Constrained Hamiltonian Monte Carlo with Nested Sampling for use in Lattice Field Theory

Candidate 8256T[1,2]

[1]*Astrophysics Group, Cavendish Laboratory, JJ Thomson Avenue, Cambridge CB3 0HE, UK*

[2]*Kavli Institute for Cosmology, Madingley Road, Cambridge CB3 0HA, UK*

Constrained Hamiltonian Monte Carlo is a novel algorithm for high dimensional nested sampling using gradients. We introduce a set of new methods which are used to demonstrate the first practical implementation incorporating gradients into nested sampling. Current state-of-the-art implementations of nested sampling become very limited for dimensions greater than $D > 1,000$. We show how our algorithm allows exploration of problems with dimension significantly higher, up to $D \sim 500,000$.

We also use the algorithm in a novel application of nested sampling to ϕ^4 lattice field theory, a simple model which exhibits a phase transition. We successfully measure observables on 512×512 sized lattices ($D = 262,144$), exploring the behaviour at criticality. We further demonstrate the resistance of our method to topological freezing and critical slowing down.

I. INTRODUCTION

Since the inception of Lattice Field Theory (LFT), numerical Monte-Carlo methods have been used with great success to study complex systems. The dominant algorithm used for simulations in LFT [1] over the past decades has been Hamiltonian Monte Carlo (HMC) [2], a Markov-Chain Monte-Carlo (MCMC) method used for *parameter estimation*. HMC has since found wide applications throughout a number of different fields in statistics [3, 4].

To recover the physical behaviour of a lattice system, the continuum limit is taken as the system approaches its critical point. At the critical point, MCMC methods suffer from *critical slowing down* [5] and *topological freezing* [6], where all samples proposed by the algorithm are highly autocorrelated. Critical slowing down is a significant problem in LFT and Lattice QCD, with large efforts devoted to combating it's effects [7–13].

A different approach to parameter estimation is with Bayesian inference [14]. A contemporary method for Bayesian inference which has found many applications in astrophysics and cosmology is with nested sampling [15, 16]. Nested sampling offers several advantages to classic MCMC algorithms, including simultaneous calculation of model *evidence*, as well as dealing effectively with multimodal distributions [15].

Modern implementations of nested sampling scale poorly with number of dimensions [17, 18], with current state-of-the-art limited to around $D \approx 1000$ dimensions [19]. This quickly renders nested sampling ineffective for solving problems in LFT, maxing out at small lattices.

We propose a novel algorithm, based off combining nested sampling with a modified constrained HMC, which aims to effectively sample in very high-dimensional parameter space, while being resistant to multimodal distributions and topological freezing.

Incorporating gradients into nested sampling poses a challenging problem which has not had significant development since the introduction of the idea over a decade ago [20, 21]. Sampling within an iso-likelihood contour, while conceptually simple, poses a practical challenge to do reliably on a computer.

Furthermore, manually supplying gradients of likelihoods can be difficult for large models such as in Cosmology [18, 22, 23]. The advent of auto differentiation [24–26] can overcome this challenge by allowing us to automatically calculate gradients without numerical error.

Section II of this paper is an introduction to Bayesian inference and lays out conventions and definitions used. In Section III we explain the nested sampling algorithm and comment previous implementations. Section IV explains Hamiltonian Monte Carlo and describes how it works as a sampler. In Section V we introduce our novel algorithm which modifies HMC for use in nested sampling. We also introduce our new methods for epsilon halving, clustering, and sampling through topological traps. Section VI discusses the necessity for adaptive parameters and describes the novel solutions we propose to achieve this.

We introduce the theoretical background of lattice field theory in Section VII and how it can be solved computationally with Bayesian inference. In Section VIII we give the numerical results for our novel application of nested sampling to ϕ^4-theory. Finally, a discussion of the performance and its comparison to existing methods is given in Section IX.

Further details such as the input parameters used are given in Appendix A, C++ code implementation in Appendix B, and results with auto differentation in Appendix C. The full derivation for metric adaptation is given in Appendix D.

Natural units $c = \hbar = 1$ are used throughout.

II. BAYESIAN INFERENCE

Bayesian inference is a robust analytic framework which allows the construction of predictive models $\mathcal{M}$ in the context of some dataset $\mathcal{D}$.

The *likelihood* is defined as the probability of observing the data given a specific parameter choice θ

$$p(\mathcal{D}|\theta, \mathcal{M}) \equiv \mathcal{L}(\theta). \tag{1}$$

A Bayesian model must also specify it's distribution of parameters before any data is known. This is termed the *prior*, de-

fined by

$$p(\theta|\mathcal{M}) \equiv \pi(\theta). \tag{2}$$

The *evidence* is the distribution of observed data marginalised over the parameters, defined as

$$p(\mathcal{D}|\mathcal{M}) \equiv \mathcal{Z} = \int \mathcal{L}(\theta)\pi(\theta)d\theta. \tag{3}$$

The evidence, or sometimes *marginalised likelihood*, is an important quantity which provides a measure on the quality of a model $\mathcal{M}$.

By using Bayes' Theorem [27], the distribution of parameters θ given our model and data can be written in terms of the quantities

$$\begin{aligned} p(\theta|\mathcal{D},\mathcal{M}) &= \frac{p(\mathcal{D}|\theta,\mathcal{M})p(\theta|\mathcal{M})}{p(\mathcal{D}|\mathcal{M})}, \\ \mathcal{P}(\theta) &= \frac{\mathcal{L}(\theta)\pi(\theta)}{\mathcal{Z}}, \end{aligned} \tag{4}$$

where $\mathcal{P}(\theta)$ is termed the *posterior*. The posterior is the distribution of parameters θ after taking the data into account.

An intuitive relationship between these quantities is that a model where the *prior* more closely resembles the *posterior* will have a greater *evidence* [28].

Calculating the posterior is in the domain of *parameter estimation*, which is often a very difficult task to solve analytically. For high-dimensional problems, we therefore resort to computationally estimating the distribution by sampling from the posterior using Markov-Chain Monte-Carlo techniques [29, 30].

Examples of such sampling algorithms include Metropolis-Hastings [31], Slice sampling [32], and Hamiltonian Monte Carlo [2, 33] which we explore further in section IV.

Single chain-based sampling algorithms struggle with multimodal distributions due to topological freezing. When one sample from the chain falls into a topological feature, it can take a very long time before it escapes and continues to explore the space. This motivates the use of multiple points in parameter space to perform sampling, as introduced by nested sampling.

III. NESTED SAMPLING

Nested sampling is an algorithm which simultaneously computes the evidence and posterior [15, 16, 19]. For a set of parameters θ, calculating the evidence through direct evaluation of the high-dimensional integral (3) becomes exponentially more expensive as the number of dimensions, D, is increased.

We define the *prior volume* as the fraction of prior contained within an iso-likelihood contour

$$X(\lambda) = \int_{\mathcal{L}(\theta)>\lambda} \pi(\theta)d\theta. \tag{5}$$

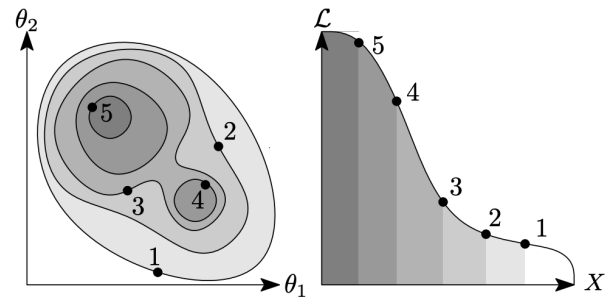

FIG. 1: Nested sampling prior volume transformation. Left: Five sequentially higher iso-likelihood contours of a two-dimensional bimodal likelihood function $\mathcal{L}(\theta)$. Each contour encloses a smaller fraction of the prior volume X. Right: Likelihood $\mathcal{L}(X)$ as a function of enclosed volume X. The Bayesian evidence is the area under the curve. (Figure used from paper [16, 18]).

Using a change of variable we write the evidence as a one-dimensional integral more feasible to calculate

$$\mathcal{Z} = \int_0^1 \mathcal{L}(X)dX. \tag{6}$$

Nested sampling introduces a population of n_{live} *live points* in the parameter space which are sorted by likelihood. These points are then iteratively updated to compress around the peaks of the posterior distribution.

Initially, n_{live} points are sampled from the prior $\pi(\theta)$. At each iteration i, the point with the lowest likelihood $\mathcal{L}_i$ is deleted and moved to the set of *dead points*. A new live point is then generated from the prior, subject to the hard constraint that its likelihood is greater than $\mathcal{L}_i$.

The prior volume on average will contract by a factor $n_{\text{live}}/(n_{\text{live}}+1)$ with each dead point, such that the expected prior volume at iteration i will be

$$\langle X_i \rangle = \left(\frac{n_{\text{live}}}{n_{\text{live}}+1}\right)^i \approx e^{-i/n_{\text{live}}}, \tag{7}$$

thus compressing exponentially for large n_{live}. Each dead point has a likelihood $\mathcal{L}_i$, a set of parameters θ_i, and a prior mass X_i, which can be used to estimate the evidence and posterior.

Current state-of-the-art algorithms for nested sampling are able to calculate model evidences up to $D \sim 1,000$ dimensions before the computational costs become overwhelming [17, 18, 34].

A. Evidence and Parameter Estimation

We can use the dead points generated by the algorithm to computationally estimate the evidence. Approximating the integral (6) using the trapezoid rule, we write

$$\mathcal{Z} = \sum_{i \in \text{dead}} w_i \mathcal{L}_i, \tag{8}$$

where $w_i = (X_{i-1} - X_{i+1})/2$ is the weight factor estimating the change in prior volume per iteration.

