of target distributions using gradients information and make distant proposals with high acceptance probabilities. It enjoyed remarkable empirical success, and (along with its popular variant No-U-Turn Sampler (NUTS) [16]) is adopted as the dominant inference strategy in many probabilistic programming systems [8, 27, 3, 25, 14, 10]. However, a fundamental limitation of HMC is that it can't be applied to distributions with mixed discrete and continuous variables.

One existing approach for addressing this limitation involves integrating out the discrete variables(e.g. in Stan[8], Pyro[3]), yet it's only applicable on a small-scale, and can't always be carried out automatically. Another approach involves alternating between updating continuous variables using HMC/NUTS and discrete variables using generic MCMC methods (e.g. in PyMC3[27], Turing.jl[14]). However, to suppress random walk behavior in HMC, long trajectories are needed. As a result, the discrete variables can only be updated infrequently, limiting the efficiency of this approach. The most promising approach involves updating the discrete and continuous variables in tandem. Since naively making MH updates of discrete variables within HMC results in incorrect samples [22], novel variants of HMC (e.g. discontinuous HMC (DHMC)[23, 29], probabilistic path HMC (PPHMC) [12]) are developed. However, these methods can't be easily generalized to complicated discrete state spaces (DHMC works best for ordinal discrete parameters, PPHMC is only applicable to phylogenetic trees), and as we show in Section 4, DHMC's embedding and algorithmic structure are inefficient.

Preprint. Under review.

In this paper, we propose mixed HMC (M-HMC), a novel family of MCMC algorithms that better addresses this limitation. M-HMC provides a general mechanism, applicable to any distributions with mixed support, to evolve the discrete and continuous variables in tandem. It allows more frequent updates of discrete variables while maintaining HMC's ability to suppress random walk behavior, and adopts an efficient implementation (using Laplace momentum) that introduces minimal overhead compared to existing HMC methods. In Section 2, we review HMC and some of its variants involving discrete variables, before presenting M-HMC and rigorously establishing its correctness. We present the efficient implementation of M-HMC with Laplace momentum in Section 3, and demonstrate M-HMC's superior performance over existing methods with numerical experiments on GMMs, variable selection in BLR and CTMs in Section 4, before concluding with discussions in Section 5.

2 Mixed Hamiltonian Monte Carlo (M-HMC)

Our goal is to sample from a target distribution $\pi(x, q^{\mathcal{C}}) \propto e^{-U(x,q^{\mathcal{C}})}$ on $\Omega \times \mathbb{R}^{N_{\mathcal{C}}}$ with mixed discrete variables $x = (x_1, \ldots, x_{N_{\mathcal{D}}}) \in \Omega$ and continuous variables $q^{\mathcal{C}} = (q^{\mathcal{C}}_1, \ldots, q^{\mathcal{C}}_{N_{\mathcal{C}}}) \in \mathbb{R}^{N_{\mathcal{C}}}$.

2.1 Review of HMC and Some Variants of HMC That Involve Discrete Variables

For a continuous target distribution $\pi(q^{\mathcal{C}}) \propto e^{-U(q^{\mathcal{C}})}$, the original HMC introduces auxiliary momentum variables $p^{\mathcal{C}} \in \mathbb{R}^{N_{\mathcal{C}}}$ associated with a kinetic energy function $K^{\mathcal{C}}$, and draws samples for $\pi(q^{\mathcal{C}})$ by simulating trajectories of Hamiltonian dynamics $\frac{\mathrm{d}q^{\mathcal{C}}(t)}{\mathrm{d}t} = \nabla K^{\mathcal{C}}(p^{\mathcal{C}})$, $\frac{\mathrm{d}p^{\mathcal{C}}(t)}{\mathrm{d}t} = -\nabla U(q^{\mathcal{C}})$ to sample from the joint distribution $\pi(q^{\mathcal{C}})\chi(p^{\mathcal{C}})$. Here $\chi(p^{\mathcal{C}}) \propto e^{-K^{\mathcal{C}}(p^{\mathcal{C}})}$.

A foundational tool in applying HMC to distributions with discrete variables is the discontinuous variant of HMC, which operates on piecewise continuous potentials. This was first studied in [24], where the authors proposed binary HMC to sample from binary distributions $\pi(x) \propto e^{-U(x)}$ for $x \in \Omega = \{-1, 1\}^{N_{\mathcal{D}}}$. The idea is to embed the binary variables x into the continuum by introducing auxiliary location variables $q^{\mathcal{D}} \in \mathbb{R}^{N_{\mathcal{D}}}$ associated with a conditional distribution $\psi(q^{\mathcal{D}}|x)$. If $sign(q^{\mathcal{D}}_i) \neq x_i$ for any $i = 1, \cdots, N^{\mathcal{D}}$, $\psi(q^{\mathcal{D}}|x) = 0$. If $sign(q^{\mathcal{D}}_i) = x_i, \forall i = 1, \cdots, N^{\mathcal{D}}$, two conditional distributions were considered: $\psi(q^{\mathcal{D}}|x) \propto e^{-\frac{1}{2}\sum_{i=1}^{N_d}(q^{\mathcal{D}}_i)^2}$ for Gaussian binary HMC, and $\psi(q^{\mathcal{D}}|x) \propto e^{-\sum_{i=1}^{N_{\mathcal{D}}}|q^{\mathcal{D}}_i|}$ for exponential binary HMC. Binary HMC introduces auxiliary momentum variables $p^{\mathcal{D}} \in \mathbb{R}^{N_{\mathcal{D}}}$ associated with a kinetic energy $K^{\mathcal{D}}(p^{\mathcal{D}}) = \sum_{i=1}^{N_{\mathcal{D}}} \frac{(p^{\mathcal{D}}_i)^2}{2}$, and operates on the joint distribution $\Psi(q^{\mathcal{D}})\nu(p^{\mathcal{D}})$ on the expanded state space $\Sigma = \mathbb{R}^{N_{\mathcal{D}}} \times \mathbb{R}^{N_{\mathcal{D}}}$. The piecewise continuous joint distribution $\Psi(q^{\mathcal{D}}) = \sum_{x\in\Omega} \pi(x)\psi(q^{\mathcal{D}}|x)$ gives rise to a piecewise continuous potential, and [24] developed a way to exactly integrate Hamiltonian dynamics for the joint distribution $\Psi(q^{\mathcal{D}})\nu(p^{\mathcal{D}})$, taking into account discontinuities in the potential. The coupling of x and $q^{\mathcal{D}}$ through the signs of $q^{\mathcal{D}}$ in ψ means we can read out samples for x from the signs of samples we get for $q^{\mathcal{D}}$ from binary HMC. We later show (see Appendix D for precise statements and proofs) that binary HMC is a special case of M-HMC, with Gaussian and exponential binary HMC corresponding to two particular choices of $k^{\mathcal{D}}$ (defined in Section 2.2) in M-HMC.

[21] later made the key observation that we can analytically integrate Hamiltonian dynamics with piecewise continuous potentials near a discontinuity while perserving the total (potential and kinetic) energy. The trick is to calculate the potential energy difference ΔE across an encountered discontinuity, and either refract (replace $p^{\mathcal{D}}_\perp$, the component of $p^{\mathcal{D}}$ that's perpendicular to the discontinuity boundary, by $\sqrt{\frac{1}{2}||p^{\mathcal{D}}_\perp||^2 - \Delta E}(p^{\mathcal{D}}_\perp/||p^{\mathcal{D}}_\perp||)$) if there's enough kinetic energy ($\frac{1}{2}||p^{\mathcal{D}}_\perp||^2 > \Delta E$), or reflect (replace $p^{\mathcal{D}}_\perp$ by $-p^{\mathcal{D}}_\perp$) if there isn't enough kinetic energy ($\frac{1}{2}||p^{\mathcal{D}}_\perp||^2 \leq \Delta E$). Reflection/refraction HMC (RRHMC) combines the above observation with the leapfrog integrator, and generalizes binary HMC to arbitrary piecewise continuous potentials with discontinuities across affine boundaries. However, RRHMC is computationally expensive due to the need to detect all encountered discontinuities, and by itself can't directly handle distributions with mixed support.

[23] proposed DHMC as an attempt to address some of the issues of RRHMC. It uses Laplace momentum to avoid the need to detect encountered discontinuities, and handles discrete variables (which it assumes take positive integer values, i.e. $x \in \mathbb{Z}_+^{N_{\mathcal{D}}}$) by an embedding into 1D spaces ($x_i = n \iff q^{\mathcal{D}}_i \in (a_n, a_{n+1}], 0 = a_1 \leq a_2 \leq \cdots$) and a coordinate-wise integrator (which is shown to be a special case of M-HMC with Laplace momentum in Section 3). In Section 4, using numerical experiments, we show that DHMC's embedding is inefficient and sensitive to ordering, and it can't easily generalize to more complicated discrete state spaces; furthermore, its need to update all discrete variables at every step makes it computationally expensive for long HMC trajectories.

2.2 The General Framework of M-HMC

Formally, M-HMC operates on the expanded state space $\Omega \times \Sigma$, where $\Sigma = \mathbb{T}^{N_\mathcal{D}} \times \mathbb{R}^{N_\mathcal{D}} \times \mathbb{R}^{N_\mathcal{C}} \times \mathbb{R}^{N_\mathcal{C}}$ with auxiliary location variables $q^\mathcal{D} \in \mathbb{T}^{N_\mathcal{D}}$ and momentum variables $p^\mathcal{D} \in \mathbb{R}^{N_\mathcal{D}}$ for $x \in \Omega$, and auxiliary momentum variables $p^\mathcal{C} \in \mathbb{R}^{N_\mathcal{C}}$ for $q^\mathcal{C} \in \mathbb{R}^{N_\mathcal{C}}$. Here $\mathbb{T}^{N_\mathcal{D}} = \mathbb{R}^{N_\mathcal{D}}/\tau\mathbb{Z}^{N_\mathcal{D}}$ denotes the $N_\mathcal{D}$-dimensional flat torus, and is identified as the hypercube $[0, \tau]^{N_\mathcal{D}}$ with the 0's and τ's in different dimensions glued together. We associate $q^\mathcal{D}$ with a flat potential $U^\mathcal{D}(q^\mathcal{D}) = 0, \forall q^\mathcal{D} \in \mathbb{T}^{N_\mathcal{D}}$ and $p^\mathcal{D}$ with a kinetic energy $K^\mathcal{D}(p^\mathcal{D}) = \sum_{i=1}^{N_\mathcal{D}} k^\mathcal{D}(p_i^\mathcal{D}), p^\mathcal{D} \in \mathbb{R}^{N_\mathcal{D}}$ where $k^\mathcal{D} : \mathbb{R} \to \mathbb{R}^+$ is some kinetic energy, and $p^\mathcal{C}$ with a kinetic energy[1] $K^\mathcal{C} : \mathbb{R}^{N_\mathcal{C}} \to \mathbb{R}^+$. Use $Q_i, i = 1, \ldots, N_\mathcal{D}$ to denote $N_\mathcal{D}$ irreducible single-site MH proposals, where $Q_i(\tilde{x}|x) > 0$ only when $\tilde{x}_j = x_j, \forall j \neq i$, and $Q_i(x|x) = 0$. In other words, Q_i only changes the value of x_i, and always moves away from x.

Intuitively, M-HMC also "embeds" the discrete variables x into the continuum (in the form of $q^\mathcal{D}$). However, the "embedding" is done by combining the original discrete state space Ω with the flat torus $\mathbb{T}^{N_\mathcal{D}}$: instead of relying on the embedding structure (e.g. the sign of $q_i^\mathcal{D}$ in binary HMC, or the value of $q_i^\mathcal{D}$ in DHMC) to determine x from $q^\mathcal{D}$, in M-HMC we explicitly record the values of x as we can't read out x from $q^\mathcal{D}$. $\mathbb{T}^{N_\mathcal{D}}$ bridges x with the continuous Hamiltonian dynamics, and functions like a "clock": the system evolves $q_i^\mathcal{D}$ with speed determined by the momentum $p_i^\mathcal{D}$ and makes an attempt to move to a different state for x_i when $q_i^\mathcal{D}$ reaches 0 or τ. Such mixed embedding makes M-HMC easily applicable to arbitrary discrete state spaces, but also prevents the use of methods like RRHMC. For this reason, M-HMC introduces probabilistic proposals Q_i's to move around Ω, and probabilistic reflection/refraction actions to handle discontinuities (which now happen at $q_i^\mathcal{D} \in \{0, \tau\}$).

