Conceptual motivation of MCMC using Hamiltonian Dynamics

Anubhav Mittal
anubhavm@iitk.ac.in
Supervised by: Prof. Satyadev Nandakumar and Prof. K. Vasudevan

***Abstract*—Hamiltonian Monte-Carlo was originally developed in the late 1980s as Hybrid Monte Carlo to tackle calculations in Lattice Quantum Chromodynamics. From there, it has found its way into the mainstream of statistical computing through the work of Prof. Radford M. Neal [1] [2] [3] and the like [4] [5].**

The aim of this review is to first show the problem of computing expectations encountered in high dimensions and show how the usual MCMC performs. Then we describe Hamiltonian dynamics, and show how to use it to construct a Markov chain Monte Carlo method. We motivate the choice of different energy functions, the different approximations involved, as well as discuss the problems that arise and how to address them.

The first step is to define a Hamiltonian function in terms of the probability distribution we wish to sample from. In addition to the variables we are interested in (the position variables), we must introduce auxiliary momentum variables, which typically have independent Gaussian distributions. The Hamiltonian Monte Carlo method alternates simple updates for these momentum variables with Metropolis updates in which a new state is proposed by computing a trajectory according to Hamiltonian dynamics, implemented with the "leapfrog method". A state proposed in this way can be distant from the current state but nevertheless have a high probability of acceptance. This bypasses the slow exploration of the state space that occurs when Metropolis updates are done using a simple random-walk proposal distribution.

We also include some of the popular extensions to Hamiltonian Monte Carlo and how they address specific problems.

I. Computing expectations by exploring probability distributions

A. Expectation of a given function

The ultimate undertaking in statistical computing is evaluating expectations with respect to some distinguished target probability distribution. For example, we might be interested in extracting information from a posterior distribution over model configuration space in Bayesian inference. Here we will be agnostic, considering only a target distribution, π, on a D-dimensional sample space, Q, and the corresponding expectations of functions, $E_\pi[f]$. Assuming the distribution is smooth over the space Q, we have:

$$E_\pi[f] = \int_Q \pi(q) f(q) dq$$

Unfortunately, for any nontrivial target distribution we will not be able to evaluate these integrals analytically, and we must instead resort to numerical methods which only approximate them. For a method to scale to the complex problems at the frontiers of applied statistics, it has to make effective use of each and every evaluation of the target density, $\pi(q)$, and relevant functions, $f(q)$. Optimizing these evaluations is a subtle problem frustrated by the natural geometry of probability distributions, especially over high-dimensional parameter spaces.

B. Typical set

One way to ensure computational inefficiency is to waste computational resources evaluating the target density and relevant functions in regions of parameter space that have negligible contribution to the desired expectation. Intuitively, we should consider the neighborhood around the mode where the density is maximized. However, expectation values are given by accumulating the integrand over a volume of parameter space and, while the density is largest around the mode, there is not much volume there. To identify the regions of parameter space that dominate expectations we need to consider the behavior of both the density and the volume. In high-dimensional spaces the volume behaves very differently from the density, resulting in a tension that concentrates the significant regions of parameter space away from either extreme.

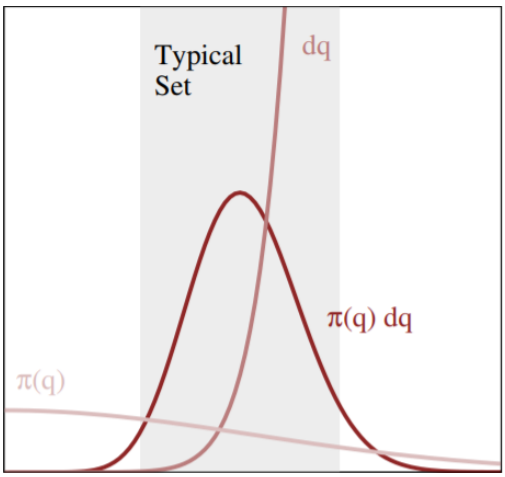

Fig. 1. Typical Set

The neighborhood immediately around the mode features large densities, but in more than a few dimensions the small volume of that neighborhood prevents it from having much contribution to any expectation. On the other hand, the complimentary neighborhood far away from the mode

features a much larger volume, but the vanishing densities lead to similarly negligible contributions expectations. The only significant contributions come from the neighborhood between these two extremes known as the **typical set**, as shown in figure 1.

Importantly, because probability densities and volumes transform oppositely under any reparameterization, the typical set is an invariant object that does not depend on the irrelevant details of any particular choice of parameters.

Consequently, in order to compute expectations efficiently, we have to be able to identify, and then focus our computational resources into, the typical set.