We can also use the dead points as samples from the posterior. Given that the i-th sample is assigned an importance weighting w_i, the posterior samples are

$$p_i = \frac{w_i \mathcal{L}_i}{\mathcal{Z}} \propto w_i \mathcal{L}_i. \tag{9}$$

We provide a further discussion of tuning nested sampling input parameters, and its effects on and evidence calculation and errors in Appendix E.

IV. HAMILTONIAN MONTE CARLO

Hamiltonian Monte Carlo (HMC) is a powerful Markov-Chain Monte-Carlo (MCMC) method used to effectively generate random samples from a probability distribution. HMC was originally developed for Lattice QCD [2], but has since found wide applications in statistical physics, computational biology, and machine learning. The most significant advantage of HMC is its ability to sample from very high dimensional ($D >$ millions) complex distributions, which is currently unmatched by other MCMC algorithms.

A. Metropolis–Hastings Algorithm

HMC is a subclass of the Metropolis–Hastings algorithm [31], a MCMC method which comprises of two steps: a proposal and a correction. Given an initial state θ, a new sample θ' is proposed using some stochastic process. The proposal is then corrected to ensure detailed-balance so the Markov-Chain converges to the target distribution $\pi(\theta)$. The correction is performed by choosing to keep the new sample with an acceptance probability

$$A(\theta'|\theta) = \min\left(1, \frac{Q(\theta|\theta')\pi(\theta')}{Q(\theta'|\theta)\pi(\theta)}\right), \tag{10}$$

where $Q(\theta|\theta')$ is the probability distribution of the proposal function. All methods we will explore have symmetric proposal distributions $Q(\theta|\theta') = Q(\theta'|\theta)$, such that the acceptance probability reduces to

$$A(\theta'|\theta) = \min\left(1, \frac{\pi(\theta')}{\pi(\theta)}\right). \tag{11}$$

With a set of N samples generated by Metropolis–Hastings, $\{\theta_i\}$, we can estimate expected values of observables of the target distribution as

$$\langle \mathcal{O} \rangle = \frac{1}{N} \sum_{i=0}^{N} \mathcal{O}(\theta_i). \tag{12}$$

B. Hamiltonian Dynamics

Hamiltonian dynamics is a formulation in classical physics based on the principle of least action. It generalises a system to its position $\mathbf{q}$ and conjugate momenta $\mathbf{p}$ in order to solve problems by working in *phase space*.

The principle of least action states that the equations of motion of a system are given by the stationary point of the action functional. The action is defined as

$$S = \int \mathcal{L}(\mathbf{q}, \dot{\mathbf{q}}) dt, \tag{13}$$

where $\mathcal{L}(\mathbf{q}, \dot{\mathbf{q}})$ is the Lagrangian (not to be confused with likelihood).

The Hamiltonian is then defined as the Legendre transform of the Lagrangian given by

$$H(\mathbf{q}, \mathbf{p}) = \dot{\mathbf{q}} \cdot \mathbf{p} - \mathcal{L} \tag{14}$$

with conjugate momenta defined as $p_i = \partial \mathcal{L} / \partial \dot{q}_i$.

Using the Euler–Lagrange equations to solve for the stationary point of the action yields the equations of motion

$$\frac{dq_i}{dt} = \frac{\partial H}{\partial p_i}, \quad \frac{dp_i}{dt} = -\frac{\partial H}{\partial q_i}, \tag{15}$$

which can be integrated to find the trajectory.

C. HMC Algorithm

HMC derives its strengths from incorporating the use of Hamiltonian dynamics to generate new samples in the Markov-Chain. The main idea is to treat the parameters θ as a position and introduce an auxiliary momentum variable p. Then, treating the target distribution as a potential energy, we can navigate the landscape by solving Hamilton's equations (15) and integrating the trajectory from (θ_0, p_0) to a new point (θ_1, p_1). Finally, a correction step is performed as described in (10) and the new sample is accepted to the Markov-Chain with a given probability.

Suppose we are trying to sample from a target distribution $\pi(\theta)$, the Hamiltonian for HMC is defined as

$$H = \frac{1}{2}\mathbf{p}^T M^{-1} \mathbf{p} - \log \pi, \tag{16}$$

where M is a mass matrix known as the *metric*, and $\mathbf{p}$ is a new momentum variable we have introduced. Intuitively, this treats the Markov-Chain point as a D dimensional particle moving through a potential well $U(\theta) = -\log \pi(\theta)$.

The momentum is stochastically drawn from a normal distribution, $\mathbf{p} \sim \mathcal{N}(0, \sigma = M)$. The metric therefore defines the region of phase space HMC explores and is an important input parameter to tune for effective sampling [35].

Once the Hamiltonian is obtained, the equations of motion (15) are numerically integrated to give a new sample $(\theta_0, p_0) \rightarrow (\theta_1, p_1)$. In order to solve the equations of motion

we therefore rely on the *gradient* of the distribution $\nabla \log \pi(\theta)$. This often presents a practical difficult as not many distributions have easily accessible gradients, however this can be alleviated using *auto differentiation* [24, 26, 36] as discussed further in Appendix C.

D. Leapfrog Integrator

When choosing a numerical integrator to solve Hamilton's equations, a few considerations must be taken into account. To maintain detailed balance in HMC, our system must exhibit time-reversibility so that the proposal distribution remains symmetric. Under T-symmetry where all momenta are reversed $\mathbf{p} \to -\mathbf{p}$, the equations of motion are time-reversible, therefore the probability of generating any two points is symmetric $(\theta_0, p_0) \leftrightarrow (\theta_1, p_1)$.

In practice however, not all numerical solvers preserve time-reversibility, and we must take care which algorithm to use. One such class of solvers is *symplectic integrators*, which guarantee time-reversibility.

The leapfrog integrator is a symplectic integrator and proves an attractive candidate due to its simplicity. Given a step size ϵ and path length L, we can update the position θ and momentum p at each iteration t.

Algorithm 1 Leapfrog Integrator

for $0 < t < L$ **do**
  {First momentum half step}
  $\mathbf{p}_{t+1/2} \leftarrow \mathbf{p}_t - \frac{1}{2}\epsilon \nabla U(\theta_t)$
  $\theta_{t+1} \leftarrow \theta_t + \epsilon \mathbf{p}_{t+1/2}$
  $\mathbf{p}_{t+1} \leftarrow \mathbf{p}_{t+1/2} - \frac{1}{2}\epsilon \nabla U(\theta_{t+1})$
end for

While theoretically the Hamiltonian satisfies the principle of energy conservation, the use of the leapfrog integration introduces an energy drift and the total energy can change during the evolution.

To correct for this, a Metropolis step is performed to choose whether to accept or reject the sample in the Markov-Chain based on it's change in energy. The acceptance probability is given by

$$A(\theta'|\theta) = \min\left(1, \frac{\exp(-H(\theta_1, q_1))}{\exp(-H(\theta_0, q_0))}\right), \tag{17}$$

which maintains detailed balance of the target distribution [37].

E. HMC Parameters

The three main input parameters for HMC are the step size ϵ, path length L, and metric M. It is essential to properly tune these parameters for each distribution to ensure functioning sampling.

The step size ϵ affects the accuracy of numerical integration and therefore the change in energy between the initial and final point $\Delta H = H(\theta_1, q_1) - H(\theta_0, q_0)$. Choosing ϵ to be too large will result in high rejection rates during the correction step, while an ϵ too small will lead to longer integration time ($L\epsilon$), both wasting computational resources.

The path length L with ϵ defines the integration time $L\epsilon$ of the numerical integration. It is important for the integration time to be long enough such that the proposal point is sufficiently decorrelated from the initial point.

Considering the Hamiltonian from an energy perspective $H(E)$, the metric defines the distribution of energy level sets which HMC explores. The metric must be properly chosen such that the momenta drawn adequately explore the target distribution level sets [35].

F. Critical Slowing Down

For a well functioning MCMC algorithm, the sequence of samples in the Markov-Chain must be uncorrelated and effectively independent. Define the autocorrelation of an observable $\mathcal{O}$ is defined as

$$C_{\mathcal{O}}(\tau) = \langle \mathcal{O}(t)\mathcal{O}(t+\tau)\rangle, \tag{18}$$

where t denotes the discrete time of the Markov-Chain, and $\langle\cdot\rangle$ denotes the time average.

The autocorrelation time characterises how long it takes for the Markov-Chain to generate an independent sample, and is defined as

$$\tau_{\mathcal{O},\text{int}} = \frac{1}{C_{\mathcal{O}}(0)} \sum_{t=1}^{T} C_{\mathcal{O}}(t). \tag{19}$$

As the target distribution approaches a critical point, the autocorrelation time dramatically increases, a phenomena known as critical slowing down.

G. Sampling within an Iso-Likelihood Contour

Nested sampling requires that samples are generated subject to the hard likelihood constraint $\mathcal{L} > \mathcal{L}_i$ at each iteration. It is a challenging problem to draw points from the constrained distribution, with current approaches such as slice sampling [32] and rejection sampling [17] scaling poorly with dimension.

HMC samples from the unconstrained distribution $\pi(\theta)$. In order to leverage its use in nested sampling, it is therefore necessary to modify the algorithm to sample given a constraint.