More concretely, M-HMC evolves according to the following dynamics: If $q^\mathcal{D} \in (0, \tau)^{N_\mathcal{D}}$, x remains unchanged, and $q^\mathcal{D}, p^\mathcal{D}$ and $q^\mathcal{C}, p^\mathcal{C}$ follow the Hamiltonian dynamics

$$\text{Discrete} \begin{cases} \frac{\mathrm{d}q_i^\mathcal{D}(t)}{\mathrm{d}t} = (k^\mathcal{D})'(p_i^\mathcal{D}), i = 1, \ldots, N_\mathcal{D} \\ \frac{\mathrm{d}p^\mathcal{D}(t)}{\mathrm{d}t} = -\nabla U^\mathcal{D}(q^\mathcal{D}) = 0 \end{cases} \quad \text{Continuous} \begin{cases} \frac{\mathrm{d}q^\mathcal{C}(t)}{\mathrm{d}t} = \nabla K^\mathcal{C}(p^\mathcal{C}) \\ \frac{\mathrm{d}p^\mathcal{C}(t)}{\mathrm{d}t} = -\nabla_{q^\mathcal{C}} U(x, q^\mathcal{C}) \end{cases} \tag{1}$$

If $q^\mathcal{D}$ hits either 0 or τ at site j (i.e. $q_j^\mathcal{D} \in \{0, \tau\}$), we propose a new $\tilde{x} \sim Q_j(\cdot|x)$, calculate $\Delta E = \log \frac{\pi(x, q^\mathcal{C}) Q_j(\tilde{x}|x)}{\pi(\tilde{x}, q^\mathcal{C}) Q_j(x|\tilde{x})}$ and either refract ($x \leftarrow \tilde{x}, q_j^\mathcal{D} \leftarrow \tau - q_j^\mathcal{D}, p_j^\mathcal{D} \leftarrow \text{sign}(p_j^\mathcal{D})(k^\mathcal{D})^{-1}(k^\mathcal{D}(p_j^\mathcal{D}) - \Delta E)$) if there's enough kinetic energy ($k^\mathcal{D}(p_j^\mathcal{D}) > \Delta E$), or reflect ($(x, q_j^\mathcal{D}, p_j^\mathcal{D}) \leftarrow (x, q_j^\mathcal{D}, -p_j^\mathcal{D})$) if there isn't enough kinetic energy ($k^\mathcal{D}(p_j^\mathcal{D}) \leq \Delta E$). For the discrete component, because of the flat potential $U^\mathcal{D}$, we can exactly integrate the Hamiltonian dynamics with arbitrary $k^\mathcal{D}$. For the continuous component, given a discrete state x and some time $t > 0$, use $I(\cdot, \cdot, t|x, U, K^\mathcal{C}) : \mathbb{R}^{N_\mathcal{C}} \times \mathbb{R}^{N_\mathcal{C}} \times \mathbb{R}^+ \to \mathbb{R}^{N_\mathcal{C}} \times \mathbb{R}^{N_\mathcal{C}}$ to denote a reversible, volume-preserving integrator[2] that's irreducible and aperiodic and approximately evolves the continuous part of the Hamiltonian dynamics in Equation 1 for time t. Given the current state $x^{(0)}, q^{\mathcal{C}(0)}$, a full M-HMC iteration first resamples the auxiliary variables $q_i^{\mathcal{D}(0)} \sim \text{Uniform}([0, \tau]), p_i^{\mathcal{D}(0)} \sim \nu(p) \propto e^{-k^\mathcal{D}(p)}$ for $i = 1, \ldots, N_\mathcal{D}$, $p^{\mathcal{C}(0)} \sim \chi(p) \propto e^{-K^\mathcal{C}(p)}$, then evolves the discrete variables (using exact integration) and continuous variables (using the integrator I) in tandem for a given time T, before making a final MH correction like in regular HMC. A detailed description of a full M-HMC iteration is given in Appendix A.

The essential idea of M-HMC is to more frequently update x within HMC, which, if done naively, results in incorrect samples. The benefits of more frequent x updates will be shown in Section 4. Note that if we use conditional distributions for Q_i, ΔE would always be 0, the discrete dynamics in Equation 1 plays no role, and M-HMC reduces to the incorrect case of naively making Gibbs updates within HMC. However, the requirement $Q_i(x|x) = 0$ (which is more efficient [19]) means Q_i is always sufficiently different from the conditional distribution and guarantees correctness of M-HMC.

2.3 M-HMC samples from the correct distribution

For notational simplicity, define $\Theta = (q^\mathcal{D}, p^\mathcal{D}, q^\mathcal{C}, p^\mathcal{C})$. To prove M-HMC samples from the correct distribution $\pi(x, q^\mathcal{C})$, we show that a full M-HMC iteration preserves the joint invariant distribution $\varphi((x, \Theta)) \propto \pi(x, q^\mathcal{C}) e^{-\left[U^\mathcal{D}(q^\mathcal{D}) + K^\mathcal{D}(p^\mathcal{D}) + K^\mathcal{C}(p^\mathcal{C})\right]}$ and establish its irreducibility and aperiodicity. At each iteration, the resampling can be seen as a Gibbs step, where we resample the auxiliary variables $q^\mathcal{D}, p^\mathcal{D}, q^\mathcal{C}$ from their conditional distribution given $x, q^\mathcal{C}$. This obviously preserves φ. So

[1] The simplest choice for $K^\mathcal{C}$ is $K^\mathcal{C}(p^\mathcal{C}) = \sum_{i=1}^{N_\mathcal{C}} \frac{(p_i^\mathcal{C})^2}{2}$, but M-HMC can work with any kinetic energy.

[2] An example is the commonly used leapfrog integrator

we only need to prove detailed balance of the evolution of x and $q^{\mathcal{C}}$ in an M-HMC iteration (described in detail in the *MixedHMC* function in Appendix A) w.r.t. φ. Formally, $\forall T > 0$, *MixedHMC* defines a transition probability kernel $R_T((x,\Theta),B) = \mathbb{P}\left(\text{MixedHMC}(x,\Theta,T) \in B\right), \forall (x,\Theta) \in \Omega \times \Sigma$ and $B \subset \Omega \times \Sigma$ measurable. For all $A \subset \Omega \times \Sigma$ measurable, $\Theta \in \Sigma$, define $A(\Theta) = \{x \in \Omega : (x,\Theta) \in A\}$.

Theorem 1. ***(Detailed Balance)*** *The MixedHMC function (Appendix A) satisfies detailed balance w.r.t. the joint invariant distribution φ, i.e. for any measurable sets $A, B \subset \Omega \times \Sigma$,*

$$\int_{\Sigma} \sum_{x \in A(\Theta)} R((x,\Theta),B)\varphi((x,\Theta))\mathrm{d}\Theta = \int_{\Sigma} \sum_{x \in B(\Theta)} R((x,\Theta),A)\varphi((x,\Theta))\mathrm{d}\Theta$$

We defer more details and the proof to the appendix. Combining the above theorem with irreducibility and aperiodicity (which follow from irreducibility and aperiodicity of integrator I, and the irreducibility of the Q_i's) proves that M-HMC samples from the correct distribution $\pi(x, q^{\mathcal{C}})$.

3 Implementation with Laplace Momentum

Algorithm 1 M-HMC with Laplace momentum

Require: U, target potential; $Q_i, i = 1, \ldots, N_{\mathcal{D}}$, single-site proposals; ε, maximum step size; L, # of times to update discrete variables; $n_{\mathcal{D}}$, # of discrete sites to update each time

input $x^{(0)}$, current discrete state; $q^{\mathcal{C}(0)}$, current continuous location; T, travel time

output x, next discrete state; $q^{\mathcal{C}}$, next continuous location

function MixedHMCLaplaceMomentum($x^{(0)}, q^{\mathcal{C}(0)}, T$)
  $k_i^{\mathcal{D}(0)} \sim \text{Exponential}(1), i = 1, \ldots, N_{\mathcal{D}}$, $p_i^{\mathcal{C}(0)} \sim N(0,1), i = 1, \ldots, N_{\mathcal{C}}$
  $x \leftarrow x^{(0)}$, $k^{\mathcal{D}} \leftarrow k^{\mathcal{D}(0)}$, $q^{\mathcal{C}} \leftarrow q^{\mathcal{C}(0)}$, $p^{\mathcal{C}} \leftarrow p^{\mathcal{C}(0)}$
  $\Lambda \sim \text{RandomPermutation}(\{1, \ldots, N_{\mathcal{D}}\})$, $(\eta, M) \leftarrow$ *GetStepSizesNSteps*$(\varepsilon, T, L, N_{\mathcal{D}}, n_{\mathcal{D}})$
  for t **from** 1 **to** L **do**
    for s from 1 to M_t **do**
      $p^{\mathcal{C}} \leftarrow p^{\mathcal{C}} - \eta_t \nabla_{q^{\mathcal{C}}} U(x, q^{\mathcal{C}})/2$; $q^{\mathcal{C}} \leftarrow q^{\mathcal{C}} + \eta_t p^{\mathcal{C}}$; $p^{\mathcal{C}} \leftarrow p^{\mathcal{C}} - \eta_t \nabla_{q^{\mathcal{C}}} U(x, q^{\mathcal{C}})/2$
    end for
    for s from 1 to $n_{\mathcal{D}}$ **do**
      $j \leftarrow \Lambda_{[(t-1)n_{\mathcal{D}}+s] \bmod N_{\mathcal{D}}}$; $\tilde{x} \sim Q_j(\cdot|x)$; $\Delta E \leftarrow \log \frac{e^{-U(x,q^{\mathcal{C}})} Q_j(\tilde{x}|x)}{e^{-U(\tilde{x},q^{\mathcal{C}})} Q_j(x|\tilde{x})}$
      if $k_j^{\mathcal{D}} > \Delta E$ **then** $x \leftarrow \tilde{x}$, $k_j^{\mathcal{D}} \leftarrow k_j^{\mathcal{D}} - \Delta E$ **end if**
    end for
  end for
  $E \leftarrow U\left(x, q^{\mathcal{C}}\right) + \sum_{i=1}^{N_{\mathcal{D}}} k_i^{\mathcal{D}} + K^{\mathcal{C}}(p^{\mathcal{C}})$, $E^{(0)} \leftarrow U\left(x^{(0)}, q^{\mathcal{C}(0)}\right) + \sum_{i=1}^{N_{\mathcal{D}}} k_i^{\mathcal{D}(0)} + K^{\mathcal{C}}(p^{\mathcal{C}(0)})$
  if Uniform($[0,1]$) $>= e^{-(E-E^{(0)})}$ **then** $x \leftarrow x^{(0)}$, $q^{\mathcal{C}} \leftarrow q^{\mathcal{C}(0)}$ **end if**
  return $x, q^{\mathcal{C}}$
end function

function GetStepSizesNSteps($\varepsilon, T, L, N_{\mathcal{D}}, n_{\mathcal{D}}$)
  $\Phi \sim \text{Dirichlet}_{N_{\mathcal{D}}+1}(1)$; $\Phi_1 \leftarrow \Phi_1 + \Phi_{N_{\mathcal{D}}+1}$
  $\eta_t \leftarrow \sum_{s=1}^{n_{\mathcal{D}}} \Phi_{[(t-1)n_{\mathcal{D}}+s] \bmod N_{\mathcal{D}}}, t = 1, \ldots, L$; $\eta_1 \leftarrow \eta_1 - \Phi_{N_{\mathcal{D}}+1}$
  $\eta_t \leftarrow T\eta_t / \sum_{s=1}^{L} \eta_s, t = 1, \ldots, L$; $M_t \leftarrow \lceil \eta_t/\varepsilon \rceil, t = 1, \ldots, L$; $\eta_t \leftarrow \eta_t / M_t, t = 1, \ldots, L$
  return η, M
end function

In this section, we present an efficient implementation of M-HMC using Laplace momentum ($k^{\mathcal{D}}(p) = |p|$). While M-HMC works with any $k^{\mathcal{D}}$, using a general $k^{\mathcal{D}}$ requires detection of all encountered discontinuities, similar to RRHMC. However, with Laplace momentum, $q_i^{\mathcal{D}}$'s speed (given by $(k^{\mathcal{D}})'(p_i^{\mathcal{D}})$) becomes a constant 1, and we can precompute the occurences of all discontinuities at the beginning of each M-HMC iteration. In particular, we no longer need to explicitly record $q^{\mathcal{D}}, p^{\mathcal{D}}$, but can instead keep track of only the kinetic energies associated with x. Note that we need to use τ to orchestrate discrete and continuous updates. Here, instead of explicitly setting τ, we propose to alternate discrete and continuous updates, specifying the total travel time T, the number of discrete updates L, and the number of discrete variables to update each time $n_{\mathcal{D}}$. The step sizes are properly scaled (effectively setting τ) to match the desired total travel time T. To reduce

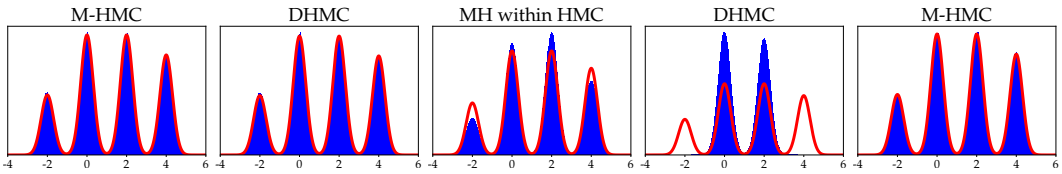


Figure 1: Visualizations of the empirical density and true density on 1D GMM. **Figure 1a**: M-HMC, 10^6 samples, $\mu = (-2, 0, 2, 4)^T$. **Figure 1b**: DHMC, 5×10^6 samples, $\mu = (-2, 0, 2, 4)^T$; **Figure 1c**: Naive MH updates within HMC, 10^7 samples, $\mu = (-2, 0, 2, 4)$.**Figure 1d**: DHMC, 10^7 samples, $\mu = (-2, 2, 0, 4)$.**Figure 1e**: M-HMC, 10^6 samples, $\mu = (-2, 2, 0, 4)$

integration error and ensure a high acceptance rate, we specify a maximum step size ε. A detailed description of the efficient implementation is given in Algorithm 1. See Appendix B for a detailed discussion on how each part of Algorithm 1 can be derived from the original *MixedHMC* function in Appendix A. The coordinate-wise integrator in DHMC corresponds to setting $n_{\mathcal{D}} = N_{\mathcal{D}}$ with Q_i's that are implicitly specified through embedding. However, the need to update all discrete variables at each step is computationally expensive for long HMC trajectories. In contrast, M-HMC can flexibly orchestrate discrete and continuous updates depending on models at hand, and introduces minimal overhead (x updates that are usually cheap) compared to existing HMC methods.

4 Numerical experiments

In this section, we empirically verify the correctness of M-HMC, and compare the performances of various samplers for GMMs, variable selection in BLR, and CTM. In addition to DHMC and M-HMC, we also compare NUTS (using Numpyro [25], for GMMs), NUTS-within-Gibbs (NwG, implemented as a compound step in PyMC3 [27]), and specialized Gibbs samplers (adapting [26] for variable selection in BLR, and adapting [9] for CTM). Our implementations of DHMC and M-HMC rely on *JAX* [6]. For Gibbs samplers, we combine *NUMBA* [28] with the package *pypolyagamma*[3].