II. Markov Chain Monte Carlo

A. Estimation using MCMC

Markov chain Monte Carlo uses a Markov chain to stochastically explore the typical set, generating a random grid across the region of high probability from which we can construct accurate expectation estimates. Given sufficient computational resources a properly designed Markov chain will eventually explore the typical set of any distribution. The more practical, and much more challenging question, however, is whether a given Markov chain will explore a typical set in the finite time available in a real analysis.

For constructing a Markov chain, we can think of a Markov transition as a conditional probability density, $T(q'|q)$, defining to which point, q', we are most likely to jump from the initial point, q.

$$\pi(q) = \int_Q dq' \pi(q') T(q|q'))$$

So long as this condition holds, at every initial point the Markov transition will concentrate towards the typical set. Consequently, no matter where we begin in parameter space the corresponding Markov chain will eventually drift into, and then across, the typical set. Given sufficient time, the history of the Markov chain, $q_0, ..., q_N$, denoted samples generated by the Markov chain, becomes a convenient quantification of the typical set. In particular, we can estimate expectations across the typical set, and hence expectations across the entire parameter space, by averaging the target function over this history,

$$\hat{f}_N = \frac{1}{N} \sum_{n=0}^{N} f(q_n)$$

If the chain is run long enough, and under certain assumption on the target density, we can claim that the estimate will converge to the true expectation:

$$\lim_{N \to \infty} \hat{f}_N = E_\pi[f]$$

Ideally, the chain will have three phases:

- . In the first phase the Markov chain converges towards the typical set from its initial position in parameter space while the Markov chain Monte Carlo estimators suffer from strong biases.
- The second phase begins once the Markov chain finds the typical set and persists through the first sojourn across the typical set. This initial exploration is extremely effective and the accuracy of Markov chain Monte Carlo estimators rapidly improves as the bias from the initial samples is eliminated.
- The third phase consists of all subsequent exploration where the Markov chain refines its exploration of the typical set and the precision of the Markov chain Monte Carlo estimators improves, albeit at a slower rate

Once the Markov chain has entered into this third phase the Markov chain Monte Carlo estimators satisfy a Central Limit Theorem:

$$\hat{f}_N^{MCMC} \sim \mathcal{N}(E_\pi[f], MCMC - SE)$$

where the *Markov Chain Monte Carlo Standard error* is given by:

$$MCMC - SE = \sqrt{\frac{Var_\pi[f]}{ESS}}$$

and *Effective sample size* is defined as:

$$ESS = \frac{N}{1 + 2\sum_{l=1}^{\infty} \rho_l}$$

where ρ_l is the lag-l autocorrelation of f over the history of the Markov chain. The effective sample size quantifies the number of exact samples from the target distribution necessary to give an equivalent estimator precision and hence the effective number of exact samples contained in the Markov chain; we can also interpret the effective sample size as the total number of sojourns the Markov chain has made across the typical set. In practice the effective sample size can be estimated from the Markov chain itself, although care must be taken to avoid biases.

Because the states of the Markov chain generated during the initial convergence phase mostly bias Markov chain Monte Carlo estimators, we can drastically improve the precision of these estimators by using only those samples generated once the Markov chain has begun to explore the typical set. Consequently, it is common practice to warm up the Markov chain by throwing away those initial converging samples before computing Markov chain Monte Carlo estimators. Warm-up can also be extended to allow for any degrees of freedom in the Markov transition to be empirically optimized without biasing the subsequent estimators.

The Metropolis-Hastings Algorithm This algorithm defines the transition function for constructing our Markov chain. The Metropolis-Hastings algorithm is comprised of two steps: a proposal and a correction. The proposal is any stochastic perturbation of the initial state while the correction rejects any proposals that stray too far away from the typical set of the target distribution. More formally, let $Q(q'|q)$ be the probability density defining each proposal. The probability of accepting a given proposal is then given by

$$a(q'|q) = \min(1, \frac{Q(q|q')\pi(q')}{Q(q'|q)\pi(q)})$$

The most common proposal distribution used is Gaussian, where the algorithm is now called Random-Walk Metropolis and due to symmetry of the gaussian distribution, the acceptance probability takes the following form:

$$a(q'|q) = \min(1, \frac{\pi(q')}{\pi(q)})$$

B. Problems

The idealized behavior of MCMC requires that the Markov transition is compatible with the structure of the target distribution. When the target distribution exhibits pathological behavior, like in a target probability distribution where the typical set pinches into a region of high curvature, Markov transitions will have trouble exploring and Markov chain Monte Carlo will fail.

Even if we have a well-behaved target distribution, we may still encounter problems. Because of its conceptual simplicity and the ease in which it can be implemented by practitioners, Random Walk Metropolis is still popular in many applications. Unfortunately, that seductive simplicity hides a performance that scales poorly with increasing dimension and complexity of the target distribution. As the dimension of the target distribution increases, the volume exterior to the typical set overwhelms the volume interior to the typical set, and almost every Random Walk Metropolis proposal will produce a point on the outside of the typical set, towards the tails. The density of these points, however, is so small, that the acceptance probability becomes negligible. In this case almost all of the proposals will be rejected and the resulting Markov chain will only rarely move.