V. CONSTRAINED HAMILTONIAN MONTE CARLO

Consider the constrained distribution subject to the hard likelihood constraint

$$\tilde{\pi}(\theta) = \begin{cases} \pi(\theta), & \mathcal{L}(\theta) > \mathcal{L}_0 \\ 0, & \text{otherwise} \end{cases} \tag{20}$$

Hamiltonian Monte Carlo can be modified to sample from a constrained prior, making it a viable technique for generating new live points in nested sampling. As described conceptually by Skilling [21] & Betancourt [20], reflecting off the iso-likelihood contours ensures that proposed samples are within the constrained distribution. When the next position θ_t in the trajectory would be below the likelihood constraint, we reflect the momentum off the boundary with normal $\mathbf{n} = -\nabla \log \mathcal{L}(\theta)$ as

$$\begin{aligned} \mathbf{n_R} &= M^{-1}\mathbf{n}, \\ \mathbf{p}' &= \mathbf{p} - 2\frac{\mathbf{p}\cdot\mathbf{n_R}}{\mathbf{n}\cdot\mathbf{n_R}}.\mathbf{n} \end{aligned} \tag{21}$$

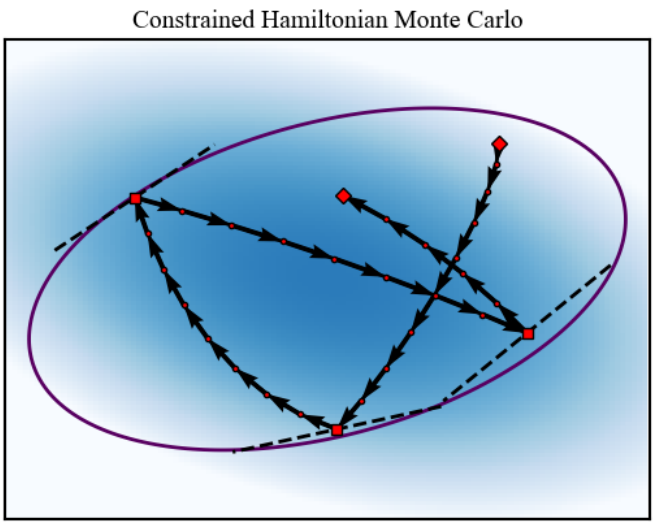

FIG. 2: Hamiltonian Monte Carlo modified to reflect off iso-likelihood boundary $\mathcal{L}_0$. Initial point with randomly sampled momentum (diamond) moving through constrained prior distribution, $\tilde{\pi}(\theta)$, shown in blue. Finite step size ϵ, shows discretisation of integrator with intermediate points (circles). Points which reflect off boundary (squares) do not fall exactly on the boundary due to discretization. After a fixed number of integration steps L, we propose a new sample from the constrained distribtion (diamond).

This sampling method can now be used for nested sampling, by using the point with the lowest likelihood at iteration i with likelihood $\mathcal{L}_i$ to seed the generation of the next sample subject to the constraint $\mathcal{L} > \mathcal{L}_i$.

Implementing this algorithm in practice poses new challenges with discretization, clustering, and parameter adaption. We propose a novel set of additional algorithms which are essential for a working implementation of Constrained HMC (CHMC).

A. Epsilon Halving

When performing reflections with a finite step size ϵ, there is no guarantee that after a reflection, the next position will be within the constrained boundary. While this scenario is uncommon for smooth boundaries, over many iterations it is likely that at one time, no valid reflection will be found, causing the algorithm to crash.

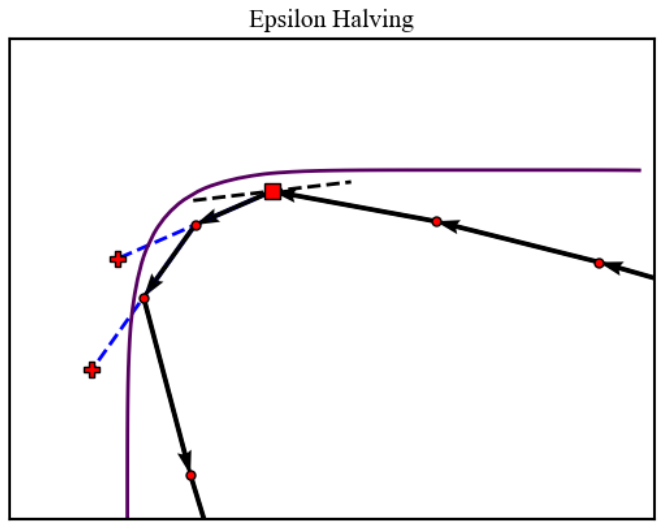

FIG. 3: For sharply curved boundaries, it is possible for the position after reflection (plus) to also lie outside the boundary constraint, leaving no valid reflections. In order for the algorithm to continue, a temporary step size is used to calculate the next position only. We iterate $\epsilon' = \epsilon/2$ until the next point lies within the constraint. After a position within the constraint is found, we return back to integrating with ϵ_0.

We propose a new reflection scheme which guarantees that a valid reflection will always be found, allowing the algorithm to continue. After a reflection, the next point is checked to see if it lies outside the boundary. If so, a temporary step size $\epsilon' = \epsilon/2$ is used to re-integrate the position of the next point. Epsilon is repeatedly halved until a position within the boundary is found. This is guaranteed to produce a valid position in the limit $\epsilon' \to 0$ as a continuous reflection is performed. The algorithm then continues with the original ϵ_0.

Algorithm 2 Epsilon Halving

```
{Momentum reflection}
p ← p − 2 (p · n̂) n̂

θ_new ← θ + ε_0 p
ε' ← ε_0

while L(θ_new) < L_0 do
    ε' ← ε'/2
    θ_new ← θ + ε' p
end while

θ ← θ_new
```

B. Ergodicity

Ergodicity and detailed balance are well known to be requirements of any MCMC algorithm [31]. In practice however, no algorithm is truly ergodic due to imperfections in computation such as floating-point errors and the cyclic nature of pseudo-random number generators. The question then becomes whether the algorithm is *sufficiently ergodic*. It is clear the process introduced by epsilon halving is not ergodic, therefore it is essential we ask whether our algorithm will maintain sufficient ergodicity.

Adaptive techniques in MCMC methods have been shown to still converge to the required distribution [38]. We argue that so long as our non-ergodic techniques are used infrequently, CHMC will be sufficiently ergodic to pass all statistical tests.

C. Clustering

Multi-modal posteriors pose a challenging problem for many MCMC sampling algorithms, with topological freezing causing significant complications [39]. In theory, nested sampling solves multi-modal distributions just as easily as uni-modal ones, however it is still important to consider the sampling method used to ensure it does not introduce a systematic bias.

For instance, if n_{live} is too low, it is possible for modes to 'die out' and be missed by nested sampling. One existing solution is to manually identify clusters of live points and treat them separately, as explored by PolyChord which uses a k-means algorithm for clustering [16].

We illustrate how CHMC naturally solves clustering in multimodal distributions. Once the likelihood constraint is increased to the point where the contour splits into isolated volumes, all live points in the separate modes evolve completely independently of each other as shown in Fig. 4.

D. Topological Traps

The natural clustering advantage of CHMC also brings rise a susceptibility to topological traps. Any live points in a local maxima will be forced to stay there, becoming further and further compressed until there is no more space new samples to be generated.

We propose a new mechanism for sampling through topological traps based on *reflection rate*. Define the reflection rate for live point i as

$$\mathcal{R}_i = \frac{r_i}{L}, \tag{22}$$

where r_i is the number of reflections in the CHMC evolution. The motivation is that as we repeatedly resample points in an ever more constrained local maximum, $\mathcal{R}$ will approach 1 for these live points as they run out of room.

We introduce a novel mechanism to move newly seeded live points from one mode into another, thereby allowing points to move from local to global maxima.

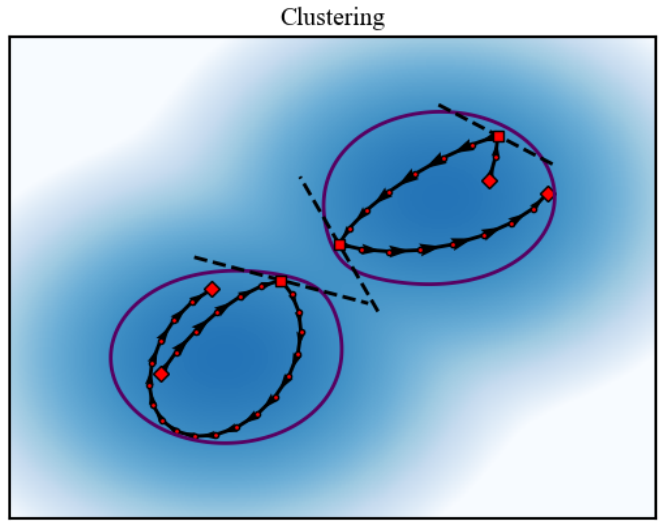

FIG. 4: Clustering in multi-modal distributions. When the iso-likelihood contour splits into two isolated modes, new samples generated are prevented from mixing with each other. The position θ is restricted to stay within the posterior region contained by the volume, with no mechanism to cross into the other mode.

Algorithm 3 Nested sampling through topological traps using reflection rate

$\mathcal{R}_0$, Reflection rate threshold
θ_r, Parameters for a random live point

θ_i, Parameters for live point with lowest likelihood at iteration i
if $\mathcal{R}(\theta_i) < \mathcal{R}_0$ **then**
  $\theta_{i+1} \leftarrow \text{CHMC}(\theta_i)$
else
  $\theta_{i+1} \leftarrow \text{CHMC}(\theta_r)$
end if
Move θ_i to dead points

If the lowest likelihood live point has a reflection rate above some threshold, $\mathcal{R}_0$, we use a random live point as the seed for the next live point, instead of the lowest likelihood live point. This ensures that given at least one live point is in the global maximum, eventually all the new live points will move into this mode where there is more 'space'.

VI. PARAMETER ADAPTION

The sampling performance of Constrained HMC is heavily dependent on a number of parameters which must be finely tuned in order to extract the full potential of the algorithm. The step size ϵ, path length L, and kinetic energy metric M are very important and their optimal value depends on the distribution being sampled [40]. Manually choosing fixed values requires significant domain knowledge and is prone to human error. Schemes have been developed to adaptively set these parameters for HMC, such as the No-U-Turn criterion for choos-

ing path lengths [41], and dual averaging for setting the step size [42].