For all three models, a common performance measure is the minimum relative effective sample size (MRESS), i.e. the minimum ESS over all dimensions, normalized by the number of samples. We use the function *ess* (with default settings) from the Python package *arviz* [18] to estimate MRESS. Our MRESS is always estimated using multiple independent chains. For experiments with M-HMC, we use $Q_j(\tilde{x}|x) \propto \pi(\tilde{x})\rho_j(\tilde{x}|x)$, where $\rho_j(\tilde{x}|x) = \begin{cases} 1 & \text{if } \tilde{x}_j \neq x_j, \tilde{x}_i = x_i, i \neq j \\ 0 & \text{otherwise} \end{cases}$ [19], as required in Section 2. Such efficient [19] proposals are also used in other samplers to ensure fair comparison.

4.1 Illustrative experiments on 1D Gaussian Mixture Model (GMM)

We start with sanity checks on a 1D GMM with 4 mixture componets (denoted by $z \in \{1, 2, 3, 4\}$). Use $x \in \mathbb{R}$ to denote the continuous variable. We are interested in $p(z, x) = \phi_z N(x|\mu_z, \Sigma)$, where $\phi_1 = 0.15, \phi_2 = \phi_3 = 0.3, \phi_4 = 0.25, \Sigma = 0.1$, and $\mu_1 = -2, \mu_2 = 0, \mu_3 = 2, \mu = 4$.

Figures 1a and 1b show that M-HMC and DHMC sample from the correct distribution for the 1D GMM. Note that, with 10^6 samples, M-HMC's empirical density already perfectly matches the true density, while for the inefficient DHMC, 5×10^6 samples are needed before we can get a good match.

We further show that naively making MH updates within an HMC step doesn't work. For illustration purposes, we put together a simple Python function *naive_mixed_hmc* (see Appendix C). *use_k=False* corresponds to naively making MH, while *use_k=True* corresponds to M-HMC. As shown in Figure 1c, using *use_k=False*, even with 10^7 samples, the empirical density still differs significantly from the true distribution. In contrast, 10^6 samples generated using *use_k=True* already gives us a perfect match, as shown in Figure 1a.

Finally, we note that DHMC is sensitive to ordering of the discrete states due to its 1D embedding. This is demonstrated with a simple experiment where we apply DHMC to the 1D GMM, but instead with $\mu_2 = 2, \mu_3 = 0$. While the underlying model remains exactly the same, as shown in Figure 1d, DHMC failed to sample all the components even after 10^7 samples, even though it can get an good fit of the true distribution with only 5×10^6 samples in the original setup. In contrast, M-HMC suffers no such issues, and works well in both cases with 10^6 samples, as shown in Figure 1e.

[3]For efficient sampling from Polya-Gamma distribution. `github.com/slinderman/pypolyagamma`

4.2 More Experiments on 24D Gaussian Mixture Model (GMM)

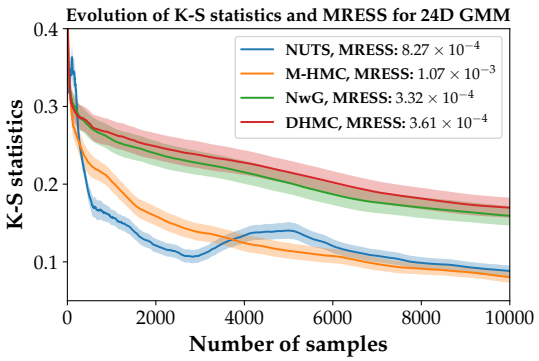


Figure 2: Evolution of K-S statistics of empirical and true samples for x_1, and MRESS for the 24D GMM. Colored regions indicate 95% confidence interval, estimated using 192 independent chains.

We experiment with a more challenging 24D GMM, also with 4 components. We again use $\phi_1 = 0.15, \phi_2 = \phi_3 = 0.3, \phi_4 = 0.25$. To avoid making the problem intractable because of multimodality, we set $\Sigma = 3I$. We use the 24 permutations of $-2, 0, 2, 4$ to specify the means of the 4 components in the 24 dimensions. We test 4 different samplers: NUTS, NwG, DHMC and M-HMC. NUTS operates on the marginal distribution $p(x)$, and serves to provide an upper bound on the performance. All other samplers operate on the joint distribution $p(z, x)$.

NUTS and NwG require no manual tuning. We favor DHMC by doing a parameter grid search and pick the setting with best MRESS for x, resulting a step-size range $(0.8, 1.0)$ and a number-of-steps range $(30, 40)$. We tune M-HMC by conducting short trial runs and inspecting the acceptance probabilities and traceplots, resulting in $\varepsilon = 1.7, L = 80, T = 136, n_{\mathcal{D}} = 1$. For each sampler, we draw 192 independent chains, with 10^4 burn-in and 10^4 actual samples in each chain.

To get a sense of the accuracy of the samplers as well as their convergence speed, we calculate the two-sided Kolmogorov-Smirnov (K-S) statistic[4] of the 24 marginal empirical distributions given by samples from the samplers and the true marginal distributions, averaged over 192 chains. We also calculate the MRESS for x to measure the efficiency of the different samplers. Figure 2 shows the evolution of the K-S statistic for x_1, with MRESS reported in legends. M-HMC clearly outperforms NwG and DHMC. Surprisingly, in terms of MRESS, M-HMC even outperforms NUTS, which explicitly integrates out z. DHMC and NwG have essentially the same performance.

4.3 Variable Selection in Bayesian Logistic Regression (BLR)

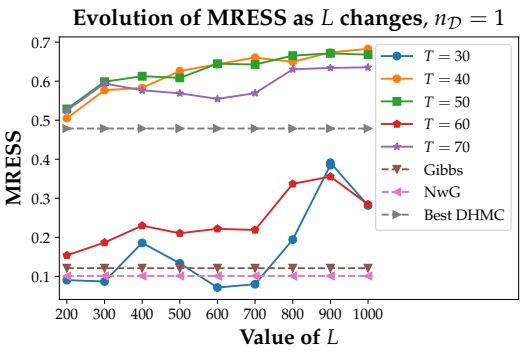
We consider the logistic regression model $y_i \sim \text{Bernoulli}\left(\sigma(X_i^T\beta)\right), i = 1, \cdots, 100$ where $X \in \mathbb{R}^{100\times 20}, \beta \in \mathbb{R}^{20}$, and $\sigma(x) = 1/(1+e^{-x})$ is the sigmoid function. For our experiments, we generate a set of synthetic data: The X_i's are generated from the multivariate Gaussian $N(0, \Sigma)$, where $\Sigma_{jj} = 3, j = 1, \cdots, 20$ and $\Sigma_{jk} = 0.3, \forall j \neq k$. For β, we set 5 randomly picked components to be 0.5, and all the other components to be 0. We generate $y_i \sim \text{Bernoulli}\left(\sigma(X_i^T\beta)\right)$. We introduce a set of binary random variables $\gamma_j, j = 1, \cdots, 20$ to indicate the presence of components of β, and put an uninformative prior $N(0, 25I)$ on β. This results in the following joint distribution on β, γ and y: $p(\beta, \gamma, y) = N(\beta|0, 25I)\prod_{i=1}^{100} p_i^{y_i}(1-p_i)^{1-y_i}$ where $p_i = \sigma(\sum_{j=1}^{20} X_{ij}\beta_j\gamma_j), i = 1, \cdots, 100$.

We are interested in a sampling-based approach to identify the relevant components of β. A natural approach [11, 30] is to sample from the posterior distribution $p(\beta, \gamma|y)$, and inspect the posterior samples of γ. This consistutes a challenging posterior sampling problem due to the lack of conjugacy and the mixed support, and prevents the wide applicability of this approach. Existing methods typically rely on data-augmentation schemes [1, 7, 17, 26]. Here we explore the applications of NwG, DHMC and M-HMC to this problem. As a baseline, we also implement a specialized Gibbs sampler, by combining the Gibbs sampler in [26] for β with a single-site systematic scan Gibbs sampler for γ.

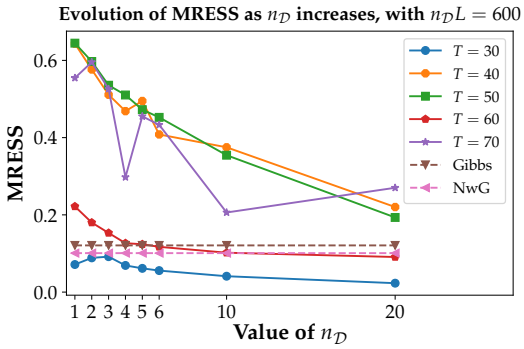
Gibbs and NwG require no manual tuning. For DHMC, we conduct a parameter grid search, and report its best performance. For M-HMC, instead of picking a particular setting, we test its performance on a variety of settings, to better understand how different components of M-HMC affect its performance. In particular, we are interested in how performance changes with the number of discrete updates L for a fixed travel time T, and with $n_{\mathcal{D}}$, the number of discrete variables to update at each discrete update while holding the total numer of single discrete variable updates $n_{\mathcal{D}}L$ a constant. For each sampler, we use 192 independent chains, each with 1000 burn-in and 2000 actual samples.

We check the accuracy of the samplers by looking at their accuracy in terms of percentage of the posterior samples for γ that agree exactly with the true model, as well as their average Hamming distance to the true model. All the tested samplers perform similarly, giving about 8.1% accuracy

[4] Calculated using *scipy.stats.ks_2samp*

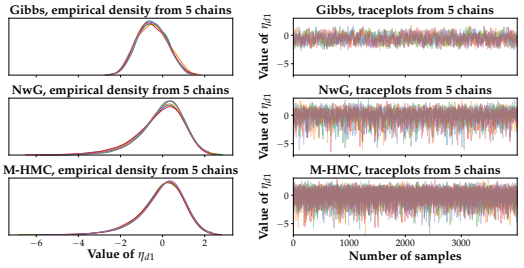


(a) Baseline MRESS for the Gibbs sampler, NwG, and best DHMC, and evolution of MRESS for M-HMC as L changes for different travel time T, with $n_{\mathcal{D}} = 1$

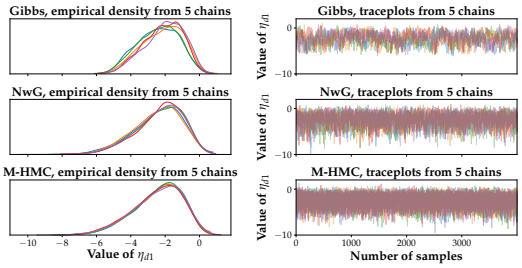


(b) Baseline MRESS for the Gibbs sampler, NwG, and evolution of MRESS for M-HMC as $n_{\mathcal{D}}$ increases for different travel time T, with $n_{\mathcal{D}}L = 600$

Figure 3: Performances (MRESS of posterior samples for β) of M-HMC as L and $n_{\mathcal{D}}$ change on variable selection for BLR, as well as baseline MRESS for the Gibbs sampler, NwG, and best DHMC



(a) Empirical density&traceplots of posterior samples from 5 chains for η_{d1}, on a document where Gibbs gives different posterior means from NwG&M-HMC



(b) Empirical density&traceplots of posterior samples from 5 chains for η_{d1}, on a document where Gibbs, NwG, M-HMC give roughly the same posterior means

Figure 4: Empirical density & traceplots of posterior samples for η_{d1} on two documents for CTM

and an average Hamming distance of around 2.2. We compare the efficiency of the 4 samplers by measuring MRESS of posterior samples for β. The results are summarized in Figures 3a, 3b. M-HMC and DHMC both significantly outperform Gibbs and NwG, demonstrating the benefits of more frequent updates of discrete variables inside HMC. However, we observe a "U-turn" [16] phenomenon, shown in Figure 3a, for both T and L: increasing T, L results in performance oscillations, suggesting that although M-HMC is capable of making distant proposals, increasing T, L beyond a certain threshold would decrease its efficiency as M-HMC starts to "double back" on itself. Nevertheless, it's clear that for fixed T, increasing L generally improves performance, again demonstrating the benefits of more frequent discrete variables updates. We also observe (Figure 3b) that $n_{\mathcal{D}} = 1$ generally gives the best performance when $n_{\mathcal{D}}L$ is held as a constant, suggesting that distributed/more frequent updates of the discrete variables is more beneficial than concentrated/less frequent updates. However, distributed/more frequent updates of discrete variables entail using a large L, which can break each leapfrog step into smaller steps, resulting in more (potentially expensive) gradients evaluations.

Although the best DHMC has good performance, we note that its algorithmic structure requires sequential updates of all discrete variables at each leapfrog step. Compared with, e.g. M-HMC with $T = 40, L = 600, n_{\mathcal{D}} = 1$, using similar implementations, the best DHMC takes 2.23 times longer with nearly 0.2 reduction in MRESS, demonstrating the superior performance of M-HMC.

4.4 Correlated Topic Model (CTM)

Topic modeling is widely used in the statistical analysis of documents collections. CTM [4] is a topic model that extends the popular Latent Dirichlet Allocation (LDA) [5] by using a logistic-normal prior to effectively model correlations among different topics. Our setup follows [4]: assume we have a CTM modeling D documents with K topics and a V-word vocabulary. The K topics are specified by a $K \times V$ matrix β. The kth row β_k is a point on the $V - 1$ simplex, defining a distribution on the vocabulary. Use $w_{d,n} \in \{1, \cdots, V\}$ to denote the nth word in the dth document, $z_{d,n} \in \{1, \cdots, K\}$ to denote the topic assignment associated with the word $w_{d,n}$, and use $\text{Categ}(p)$ to denote a categorical

distribution with distribution p. Define $f : \mathbb{R}^K \to \mathbb{R}^K$ to be $f_i(\eta) = e^{\eta_i} / \sum_{j=1}^K e^{\eta_k}$. Given the topics β, a vector $\mu \in \mathbb{R}^K$ and a $K \times K$ covariance matrix Σ, for the dth document with N_d words, CTM first samples $\eta_d \sim N(\mu, \Sigma)$; then for each $n \in \{1, \cdots, N_d\}$, CTM draws topic assignment $z_{d,n}|\eta_d \sim \text{Categ}(f(\eta_d))$, before finally drawing word $w_{d,n}|z_{d,n}, \beta \sim \text{Categ}(\beta_{z_{d,n}})$.