Consequently, if want to scale Markov chain Monte Carlo to the high-dimensional probability distributions of practical interest then we need a better way of exploring the typical set. In particular, we need to better exploit the geometry of the typical set itself.

C. Need for better transition functions

As seen in the problems encountered above, we need to define a different transition function. We need to exploit information about the geometry of the typical set. Specifically, we need transitions that can follow those contours of high probability mass, coherently gliding through the typical set.

When the sample space is continuous, a natural way of encoding this direction information is with a vector field aligned with the typical set. A vector field is the assignment of a direction at every point in parameter space, and if those directions are aligned with the typical set then they act as a guide through this neighborhood of largest target probability.

To construct such a vector field, we exploit the differential structure of the target distribution which we can query through the gradient of the target probability density function. In particular, the gradient defines a vector field in parameter space sensitive to the structure of the target distribution.

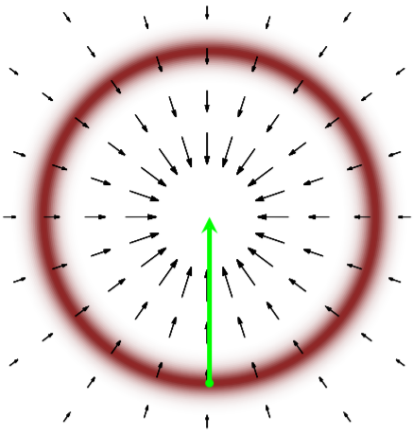

Fig. 2. Direction of the gradient pulls away from the typical set

Unfortunately, that sensitivity is not sufficient as the gradient will never be aligned with the typical set. Following the guidance of the gradient pulls us away from the typical set and towards the mode of the target density. To utilize the information in the gradient we need to complement it with additional geometric constraints, carefully removing the dependence on any particular parameterization while twisting the directions to align with the typical set.

III. Hamiltonian Dynamics

Hamiltonian dynamics has a physical interpretation that can provide useful intuitions. In two dimensions, we can visualize the dynamics as that of a frictionless puck that slides over a surface of varying height. The state of this system consists of the position of the puck, given by a 2D vector q, and the momentum of the puck (its mass times its velocity), given by a 2D vector p. The potential energy, $U(q)$, of the puck is proportional to the height of the surface at its current position, and its kinetic energy, $K(p)$, is equal to $|p|^2/(2m)$, where m is the mass of the puck. On a level part of the surface, the puck moves at a constant velocity, equal to p/m. If it encounters a rising slope, the pucks momentum allows it to continue, with its kinetic energy decreasing and its potential energy increasing, until the kinetic energy (and hence p) is zero, at which point it will slide back down (with kinetic energy increasing and potential energy decreasing).

In non-physical MCMC applications of Hamiltonian dynamics, the position will correspond to the variables of interest. The potential energy will be minus the log of the probability density for these variables. Momentum variables, one for each position variable, will be introduced artificially.

A. Hamilton's equations

Hamiltonian dynamics operates on a d-dimensional position vector, q, and a d-dimensional momentum vector, p, so that the full state space has $2d$ dimensions. The system is described by a function of q and p known as the Hamiltonian, $H(q, p)$.

Equations of motion The partial derivatives of the Hamiltonian determine how q and p change over time, t, according to Hamiltons equations:

$$\frac{dq_i}{dt} = \frac{\partial H}{\partial p_i}$$
$$\frac{dp_i}{dt} = -\frac{\partial H}{\partial q_i}$$

for $i = 1, ..., d$.

Potential and kinetic energy For Hamiltonian Monte Carlo, we usually use Hamiltonian functions that can be written as follows:

$$H(q, p) = U(q) + K(p)$$

Here, $U(q)$ is called the potential energy, and will be defined to be minus the log probability density of the distribution for q that we wish to sample, plus any constant that is convenient. $K(p)$ is called the kinetic energy, and is usually defined as

$$K(p) = \frac{p^T M^1 p}{2}$$

Here, M is a symmetric, positive-definite mass matrix, which is typically diagonal, and is often a scalar multiple of the identity matrix.