Furthermore, in the case of CHMC for nested sampling it is in fact not possible to use a fixed value for ϵ and metric M, which maintain performance until termination.

In nested sampling, as the iso-likelihood boundary is contracted after each iteration, the posterior distribution is implicitly changed as it is sampled. Even if an optimal value for the step size is initially chosen, as the posterior space is compressed, the step size will unsuspectingly become too large and the integrator will lose efficiency until all new samples are rejected.

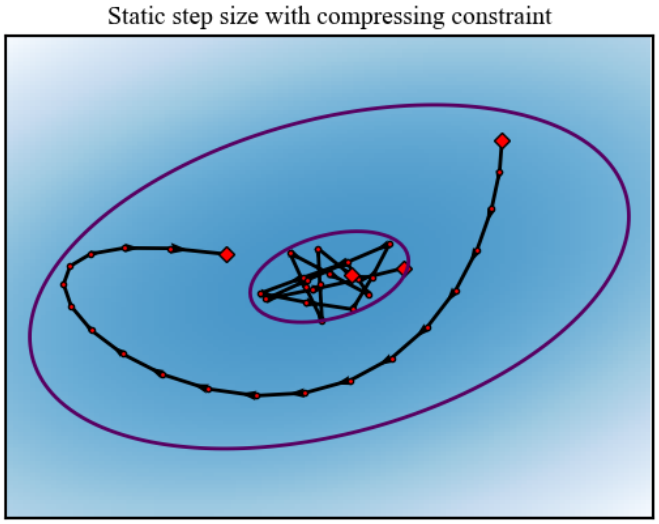

FIG. 5: Trajectories for static step size ϵ with different iso-likelihood contours. ϵ may initially begin well tuned to explore the posterior, but as nested sampling iterates, the space compresses and sampling becomes inefficient. Eventually, CHMC will no longer be able to generate new samples with the given step size.

A. Epsilon Dual Averaging

We can use the Markov-Chain samples themselves to adapt ϵ dynamically to give a target Metropolis acceptance rate, as discussed in section IV D. Employing the dual averaging scheme of Nesterov [42] has been shown to be well suited to MCMC samplers due the convexity of giving proportionate weight to *later* samples [41].

Given statistics H_t, we want to find a parameter x such that $\mathbb{E}_t[H_t|x] = 0$. We can apply the updates

$$\begin{aligned} x_{t+1} &\leftarrow \mu - \frac{1}{\gamma\sqrt{t}}\sum_{i=1}^{t} H_i, \\ \bar{x}_{t+1} &\leftarrow \eta_t x_{t+1} + (1-\eta_{t+1})\bar{x}_t, \end{aligned} \tag{23}$$

where μ is a user defined parameter that x_t is shrunk towards, γ is a free parameter that controls the amount of shrinkage to μ, and $\eta_t \equiv t^{-\kappa}$ is the shrinkage step size.

To apply this averaging scheme to CHMC we set $x = \log\epsilon$ and $H_t = \delta - P_t$, where P_t is the metropolis acceptance probability of sample t and $\delta \in (0, 1)$. The values for μ, κ, δ used for our implementations are listed in Appendix A.

Averaging in HMC is generally performed for a fixed number of steps during the burn-in phase, after which the tuned value of epsilon is chosen and kept constant. As illustrated above, this scheme is modified for CHMC and we continuously adapt ϵ accordingly until termination.

B. Kinetic Energy Adaption

The kinetic energy term in the Hamiltonian (16) is expressed as

$$T = \frac{1}{2}\mathbf{p}^T M^{-1}\mathbf{p}, \tag{24}$$

with metric M, where the momentum is drawn from the distribution $\mathbf{p} \sim \mathcal{N}(0, \sigma = M)$.

Rephrasing the Hamiltonian in terms of energy $H(E)$, the metric defines which energy level sets will be explored by the HMC Markov-Chain. A poorly chosen metric will very slowly explore the energy levels of the target distribution $\pi(E)$. Schemes to diagnose suboptimal metrics for HMC have been developed using Bayesian information arguments [35], and geometric approaches [43].

We propose a new algorithm to continuously adapt the metric in CHMC for nested sampling, using motivations from the *equipartition theorem*. The equipartition theorem states that in thermal equilibrium, the energy is shared equally amongst a system's degrees of freedom [44].

Considering the set of live points as a thermodynamic ensemble, we can therefore relate the *variance* of the kinetic energy and potential energy per dimension as

$$\mathrm{Var}[T] = \frac{1}{D}\mathrm{Var}[U]. \tag{25}$$

Following the derivation in Appendix D we arrive at the result for the metric

$$M = \frac{2}{D}\sqrt{\mathrm{Var}[U]}\mathbb{1}, \tag{26}$$

Where D is the dimensionality of the problem, and we use a scaled unit metric for simplicity.

In order to computationally estimate the potential energy variance during the CHMC algorithm, we use the set of live points as an ensemble sample of the distribution, $\mathrm{Var}[U] = \mathrm{Var}[\log\pi(\theta_i)]$.

Using this scheme, we can update our metric every $\mathcal{O}(n_{\mathrm{live}})$ iterations with very little computational overhead.

We compared this method for kinetic energy adaption to other suggested approaches such as a BFMI diagnostic [35].

We observe heuristically that our equipartition scheme performs better than BFMI, with lower computational cost, as we exploit the natural information contained by the set of live points.

VII. LATTICE FIELD THEORY

A. Lattice Field Theory for Markov-Chain Monte Carlo methods

The path integral formalism for quantum mechanics provides a powerful tool for studying relativistic quantum field theories [45]. It is written as a functional integral

$$\mathcal{Z} = \int \mathcal{D}\phi e^{iS[\phi]}, \tag{27}$$

where $\mathcal{D}\phi$ is a measure over all field configurations ϕ, and $e^{iS[\phi]}$ weights the contribution of each configuration. In the classical limit, all the 'unphysical' fields with very a large action cancel each other, and we reduce to Hamilton's least action principle.

The path integral approach provides a useful starting point for lattice field theory. We begin by discretizing space-time, such that the field $\phi(x)$ lives on a finite lattice rather than in a continuous space. However, this still leaves a sum over an infinite number of possible field configurations. In order to computationally measure observables of a field theory system, we must employ MCMC methods to sample a number of field configurations and estimate the path integral.

To interpret the field weight factor $e^{iS[\phi]}$ as a probability for a given configuration, we can perform a Wick-rotation by redefining time as imaginary [46].

$$\begin{aligned} t &\to i\tau, \\ \mathrm{e}^{iS[\phi]} &\to e^{-S_E[\phi]}, \end{aligned} \tag{28}$$

where from now on we only refer to the Euclidean action as S.

This factor is now a well-defined probability distribution as it is bounded and normalised $P[\phi] \equiv e^{-S[\phi]}/\mathcal{Z} > 0$. Therefore, it is possible to apply MCMC methods to sample from this distribution and calculate observables.

Given a proposal field ϕ' from initial field ϕ, we can sample using the Metropolis algorithm (11). The acceptance probability is given by

$$A(\phi'|\phi) = \min\left(1, \frac{\exp(-S[\phi'])}{\exp(-S[\phi])}\right), \tag{29}$$

as the desired probability distribution we are sampling from is $P[\phi]$.

B. Scalar ϕ^4 Theory

We work with a real, scalar ϕ^4-theory in two-dimensions, discretized to a Euclidean square lattice of length N. The dimensionless action for the theory is given by

$$\begin{aligned} S = \sum_{x\in\Lambda}\Bigg[-2\kappa \sum_{\mu=1}^{d}\phi(x)\phi(x+\hat{\mu}) \\ + \lambda\phi(x)^4 + (1-2\lambda)\phi(x)^2\Bigg], \end{aligned} \tag{30}$$

where Λ is the set of lattice points, $\hat{\mu}$ is the unit vector in μ direction, and $\phi(x)$ is the field value at lattice site x [47]. κ is the kinetic coupling for neighbour interactions and λ is the coupling strength of the interaction.

ϕ^4 theory lives in the same universality class as the 2D Ising model, and we fully recover the Ising model in the limit $\lambda \to \infty$. As such, the model exhibits a second-order phase transition in the κ & λ parameters, associated with the breaking of Z_2 symmetry.

The order parameter which undergoes a phase transition is the mean magnetization, defined as the normalised expectation of absolute field value

$$\langle M \rangle = \frac{1}{N^2}\Big\langle \sum_{x\in\Lambda} |\phi(x)| \Big\rangle. \tag{31}$$

C. Nested Sampling for ϕ^4 Theory

As described in section VII A, we sample from the distribution with probability $P[\phi] = e^{-S[\phi]}$. Therefore, defining the log likelihood function as $\log\mathcal{L} = -S[\phi]$ the problem is reframed in Bayesian inference terms, allowing us to apply the new algorithm we have introduced in section V.

The model of ϕ^4-theory is a good candidate to test CHMC for nested sampling for a number of reasons. Below the critical point, the loglikelihood is a bimodal distribution allowing us to investigate the clustering behaviour of the algorithm. The likelihood gradient is also analytic

$$\begin{aligned} \frac{\partial \log\mathcal{L}}{\partial\phi(x)} = -2\kappa \sum_{\mu}\phi(x+\hat{\mu}) \\ + 4\lambda\phi(x)^3 + 2(1-2\lambda)\phi(x) \end{aligned} \tag{32}$$

and well-behaved. Furthermore, this model exhibits *critical slowing down*, allowing us to test the algorithm's resilience to this phenomenon.