While CTM has proved to be a better topic model than LDA [4], its use of the non-conjugate logistic-normal prior makes efficient posterior inference of $p(\eta, z|w; \beta, \mu, \Sigma)$ highly challenging. In [4], the authors resorted to a variational inference method with highly idealized mean-field approximations. There has been efforts on developing more efficient inference methods using a sampling-based approach, e.g. specialized Gibbs samplers [20, 9]. In this section, we explore the applications of NwG, DHMC and M-HMC to the posterior inference problem $p(\eta, z|w; \beta, \mu, \Sigma)$ in CTM.

We use the Associated Press (AP) dataset [15][5], which consists of 2246 documents. Since we are interested in comparing the performance of different samplers, we train a CTM using *ctm-c*[6], with the default settings, $K = 10$ topics and the given vocabulary of $V = 10473$ words. To establish a baseline, we use the Gibbs sampler developed in [9], which was empirically demonstrated to be highly effective. Note that unlike [9], there's no Dirichlet prior on β in our setup; moreover, for K topics, *ctm-c* handles the issue of non-identifiability by using $\eta_d \in \mathbb{R}^{K-1}$ and assuming the first dimension to be 0. Nevertheless, it's straightforward to adapt [9] to our setup.

After training the model with *ctm-c*, we pick 20 random documents, and apply the 4 different samplers for posterior sampling of z and η. For each sampler, we generate 96 independent chains, and use 1000 burn-in and 4000 actual samples in each chain. Gibbs and NwG require no manual tuning. For DHMC, we conduct a grid search of step-size range and number-of-steps range. For M-HMC, we inspect traceplots and acceptance probabilities with short trial runs on a document outside the picked 20, and fix $T = 600$ and $n_{\mathcal{D}} = 1$; we set $L = 80 \times N_d$ for document d; empirically, we find it's important to use different step sizes for different dimensions of η_d (i.e. to use a non-identity mass matrix for the kinetic energy $K^{\mathcal{C}}$). For our experiments, we use step size $\frac{4\Sigma_{ii}}{\sum_{j=1}^9 \Sigma_{jj}}$ for $\eta_{d,i}$.

We first compare the accuracy of the 4 different samplers, by inspecting the posterior means of η_d using samples from the 4 different samplers on the 20 randomly picked documents. Likely due to its inability to generalize to complicated discrete state spaces, the sample means for η_d from DHMC differ significantly from the 3 other samplers on all 20 documents. NwG and M-HMC agree on all 20 documents, while Gibbs agrees ($\pm 5\%$ relative error) with them on 17 out of the 20 documents.

We additionally inspect the empirical density and traceplots of posterior samples for η_{d1} on a document where Gibbs disagrees with the other 2 samplers (Figure 4a), and a document where it agrees (Figure 4b). In both cases, M-HMC clearly mixes the fastest, with NwG also outperforming Gibbs. Moreover, in both cases (but especially in Figure 4a), NwG and M-HMC explore the state space much more thoroughly, suggesting that Gibbs gives different posterior means for η_d on the 3 documents because of its inability to effectively explore the state spaces.

On the 17 documents where the 3 samplers do agree, we further calculate the MRESS of η_d. Without much tunning, M-HMC already demonstrates significant advantages over the specialized Gibbs sampler and the highly-optimized NwG: among the 3 samplers, M-HMC has the largest MRESS for all 17 documents; moreover, its MRESS is on average **22.48** times larger than that of Gibbs, and **3.38** times larger than that of NwG. NwG also outperforms Gibbs on all 17 documents, and has on average **8.48** times larger MRESS than Gibbs. Note that the poor performance of Gibbs is not surprising, as it's sequentially updating each component of z and η, which likely causes the slow mixing.

5 Discussions and Future Directions

M-HMC provides a general mechanism that can be easily implemented to make more frequent updates of discrete variables within HMC. Such updates are usually inexpensive (when compared to gradients evaluations) yet highly beneficial as shown in our numerical experiments in Section 4. This makes M-HMC an appealing option for models with mixed support. Some interesting future directions include exploring an extension of M-HMC in a NUTS-like way, automatically setting the involved parameters (as HMC is known to be sensitive to choices of step sizes and number of steps), and the applications of M-HMC for developing stochastic gradient MCMC methods.

[5] The dataset can be downloaded at `http://www.cs.columbia.edu/~blei/lda-c/ap.tgz`

[6] `https://github.com/blei-lab/ctm-c`

Appendix

A Algorithm and theory

A.1 Detailed description of a full M-HMC iteration

See Algorithm 2 for a detailed description of a full M-HMC iteration.

Algorithm 2 Core step of M-HMC

Require: U, potential for the target distribution π; $Q_i, i = 1, \ldots, N_{\mathcal{D}}$, single-site proposals; $k^{\mathcal{D}}$, kinetic energy for discrete component; $I(\cdot, \cdot, \cdot | x, U, K^{\mathcal{C}})$, reversible and volume-preserving integrator for continuous component; τ, interval length in $\mathbb{T}^{N_{\mathcal{D}}}$

input $x^{(0)}$, discrete state; $q^{\mathcal{D}(0)}, p^{\mathcal{D}(0)}$, auxiliary location and momentum for discrete state; $q^{\mathcal{C}(0)}$, continuous location; $p^{\mathcal{C}(0)}$, auxiliary momentum for continuous state; T, travel time

output x, next discrete state; $q^{\mathcal{D}}, p^{\mathcal{D}}$, next auxiliary location and momentum for discrete state; $q^{\mathcal{C}}$, next continuous location; $p^{\mathcal{C}}$, next auxiliary momentum for continuous state

function MixedHMC($x^{(0)}, q^{\mathcal{D}(0)}, p^{\mathcal{D}(0)}, q^{\mathcal{C}(0)}, p^{\mathcal{C}(0)}, T$)
  $x \leftarrow x^{(0)}, q^{\mathcal{D}} \leftarrow q^{\mathcal{D}(0)}, p^{\mathcal{D}} \leftarrow p^{\mathcal{D}(0)}$
  $q^{\mathcal{C}} \leftarrow q^{\mathcal{C}(0)}, p^{\mathcal{C}} \leftarrow p^{\mathcal{C}(0)}$
  $v_i \leftarrow (k^{\mathcal{D}})'(p_i^{\mathcal{D}}), i = 1, \ldots, N_{\mathcal{D}}$
  $t_i \leftarrow \frac{\tau(\text{sign}(v_i)+1) - 2q_i^{\mathcal{D}}}{2v_i}, i = 1, \ldots, N_{\mathcal{D}}$
  while $T > 0$ **do**
    $j \leftarrow \text{argmin}_i\{t_i, i = 1, \ldots, N_{\mathcal{D}}\}$
    $\varepsilon = \min\{t_j, T\}$
    $q_i^{\mathcal{D}} \leftarrow q_i^{\mathcal{D}} + \varepsilon v_i, i = 1, \ldots, N_{\mathcal{D}}$
    $(q^{\mathcal{C}}, p^{\mathcal{C}}) \leftarrow I(q^{\mathcal{C}}, p^{\mathcal{C}}, \varepsilon | x, U, K^{\mathcal{C}})$
    $T \leftarrow T - \varepsilon$
    if $\varepsilon = t_j$ **then**
      $t_i \leftarrow t_i - t_j, i = 1, \ldots, N_{\mathcal{D}}$
      $\tilde{x} \sim Q_j(\cdot | x)$
      $\Delta E \leftarrow \log \frac{e^{-U(x, q^{\mathcal{C}})} Q_j(\tilde{x} | x)}{e^{-U(\tilde{x}, q^{\mathcal{C}})} Q_j(x | \tilde{x})}$
      if $k^{\mathcal{D}}(p_j^{\mathcal{D}}) > \Delta E$ **then**
        $x \leftarrow \tilde{x}, q_j^{\mathcal{D}} \leftarrow \tau - q_j^{\mathcal{D}}$
        $p_j^{\mathcal{D}} \leftarrow \text{sign}(p_j^{\mathcal{D}})(k^{\mathcal{D}})^{-1}(k^{\mathcal{D}}(p_j^{\mathcal{D}}) - \Delta E)$
        $v_j \leftarrow (k^{\mathcal{D}})'(p_j^{\mathcal{D}})$
      else
        $p_j^{\mathcal{D}} \leftarrow -p_j^{\mathcal{D}}, v_j \leftarrow -v_j$
      end if
      $t_j \leftarrow \frac{\tau(\text{sign}(v_j)+1) - 2q_j^{\mathcal{D}}}{2v_j}$
    end if
  end while
  $E = U\left(x, q^{\mathcal{C}}\right) + K^{\mathcal{D}}(p^{\mathcal{D}}) + K^{\mathcal{C}}(p^{\mathcal{C}})$
  $E^{(0)} = U\left(x^{(0)}, q^{\mathcal{C}(0)}\right) + K^{\mathcal{D}}(p^{\mathcal{D}(0)}) + K^{\mathcal{C}}(p^{\mathcal{D}(0)})$
  if Uniform($[0, 1]$) $< e^{-(E - E^{(0)})}$ **then**
    $p^{\mathcal{D}} \leftarrow -p^{\mathcal{D}}, p^{\mathcal{C}} \leftarrow -p^{\mathcal{C}}$
  else
    $x \leftarrow x^{(0)}, q^{\mathcal{D}} \leftarrow q^{\mathcal{D}(0)}, p^{\mathcal{D}} \leftarrow p^{\mathcal{D}(0)}$
    $q^{\mathcal{C}} \leftarrow q^{\mathcal{C}(0)}, p^{\mathcal{C}} \leftarrow p^{\mathcal{C}(0)}$
  end if
  return $x, q^{\mathcal{D}}, p^{\mathcal{D}}, q^{\mathcal{C}}, p^{\mathcal{C}}$
end function

A.2 Proof of Theorem 1

A.2.1 Proof of the Theorem

Theorem 1. ***(Detailed Balance)*** *The MixedHMC function in Algorithm 2 satisfies detailed balance w.r.t. the joint invariant distribution φ, i.e. for any measurable sets $A, B \subset \Omega \times \Sigma$,*

$$\begin{aligned}&\int_{\Sigma} \sum_{x \in A(\Theta)} R((x,\Theta), B)\varphi((x,\Theta))\mathrm{d}\Theta \\ =&\int_{\Sigma} \sum_{x \in B(\Theta)} R((x,\Theta), A)\varphi((x,\Theta))\mathrm{d}\Theta\end{aligned}$$

Proof. For notational simplicity, we use $s = (x, q^{\mathcal{D}}, p^{\mathcal{D}}, q^{\mathcal{C}}, p^{\mathcal{C}})$ and $s' = (x', q^{\mathcal{D}\prime}, p^{\mathcal{D}\prime}, q^{\mathcal{C}\prime}, p^{\mathcal{C}\prime})$ to denote two points in $\Omega \times \Sigma$.

Sequence of Proposals and Probabilistic Paths

If we start from $s \in \Omega \times \Sigma$, for a given travel time T, a concrete run of the *MixedHMC* function would involve a finite sequence of random proposals. Assume the length of the sequence is M. The sequence of random proposals Y can be denoted as

$$Y = (y^{(0)}, y^{(1)}, \ldots, y^{(M-1)}), y^{(m)} \in \Omega, m = 0, \ldots, M-1$$

This sequence of proposals indicates that, for this particular run of *MixedHMC*, we reach 0 or τ at individual sites M times, and each time the system makes a proposal to go to the discrete state $y^{(m)} \in \Omega, m = 0, \cdots, M-1$ from the current discrete state.

If we fix Y, the *MixedHMC* function (without the final accept/reject step) in fact specifies a deterministic mapping, and would map s to a single point $s' \in \Omega \times \Sigma$. For each such sequence of proposals Y, we introduce an associated probabilistic path $\omega(s, T, Y)$, which contains all the information of the system going from s to s' in time T through the function *MixedHMC*. Formally, $\omega(s, T, Y)$ is specified by

- The sequence of random proposals Y
$$Y = (y^{(0)}, y^{(1)}, \ldots, y^{(M-1)}), y^{(m)} \in \Omega, m = 0, \ldots, M-1$$
- The indices of the sites for the M site visitations $j^{(0)}, j^{(1)}, \ldots, j^{(M-1)} \in \{1, \ldots, N_{\mathcal{D}}\}$
- The times of the M site visitations $0 \leqslant t^{(0)} < t^{(1)} < \ldots < t^{(M-1)} \leqslant T$
- The discrete states of the system at M site visitations $x = x^{(0)}, x^{(1)}, \ldots, x^{(M-1)} \in \Omega$
- Accept/reject decisions for the M site visitations $a^{(m)} = \mathbb{1}_{\{y^{(m)} = x^{(m+1)}\}}$, where $x^{(M)} = x'$
- The evolution of the location variables $q^{\mathcal{D}}(t), q^{\mathcal{C}}(t)$ and the momentum variables $p^{\mathcal{D}}(t), p^{\mathcal{C}}(t), 0 \leqslant t \leqslant T$. Note that we might have discontinuities in $p^{\mathcal{D}}(t)$. We use $p^{\mathcal{D}}(t^-)$ to denote the left limit and $p^{\mathcal{D}}(t^+)$ to denote the right limit.

Countable Number of Probabilistic Paths

In order for a probabilistic path $\omega(s, T, Y)$ to be valid, the different components of $\omega(s, T, Y)$ have to interact with each other in a way as determined by the *MixedHMC* function. For example, we should have $y_i^{(m)} = x_i^{(m)}, \forall i \neq j^{(m)}$ and

$$x^{(m+1)} = \begin{cases} y^{(m)} & \text{if } k^{\mathcal{D}}(p^{\mathcal{D}}(t^{(m)-})) > \log \frac{\pi(x^{(m)}, q^{\mathcal{C}}(t^{(m)}))Q_{j^{(m)}}(y^{(m)}|x^{(m)})}{\pi(y^{(m)}, q^{\mathcal{C}}(t^{(m)}))Q_{j^{(m)}}(x^{(m)}|y^{(m)})} \\ x^{(m)} & \text{otherwise} \end{cases}$$

For $s \in \Omega \times \Sigma$ and some given travel time T, we say a sequence of proposals Y is compatible with s, T and *MixedHMC* if we can find a corresponding probabilistic path $\omega(s, T, Y)$ that's valid.

