

# A Multi-Trajectory Monte Carlo Sampler

Xiaopeng Xu<sup>1</sup>, Chuancai Liu<sup>1</sup>, Hongji Yang<sup>2\*</sup>, Xiaochun Zhang<sup>3</sup>

<sup>1</sup> School of Computer Science and Engineering, Nanjing University of Science and Technology, China

<sup>2</sup> School of Computing and Mathematical Sciences, University of Leicester, U.K.

<sup>3</sup> School of Management Science and Engineering, Anhui University of Finance and Economics, China  
xiaopeng.xu@aufe.edu.cn, chuancailiu@njust.edu.cn, Hongji.Yang@Leicester.ac.uk, xiaochun.zhang@aufe.edu.cn

## Abstract

Markov Chain Monte Carlo techniques based on Hamiltonian dynamics can sample the first or last principal components of multivariate probability models using simulated trajectories. However, when components' scales span orders of magnitude, these approaches may be unable of accessing all components adequately. While it is possible to reconcile the first and last components by alternating between two different types of trajectories, the sampling of intermediate components may be imprecise. In this paper, a function generalizing the kinetic energies of Hamiltonian Monte Carlo and Riemannian Manifold Hamiltonian Monte Carlo is proposed, and it is found that the methods based on a specific form of the function can more accurately sample normal distributions. Additionally, the multi-particle algorithm's reasoning is given after a review of some statistical ideas.

**Keywords:** Hamiltonian dynamics, Kinetic energy, Multi-particle system, Positive definite, Hessian

## 1 Introduction

It is well established that MCMC's spatial random walk is inefficient for high-dimensional models of significant non-normality and correlation [1-2]. Proposal distributions present another issue. Hamiltonian Monte Carlo (HMC) is a flexible solutions for multivariate probability models [3-4]. Its random walk is performed on simulation trajectories, overcoming the restrictions of Gibbs sampling's component-wise operation. As a result, the approach is theoretically applicable to any multivariate probability models of continuous variables.

For HMC, both accuracy and stability may be enhanced. One characteristic is inherent instability as a result of the violation of conservation of energy, which arises primarily from the irreconcilable conflict between the stability and exploration in the single-particle systems. The method is meant to prioritize exploration above stability. Its theoretical probability of acceptance is fixed at one, resulting in practically uncontrollable sampling operations. Currently, the majority of HMC algorithms are based on single-particle systems, which are unstable, and sophisticated approaches for tuning run-time parameters are used [5-6].

If the total energy of a single-particle system is conserved, the mutual proposal (jump) probability between two terminals

of a trajectory are equal [7]. Due to the fact that limiting the total energy of single-particle systems will limit exploration significantly, multi-particle systems with collision effect can be used to reconcile exploration and stability. As with single-particle systems, the intuitive notion is that multi-particle systems with collision effects will retain equal mutual jump probabilities between two terminals of the trajectory, which is appropriate for the Metropolis algorithm.

The multi-particle system with collisions is an ideal statistical mechanics model. On the assumption that momenta have equiprobable directions, it is possible to deduce that their magnitudes follow the Maxwell-Boltzmann distribution. Indeed, the assumption is true for a broader class of models, namely closed multi-particle systems with non-physical kinetic and potential energies. The model can serve as a foundation for developing effective and resilient MCMC samplers.

One issue occurs when the correlation coefficients of multivariate normal approaches 1 or -1. Similarly, variables can be spread across the manifold (for example, the surface) of a high-dimensional space. The models have proven to be intractable for classical samplers. To address these issues, the Riemannian Manifold HMC (RMHMC) was suggested with non-physical kinetic energies based on the Hessian of potential energies [8].

The Hessian matrices of potential energies are positive-definite for multivariate normal distributions, and stable simulation trajectories can be formed using physical kinetic energy. In general, the Hessian of potential energies can vary continuously and can be positive, negative, or indefinite, depending on the current locations. This presents a difficulty for simulating algorithms. Because the Hessian of potential energies is not always positive definite, efforts have been made to convert the matrices to positive definite forms [9-10]. Additionally, the convoluted of potential surfaces may trap sampling on a local scale [11]. Contrary to popular belief, a non-positive definite Hessian of potential energies does not necessarily result in divergence. For negative quadratic functions, for example, simulation using physical kinetic energy cannot produce stable trajectories; however, stable simulation using non-physical kinetic energies is possible.

Numerous existing difficulties are found to be related to kinetic energies, and the kinetic energies that are suitable for various conditions all belong to the same family, with the kinetic energy RMHMC (as well as HMC) being a specific case. The Hessian of potential energies can be used to define kinetic energies that can yield trajectories suited for a wide

variety of situations. As a result, parameter estimation for general probability density functions is possible. For approximately normal models with weak correlations, current samplers may be adequate. However, it is important to generalize these methods for other multivariate probability models.

## 2 Multi-Particle Sampler

The fact that the suggested sampler is based on a multi-particle system will be discussed in detail.

### 2.1 Maxwell-Boltzmann Distribution

Assume a closed system containing  $N$  high-speed particles colliding to exchange energy, and disregard potential energies for the time being. In three-dimensional space, a  $3 \times N$  vector is formed by the initial momenta of the  $N$  particles. Due to the fact that each particle's travelling direction is equally likely after a collision, the  $3 \times N$  vector is uniformly distributed over a hypersphere whose magnitude is determined solely by the total kinetic energy of the system. Due to these characteristics being unique to the normal distribution, each dimension of the  $3 \times N$  vector follows the same zero-mean normal distribution [12].