When using CHMC and nested sampling for ϕ^4 theory, we must define a prior distribution $\pi(\theta)$. To use CHMC, we already provide the likelihood function gradient for reflections. Therefore, we can set the prior to be the posterior by using the likelihood function as the prior $\pi(\theta) = \mathcal{L}(\theta)$. This removes the need for any new gradients and maximises the speed of convergence, as sampling directly from the posterior will be most efficient.

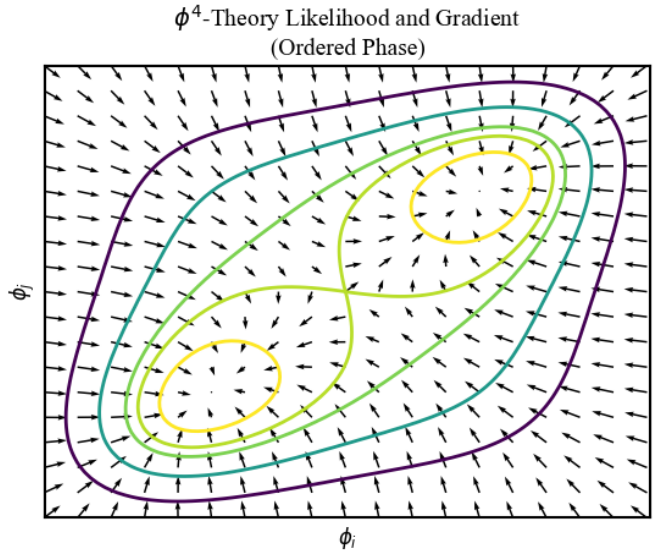

FIG. 6: In the ordered phase, neighbouring fields prefer aligning in the same direction as each other. This results in a bimodal distribution with a positive mean magnetisation $\langle M \rangle$.

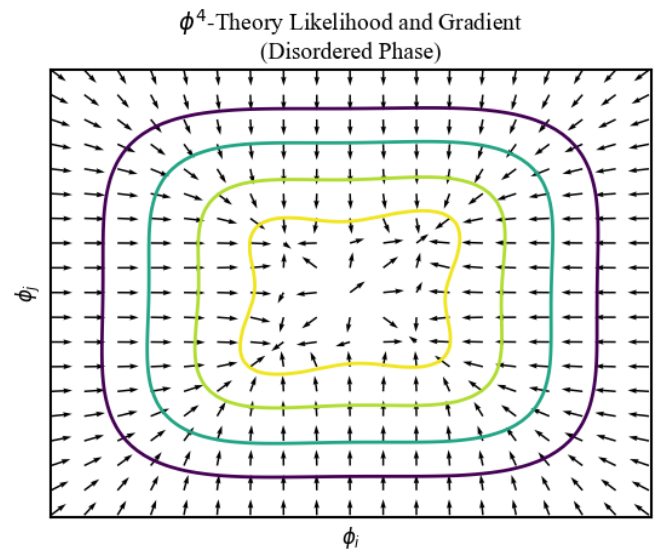

FIG. 7: In the disordered phase, neighbouring fields are uncorrelated. The likelihood is a unimodal distribution with zero mean magnetisation $\langle M \rangle$.

VIII. NUMERICAL RESULTS

The CHMC for nested sampling algorithm was implemented as an optimised C++ library in order to run large scale, high dimensional tests. We provide a novel application of nested sampling to ϕ^4-theory to perform experiments. The ϕ^4 action was given as the likelihood function as defined in Section VII C.

All input parameter values are listed in Appendix A and full details about the implementation can be found in Appendix B. All results were gathered using a single core on the CSD3 compute cluster.

A. Phase Transitions

To investigate the phase transition in ϕ^4-theory, we aim to measure the mean magnetisation (31) of the field for a wide range of κ & λ values.

We use a 32×32 lattice ($D = 1024$) and set $n_{\text{live}} = 500$ with precision criteria $p = 0.1$. For this setup, the average number of nested sampling iterations to converge is $i_{\text{max}} \approx 15,000$.

The mean magnetisation is calculated as the mean value of absolute field value. We show the distribution of $\phi(x)$ for an action in the disordered phase in Fig. 8.

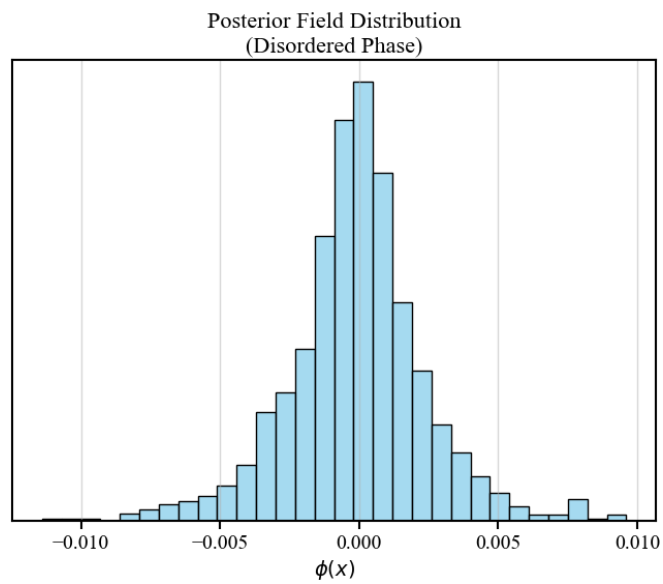

FIG. 8: The posterior distribution of $\phi(x)$ for an action in the disordered phase. For $\kappa = 0.1, \lambda = 0.02$, the distribution is unimodal with narrow variance. The magnetisation is the mean of the absolute field value, where $\langle M \rangle \approx 0$ for the above action.

We vary the action (30) across the range $\kappa \in (0, 0.3)$, $\lambda \in (0, 0.03)$ for a total of 1,000 unique actions, and calculate $\langle M \rangle$ for each one [48], to construct a phase diagram shown in Fig. 9.

This phase diagram accurately reconstructs the theoretical result and is verified against other simulations [7].

B. Correlation Functions

The two-point correlation functions are a key quantity related to physical observables of the quantum field theory. For example, the renormalised mass parameter and correlation length can both be measured with correlation functions [47, 49].

In the context of a Euclidean ϕ^4 theory, we define the spatial (equal-time) correlation functions as

$$C(x_1, x_2) = \langle \phi(x_1)\phi(x_2) \rangle, \tag{33}$$

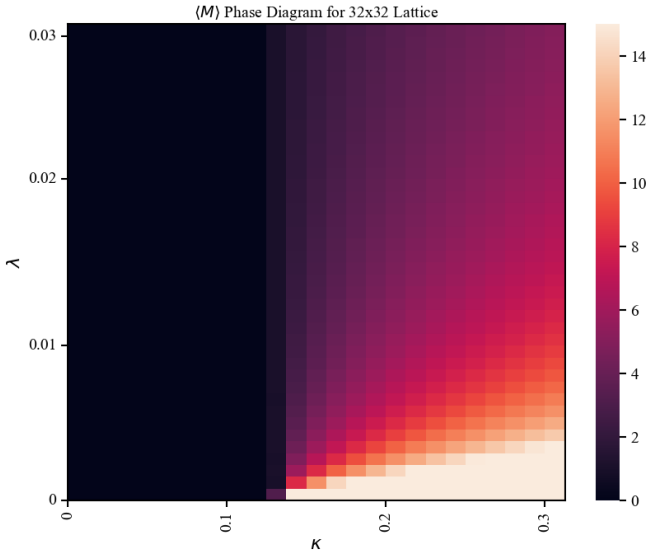

FIG. 9: κ vs λ phase diagram for 32x32 lattice ϕ^4-theory. For $\kappa \lesssim 0.125$, we see the expression of the disordered phase with $\langle M \rangle = 0$. The second-order phase transition is apparent along this boundary, as $\langle M \rangle$ grows suddenly across it.
In the ordered phase ($\kappa \gtrsim 0.125$), the mean magnetisation is strongly dependent on the field coupling λ. With a weak coupling, the kinetic energy dominates and the field seperates, leading to large $\langle M \rangle$. We also note that λ has a minor influence on the exact critical point in κ, with the critical line having a slight negative slope.

where $\langle . \rangle$ denotes the Markov-Chain ensemble average (12).

Exploiting the translational and rotational symmetry of the ϕ^4 action, we can fully characterise the two-point correlation function in terms of the distance between two points r

$$C(r) = \langle \phi(x)\phi(x+r) \rangle, \tag{34}$$

where we now also average over all lattice points x.

Directly calculating correlation functions numerically is prohibitively expensive, requiring $\mathcal{O}(N^3)$ operations for each microstate. We use a method of fourier transform and convolution theorem to speed up calculation [50].

In the continuous limit $N \to \infty$ we can write the correlation function for a single microstate at Markov-Chain iteration i as

$$C_i(r) = \int_{-\infty}^{\infty} \phi(x)\phi(x+r)dx. \tag{35}$$

Applying the convolution theorem gives

$$C_i(r) = \mathcal{F}^{-1}\{|\tilde{\phi}(k)|^2\} \tag{36}$$

providing a new way to evaluate the correlation function in $\mathcal{O}(N^2 \log N)$ using fast fourier transform.

Below the critical point, the correlation function decays exponentially as $C(r) \sim \exp(-r/\xi)$, where ξ is defined as the correlation length. As we approach the critical point, the correlation length diverges and becomes infinite in the ordered phase.

We measure the correlation functions on a 128×128 lattice ($D = 16,384$). Using $n_{\text{live}} = 1000$ with fixed $\lambda = 0.03$, we show the measured correlation functions for different κ in Fig. 10.