Not all sequences of proposals correspond to valid probabilistic paths. But even if we don't consider the compatibility of the sequence of proposals with s, T and *MixedHMC*, the set of all possible such sequences has only a countable number of elements. This is because we only need to look at sequences of finite length (because of the fixed travel time T), and all the individual proposals are on discrete state spaces with a finite number of states.

The above analysis indicates that for some starting point $s \in \Omega \times \Sigma$ and travel time T, running the *MixedHMC* function would result in only a countable number of possible destinations s'. Furthermore, $\forall s, s' \in \Omega \times \Sigma$ for which $R_T(s, \{s'\}) > 0$, there are at most a countable number of probabilistic paths which bring s to s' in time T through *MixedHMC*.

Formally, given some travel time T and a sequence of proposals Y, define

$$\mathcal{D}(T, Y) = \{s \in \Omega \times \Sigma : Y \text{ is compatible with } s, T \text{ and } \textit{MixedHMC}\}$$

Use $\mathcal{T}_{T,Y} : \mathcal{D}(T, Y) \to \Omega \times \Sigma$ to denote the deterministic mapping defined by *MixedHMC* (without the final accept/reject step) for the given Y in time T (so that $\mathcal{D}(T, Y)$ represents the domain of the mapping $\mathcal{T}_{T,Y}$), and use

$$\mathcal{I}(T, Y) = \{s' \in \Omega \times \Sigma : \exists s \in \mathcal{D}(T, Y), s.t. \mathcal{T}_{T,Y}(s) = s'\}$$

to denote the image of the mapping $\mathcal{T}_{T,Y}$. For a given $x \in \Omega$, use

$$\mathcal{T}_{T,Y,x} : \{(q^{\mathcal{D}}, p^{\mathcal{D}}, q^{\mathcal{C}}, p^{\mathcal{C}}) \in \Sigma : s = (x, q^{\mathcal{D}}, p^{\mathcal{D}}, q^{\mathcal{C}}, p^{\mathcal{C}}) \in \mathcal{D}(T, Y)\} \to \Sigma$$

to denote the deterministic mapping induced by $\mathcal{T}_{T,Y}$ on Σ. In other words,

$$\forall s = (x, q^{\mathcal{D}}, p^{\mathcal{D}}, q^{\mathcal{C}}, p^{\mathcal{C}}) \in \mathcal{D}(T, Y), \mathcal{T}_{T,Y,x}((q^{\mathcal{D}}, p^{\mathcal{D}}, q^{\mathcal{C}}, p^{\mathcal{C}})) = (q^{\mathcal{D}\prime}, p^{\mathcal{D}\prime}, q^{\mathcal{C}\prime}, p^{\mathcal{C}\prime})$$

where $s' = (x', q^{\mathcal{D}\prime}, p^{\mathcal{D}\prime}, q^{\mathcal{C}\prime}, p^{\mathcal{C}\prime}) = \mathcal{T}_{T,Y}(s)$. Define

$$(\Omega \times \Sigma)(s, T) = \{s' = (x', q^{\mathcal{D}\prime}, p^{\mathcal{D}\prime}, q^{\mathcal{C}\prime}, p^{\mathcal{C}\prime}) \in \Omega \times \Sigma : R_T(s, \{s'\}) > 0\}$$

$\forall s, s' \in \Omega \times \Sigma$ for which $R_T(s, \{s'\}) > 0$, further define

$$\mathcal{P}(s, s', T) = \{Y \text{ a sequence of proposals: } s \in \mathcal{D}(T, Y) \text{ and } \mathcal{T}_{T,Y}(s) = s'\}$$

Then both $(\Omega \times \Sigma)(s, T)$ and $\mathcal{P}(s, s', T)$ have at most a countable number of elements.

Proof of Detailed Balance

First, we note that it's trivially true that

$$\varphi(s) R_T(s, \{s\}) = \varphi(s) R_T(s, \{s\}) \tag{2}$$

Next, we consider $s' \neq s$. For a given travel time T and a sequence of proposals Y, $\forall s \in \mathcal{D}(T, Y)$, we use $r_{T,Y}(s, s')$ to denote the probability of going from s to s' through the probabilistic path $\omega(s, T, Y)$. Since *MixedHMC* (without the final accept/reject step) defines a deterministic mapping $\mathcal{T}_{T,Y}$ for given T and Y, considering all $s' \neq s$, the only non-zero term is $r_{T,Y}(s, \mathcal{T}_{T,Y}(s))$. For all $s' \neq s, \mathcal{T}_{T,Y}(s)$, we have $r_{T,Y}(s, s') = 0$.

Using the above notation, $\forall s \in A$ and $B \subset \Omega \times \Sigma$ measurable for which $s \notin B$, we can write $R_T(s, B)$ as

$$\begin{aligned} R_T(s, B) &= \sum_{s' \in B \cap (\Omega \times \Sigma)(s,T)} R_T(s, \{s'\}) \\ &= \sum_{s' \in B \cap (\Omega \times \Sigma)(s,T)} \sum_{Y \in \mathcal{P}(s,s',T)} r_{T,Y}(s, s') \\ &= \sum_{s' \in B \cap (\Omega \times \Sigma)(s,T)} \sum_{Y \in \mathcal{P}(s,s',T)} r_{T,Y}(s, \mathcal{T}_{T,Y}(s)) \end{aligned}$$

For a given travel time T, $\forall s, s' \in \Omega \times \Sigma, s \neq s'$, if $R_T(s, \{s'\}) > 0$, then $\mathcal{P}(s, s', T) \neq \emptyset$. In Lemma 3, we prove that $\forall Y \in \mathcal{P}(s, s', T)$, the absolute value of the determinant of the Jacobian of

$\mathcal{T}_{T,Y,x}$ is $|\det \mathcal{J}\mathcal{T}_{T,Y,x}| = 1$, for all $x \in \Omega$. Furthermore, the deterministic mapping $\mathcal{T}_{T,Y}$ is reversible, and there exists a sequence of proposals $\tilde{Y} \in \mathcal{P}(s', s, T)$, s.t. $s = \mathcal{T}_{T,Y}^{-1}(s') = \mathcal{T}_{T,\tilde{Y}}(s')$.

In Lemma 4, we prove that, $\forall s' = \mathcal{T}_{T,Y}(s) \neq s$,

$$\varphi(s) r_{T,Y}(s, s') = \varphi(s) r_{T,Y}(s, \mathcal{T}_{T,Y}(s)) = \varphi(s') r_{T,\tilde{Y}}(s', \mathcal{T}_{T,\tilde{Y}}(s')) = \varphi(s') r_{T,\tilde{Y}}(s', s)$$

Using the above results, it's not hard to see that, for the case where $A \cap B = \emptyset$,

$$\begin{aligned}
&\int_{\Sigma} \sum_{x \in A(\Theta)} R_T(s, B) \varphi(s) \mathrm{d}\Theta \\
=&\int_{\Sigma} \sum_{x \in A(\Theta)} \sum_{s' \in B \cap (\Omega \times \Sigma)(s,T)} \sum_{Y \in \mathcal{P}(s,s',T)} r_{T,Y}(s, s') \varphi(s) \mathrm{d}\Theta \\
=&\int_{\Sigma} \sum_{x \in A(\Theta)} \sum_{s' \in B \cap (\Omega \times \Sigma)(s,T)} \sum_{Y \in \mathcal{P}(s,s',T)} r_{T,\tilde{Y}}(s', s) \varphi(s') \mathrm{d}\Theta \\
\overset{\text{change of variables}}{=}&\int_{\Sigma} \sum_{x' \in B(\Theta')} \sum_{s \in A \cap (\Omega \times \Sigma)(s',T)} \sum_{\tilde{Y} \in \mathcal{P}(s',s,T)} r_{T,\tilde{Y}}(s', s) \varphi(s') \frac{1}{|det \mathcal{J}\mathcal{T}_{T,Y,x}|} \mathrm{d}\Theta' \\
=&\int_{\Theta} \sum_{x' \in B(\Theta')} R_T(s', A) \varphi(s') \mathrm{d}\Theta'
\end{aligned}$$

Combining the above reasoning with Equation 2, the same result can be established for the case where $A \cap B \neq \emptyset$. This proves the desired detailed balance property of ***MixedHMC*** w.r.t. φ

$$\begin{aligned}
&\int_{\Sigma} \sum_{x \in A(\Theta)} R((x, \Theta), B) \varphi((x, \Theta)) \mathrm{d}\Theta \\
=&\int_{\Sigma} \sum_{x \in B(\Theta)} R((x, \Theta), A) \varphi((x, \Theta)) \mathrm{d}\Theta
\end{aligned}$$

□

A.2.2 Useful Lemmas

In this section, we prove a few useful lemmas to complete the proof of Theorem 1. W.l.o.g. we assume $\tau = 1$ in this section. The proof can be trivially modified to be applicable to arbitrary τ.

First, we prove two lemmas, similar to Lemma 1 and Lemma 2 in Section 5.1 of [21].

Lemma 1. ***(Refraction)*** *Let $\mathcal{T} : \mathbb{T}^{N_\mathcal{D}} \times \mathbb{R}^{N_\mathcal{D}} \to \mathbb{T}^{N_\mathcal{D}} \times \mathbb{R}^{N_\mathcal{D}}$ be a transformation in $\mathbb{T}^{N_\mathcal{D}}$ that takes a unit mass located at $q^\mathcal{D} = (q_1^\mathcal{D}, \ldots, q_{N_\mathcal{D}}^\mathcal{D})$ and moves it with constant velocity $v = ((k^\mathcal{D})'(p_1^\mathcal{D}), \ldots, (k^\mathcal{D})'(p_{N_\mathcal{D}}^\mathcal{D}))$. Assume it reaches 0 or 1 at site j first. Subsequently $q_j^\mathcal{D}$ is changed to $1 - q_j^\mathcal{D}$, and $p_j^\mathcal{D}$ is changed to sign$(p_j^\mathcal{D})(k^\mathcal{D})^{-1}(k^\mathcal{D}(p_j^\mathcal{D}) - \Delta E)$ (where ΔE is a constant and satisfies $\Delta E < k^\mathcal{D}(p_j^\mathcal{D})$). The move is carried on, with the velocity v_j changed to $(k^\mathcal{D})'$(sign$(p_j^\mathcal{D})(k^\mathcal{D})^{-1}(k^\mathcal{D}(p_j^\mathcal{D}) - \Delta E)$), for the total time period μ till it ends in location $q^{\mathcal{D}\prime}$ and momentum $p^{\mathcal{D}\prime}$, before it reaches 0 or 1 again at any sites. Then $\mathcal{T}$ is volume preserving, i.e. the absolute value of the determinant of its Jacobian $|det \mathcal{J}\mathcal{T}| = 1$.*

Proof. Following the same argument as in the proof of Lemma 1 of [21], we have

$$|\det \mathcal{J}\mathcal{T}| = \left| \det \begin{pmatrix} \frac{\partial q^{\mathcal{D}_j\prime}}{\partial q_j^\mathcal{D}} & \frac{\partial q^{\mathcal{D}_j\prime}}{\partial p_j^\mathcal{D}} \\ \frac{\partial p^{\mathcal{D}_j\prime}}{\partial q_j^\mathcal{D}} & \frac{\partial p^{\mathcal{D}_j\prime}}{\partial p_j^\mathcal{D}} \end{pmatrix} \right|$$

If we define $t_j = \frac{\text{sign}(v_j)+1-2q_j^{\mathcal{D}}}{2(k^{\mathcal{D}})'(p_j^{\mathcal{D}})} = \frac{\text{sign}(p_j^{\mathcal{D}})+1-2q_j^{\mathcal{D}}}{2(k^{\mathcal{D}})'(p_j^{\mathcal{D}})}$, then

$$\begin{aligned} p^{\mathcal{D}_j\prime} &= \text{sign}(p_j^{\mathcal{D}})(k^{\mathcal{D}})^{-1}(k^{\mathcal{D}}(p_j^{\mathcal{D}}) - \Delta E) \\ q^{\mathcal{D}_j\prime} &= \frac{1-\text{sign}(p_j^{\mathcal{D}})}{2} + (k^{\mathcal{D}})'(p^{\mathcal{D}_j\prime})(\mu - t_j) \\ &= \frac{1-\text{sign}(p_j^{\mathcal{D}})}{2} + (k^{\mathcal{D}})'(p^{\mathcal{D}_j\prime})\left(\mu - \frac{\text{sign}(p_j^{\mathcal{D}})+1-2q_j^{\mathcal{D}}}{2(k^{\mathcal{D}})'(p_j^{\mathcal{D}})}\right) \end{aligned}$$

This implies

$$\begin{aligned} |\text{det}\mathcal{J}\mathcal{T}| &= \left|\text{det}\begin{pmatrix} \frac{\partial q^{\mathcal{D}_j\prime}}{\partial q_j^{\mathcal{D}}} & \frac{\partial q^{\mathcal{D}_j\prime}}{\partial p_j^{\mathcal{D}}} \\ \frac{\partial p^{\mathcal{D}_j\prime}}{\partial q_j^{\mathcal{D}}} & \frac{\partial p^{\mathcal{D}_j\prime}}{\partial p_j^{\mathcal{D}}} \end{pmatrix}\right| = \left|\text{det}\begin{pmatrix} \frac{\partial q^{\mathcal{D}_j\prime}}{\partial q_j^{\mathcal{D}}} & \frac{\partial q^{\mathcal{D}_j\prime}}{\partial p_j^{\mathcal{D}}} \\ 0 & \frac{\partial p^{\mathcal{D}_j\prime}}{\partial p_j^{\mathcal{D}}} \end{pmatrix}\right| \\ &= \left|\frac{\partial q^{\mathcal{D}_j\prime}}{\partial q_j^{\mathcal{D}}}\frac{\partial p^{\mathcal{D}_j\prime}}{\partial p_j^{\mathcal{D}}}\right| = \left|\frac{(k^{\mathcal{D}})'(p^{\mathcal{D}_j\prime})}{(k^{\mathcal{D}})'(p_j^{\mathcal{D}})}\frac{(k^{\mathcal{D}})'(p_j^{\mathcal{D}})}{(k^{\mathcal{D}})'(p^{\mathcal{D}_j\prime})}\right| = 1 \end{aligned}$$