With these forms, the Hamilton's equations reduce to the following:

$$\frac{dq_i}{dt} = [M^{-1}p]_i$$
$$\frac{dp_i}{dt} = -\frac{\partial U}{\partial q_i}$$

B. Discretizing the Hamilton's equations

For simplicity, we assume that M is diagonal, with diagonal elements

$$m_1, ..., m_d$$

, so that:

$$K(p) = \sum_{i=1}^{d} \frac{p_i^2}{m_i}$$

Eulers method For Hamiltons equations, this method performs the following steps, for each component of position and momentum, indexed by $i = 1, ..., d$:

$$p_i(t + \epsilon) = p_i(t) + \epsilon \frac{dp_i}{dt}(t) = p_i(t) - \epsilon \frac{\partial U}{\partial q_i}(t)$$
$$q_i(t + \epsilon) = q_i(t) + \epsilon \frac{dq_i}{dt}(t) = q_i(t) + \epsilon \frac{p_i(t)}{m_i}$$

In practise, however, the results are not that good, so we slightly modify the method to our liking.

A modification of Eulers method Much better results can be obtained by slightly modifying Eulers method, as follows:

$$p_i(t + \epsilon) = p_i(t) - \epsilon \frac{\partial U}{\partial q_i}(t)$$
$$q_i(t + \epsilon) = q_i(t) + \epsilon \frac{p_i(t + \epsilon)}{m_i}$$

We simply use the new value for the momentum variables, p_i , when computing the new value for the position variables, q_i . A method with similar performance can be obtained by instead updating the q_i first and using their new values to update the p_i . We get better results using this method, but we can still improve further with almost no extra overhead, as described below.

The leapfrog method Even better results can be obtained with the leapfrog method, which works as follows:

$$p_i(t + \frac{\epsilon}{2}) = p_i(t) - \frac{\epsilon}{2} \frac{\partial U}{\partial q_i}(t)$$
$$q_i(t + \epsilon) = q_i(t) + \epsilon \frac{p_i(t + \frac{\epsilon}{2})}{m_i}$$
$$p_i(t + \epsilon) = p_i(t + \frac{\epsilon}{2}) - \frac{\epsilon}{2} \frac{\partial U}{\partial q_i}(t + \epsilon)$$

IV. MCMC from Hamiltonian dynamics

Using Hamiltonian dynamics to sample from a distribution requires translating the density function for this distribution to a potential energy function and introducing momentum variables to go with the original variables of interest (now seen as position variables). We can then simulate a Markov chain in which each iteration resamples the momentum and then does a Metropolis update with a proposal found using Hamiltonian dynamics.

A. Canonical distributions

The distribution we wish to sample can be related to a potential energy function via the concept of a canonical distribution from statistical mechanics. Given some energy function, $E(x)$, for the state, x, of some physical system, the canonical distribution over states has probability or probability density function:

$$P(x) = \frac{1}{Z} exp(-E(x)/T)$$

Viewing this the opposite way, if we are interested in some distribution with density function $P(x)$, we can obtain it as a canonical distribution with $T = 1$ by setting $E(x) = logP(x)logZ$, where Z is any convenient positive constant.

$$P(q, p) = \frac{1}{Z} exp(-E(q, p))$$

Since $H(q, p) = U(q) + K(p)$,

$$P(q, p) = \frac{1}{Z} exp(-U(q))exp(-K(p))$$

we see that q and p are independent, and each have canonical distributions, with energy functions $U(q)$ and $K(p)$. We will use q to represent the variables of interest, and introduce p just to allow Hamiltonian dynamics to operate.

In Bayesian statistics, the posterior distribution for the model parameters is the usual focus of interest, and hence these parameters will take the role of the position, q. We can express the posterior distribution as a canonical distribution using a potential energy function defined as follows:

$$U(q) = -\log[\pi(q)L(q|D)]$$

where $\pi(q)$ is the prior density, and $L(q|D)$ is the likelihood function given data D.

B. The algorithm for Hamiltonian MCMC

We make some general assumptions for ease of notation. We assume that the density is non-zero everywhere. We must also be able to compute the partial derivatives of the log of the density function. These derivatives must therefore exist, except perhaps on a set of points with probability zero, for which some arbitrary value could be returned.

We use the values of $K(p)$ and $U(q)$ as defined before. There are two main steps in the algorithm:

1) In the first step, new values for the momentum variables are randomly drawn from their Gaussian distribution, independently of the current values of the position variables. For the kinetic energy, the d momentum variables are independent, with p_i having mean zero and variance m_i . Since q isnt changed, and p is drawn from its correct conditional distribution given q (the same as its marginal distribution, due to independence), this step obviously leaves the canonical joint distribution invariant.
2) For the second step, we need some background for the details. There is one important exception to performance of the leapfrog method. Long time accuracy can be compromised when the exact energy level sets feature neighborhoods of high curvature that the finite time discretization is not able to resolve. These neighborhoods induce a divergence that almost immediately propels the numerical trajectory towards infinite energies. This distinctive behavior proves beneficial in practice, however, because it makes the failures of the method straightforward to identify and hence diagnose. An intuitive way to correct this is to treat the Hamiltonian transition as the proposal for a Metropolis Hastings scheme on phase space.
   In the second step, a Metropolis update is performed, using Hamiltonian dynamics to propose a new state. Starting with the current state, (q, p), Hamiltonian dynamics is simulated for L steps using the Leapfrog method (or some other reversible method that preserves volume), with a stepsize of ϵ. Here, L and ϵ are parameters of the algorithm, which need to be tuned to obtain good performance.
   We encounter a problem here.Because Hamiltonian trajectories, and their numerical approximations, are deterministic and non-reversible, Metropolis-Hastings proposals are always rejected. In particular, we have positive proposal probabilities going forwards in time but vanishing proposal probabilities going backwards in time which renders the Metropolis-Hastings acceptance probability identically zero.
   If we modify the Hamiltonian transition to be reversible, however, then the ratio of proposal densities becomes non-zero and we achieve a useful correction scheme. The simplest way of achieving a reversible proposal is to augment the the numerical integration with a negation step that flips the sign of momentum,

$$(p_L, q_L) \to (p_L, -q_L)$$

   Hence, the momentum variables at the end of the L-step trajectory are then negated, giving a proposed state $(q*, p*)$. This proposed state is accepted as the next state of the Markov chain with probability:

$$min[1, exp(-H(q*, p*) + H(q, p)]$$
$$= min[1, exp(-U(q*) + U(q) - K(p*) + K(p)]$$

If the proposed state is not accepted (ie, it is rejected), the next state is the same as the current state (and is counted again when estimating the expectation of some function of state by its average over states of the Markov chain).

Importance of first step The resampling of the momentum variables is still crucial to obtaining the proper distribution for q. Without resampling, $H(q, p) = U(q) + K(p)$ will be (nearly) constant, and since $K(p)$ and $U(q)$ are non-negative, $U(q)$ could never exceed the initial value of $H(q, p)$ if no resampling for p were done.

C. Avoiding random walks

Avoidance of random-walk behaviour is one major benefit of Hamiltonian Monte Carlo. To see this, note that the variance of the position after n iterations of random walk Metropolis from some start state will grow in proportion to n (until this variance becomes comparable to the overall variance of the state), since the position is the sum of mostly independent movements for each iteration. The standard deviation of the amount moved (which gives the typical amount of movement) is therefore proportional to $\sqrt{n}$.

The stepsize used for the leapfrog steps is similarly limited by the most constrained direction, but the movement will be in the same direction for many steps. The distance moved after n steps will therefore tend to be proportional to n, until the distance moved becomes comparable to the overall width of the distribution. The advantage compared to movement by a random walk will be a factor roughly equal to the ratio of the standard deviations in the least confined direction and most confined direction.

V. Efficient Hamiltonian Monte Carlo

:

An immediate complication with this foundational construction is that it does not define a unique Markov transition but rather an infinity of them. Every choice of kinetic energy and integration time yields a new Hamiltonian transition that will interact differently with a given target distribution. Unfortunately, these interactions will usually lead to suboptimal performance and we are left with a delicate tuning problem. When these degrees of freedom are well-chosen, the resulting implementation of Hamiltonian Monte Carlo will perform well on even the challenging, high-dimensional problems of applied interest. When they are poorly-chosen, however, the performance can suffer dramatically.

In order to be able to optimize the application of the Hamiltonian Monte Carlo method and ensure robust performance, we need to understand exactly how these degrees of freedom interact with the target distribution. Although this seems like

a daunting task, we can facilitate it by exploiting the latent geometry of Hamiltonian Monte Carlo itself. In particular, the analysis is make much easier by considering a different view of phase space.

A. *The Natural Geometry of Phase Space*

One of the characteristic properties of Hamiltons equations is that they conserve the value of the Hamiltonian. In other words, every Hamiltonian trajectory is confined to an energy level set,

$$H^{-1}(E) = \{\ q, p\ | H(q,p) = E\}$$

which, save for some ignorable exceptions, are all $(2D1)$ dimensional, compact surfaces in phase space. In fact, once weve removed any singular level sets, the entirety of phase space neatly decomposes, or foliates into concentric level sets:

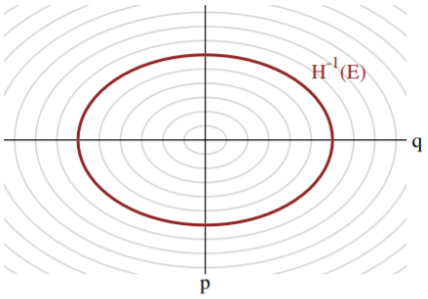

Fig. 3. Phase space naturally decomposes into level sets of the Hamiltonian, $H^{-1}(E)$

Consequently, we can specify any point in phase space by first specifying the energy of the level set it falls on, E, and the position within that level set, θ_E.