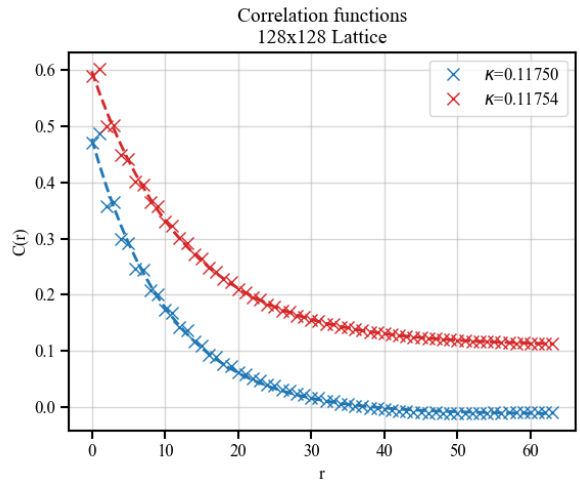

FIG. 10: Correlation functions for 128×128 lattice in ϕ^4-theory for ordered phase (red) and disordered phase (blue). The fitted exponentials show the expected decay characteristic for both correlation functions. In the ordered phase, the correlation function does not decay to 0, but instead shows there is a mean magnetisation throughout the field.

We can estimate the correlation length ξ for each value of κ, and plot the results in Fig. 11 to visualise the phase transition as a divergence in ξ.

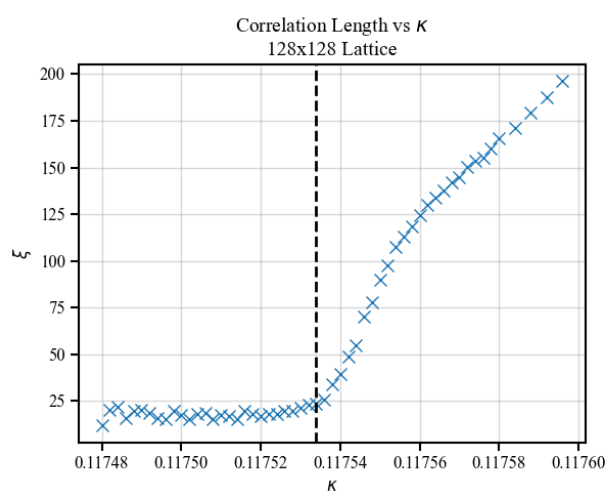

FIG. 11: Correlation length (ξ) vs κ shows the second-order phase transition as the correlation length diverges. In the disordered phase the correlation length is 0, then begins to increase until it exceeds the size of the physical lattice, which marks the ordered phase.

We also calculate the correlation function for a much larger lattice of size 512×512 ($D = 262,144$), with $n_{\text{live}} = 500$,

shown in Fig. 12. The phase transition for such a large lattice is very sharp, with the change between ordered and disordered phase occurring over $\Delta\kappa \approx 2 \times 10^{-6}$, which our algorithm correctly identifies.

For the correlation functions in Fig. 12, there is unexpected cyclic behaviour in the decay. We are not sure as to the cause of this phenomenon and further work is required to full understand it.

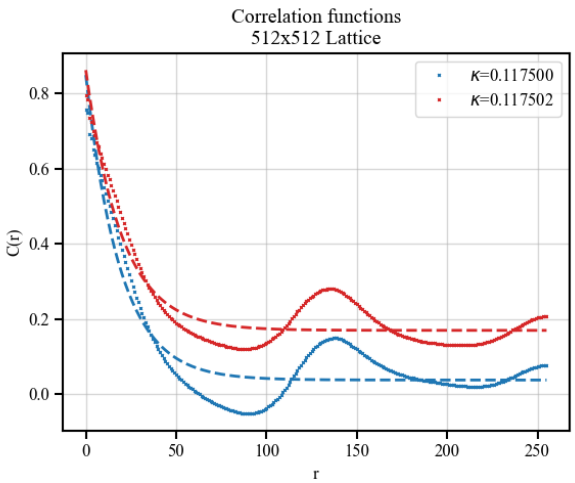

FIG. 12: Correlation functions for 512×512 lattice for ordered phase (red) and disordered phase (blue). Note the phase transition is very sharp, occuring over a range $\Delta\kappa \approx 2\times10^{-6}$. The correlation functions exhibit cyclic behaviour in decay, which is unexpected and the cause is not well understood. The calculation of these results demonstrate it is possible to perform nested sampling in dimensions much larger ($D = 262,144$) than explored previously.

The numerical results clearly demonstrate that Constrained HMC with nested sampling is a viable tool for lattice field theory. The algorithm successfully samples and estimates observables accurately at the critical point, without any modifications or difficulty. This confirms that leveraging nested sampling for lattice field theory provides strong resistance to topological freezing.

IX. PERFORMANCE

We aim to showcase the performance strength of our algorithm as a high dimensional sampler by comparing to existing solutions such as PolyChord [18]. It is important to note that other samplers do not require the use of gradients unlike CHMC, and can be used for likelihoods where no analytic gradient exists, or auto differentiation (C) fails.

A significant advantage of CHMC is that the number of likelihood & gradient evaluations to generate a sample is completely independent of dimension. It is defined by the path length alone such that number of calls is $N_{\mathcal{L},\nabla} = L$ per iteration.

Therefore, all the added computational expense comes from longer likelihood evaluation times for more parameters. We demonstrate this fact by measuring the average time per sample over a nested sampling run, for a range of dimensions up to $D \approx 600,000$ in Fig. 13.

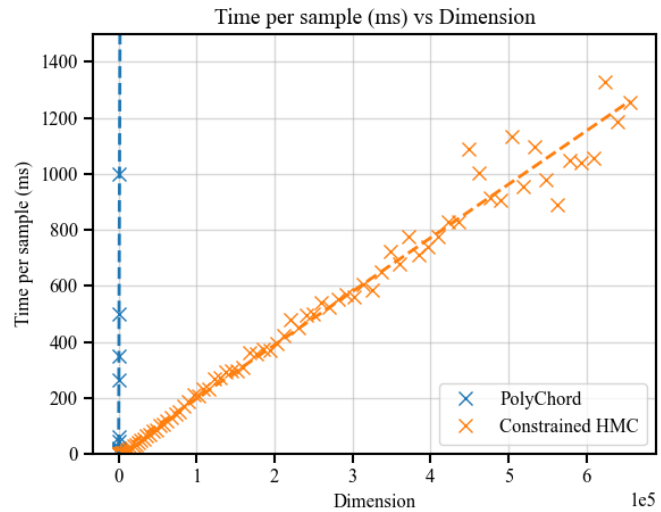

FIG. 13: Average time to generate sample vs dimension for a ϕ^4-theory likelihood. We compare PolyChord nested sampling versus our Constrained HMC algorithm. Our algorithm scales significantly better with dimension, and can generate a sample every second up to $D \approx 5 \times 10^5$. PolyChord exceeds one second per sample after $D \approx 1,000$.

We also compare the speed per sample using identical setups between PolyChord and our algorithm, shown in Fig. 13. For high-dimensional problems, it is clear that the added complexity of using gradients for CHMC is well worth the performance benefits.

X. CONCLUSION

We have now introduced our new algorithm and implementation for Constrained Hamiltonian Monte Carlo for nested sampling. By drawing concepts from classical mechanics together with Bayesian inference we provide a powerful set of tools which are very effectively applied to lattice field theory.

A summary of our novel algorithms and results include

- Constrained Hamiltonian Monte Carlo
  - Epsilon halving: a new algorithm for guaranteeing a valid reflection in discretized solvers.
  - Reflection rate hopping: a new mechanism for moving dead points from local to global maxima.
  - Epsilon adaption: a dual averaging scheme for continuously adapting the step size throughout evolution.
  - Metric adaption: A new method to adapt the kinetic energy metric based off the equipartition theorem, outperforming existing adaption schemes

such as BFMI.

- First application of nested sampling to ϕ^4 theory to calculate a numerically verified phase diagram.
- Accurate calculation of magnetisation and correlation functions for 512×512 sized lattices with nested sampling.
- Demonstrated computationally viable nested sampling for likelihoods up to dimensions $D = 500,000$.

Further work can be done to investigate the cause of cyclic decay in correlation functions in Fig. 12, whether it be due to boundary conditions, or computational methods. The C++ implementation would greatly benefit from parallelisation, allowing it to take full advantage of modern compute clusters. Using auto differentiation will also require further investigation to ensure no pathological behaviour arises for complex likelihoods.

By exploiting gradients we believe it is possible to use this algorithm for a new class of Bayesian problems previously inaccessible to machines.

ACKNOWLEDGEMENTS

Except where specific reference is made to the work of others, this work is original and has not been already submitted either wholly or in part to satisfy any degree requirement at this or any other university.

I would like to acknowledge my supervisors, Dr. Will Barker and Dr. David Yallup, who have provided invaluable guidance and support throughout my foray into physics research.

I would also like to acknowledge Dr. Will Handley for his helpful insights and for staying to answer my questions after his lectures.

I would like to thank Dr. Tyson Jones, who first hooked me to the beautiful pursuit of physics many years ago.

Finally, I would like to thank my housemates, who all also study physics. They have proved invaluable friends who are always there to make you laugh while debugging code together at midnight.