□

Lemma 2. ***(Reflection)*** *Let $\mathcal{T} : \mathbb{T}^{N_{\mathcal{D}}} \times \mathbb{R}^{N_{\mathcal{D}}} \to \mathbb{T}^{N_{\mathcal{D}}} \times \mathbb{R}^{N_{\mathcal{D}}}$ be a transformation in $\mathbb{T}^{N_{\mathcal{D}}}$ that takes a unit mass located at $q^{\mathcal{D}} = (q_1^{\mathcal{D}}, \ldots, q_N^{\mathcal{D}})$ and moves it with constant velocity $v = ((k^{\mathcal{D}})'(p_1^{\mathcal{D}}), \ldots, (k^{\mathcal{D}})'(p_{N_{\mathcal{D}}}^{\mathcal{D}}))$. Assume it reaches 0 or 1 at site j first. Subsequently $p_j^{\mathcal{D}}$ is changed to $-p_j^{\mathcal{D}}$. The move is carried on, with the velocity v_j changed to $-v_j$, for the total time period μ till it ends in location $q^{\mathcal{D}\prime}$ and momentum $p^{\mathcal{D}\prime}$, before it reaches 0 or 1 at any sites again. Then $\mathcal{T}$ is volume preserving, i.e. the absolute value of the determinant of its Jacobian $|det\mathcal{J}\mathcal{T}| = 1$.*

Proof. Following the same argument as in the proof of Lemma 2 of [21], we have

$$|\text{det}\mathcal{J}\mathcal{T}| = \left|\text{det}\begin{pmatrix} \frac{\partial q^{\mathcal{D}_j\prime}}{\partial q_j^{\mathcal{D}}} & \frac{\partial q^{\mathcal{D}_j\prime}}{\partial p_j^{\mathcal{D}}} \\ \frac{\partial p^{\mathcal{D}_j\prime}}{\partial q_j^{\mathcal{D}}} & \frac{\partial p^{\mathcal{D}_j\prime}}{\partial p_j^{\mathcal{D}}} \end{pmatrix}\right|$$

If we define $t_j = \frac{\text{sign}(v_j)+1-2q_j^{\mathcal{D}}}{2(k^{\mathcal{D}})'(p_j^{\mathcal{D}})} = \frac{\text{sign}(p_j^{\mathcal{D}})+1-2q_j^{\mathcal{D}}}{2(k^{\mathcal{D}})'(p_j^{\mathcal{D}})}$, then

$$\begin{aligned} p^{\mathcal{D}_j\prime} &= -p_j^{\mathcal{D}} \\ q^{\mathcal{D}_j\prime} &= \frac{1+\text{sign}(p_j^{\mathcal{D}})}{2} - (k^{\mathcal{D}})'(p_j^{\mathcal{D}})(\mu - t_j) \\ &= \frac{1+\text{sign}(p_j^{\mathcal{D}})}{2} - (k^{\mathcal{D}})'(p_j^{\mathcal{D}})\left(\mu - \frac{\text{sign}(p_j^{\mathcal{D}})+1-2q_j^{\mathcal{D}}}{2(k^{\mathcal{D}})'(p_j^{\mathcal{D}})}\right) \\ &= 1 + \text{sign}(p_j^{\mathcal{D}}) - (k^{\mathcal{D}})'(p_j^{\mathcal{D}})\mu - q_j^{\mathcal{D}} \end{aligned}$$

This implies

$$|\text{det}\mathcal{J}\mathcal{T}| = \left|\text{det}\begin{pmatrix} \frac{\partial q^{\mathcal{D}_j\prime}}{\partial q_j^{\mathcal{D}}} & \frac{\partial q^{\mathcal{D}_j\prime}}{\partial p_j^{\mathcal{D}}} \\ \frac{\partial p^{\mathcal{D}_j\prime}}{\partial q_j^{\mathcal{D}}} & \frac{\partial p^{\mathcal{D}_j\prime}}{\partial p_j^{\mathcal{D}}} \end{pmatrix}\right| = \left|\text{det}\begin{pmatrix} -1 & \frac{\partial q^{\mathcal{D}_j\prime}}{\partial p_j^{\mathcal{D}}} \\ 0 & -1 \end{pmatrix}\right| = 1$$

□

Lemma 3. *Given travel time T, $\forall s, s' \in \Omega \times \Sigma, s \neq s'$ for which $R_T(s, \{s'\}) > 0$, $\mathcal{P}(s, s', T) \neq \emptyset$. $\forall Y \in \mathcal{P}(s, s', T)$, the absolute value of the determinant of the Jacobian of $\mathcal{T}_{T,Y,x}$ is $|det\mathcal{J}\mathcal{T}_{T,Y,x}| = 1$, for all $x \in \Omega$ where $\mathcal{T}_{T,Y,x}$ is well-defined. Furthermore, the deterministic mapping $\mathcal{T}_{T,Y}$ is reversible, and there exists a sequence of proposals $\tilde{Y} \in \mathcal{P}(s', s, T)$, s.t. $s = \mathcal{T}_{T,Y}^{-1}(s') = \mathcal{T}_{T,\tilde{Y}}(s')$*

Proof. Given travel time T, $\forall s, s' \in \Omega \times \Sigma$, if $R_T(s, \{s'\}) > 0$, then by definition $\mathcal{P}(s, s', T) \neq \emptyset$. $\forall Y \in \mathcal{P}(s, s', Y)$, for some $x \in \Omega$, if the deterministic mapping $\mathcal{T}_{T,Y,x}$ is well-defined, then $\mathcal{T}_{T,Y,x}$ can be be written as the composition of a sequence of deterministic mappings

$$\mathcal{T}_{T,Y,x} = \mathcal{T}^{(0)}_{T,Y,x} \circ \mathcal{T}^{(1)}_{T,Y,x} \circ \cdots \circ \mathcal{T}^{(M-1)}_{T,Y,x}$$

Each one of the mappings $\mathcal{T}^{(m)}_{T,Y,x}, m = 0, \ldots, M-1$ consists of two parts that don't interact: a discrete part that operates on $q^{\mathcal{D}}, p^{\mathcal{D}}$, and a continuous part that operates on $q^{\mathcal{C}}, p^{\mathcal{C}}$. The discrete part is either a refraction mapping as described in Lemma 1, or a reflection mapping as described in Lemma 2. The continuous part is given by the integrator I, which is reversible and volume-preserving. Using Lemma 1 and Lemma 2 and the properties of the integrator I, it's easy to see that the absolute value of the determinant of the Jacobian

$$|\text{det}\mathcal{J}\mathcal{T}_{T,Y,x}| = \prod_{m=0}^{M-1} |\text{det}\mathcal{J}\mathcal{T}^{(m)}_{T,Y,x}| = 1$$

$\forall Y \in \mathcal{P}(s, s', Y)$, define a new sequence of proposals $\tilde{Y} = (\tilde{y}^{(0)}, \tilde{y}^{(1)}, \ldots, \tilde{y}^{(M-1)})$ where

$$\tilde{y}^{(m)} = \begin{cases} x^{(M-m-1)} & \text{if } a^{(M-m-1)} = 1 (\text{i.e. } y^{(M-m-1)} = x^{(M-m)}) \\ y^{(M-m-1)} & \text{otherwise (i.e. } y^{(M-m-1)} \neq x^{(M-m)}, \text{which means } x^{(M-m-1)} = x^{(M-m)}) \end{cases}$$

We claim that $\tilde{Y} \in \mathcal{P}(s, s', T)$, and $\mathcal{T}_{T,\tilde{Y}}(s') = s$. To see $\tilde{Y}$ has these desired properties, we look at its corresponding probabilistic path $\omega(s', T, \tilde{Y})$. The corresponding discrete states of the system at M site visitations $\tilde{x}^{(m)}, m = 0, \ldots, M$ and the indices of the sites for the M site visitations $\tilde{j}^{(m)}, m = 0, \ldots, M-1$ are given by simple reversals of the original sequence of discrete states $x^{(m)}, m = 0, \ldots, M$ and the original sequence of indices for visited sites $j^{(m)}, m = 0, \ldots, M-1$:

$$\begin{aligned} \tilde{j}^{(m)} &= j^{(M-m-1)}, m = 0, \ldots, M-1 \\ \tilde{x}^{(m)} &= x^{(M-m)}, m = 0, \ldots, M \end{aligned}$$

The corresponding sequence of accept/reject decisions $\tilde{a}^{(m)}, m = 0, \ldots, M-1$ is also a simple reversal of the original sequence of accept/reject decisions $a^{(m)}, m = 0, \ldots, M-1$

$$\tilde{a}^{(m)} = \mathbb{1}_{\{\tilde{y}^{(m)} = \tilde{x}^{(m+1)}\}} = \begin{cases} \mathbb{1}_{\{x^{(M-m-1)} = x^{(M-m-1)}\}} = 1 & \text{if } a^{(M-m-1)} = 1 \\ \mathbb{1}_{\{y^{(M-m-1)} = x^{(M-m-1)}\}} = 0 & \text{if } a^{(M-m-1)} = 0 \end{cases} = a^{(M-m-1)}$$

It's straightforward to verify that $\omega(s', T, \tilde{Y})$ is a valid probabilistic path that brings s' back to s in time T through ***MixedHMC***. In particular, note the importance of the momentum negating step in ensuring the existence of such a probabilistic path. This proves our claim.

□

Lemma 4. *$\forall s, s' \in \Omega \times \Sigma, s \neq s'$ for which $R_T(s, \{s'\}) > 0$, for $Y \in \mathcal{P}(s, s', T)$, we have*

$$\varphi(s) r_{T,Y}(s, s') = \varphi(s) r_{T,Y}(s, \mathcal{T}_{T,Y}(s)) = \varphi(s') r_{T,\tilde{Y}}(s', \mathcal{T}_{T,\tilde{Y}}(s')) = \varphi(s') r_{T,\tilde{Y}}(s', s)$$

where $\tilde{Y}$ is defined as in Lemma 3.

Proof. We can directly calculate the transition probability $r_{T,Y}(s, s')$. Define

$$E = U(x, q^{\mathcal{C}}) + K^{\mathcal{D}}(p^{\mathcal{D}}) + K^{\mathcal{C}}(p^{\mathcal{C}}), E' = U(x', q^{\mathcal{C}\prime}) + K^{\mathcal{D}}(p^{\mathcal{D}\prime}) + K^{\mathcal{C}}(p^{\mathcal{C}\prime})$$

Then

$$r_{T,Y}(s, s') = \prod_{m=0}^{M-1} Q_{j^{(m)}}(y^{(m)} | x^{(m)}) \min\{1, e^{-(E'-E)}\}$$

Correspondingly, we can also calculate the transition probability $r_{T,\tilde{Y}}(s', s)$.

$$r_{T,\tilde{Y}}(s', s) = \prod_{m=0}^{M-1} Q_{\tilde{j}^{(m)}}(\tilde{y}^{(m)} | \tilde{x}^{(m)}) \min\{1, e^{-(E-E')}\}$$

Note that

$$
\begin{aligned}
\frac{r_{T,Y}(s,s')}{\min\{1,e^{-(E'-E)}\}} &= \prod_{m=0}^{M-1} Q_{j^{(m)}}^{a^{(m)}}(y^{(m)}|x^{(m)}) \prod_{m=0}^{M-1} Q_{j^{(m)}}^{1-a^{(m)}}(y^{(m)}|x^{(m)}) \\
&= \prod_{m:a^{(m)}=1} Q_{j^{(m)}}(y^{(m)}|x^{(m)}) \prod_{m:a^{(m)}=0} Q_{j^{(m)}}(y^{(m)}|x^{(m)}) \\
\frac{r_{T,\tilde{Y}}(s',s)}{\min\{1,e^{-(E-E')}\}} &= \prod_{m=0}^{M-1} Q_{\tilde{j}^{(m)}}^{\tilde{a}^{(m)}}(\tilde{y}^{(m)}|\tilde{x}^{(m)}) \prod_{m=0}^{M-1} Q_{\tilde{j}^{(m)}}^{1-\tilde{a}^{(m)}}(\tilde{y}^{(m)}|\tilde{x}^{(m)}) \\
&= \prod_{m:\tilde{a}^{(m)}=1} Q_{\tilde{j}^{(m)}}(\tilde{y}^{(m)}|\tilde{x}^{(m)}) \prod_{m:\tilde{a}^{(m)}=0} Q_{\tilde{j}^{(m)}}(\tilde{y}^{(m)}|\tilde{x}^{(m)}) \\
&= \prod_{m:a^{(M-m-1)}=1} Q_{j^{(M-m-1)}}(x^{(M-m-1)}|y^{(M-m-1)}) \\
&\times \prod_{m:a^{(M-m-1)}=0} Q_{j^{(M-m-1)}}(y^{(M-m-1)}|x^{(M-m)}) \\
&= \prod_{m:a^{(M-m-1)}=1} Q_{j^{(M-m-1)}}(x^{(M-m-1)}|y^{(M-m-1)}) \\
&\times \prod_{m:a^{(M-m-1)}=0} Q_{j^{(M-m-1)}}(y^{(M-m-1)}|x^{(M-m-1)}) \\
&= \prod_{m:a^{(m)}=1} Q_{j^{(m)}}(x^{(m)}|y^{(m)}) \prod_{m:a^{(m)}=0} Q_{j^{(m)}}(y^{(m)}|x^{(m)})
\end{aligned}
$$

By following the probabilistic path $\omega(s,T,Y)$ and doing explicit calculations, we can show that

$$\varphi(s) r_{T,Y}(s,s') = \varphi(s') r_{T,\tilde{Y}}(s',s)$$

□

B Details on implementation with Laplace momentum

In what follows, line numbers refer to lines in Algorithm 2. Under Laplace momentum, $v_i = \text{sign}(p_i^{\mathcal{D}}) \in \{1,-1\}$. As a result, different $q_i^{\mathcal{D}}$ always evolve with a constant speed 1, and we no longer need the argmin in Line 7. Site visitation order is completely determined by the initial sampling of $q^{\mathcal{D}}, p^{\mathcal{D}}$. Furthermore, we can precompute all the involved step sizes (in Line 8). These step sizes are in fact differences of neighboring order statistics of $N^{\mathcal{D}}$ uniform samples on $[0,\tau]$, and as a result have the Dirichlet distribution as the joint distribution. The initial momentum is given by $p_i^{\mathcal{D}(0)} \sim \nu(p) \propto e^{-|p|}$, which corresponds to the initial kinetic energy $k^{\mathcal{D}}(p_i^{\mathcal{D}(0)}) \sim \text{Exponential}(1)$.