Correspondingly the canonical distribution on phase space admits a microcanonical decomposition,

$$\pi(q,p) = \pi(\theta_E \mid E)\pi(E)$$

across this foliation. The conditional distribution over each level set, $\pi(\theta_E \mid E)$, is called the microcanonical distribution, while the distribution across the level sets, $\pi(E)$, is called the marginal energy distribution. Because they are derived from the same geometry, this microcanonical decomposition is particularly well-suited to analyzing the Hamiltonian transition.

To understand this more clearly, consider a Hamiltonian Markov chain consisting of multiple transitions. Each Hamiltonian trajectory explores a level set while the intermediate projections and lifts define a random jump between the level sets themselves. Consequently, the entire Hamiltonian Markov chain decouples into two distinct phases: deterministic exploration of individual level sets and a stochastic exploration between the level sets themselves.

This decoupling makes it particularly convenient to analyze the efficiency of each phase, and hence the efficiency of the overall Hamiltonian Markov transition. For example, the efficacy of the deterministic exploration is determined by how long the Hamiltonian trajectories are integrated and, consequently, how completely they explore the corresponding level sets. The cost of this phase, however, is ultimately proportional to the total integration time. The integration time needed to explore just enough of each level set, and hence the overall efficiency of the deterministic exploration, depends on the geometry of the energy level sets. The more uniform and regular the level sets, the faster the trajectories will explore for a given integration time.

Similarly, the performance of the stochastic exploration is determined by how quickly the random walk can diffuse across the energies typical to the marginal energy distribution. Writing $\pi(E \mid q)$ as the transition distribution of energies induced by a momentum resampling at a given position, q, the diffusion speed depends on how heavy-tailed the marginal energy distribution is relative to $\pi(E \mid q)$. For example, if this energy transition distribution is narrow relative to the marginal energy distribution, then the random walk will proceed very slowly, taking many costly transitions to completely explore the target distribution. If the energy transition distribution is similar to the marginal energy distribution, however, then we will generate nearly-independent samples from the marginal energy distribution at every transition, rapidly surveying the relevant energies with maximal efficiency.

By analyzing how these algorithmic degrees of freedom in the Hamiltonian Markov transition interact with the target distribution to determine the microcanonical geometry,we can determine how they affect the performance of the resulting Hamiltonian Markov chain. In particular, we can construct criteria that identify the optimal choices of these degrees of freedom, which then motivate the effective tuning of the method, even for the complex target distributions encountered in practice.

B. *Optimizing the Choice of Kinetic Energy*

The first substantial degree of freedom in the Hamiltonian Monte Carlo method that we can tune is the choice of the conditional probability distribution over the momentum or, equivalently, the choice of a kinetic energy function. Along with the target distribution, this choice completes the probabilistic structure on phase space which then determines the geometry of the microcanonical decomposition. Consequently, the ideal kinetic energy will interact with the target distribution to ensure that the energy level sets are as uniform as possible while the energy transition distribution matches the marginal energy distribution as well as possible.

Unfortunately, there is an infinity of possible kinetic energies and it would be impossible to search through them all to find an optimal choice for any given target distribution. It is much more practical to search through a restricted family of kinetic energies, especially if that family is built from the structure of the problem itself.

Euclidean-Gaussian Kinetic Energies:

For example, in many problems the sample space is endowed with a Euclidean metric, g, that allows us to measure, amongst other quantities, the distance between any two points.

In a given parameterization, g is represented with a $D\ X\ D$ matrix from which we can compute distances as

$$\delta(q, q') = (q - q')^T g (q - q')$$

Moreover, we can construct an entire family of modified Euclidean metrics, M, by scaling and then rotating this natural metric,

$$M = R\ S\ g\ S^T\ R^T$$

where S is a diagonal scaling matrix and R is an orthogonal rotation matrix.

Any such Euclidean structure on the target parameter space immediately induces an inverse structure on the momentum space, allowing us to measure distances between momenta,

$$\delta(p, p') = (p - p')^T M^{-1} (p - p')$$

Finally, distances in momentum space allow us to construct many common probability distributions over the momentum, such as a Gaussian distribution centered at 0,

$$\pi(p|q) = \mathcal{N}(p \mid 0, M)$$

This particular choice defines a Euclidean-Gaussian kinetic energy,

$$K(q, p) = \frac{1}{2} p^T M^{-1} p + \log |M| + const$$

In the physical perspective the Euclidean metric is known as the mass matrix, a term that has consequently become common in the Hamiltonian Monte Carlo literature.