[1] S. Borsanyi, Z. Fodor, J. Guenther, C. Hoelbling, S. Katz, L. Lellouch, T. Lippert, K. Miura, L. Parato, K. Szabo, *et al.*, Nature **593**, 51 (2021).
[2] S. Duane, A. Kennedy, B. J. Pendleton, and D. Roweth, Physics Letters B **195**, 216 (1987).
[3] M. Girolami and B. Calderhead, Journal of the Royal Statistical Society: Series B (Statistical Methodology) **73**, 123 (2011).
[4] A. Kramer, B. Calderhead, and N. Radde, BMC bioinformatics **15**, 1 (2014).
[5] U. Wolff, Nuclear Physics B - Proceedings Supplements **17**, 93 (1990).
[6] M. Hasenbusch, EPJ Web of Conferences **175**, 02004 (2018).
[7] J. M. Pawlowski and J. M. Urban, Machine Learning: Science and Technology **1**, 045011 (2020).
[8] K. Jansen, E. H. Müller, and R. Scheichl, Physical Review D **102** (2020), 10.1103/physrevd.102.114512.
[9] M. Albergo, G. Kanwar, and P. Shanahan, Physical Review D **100** (2019), 10.1103/physrevd.100.034515.
[10] D. C. Hackett, C.-C. Hsieh, M. S. Albergo, D. Boyda, J.-W. Chen, K.-F. Chen, K. Cranmer, G. Kanwar, and P. E. Shanahan, (2021), arXiv:2107.00734 [hep-lat].
[11] R. Abbott *et al.*, PoS **LATTICE2022**, 036 (2023), arXiv:2208.03832 [hep-lat].
[12] M. S. Albergo, D. Boyda, K. Cranmer, D. C. Hackett, G. Kanwar, S. Racanière, D. J. Rezende, F. Romero-López, P. E. Shanahan, and J. M. Urban, Phys. Rev. D **106**, 014514 (2022), arXiv:2202.11712 [hep-lat].
[13] X. Gao and L.-M. Duan, Nature communications **8**, 662 (2017).
[14] R. van de Schoot, S. Depaoli, R. King, B. Kramer, K. Märtens, M. G. Tadesse, M. Vannucci, A. Gelman, D. Veen, J. Willemsen, *et al.*, Nature Reviews Methods Primers **1**, 1 (2021).
[15] J. Skilling, Bayesian Analysis **1**, 833 (2006).
[16] W. J. Handley, M. P. Hobson, and A. N. Lasenby, Monthly Notices of the Royal Astronomical Society **453**, 4385 (2015).
[17] F. Feroz, M. P. Hobson, and M. Bridges, Monthly Notices of the Royal Astronomical Society **398**, 1601 (2009).
[18] W. J. Handley, M. P. Hobson, and A. N. Lasenby, Monthly Notices of the Royal Astronomical Society: Letters **450**, L61 (2015).
[19] G. Ashton, N. Bernstein, J. Buchner, X. Chen, G. Csányi, A. Fowlie, F. Feroz, M. Griffiths, W. Handley, M. Habeck, E. Higson, M. Hobson, A. Lasenby, D. Parkinson, L. B. Pártay, M. Pitkin, D. Schneider, J. S. Speagle, L. South, J. Veitch, P. Wacker, D. J. Wales, and D. Yallup, Nature Reviews Methods Primers **2** (2022), 10.1038/s43586-022-00121-x.
[20] M. Betancourt, AIP Conference Proceedings **1305**, 165 (2011), https://pubs.aip.org/aip/acp/article-pdf/1305/1/165/11567522/165_1_online.pdf.
[21] J. Skilling, AIP Conference Proceedings **1443**, 145 (2012), https://aip.scitation.org/doi/pdf/10.1063/1.3703630.
[22] and N. Aghanim, Y. Akrami, M. Ashdown, J. Aumont, C. Baccigalupi, M. Ballardini, A. J. Banday, R. B. Barreiro, N. Bartolo, S. Basak, R. Battye, K. Benabed, J.-P. Bernard, M. Bersanelli, P. Bielewicz, J. J. Bock, J. R. Bond, J. Borrill, F. R. Bouchet, F. Boulanger, M. Bucher, C. Burigana, R. C. Butler, E. Calabrese, J.-F. Cardoso, J. Carron, A. Challinor, H. C. Chiang, J. Chluba, L. P. L. Colombo, C. Combet, D. Contreras, B. P. Crill, F. Cuttaia, P. de Bernardis, G. de Zotti, J. Delabrouille, J.-M. Delouis, E. D. Valentino, J. M. Diego, O. Doré , M. Douspis, A. Ducout, X. Dupac, S. Dusini, G. Efstathiou, F. Elsner, T. A. Enßlin, H. K. Eriksen, Y. Fantaye, M. Farhang, J. Fergusson, R. Fernandez-Cobos, F. Finelli, F. Forastieri, M. Frailis, A. A. Fraisse, E. Franceschi, A. Frolov, S. Galeotta, S. Galli, K. Ganga, R. T. Génova-Santos, M. Gerbino, T. Ghosh, J. González-Nuevo, K. M. Górski, S. Gratton, A. Gruppuso, J. E. Gudmundsson, J. Hamann, W. Handley, F. K. Hansen, D. Herranz, S. R. Hildebrandt, E. Hivon, Z. Huang, A. H. Jaffe, W. C. Jones, A. Karakci, E. Keihänen, R. Keskitalo, K. Kiiveri, J. Kim, T. S. Kisner, L. Knox, N. Krachmalnicoff, M. Kunz, H. Kurki-Suonio, G. Lagache, J.-M. Lamarre, A. Lasenby, M. Lattanzi,

C. R. Lawrence, M. L. Jeune, P. Lemos, J. Lesgourgues, F. Levrier, A. Lewis, M. Liguori, P. B. Lilje, M. Lilley, V. Lindholm, M. López-Caniego, P. M. Lubin, Y.-Z. Ma, J. F. Macías-Pérez, G. Maggio, D. Maino, N. Mandolesi, A. Mangilli, A. Marcos-Caballero, M. Maris, P. G. Martin, M. Martinelli, E. Martínez-González, S. Matarrese, N. Mauri, J. D. McEwen, P. R. Meinhold, A. Melchiorri, A. Mennella, M. Migliaccio, M. Millea, S. Mitra, M.-A. Miville-Deschênes, D. Molinari, L. Montier, G. Morgante, A. Moss, P. Natoli, H. U. Nørgaard-Nielsen, L. Pagano, D. Paoletti, B. Partridge, G. Patanchon, H. V. Peiris, F. Perrotta, V. Pettorino, F. Piacentini, L. Polastri, G. Polenta, J.-L. Puget, J. P. Rachen, M. Reinecke, M. Remazeilles, A. Renzi, G. Rocha, C. Rosset, G. Roudier, J. A. Rubiño-Martín, B. Ruiz-Granados, L. Salvati, M. Sandri, M. Savelainen, D. Scott, E. P. S. Shellard, C. Sirignano, G. Sirri, L. D. Spencer, R. Sunyaev, A.-S. Suur-Uski, J. A. Tauber, D. Tavagnacco, M. Tenti, L. Toffolatti, M. Tomasi, T. Trombetti, L. Valenziano, J. Valiviita, B. V. Tent, L. Vibert, P. Vielva, F. Villa, N. Vittorio, B. D. Wandelt, I. K. Wehus, M. White, S. D. M. White, A. Zacchei, and A. Zonca, Astronomy &amp Astrophysics **641**, A6 (2020).

[23] P. Mukherjee, D. Parkinson, and A. R. Liddle, The Astrophysical Journal **638**, L51 (2006).

[24] W. Moses and V. Churavy, in *Advances in Neural Information Processing Systems*, Vol. 33, edited by H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin (Curran Associates, Inc., 2020) pp. 12472–12485.

[25] W. S. Moses, V. Churavy, L. Paehler, J. Hückelheim, S. H. K. Narayanan, M. Schanen, and J. Doerfert, in *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, SC '21 (Association for Computing Machinery, New York, NY, USA, 2021).

[26] W. S. Moses, S. H. K. Narayanan, L. Paehler, V. Churavy, M. Schanen, J. Hückelheim, J. Doerfert, and P. Hovland, in *Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis*, SC '22 (IEEE Press, 2022).

[27] C. M. Bishop, *Pattern Recognition and Machine Learning* (Springer, New York, 2006).

[28] D. J. MacKay, *Information Theory, Inference, and Learning Algorithms* (Cambridge University Press, Cambridge, 2003).

[29] A. Gupta and J. B. Rawlings, AIChE Journal **60** (2014), 10.1002/aic.14409.

[30] P. Del Moral, *Mean Field Simulation for Monte Carlo Integration* (Chapman Hall/CRC Press, 2013).

[31] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, J. Chem. Phys. **21**, 1087 (1953).

[32] R. M. Neal, Annals of Statistics **31** (2003), 10.1214/aos/1056562461.

[33] R. M. Neal, in *Bayesian Learning for Neural Networks* (Springer, 1996).

[34] B. J. Brewer, L. B. Pártay, and G. Csányi, "Diffusive nested sampling," (2010), arXiv:0912.2380 [stat.CO].

[35] M. Betancourt, "Diagnosing suboptimal cotangent disintegrations in hamiltonian monte carlo," (2016), arXiv:1604.00695 [stat.ME].

[36] B. Carpenter, M. D. Hoffman, M. Brubaker, D. Lee, P. Li, and M. Betancourt, "The stan math library: Reverse-mode automatic differentiation in c++," (2015), arXiv:1509.07164 [cs.MS].

[37] M. Betancourt, "A conceptual introduction to hamiltonian monte carlo," (2018), arXiv:1701.02434 [stat.ME].

[38] C. Andrieu and É. Moulines, The Annals of Applied Probability **16**, 1462 (2006).

[39] O. Mangoubi, N. S. Pillai, and A. Smith, "Does hamiltonian monte carlo mix faster than a random walk on multimodal densities?" (2018), arXiv:1808.03230 [math.PR].

[40] R. M. Neal, (2012), arXiv:1206.1901.

[41] M. D. Hoffman and A. Gelman, "The no-u-turn sampler: Adaptively setting path lengths in hamiltonian monte carlo," (2011), arXiv:1111.4246 [stat.CO].

[42] Y. Nesterov, (2007), 10.1007/s10107-007-0149-x.

[43] B. Bales, A. Pourzanjani, A. Vehtari, and L. Petzold, "Selecting the metric in hamiltonian monte carlo," (2019), arXiv:1905.11916 [stat.CO].