The above observations indicate that, using Laplace momentum, we no longer need to keep track of $q^{\mathcal{D}}, p^{\mathcal{D}}$. Instead, at the beginning of each iteration, we can sample the site visitation order as a random permutation, the step sizes from a Dirichlet distribution, and the kinetic energies from independent exponential distributions. In each iteration, we simply evolve the system according to the step sizes, visit each site in order, and keep track of changes in kinetic energies. These simplications results in the efficient implementation described in Algorithm 1 in the main text.

C Python function comparing M-HMC with naive MH updates within HMC

Code for reproducing the results in the paper is available at `https://github.com/StannisZhou/mixed_hmc`. In particular, we include below a illustrative python function for comparing M-HMC with naive Metropolis updates within HMC. Experimental results using this function can be reproduced using the script *test_naive_mixed_hmc.py* under *scripts/simple_gmm*.

```
import numba
import numpy as np
from tqdm import tqdm
```

```
def naive_mixed_hmc(
    z0, q0, n_samples, epsilon, L, pi, mu_list, sigma_list, use_k=True
):
    """Function for comparing M-HMC and naive MH updates within HMC

    Parameters
    ----------
    z0 : int
        Discrete variable for the mixture component
    q0 : float
        Continuous variable for the state of GMM
    n_samples : int
        Number of samples to draw
    epsilon : float
        Step size
    L : int
        Number of steps
    pi : np.array
        Array of shape (n_components,).
        The probabilities for different components
    mu_list : np.array
        Array of shape (n_components,).
        Means of different components
    sigma_list : np.array
        Array of shape (n_components,).
        Standard deviations of different components
    use_k : bool
        True if we use M-HMC.
        False if we make naive MH updates within HMC

    Returns
    -------
    z_samples : np.array
        Array of shape (n_samples,). Samples for z
    x_samples : np.array
        Array of shape (n_samples,). Samples for x
    accept_list : np.array
        Array of shape (n_samples,).
        Records whether we accept or reject at each step
    """

    @numba.jit(nopython=True)
    def potential(z, q):
        potential = (
            -np.log(pi[z])
            + 0.5 * np.log(2 * np.pi * sigma_list[z] ** 2)
            + 0.5 * (q - mu_list[z]) ** 2 / sigma_list[z] ** 2
        )
        return potential

    @numba.jit(nopython=True)
    def grad_potential(z, q):
        grad_potential = (q - mu_list[z]) / sigma_list[z] ** 2
        return grad_potential

    @numba.jit(nopython=True)
    def take_naive_mixed_hmc_step(z0, q0, epsilon, L, n_components):
```

```
    # Resample momentum
    p0 = np.random.randn()
    k0 = np.random.exponential()
    # Initialize q, k
    z = z0
    q = q0
    p = p0
    k = k0
    # Take L steps
    for ii in range(L):
        q, p = leapfrog_step(z=z, q=q, p=p, epsilon=epsilon)
        z, k = update_discrete(
            z0=z, k0=k, q=q, n_components=n_components
        )

    # Accept or reject
    current_U = potential(z0, q0)
    current_K = k0 + 0.5 * p0 ** 2
    proposed_U = potential(z, q)
    proposed_K = k + 0.5 * p ** 2
    accept = np.random.rand() < np.exp(
        current_U - proposed_U + current_K - proposed_K
    )
    if not accept:
        z, q = z0, q0

    return z, q, accept

@numba.jit(nopython=True)
def leapfrog_step(z, q, p, epsilon):
    p -= 0.5 * epsilon * grad_potential(z, q)
    q += epsilon * p
    p -= 0.5 * epsilon * grad_potential(z, q)
    return q, p

@numba.jit(nopython=True)
def update_discrete(z0, k0, q, n_components):
    z = z0
    k = k0
    distribution = np.ones(n_components)
    distribution[z] = 0
    distribution /= np.sum(distribution)
    proposal_for_ind = np.argmax(
        np.random.multinomial(1, distribution)
    )
    z = proposal_for_ind
    delta_E = potential(z, q) - potential(z0, q)
    # Decide whether to accept or reject
    if use_k:
        accept = k > delta_E
        if accept:
            k -= delta_E
        else:
            z = z0
    else:
        accept = np.random.exponential() > delta_E
        if not accept:
            z = z0
```

```
    return z, k

z, q = z0, q0
z_samples, x_samples, accept_list = [], [], []
for _ in tqdm(range(n_samples)):
    z, q, accept = take_naive_mixed_hmc_step(
        z0=z, q0=q, epsilon=epsilon, L=L, n_components=pi.shape[0]
    )
    z_samples.append(z)
    x_samples.append(q)
    accept_list.append(accept)

z_samples = np.array(z_samples)
x_samples = np.array(x_samples)
accept_list = np.array(accept_list)
return z_samples, x_samples, accept_list
```

D Binary HMC Samplers are special cases of M-HMC

Formally, we have the following equivalence between binary HMC and M-HMC:

Proposition 1. *Binary HMC is equivalent to a variant of M-HMC (where $q^{\mathcal{D}}$ is initialized at the start and not resampled at each iteration) with $\tau = 1$ and deterministic proposals $Q_i, i = 1, \ldots, N_{\mathcal{D}}$*

$$Q_i(\tilde{x}|x) = \begin{cases} 1, \textit{ if } \tilde{x}_i = -x_i, \tilde{x}_j = x_j, \forall j \neq i \\ 0, \textit{ otherwise} \end{cases}$$

Gaussian and exponential binary HMC correspond to $k^{\mathcal{D}}(p) = |p|$ and $k^{\mathcal{D}}(p) = |p|^{\frac{2}{3}}$ respectively.

Since no continuous component is involved in a binary distribution, for notational simplicity, we drop all the superscript $\mathcal{D}$ in the following discussions. We consider the family of kinetic energies $K_\beta(p) = |p|^\beta$, and define the corresponding distribution to be $\nu_\beta(p) \propto e^{-K_\beta(p)}$. We want to show that the binary HMC samplers are special cases of a variant of M-HMC. In what follows, we use M-HMC to refer to the variant of M-HMC where q is initialized at the start and not resampled at each iteration.

In order to establish the equivalence between binary HMC and M-HMC, we need to study:

1. For site j, the distribution on the initial time it takes to visit site j, which we denote by $t_j^{(0)}$.
   - As shown in Algorithm 1, in M-HMC
   $$t_j^{(0)} = \frac{\text{sign}(v_j^{(0)}) + 1 - 2q_j^{(0)}}{2v_j^{(0)}}$$
   where $v_j^{(0)} = K_\beta'(p_j^{(0)}) = \text{sign}(p_j^{(0)})\beta|p_j^{(0)}|^{\beta-1}$ is the velocity at site j, and
   $$q_j^{(0)} \sim U([0,1]), p_j^{(0)} \sim \nu_\beta(p_j^{(0)})$$
   - For the Gaussian binary HMC sampler,
   $$t_j^{(0)} = \begin{cases} -\arctan\left(\frac{q_j^{(0)}}{p_j^{(0)}}\right) & \text{if } \frac{q_j^{(0)}}{p_j^{(0)}} \leqslant 0 \\ \pi - \arctan\left(\frac{q_j^{(0)}}{p_j^{(0)}}\right) & \text{if } \frac{q_j^{(0)}}{p_j^{(0)}} > 0 \end{cases}$$
   where $q_j^{(0)}, p_j^{(0)} \sim N(0,1)$.
   - For the exponential binary HMC sampler,
   $$t_j^{(0)} = p_j^{(0)} + \sqrt{(p_j^{(0)})^2 + 2q_j^{(0)}}$$
   where $q_j^{(0)} \sim \exp(1), p_j^{(0)} \sim N(0,1)$.

2. For site j, the distribution on the initial total energy, which we denote by $k_j^{(0)}$.
   - For M-HMC, $k_j^{(0)} = K_\beta(p_j^{(0)})$, where $p_j^{(0)} \sim \nu_\beta(p_j^{(0)})$.
   - For the Gaussian binary HMC sampler,
   $$k_j^{(0)} = \frac{1}{2}(q_j^{(0)})^2 + \frac{1}{2}(p_j^{(0)})^2$$
   where $q_j^{(0)}, p_j^{(0)} \sim N(0,1)$.
   - For the exponential binary HMC sampler,
   $$k_j^{(0)} = q_j^{(0)} + \frac{1}{2}(p_j^{(0)})^2$$
   where $q_j^{(0)} \sim \exp(1), p_j^{(0)} \sim N(0,1)$.
3. For site j, after we reach 0 or 1, if we have total energy k, the time it takes to hit a boundary again at this site. We denote this time by $t_j(k)$.
   - For M-HMC, $t_j(k) = \frac{1}{\beta k^{1-\frac{1}{\beta}}}$
   - For the Gaussian binary HMC, $t_j(k) = \pi$
   - For the exponential binary HMC, $t_j(k) = 2\sqrt{2k}$

Since different dimensions are independent of each other, we only need to look at one particular dimension j. We can prove the corresponding propositions if we can establish suitable equivalence concerning the joint distribution on $(t_j^{(0)}, k_j^{(0)})$, and the function $t_j(k)$.

D.1 Proof of Proposition 1 for Gaussian binary HMC

In order to prove Proposition 1 for Gaussian binary HMC, we first prove a lemma

Lemma 5. *Assume $q, p \sim N(0,1)$ are two independent standard normal random variables. Then $\frac{q}{p}$ and q^2+p^2 are independent. Furthermore, $\arctan\left(\frac{q}{p}\right)$ follows the uniform distribution $U\left(\left[-\frac{\pi}{2}, \frac{\pi}{2}\right]\right)$, and $\frac{q^2+p^2}{2}$ follows the exponential distribution $\exp(1)$.*

Proof. We calculate the characteristic function of the random vector $\left(\frac{q}{p}, q^2+p^2\right)$:

$$\begin{aligned}
& \mathbb{E}_{q,p\sim N(0,1)}\left[e^{i\left[t_1\frac{q}{p}+t_2(q^2+p^2)\right]}\right] \\
= & \frac{1}{2\pi}\int_{\mathbb{R}^2} e^{it_1\frac{q}{p}+it_2(q^2+p^2)} e^{-\frac{q^2+p^2}{2}}\mathrm{d}q\mathrm{d}p \\
= & \frac{1}{2\pi}\int_0^{+\infty}\int_0^{2\pi} e^{it_1\tan\theta} e^{it_2 r^2} e^{-\frac{r^2}{2}} r\mathrm{d}r\mathrm{d}\theta \\
= & \left[\int_0^{2\pi} e^{it_1\tan\theta}\frac{1}{2\pi}\mathrm{d}\theta\right]\left[\int_0^{+\infty} e^{it_2r^2-\frac{r^2}{2}} r\mathrm{d}r\right] \\
= & \left[\int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} e^{it_1\tan\theta}\frac{1}{\pi}\mathrm{d}\theta\right]\left[\int_0^{+\infty} e^{it_2x}\frac{1}{2}e^{-2x}\mathrm{d}x\right] \\
= & \left[\int_{-\infty}^{+\infty} e^{it_1x}\frac{1}{\pi(1+x^2)}\mathrm{d}x\right]\left[\int_0^{+\infty} e^{it_2x}\frac{1}{2}e^{-2x}\mathrm{d}x\right] \\
= & \mathbb{E}_{x\sim\text{Cauchy}(0,1)}[e^{it_1x}]\mathbb{E}_{x\sim\exp(2)}[e^{it_2x}]
\end{aligned}$$

This calculation implies that $\frac{q}{p}$ and q^2+p^2 are independent, and that $\frac{q}{p} \sim \text{Cauchy}(0,1)$, $q^2+p^2 \sim \exp(2)$. Since the cumulative distribution function (CDF) of $\text{Cauchy}(0,1)$ is given by

$$\frac{1}{\pi}\arctan(x) + \frac{1}{2}$$

we have $\frac{1}{\pi}\arctan\left(\frac{q}{p}\right)+\frac{1}{2}\sim U([0,1])$, which implies that $\arctan\left(\frac{q}{p}\right)\sim U\left(\left[-\frac{\pi}{2},\frac{\pi}{2}\right]\right)$. From $q^2+p^2\sim\exp(2)$, it's easy to deduce that $\frac{q^2+p^2}{2}\sim\exp(1)$. □

Proof. **(Proposition 1 for Gaussian binary HMC)** For the Gaussian binary HMC sampler, using Lemma 5 and the expressions we derived in Section D, given a dimension j, it's easy to see that $t_j^{(0)}$ and $k_j^{(0)}$ are independent, and that $t_j^{(0)}\sim U([0,\pi])$, $k_j^{(0)}\sim\exp(1)$. For M-HMC with $\beta=1$, it's easy to see that we also have $t_j^{(0)}$ and $k_j^{(0)}$ are independent, and that $t_j^{(0)}\sim U([0,1]), k_j^{(0)}\sim\exp(1)$. This implies that the random vector $\left(\frac{t_j^{(0)}}{\pi},k_j^{(0)}\right)$ from the Gaussian binary HMC sampler has the same joint distribution as the random vector $(t_j^{(0)},k_j^{(0)})$ from M-HMC with $\beta=1$.