Because the Euclidean structure over the momentum is dual to the Euclidean structure over the parameters, its interactions with the target distribution are straightforward to derive. Applying the transformation $p' = \sqrt{M^{-1}} p$ simplifies the kinetic energy, but remember that we have to apply the opposite transformation to the parameters, $q' = \sqrt{M} q$ to preserve the Hamiltonian geometry. Consequently, a choice of M^{-1} effectively rotates and then rescales the target parameter space, potentially correlating or de-correlating the target distribution and correspondingly warping the energy level sets.

In particular, as the inverse Euclidean metric more closely resembles the covariance of the target distribution it de-correlates the target distribution, resulting in energy level sets that are more and more uniform and hence easier to explore. We can then readily optimize over the family of Euclidean-Gaussian kinetic energies by setting the inverse Euclidean metric to the target covariances,

$$M^{-1} = E_\pi[(q - \mu)(q - \mu)^T]$$

In practice we can compute an empirical estimate of the target covariance using the Markov chain itself in an extended warm-up phase. After first converging to the typical set we run the Markov chain using a default Euclidean metric for a short window to build up an initial estimate of the target covariance, then update the metric to this estimate before running the now better-optimized chain to build up an improved estimate. A few iterations of this adaptation will typical yield an accurate estimate of the target covariance and hence a near-optimal metric.

Riemannian-Gaussian Kinetic Energies:

Unless the target distribution is exactly Gaussian, however, no global rotation and rescaling will yield completely uniform level sets; locally the level sets can still manifest strong curvature that slows the exploration of the Hamiltonian trajectories. To improve further we need to introduce a Riemannian metric which, unlike the Euclidean metric, varies as we move through parameter space. A Riemannian structure allows us to construct a Gaussian distribution over the momentum whose covariance depends on our current position in parameter space,

$$\pi(p|q) = \mathcal{N}(p, 0, \Sigma(q))$$

which then defines a Riemannian-Gaussian kinetic energy,

$$K(q, p) = \frac{1}{2} p^T \Sigma^{-1}(q) p + \frac{1}{2} \log |\Sigma(q)| + const$$

The resulting implementation of Hamiltonian Monte Carlo is known as Riemannian Hamiltonian Monte Carlo. Further details have been omitted.

Non-Gaussian Kinetic Energies:

In theory we are not limited to Gaussian distributions over the momentum a Euclidean or Riemannian structure allows us to construct any distribution with quadratic sufficient statistics. In particular, why should we not consider momentum distributions with particularly heavy or light tails? Although there is not much supporting theory, empirically non-Gaussian kinetic energies tend to perform poorly, especially in high-dimensional problems. Some intuition for the superiority of Gaussian kinetic energies may come from considering the asymptotics of the marginal energy distribution. As we target higher and higher dimensional models, the marginal energy distribution becomes a convolution of more and more parameters and, under relatively weak conditions, it tends to follow a central limit theorem. When the marginal energy distribution converges towards a Gaussian, only a Gaussian distribution over the momentum will yield the optimal energy transition.

C. Optimizing the Choice of Integration Time

The choice of a kinetic energy completely specifies the microcanonical geometry and, consequently, the shape of the energy level sets. How effectively each Hamiltonian trajectory explores those level sets is then determined entirely by the choice of integration times across phase space, $T(q, p)$. Intuitively, if we integrate for only a short time then we dont take full advantage of the coherent exploration of the Hamiltonian trajectories and we will expect performance to suffer. On the other hand, because the level sets are topologically compact in well-behaved problems, trajectories will eventually return to previously explored neighborhoods and integrating too long can suffer from diminishing returns.

This intuition is formalized in the notion of dynamic ergodicity. Here we consider the orbit, , of a trajectory, consisting of all points that the trajectory will reach as the integration time is increased to infinity. The orbit might encompass the entire level set or it could be limited to a just subset of the level set, but in either case any trajectory will explore the microcanonical distribution restricted to its orbit. Dynamic

ergodicity guarantees that a uniform sample from a trajectory will more closely resemble a sample from this restricted microcanonical distribution as the integration time is increased and the trajectory grows. In other words, as the integration time grows the temporal expectation over the trajectory converges to the spatial expectation over its orbit.

The performance of the Hamiltonian transition, however, depends on the rate at which these expectations converge. Given typical regularity conditions, the temporal expectation will initially converges toward the spatial expectation quite rapidly, consistent with our intuition that coherent exploration is extremely effective. Eventually, however, that convergence slows, and we enter an asymptotic regime where any benefit of exploration comes at an ever increasing cost. The optimal integration time straddles these two regimes, exploiting the coherent exploration early on but not wasting computation on the diminishing returns of long integration times.

This optimization criterion also has a helpful geometric interpretation. The superlinear regime corresponds to the first sojourn around the orbit of the trajectory, where every new step forwards is new and informative. Eventually, however, the trajectory returns to neighborhoods it has already explored and enters into the asymptotic regime. This additional exploration refines the exploration of the orbit, improving the accuracy of the temporal expectation only very slowly.