[44] L. D. Landau and E. M. Lifshitz, *Theoretical Physics, Volume I: Mechanics* (Nauka, Moscow, 1972).

[45] R. P. Feynman, Reviews of Modern Physics **20**, 367 (1948).

[46] H. J. Rothe, *Lattice Gauge Theories: An Introduction* (World Scientific, Singapore, 2005).

[47] A. Maas, "Lattice gauge theory: From basics to advanced concepts," (2020), lecture series held at the University of Graz, Austria.

[48] W. Handley, The Journal of Open Source Software **4**, 1414 (2019).

[49] R. C. Brower, M. Cheng, E. S. Weinberg, G. T. Fleming, A. D. Gasbarro, T. G. Raben, and C.-I. Tan, Physical Review D **98**, 014502 (2018).

[50] C. Ruge, P. Zhu, and F. Wagner, (1994), arXiv:hep-lat/9403009.

[51] C. R. Keeton, Monthly Notices of the Royal Astronomical Society **414**, 1418 (2011).

Appendix A: Constrained HMC Parameter Values

The below table shows the values used for all input parameters into CHMC as defined in the paper.

Symbol	Description	Value
ϵ_0	Initial step size	0.1
L	Path Length	100
δ	Target Acceptance Rate	0.8
γ	Adaption scaling	0.05
κ	Adaption shrinkage	0.75
μ	Asymptotic Target	-1

These parameters were selected based on those used in the paper [41] and manual tweaking.

Appendix B: Code Implementation

The code was implemented in C++ and optimised for speed and modularity. We use an object-oriented interface design based on *dependency injection*. This allows the code base to be easily compartmentalised and extended by other authors, while also keeping an easy plug-in design for any likelihood function.

The EIGEN3 library was used for heavy all calculations, which is an optimised scientific linear algebra library.

To ensure correctness, we implemented a set of unit tests using the GOOGLETEST framework to rigorously check the functionality of each class.

The full C++ code can be found at `https://github.com/BorisDeletic/CHMC-Nested-Sampling`.

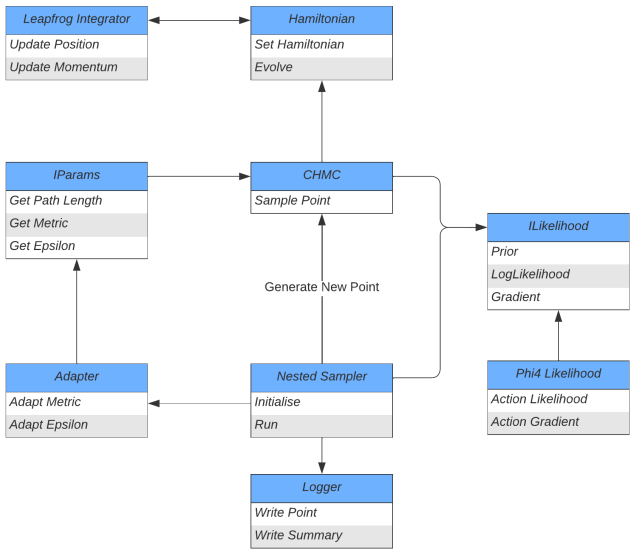

FIG. 14: UML Diagram illustrating the relationship between each objects. Each class was implemented as an independent code base hiding implementation details, then injected into each other to expose the desired functions. An interface design is used for the likelihood class, so the program does not require any specific knowledge of the likelihood. This allows one to simply implement the desired likelihood, such as ϕ^4 theory, without making any additional modifications.

Appendix C: Auto Differentiation

Auto differentiation is a modern approach to numerically calculating gradients with no numerical error [36]. The essential idea is that a compiler can break down a program to its base operations and automatically apply the chain-rule repeatedly to calculate its gradient. Recent advancements in machine learning have greatly increased the demand for auto differentiation tools and now highly optimised frameworks exist to calculate gradients for any code base.

One such example is ENZYME [24–26], a tool that takes arbitrary existing code base in a variety of languages (including C++) and computes the gradient of that function. We demonstrate this capability on the Rosenbrock function which allows CHMC to automatically run without needing to specify any gradients.

The D-dimensional Rosenbrock function is defined as

$$f(\theta) = AD + \sum_{i=0}^{D} \left(\theta_i^2 - A\cos(2\pi\theta_i)\right) \tag{C1}$$

In code block below, we show how to use ENZYME to automatically calculate the gradient of the Rosenbrock function.

```
//Rosenbrock Likelihood
double Likelihood::likelihood(double* theta, int size, double A) {
    double f = A * size;
    for (int i = 0; i < size; i++) {
        f += pow(theta[i], 2) - A * cos(2 * M_PI * theta[i]);
    }
    return f;
}

\\ Rosenbrock Autogradient
int enzyme_dup, enzyme_const;
extern double __enzyme_autodiff(...);
double Likelihood::gradient(double* theta, double* d_theta, int size, double A) {
    return __enzyme_autodiff(likelihood,
                             enzyme_dup, theta, d_theta,
                             enzyme_const, size, A);
}
```

The result in Fig. 15 shows the function contours and auto gradient, demonstrating how auto differentiation is a viable solution for CHMC with no numerical rounding error.

Appendix D: Metric Adaption with the Equipartition Theorem

Consider a D dimensional Hamiltonian with kinetic and potential energy

$$H = \frac{1}{2}\mathbf{p}^T M^{-1}\mathbf{p} + U, \tag{D1}$$

with momentum distributed as $\mathbf{p} \sim \mathcal{N}(0, \sigma = M)$. For simplicity, we use a scaled unit metric $M = \alpha\mathbb{1}$. According to

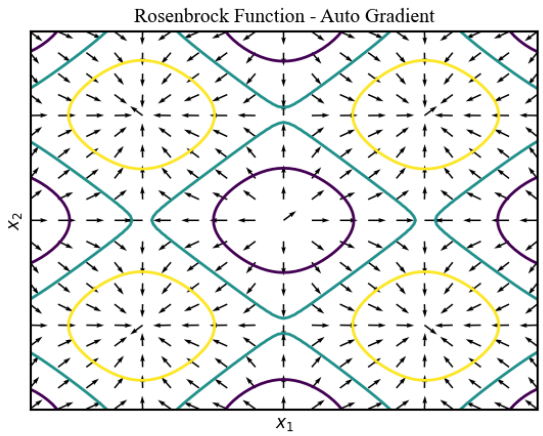

FIG. 15: Rosenbrock function (C1) and automatic differentiation gradient field plot. We demonstrate the validity of auto differentiation as a viable solution for any CHMC likelihood function.

the equipartition theorem, the energy is equally shared among the degrees of freedom. Therefore, we match the variance in potential energy per dimension to the kinetic energy.

$$\begin{aligned}
\frac{1}{D}\mathrm{Var}[U] &= \mathrm{Var}[T] \\
\mathrm{Var}[U] &= D\mathrm{Var}\left[\frac{1}{2}\mathbf{p}^T M^{-1}\mathbf{p}\right] \\
\mathrm{Var}[U] &= \frac{D}{4\alpha^2}\mathrm{Var}\left[\mathbf{p}^T\mathbf{p}\right] \\
\mathrm{Var}[U] &= \frac{D}{4\alpha^2}\mathrm{Var}\left[\sum_i^D \mathbf{p}_i^2\right] \\
\mathrm{Var}[U] &= \frac{D^2}{4\alpha^2}\alpha^4 \\
\mathrm{Var}[U] &= \frac{D^2\alpha^2}{4} \\
\alpha &= \frac{2}{D}\sqrt{\mathrm{Var}[U]}
\end{aligned} \tag{D2}$$

Which gives the result for the metric

$$M = \frac{2}{D}\sqrt{\mathrm{Var}[U]}\mathbb{1}, \tag{D3}$$

Appendix E: Nested Sampling Parameters

1. Termination criteria

The stopping criteria for nested sampling as suggested by Handley et al. [18], is determined by the remaining evidence in the live points. We terminate the algorithm once the evidence in the live points is a small fraction of the total accumulated evidence, determined by the precision criterion $\mathcal{Z}_{\mathrm{live}}/\mathcal{Z}$.

The remaining evidence can be estimated by assuming all the live points have the same prior volume at iteration i

$$\mathcal{Z}_{\mathrm{live}} \approx \langle\mathcal{L}\rangle_{\mathrm{live}} X_i. \tag{E1}$$

As this approximation will overestimate the live evidence, this will generally not cause early stopping of the algorithm. Further discussion of this can be found in Feroz [17], Handley [18], and Keeton [51].

2. Choosing n_{live}

The input parameter for the number of live points n_{live}, has a significant impact on the quality of results generated. More live points will increase the precision of the evidence calculation (8) and force more iterations to contract onto the posterior. However, more computational time will be required to complete the algorithm.

The remaining evidence $\mathcal{Z}_{\mathrm{live}}$ can be used to calculate the error in final evidence [18, 51] which will also be influenced by n_{live}. Therefore, a balance for choosing n_{live} must be found between accuracy and computational resources.

Furthermore, we argue that the required number of live points is dependent on the topology of the posterior. For unimodal geometries, all live points are guaranteed to be in the global likelihood maximum. However, for topologies with multi-modal distributions and topological traps in high dimensions, it is possible for live points to miss features of the posterior. As we discuss in Section V D, for certain geometries it is important to set n_{live} sufficiently high such that the global features are not completely missed.

For certain well-behaved likelihoods, it is not even necessary to scale $n_{\mathrm{live}} \sim D$ as suggested by Handley et al. [18], as shown in Section VIII where the results given have $D \gg n_{\mathrm{live}}$.

Further discussion of the effect of parameters on evidence calculation and the associated errors can be found in Handley et al. [18, 19].