For the Gaussian binary HMC sampler, $t_j(k)=\pi$, which is a constant function and is independent of the value of k. For M-HMC with $\beta=1$, it's easy to see that $t_j(k)=1$, which is also a constant function. This implies that $\forall k$, $\frac{t_j(k)}{\pi}$ for the Gaussian binary HMC sampler is equivalent to $t_j(k)$ for M-HMC with $\beta=1$.

The above equivalences imply that the Gaussian binary HMC has exactly the same behavior as M-HMC with $\beta=1$. In fact, the Gaussian binary HMC sampler behaves like scaling the time of M-HMC with $\beta=1$ by π. □

D.2 Proof of Proposition 1 for exponential binary HMC

Proof. **(Proposition 1 for exponential binary HMC)** Using the expressions we derived in Section D, we can see that, at a given site j,

- For the exponential binary HMC sampler, the joint distribution of the random vector $(t_j^{(0)},k_j^{(0)})$ is the same as the random vector $\left(p+\sqrt{p^2+2q},q+\frac{1}{2}p^2\right)$, where $q\sim\exp(1), p\sim N(0,1)$ are independent. For a given total energy level k, $t_j(k)=2\sqrt{2k}$.
- For M-HMC with $\beta=\frac{2}{3}$, the joint distribution of the random vector $(t_j^{(0)},k_j^{(0)})$ is the same as the random vector $\left(\frac{3}{2}q|p|^{\frac{1}{3}},|p|^{\frac{2}{3}}\right)$, where $q\sim U([0,1]), p\sim G\left(0,1,\frac{2}{3}\right)$ are independent. For a given total energy level k, $t_j(k)=\frac{3}{2}\sqrt{k}$.

In order to establish the equivalence between these two samplers, we calculate the characteristic functions of two random vectors. We first calculate the characteristic function of the random vector $\left(p+\sqrt{p^2+2q},q+\frac{1}{2}p^2\right)$, where $q\sim\exp(1), p\sim N(0,1)$ are independent:

$$
\begin{aligned}
& \mathbb{E}_{q\sim\exp(1),p\sim N(0,1)}\left[e^{i\left[t_1\left(p+\sqrt{p^2+2q}\right)+t_2\left(q+\frac{1}{2}p^2\right)\right]}\right] \\
= \quad & \frac{1}{\sqrt{2\pi}}\int_0^{+\infty}\int_{\mathbb{R}} e^{it_1\left(p+\sqrt{p^2+2q}\right)+it_2\left(q+\frac{p^2}{2}\right)}e^{-q}e^{-\frac{p^2}{2}}\mathrm{d}p\mathrm{d}q \\
= \quad & \frac{1}{2\sqrt{2\pi}}\int_{\mathbb{R}^2} e^{it_1\left(p+\sqrt{p^2+2|q|}\right)+it_2\left(|q|+\frac{p^2}{2}\right)}e^{-|q|}e^{-\frac{p^2}{2}}\mathrm{d}p\mathrm{d}q \\
\overset{p=r\cos\theta,q=\mathrm{sign}(\sin\theta)\frac{r^2\sin^2\theta}{2}}{=} \quad & \frac{1}{2\sqrt{2\pi}}\int_0^{+\infty}\int_0^{2\pi} e^{it_1r(1+\cos\theta)+it_2\frac{r^2}{2}}e^{-\frac{r^2}{2}}r^2\sin\theta\mathrm{d}\theta\mathrm{d}r
\end{aligned}
$$

Next we calculate the characteristic function of the random vector $\left(2\sqrt{2}q|p|^{\frac{1}{3}}, |p|^{\frac{2}{3}}\right)$, where $q \sim U([0,1]), p \sim G\left(0,1,\frac{2}{3}\right)$ are independent:

$$
\begin{aligned}
& \mathbb{E}_{q\sim U([0,1]),p\sim G\left(0,1,\frac{2}{3}\right)}\left[e^{i\left(t_1 2\sqrt{2}q|p|^{\frac{1}{3}}+t_2|p|^{\frac{2}{3}}\right)}\right] \\
= & \frac{\frac{2}{3}}{2\Gamma\left(\frac{3}{2}\right)}\int_0^1\int_{\mathbb{R}} e^{it_1 2\sqrt{2}q|p|^{\frac{1}{3}}+it_2|p|^{\frac{2}{3}}}e^{-|p|^{\frac{2}{3}}}\mathrm{d}p\mathrm{d}q \\
= & \frac{2}{3\sqrt{\pi}}\int_0^1\int_{\mathbb{R}} e^{it_1 2\sqrt{2}q|p|^{\frac{1}{3}}+it_2|p|^{\frac{2}{3}}}e^{-|p|^{\frac{2}{3}}}\mathrm{d}p\mathrm{d}q \\
= & \frac{4}{3\sqrt{\pi}}\int_0^1\int_0^{+\infty} e^{it_1 2\sqrt{2}qp^{\frac{1}{3}}+it_2p^{\frac{2}{3}}}e^{-p^{\frac{2}{3}}}\mathrm{d}p\mathrm{d}q \\
\overset{q=\frac{1+\cos\theta}{2},p=\frac{r^3}{2^{\frac{3}{2}}}}{=} & \frac{4}{3\sqrt{\pi}}\int_0^{\pi}\int_0^{+\infty} e^{it_1 r(1+\cos\theta)+it_2\frac{r^2}{2}}e^{-\frac{r^2}{2}}\frac{3}{2^{\frac{5}{2}}}r^2\sin\theta\mathrm{d}r\mathrm{d}\theta \\
= & \frac{1}{\sqrt{2\pi}}\int_0^{+\infty}\left[\int_0^{\pi} e^{it_1 r(1+\cos\theta)}\sin\theta\mathrm{d}\theta\right]e^{it_2\frac{r^2}{2}-\frac{r^2}{2}}r^2\mathrm{d}r \\
= & \frac{1}{2\sqrt{2\pi}}\int_0^{+\infty}\left[\int_0^{2\pi} e^{it_1 r(1+\cos\theta)}\sin\theta\mathrm{d}\theta\right]e^{it_2\frac{r^2}{2}-\frac{r^2}{2}}r^2\mathrm{d}r \\
= & \frac{1}{2\sqrt{2\pi}}\int_0^{+\infty}\int_0^{2\pi} e^{it_1 r(1+\cos\theta)+it_2\frac{r^2}{2}}e^{-\frac{r^2}{2}}r^2\sin\theta\mathrm{d}\theta\mathrm{d}r
\end{aligned}
$$

The above calculations indicate that the joint distribution of $(t_j^{(0)}, k_j^{(0)})$ for the exponential binary HMC sampler is equivalent to the joint distribution of $\left(\frac{4\sqrt{2}}{3}t_j^{(0)}, k_j^{(0)}\right)$ for M-HMC with $\beta = \frac{2}{3}$. Furthermore, if we multiply the $t_j(k)$ function of M-HMC with $\beta = \frac{2}{3}$ by $\frac{4\sqrt{2}}{3}$, we get the function $2\sqrt{2k}$, which is exactly the $t_j(k)$ function for the exponential binary HMC sampler.

The above equivalences imply that the exponential binary HMC has exactly the same behavior as M-HMC with $\beta = \frac{2}{3}$. In fact, the exponential binary HMC sampler behaves like scaling the time of M-HMC with $\beta = \frac{2}{3}$ by $\frac{3}{4\sqrt{2}}$. □

References

[1] James H Albert and Siddhartha Chib. Bayesian analysis of binary and polychotomous response data. *J. Am. Stat. Assoc.*, 88(422):669–679, June 1993.

[2] Michael Betancourt. A conceptual introduction to hamiltonian monte carlo. *arXiv:1701.02434*, July 2018.

[3] Eli Bingham, Jonathan P. Chen, Martin Jankowiak, Fritz Obermeyer, Neeraj Pradhan, Theofanis Karaletsos, Rohit Singh, Paul Szerlip, Paul Horsfall, and Noah D. Goodman. Pyro: Deep Universal Probabilistic Programming. *arXiv preprint arXiv:1810.09538*, 2018.

[4] David M Blei and John D Lafferty. A correlated topic model of science. August 2007.

[5] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3(Jan):993–1022, 2003.

[6] James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Chris Leary, Dougal Maclaurin, and Skye Wanderman-Milne. JAX: composable transformations of Python+NumPy programs, 2018.

[7] Bradley P Carlin and Siddhartha Chib. Bayesian model choice via markov chain monte carlo methods. *J. R. Stat. Soc. Series B Stat. Methodol.*, 57(3):473–484, 1995.

[8] Bob Carpenter, Andrew Gelman, Matthew Hoffman, Daniel Lee, Ben Goodrich, Michael Betancourt, Marcus Brubaker, Jiqiang Guo, Peter Li, and Allen Riddell. Stan: A probabilistic programming language. *Journal of Statistical Software, Articles*, 76(1):1–32, 2017.

[9] Jianfei Chen, Jun Zhu, Zi Wang, Xun Zheng, and Bo Zhang. Scalable inference for Logistic-Normal topic models. In C J C Burges, L Bottou, M Welling, Z Ghahramani, and K Q Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 2445–2453. Curran Associates, Inc., 2013.

[10] Marco F. Cusumano-Towner, Feras A. Saad, Alexander K. Lew, and Vikash K. Mansinghka. Gen: A general-purpose probabilistic programming system with programmable inference. In *Proceedings of the 40th ACM SIGPLAN Conference on Programming Language Design and Implementation*, PLDI 2019, pages 221–236, New York, NY, USA, 2019. ACM.

[11] Petros Dellaportas, Jonathan J Forster, and Ioannis Ntzoufras. Bayesian variable selection using the gibbs sampler. *BIOSTATISTICS-BASEL-*, 5:273–286, 2000.

[12] Vu Dinh, Arman Bilge, Cheng Zhang, and Frederick A Matsen, IV. Probabilistic path hamiltonian monte carlo. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, pages 1009–1018, Sydney, NSW, Australia, 2017. JMLR.org.

[13] Simon Duane, A D Kennedy, Brian J Pendleton, and Duncan Roweth. Hybrid monte carlo. *Phys. Lett. B*, 195(2):216–222, September 1987.

[14] Hong Ge, Kai Xu, and Zoubin Ghahramani. Turing: a language for flexible probabilistic inference. In *International Conference on Artificial Intelligence and Statistics, AISTATS 2018, 9-11 April 2018, Playa Blanca, Lanzarote, Canary Islands, Spain*, pages 1682–1690, 2018.

[15] Donna Harman. Overview of the first TREC conference. In *Proceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 36–47. dl.acm.org, 1993.

[16] M D Hoffman and A Gelman. The No-U-Turn sampler: adaptively setting path lengths in hamiltonian monte carlo. *J. Mach. Learn. Res.*, 2014.

[17] Chris C Holmes and Leonhard Held. Bayesian auxiliary variable models for binary and multinomial regression. *Bayesian Anal.*, 1(1):145–168, March 2006.

[18] Ravin Kumar, Carroll Colin, Ari Hartikainen, and Osvaldo A. Martin. ArviZ a unified library for exploratory analysis of Bayesian models in Python. *The Journal of Open Source Software*, 2019.

[19] Jun S Liu. Peskun's theorem and a modified discrete-state gibbs sampler. *Biometrika*, 83(3):681–682, September 1996.

[20] David Mimno, Hanna Wallach, and Andrew McCallum. Gibbs sampling for logistic normal topic models with graph-based priors. In *NIPS Workshop on Analyzing Graphs*, volume 61. people.cs.umass.edu, 2008.

[21] Hadi Mohasel Afshar and Justin Domke. Reflection, refraction, and hamiltonian monte carlo. In C Cortes, N D Lawrence, D D Lee, M Sugiyama, and R Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 3007–3015. Curran Associates, Inc., 2015.

[22] Radford M. Neal. MCMC using Hamiltonian dynamics. *Handbook of Markov Chain Monte Carlo*, 54:113–162, 2010.

[23] Akihiko Nishimura, David Dunson, and Jianfeng Lu. Discontinuous hamiltonian monte carlo for discrete parameters and discontinuous likelihoods. *arXiv:1705.08510*, August 2018.

[24] Ari Pakman and Liam Paninski. Auxiliary-variable exact hamiltonian monte carlo samplers for binary distributions. In C J C Burges, L Bottou, M Welling, Z Ghahramani, and K Q Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 2490–2498. Curran Associates, Inc., 2013.

[25] Du Phan, Neeraj Pradhan, and Martin Jankowiak. Composable effects for flexible and accelerated probabilistic programming in NumPyro. December 2019.

[26] Nicholas G Polson, James G Scott, and Jesse Windle. Bayesian inference for logistic models using Pólya–Gamma latent variables. *J. Am. Stat. Assoc.*, 108(504):1339–1349, December 2013.

[27] John Salvatier, Thomas V. Wiecki, and Christopher Fonnesbeck. Probabilistic programming in python using PyMC3. *PeerJ Computer Science*, 2:e55, apr 2016.

[28] Siu Kwan Lam Continuum Analytics, Austin, Texas, Antoine Pitrou Continuum Analytics,, and Stanley Seibert Continuum Analytics,. Numba | proceedings of the second workshop on the LLVM compiler infrastructure in HPC. https://dl.acm.org/doi/pdf/10.1145/2833157.2833162. Accessed: 2020-2-6.

[29] Yuan Zhou, Bradley J Gram-Hansen, Tobias Kohn, Tom Rainforth, Hongseok Yang, and Frank Wood. LF-PPL: A Low-Level first order probabilistic programming language for Non-Differentiable models. March 2019.

[30] Manuela Zucknick and Sylvia Richardson. MCMC algorithms for bayesian variable selection in the logistic regression model for large-scale genomic applications. February 2014.