In general, this optimal integration time will vary strongly depending on which trajectory we are considering: no single integration time will perform well everywhere. We can make this explicit in one dimension where the optimal integration times can be identified analytically. For example, for the family of target densities,

$$\pi_\beta(q) \propto e^{-|q|^\beta}$$

with the Euclidean-Gaussian kinetic energy

$$\pi(p|q) - \mathcal{N}(0, 1),$$

the optimal integration time scales with the energy of the level set containing the trajectory,

$$T_{\text{optimal}}(q, p) \propto (H(q, p))^{\frac{2-\beta}{2\beta}}$$

In particular, when the target distribution is heavy-tailed, $\beta < 2$, the optimal integration time will quickly grow as we move from trajectories exploring the bulk to trajectories exploring the tails. Consequently, the exploration generated by trajectories with any static integration time will decay and the Hamiltonian Markov chain will slow to a crawl.

Hence if we want to fully exploit these Hamiltonian trajectories then we need to identify the optimal integration time dynamically, as we generate the trajectories themselves. How exactly do we identify when a trajectory has reached its optimal length? One heuristic is the No-U-Turn termination criterion which, like the kinetic energies discussed before, utilizes a Euclidean or Riemannian structure on the target parameter space. The explicit form is omitted here.

In some simple cases the near-optimality of the No-U-Turn criterion can be shown rigorously, but it has proven a empirical success on an incredibly diverse set of target distributions encountered in applied problems. More recently proposed possibilities are exhaustive termination criteria which utilize the microcanonical geometry itself to identify the optimal stopping time. Exhaustive termination criteria can be more robust than the No-U-Turn termination criterion, but they require careful tuning which is an open topic of research.

VI. Conclusion

By exploiting the geometry of the typical set, Hamiltonian Monte Carlo generates coherent exploration of smooth target distributions. This effective exploration yields not only better computational efficiency than other Markov chain Monte Carlo algorithms, but also stronger guarantees on the validity of the resulting estimators. Moreover, careful analysis of this geometry facilitates principled strategies for automatically constructing optimal implementations of the method, allowing users to focus their expertise into building better models instead of wrestling with the frustrations of statistical computation. Consequently, Hamiltonian Monte Carlo is uniquely suited to tackling the most challenging problems at the frontiers of applied statistics, as demonstrated by the huge success of tools like Stan.

In hindsight, this success should not be particularly surprising. The ultimate challenge in estimating probabilistic expectations is quantifying the typical set of the target distribution, a set which concentrates near a complex surface in parameter space. What more natural way is there to quantify and then exploit information about a surface than through its geometry?

Continuing to leverage this geometric perspective will be critical to further advancing our understanding of the Hamiltonian Monte Carlo method and its optimal implementations. This includes everything from improving geometric ergodicity analysis, refining the existing implementation techniques, and motivating the appropriate use of non-Gaussian and nonEuclidean kinetic energies.

References

[1] R. M. Neal *et al.*, "Mcmc using hamiltonian dynamics," *Handbook of markov chain monte carlo*, vol. 2, no. 11, p. 2, 2011.

[2] R. M. Neal, *Bayesian learning for neural networks*, vol. 118. Springer Science & Business Media, 2012.

[3] R. M. Neal, "An improved acceptance procedure for the hybrid monte carlo algorithm," *Journal of Computational Physics*, vol. 111, no. 1, pp. 194–203, 1994.

[4] M. Betancourt, "A conceptual introduction to hamiltonian monte carlo," 2017.

[5] M. Betancourt, "A general metric for riemannian hamiltonian monte carlo," in *First International Conference on the Geometric Science of Information (F. Nielsen and F. Barbaresco, eds.). Lecture Notes in Computer Science*, vol. 8085, 2013.

[6] S. Holmes, S. Rubinstein-Salzedo, and C. Seiler, "Curvature and concentration of hamiltonian monte carlo in high dimensions," *arXiv preprint arXiv:1407.1114*, 2014.

[7] W. K. Hastings, "Monte carlo sampling methods using markov chains and their applications," 1970.

[8] M. Betancourt, "Identifying the optimal integration time in hamiltonian monte carlo," *arXiv preprint arXiv:1601.00225*, 2016.

[9] M. Betancourt, S. Byrne, and M. Girolami, "Optimizing the integrator step size for hamiltonian monte carlo," *arXiv preprint arXiv:1411.6669*, 2014.

[10] M. J. Betancourt, "Generalizing the no-u-turn sampler to riemannian manifolds," *arXiv preprint arXiv:1304.1920*, 2013.

[11] P. B. Mackenze, "An improved hybrid monte carlo method," *Physics Letters B*, vol. 226, no. 3-4, pp. 369–371, 1989.