The infinitesimal element in spherical and Euclidean coordinate systems has the following properties:

$$dp_x dp_y dp_z = p^2 \sin(\phi) dp d\theta d\phi. \quad (1)$$

When a spherical coordinate is substituted for the Euclidean coordinate in (1), the infinitesimal probability under the Euclidean coordinate can be transformed to a spherical coordinate

$$\frac{e^{-\frac{p_x^2}{2\sigma^2} - \frac{p_y^2}{2\sigma^2} - \frac{p_z^2}{2\sigma^2}}}{2\sqrt{2}\pi^{3/2}\sigma^3} dp_x dp_y dp_z = \frac{p^2 \sin(\phi) e^{-\frac{p^2}{2\sigma^2}}}{2\sqrt{2}\pi^{3/2}\sigma^3} dp d\phi d\theta. \quad (2)$$

Integrate  $\phi$  and  $\theta$  for the right of (2) in the ranges of  $[0, \pi]$  and  $[0, 2\pi]$ , respectively, to obtain the Maxwell-Boltzmann distribution.

$$\frac{\sqrt{\frac{2}{\pi}} p^2 e^{-\frac{p^2}{2\sigma^2}}}{\sigma^3} dp. \quad (3)$$

The Maxwell-Boltzmann distribution is motivated by the concept of equal-probable directions for momenta, which is applicable to systems with any type of potential energy in principle. Because the total energy of a closed system is conserved, if the initial positions of all particles are known, the system's total potential energy is likewise known, and thus the system's total kinetic energy is determined. Thus, the system's  $3 \times N$  vector of starting momenta is evenly distributed on a spherical surface with  $3 \times N - 1$  dimensions, and its magnitude is governed by the system's total kinetic energy. Each point on the spherical surface corresponds to a spatial state that defines all particles' locations. The concept of equal a priori probabilities, stipulates that every point on the spherical surface is equally likely to be chosen. Due to the fact that the subsequent state is equally probable given the initial

state, the probability of mutual proposal (jump) between any two states will be equal.

### 2.2 Metropolis Algorithm

Collisions between the particles in a multi-particle system can exchange energy, resulting in variable total energies for each particle. For the time being, momenta's magnitudes can be ignored because they are governed by kinetic energies. Each dimension of momentum is distributed according to a standard normal distribution; thus, the momentum of a  $D$ -dimensional particle is distributed according to a  $D$ -dimensional standard normal distribution, while the momenta of  $N$  particles are distributed according to an  $N \times D$ -dimensional standard normal distribution. Thus, the initial momenta of all  $N$  particles can be randomly chosen from the surface of an  $N \times D - 1$ -dimensional hypersphere. Because initial momenta and final terminals of simulation trajectories are bijective, each final terminal (an  $N \times D$ -dimensional vector) has an equal chance. Thus, given the initial positions, the final positions follow a uniform distribution. As a result, when  $N$  particles are treated as a single entity, the mutual jump probabilities between the beginning and final positions are equal

$$\begin{aligned} P(q_{1,0}, \dots, q_{N,0} \rightarrow q_{1,1}, \dots, q_{N,1}) = \\ P(q_{1,1}, \dots, q_{N,1} \rightarrow q_{1,0}, \dots, q_{N,0}). \end{aligned} \quad (4)$$

Due to the fact that the particles are only related via the system's energy, which is a scalar quantity, the probability can be assumed to be independent

$$\prod_{i=1}^N P(q_{i,0} \rightarrow q_{i,1}) \approx \prod_{i=1}^N P(q_{i,1} \rightarrow q_{i,0}). \quad (5)$$

The simplest condition satisfying the preceding statement is that the mutual jump probabilities between the initial and final positions of each particle are equal

$$P(q_{i,0} \rightarrow q_{i,1}) \approx P(q_{i,1} \rightarrow q_{i,0}). \quad (6)$$

In summary, any two  $N \times D$ -dimensional positions in the entire system have identical mutual jump probability, which implies that the mutual jump probabilities between any two points of any particle are equal. This means that in a collision-prone system, even if the total energies of a particle's two terminal positions are different, their mutual jump probabilities are equal. As a result, each particle can be subjected to the Metropolis algorithm by conditionally accepting a new position based on the acceptance probability

$$\alpha = \min(1, e^{U(q_{i,0}) - U(q_{i,1})}). \quad (7)$$

The magnitude of a particle's momenta can be determined using the Euclidean norm for kinetic energy  $K_0$

$$\|p\| = \sqrt{p^T p} = \sqrt{2K_0}. \quad (8)$$

Thus, on the surface of a hypersphere, momenta are uniformly distributed. When we consider the biject between the initial momenta and end terminals of simulation trajectories, we see that all final spatial candidates are equiprobable given an initial spatial position. As a result, under the assumption of total energy conservation, the (new)

spatial positions have identical a priori probabilities. The Mahalanobis norm can be used to define the magnitudes of momenta for general kinetic energies  $K_r$ ,

$$\|p\| = \sqrt{p^T \left(\frac{d^2U}{dq^2}\right)^{-r} p} = \sqrt{2K_r}. \tag{9}$$

The momenta of the required kinetic energy are located on the surface of a hyper-ellipsoid, with equal (Mahalanobis) magnitudes in all directions. Thus, the momenta are effectively dispersed on the surface of a hyper-sphere, resulting in equiprobable end terminals for trajectories, akin to  $K_0$ . As a result, introducing kinetic  $K_r$  does not alter the discussion of  $K_0$ , as only the form of trajectories is altered.

### 2.3 Principal Components of Probability Distributions

Multivariate probability density functions or associated potential functions can be used to define general probability models. The eigenvalues and eigenvectors of the inverse of the Hessian matrix can be used to characterize the local span and direction of each spatial point. The inverse of Hessian matrices produces covariance matrices with eigenvalues for variances of principal components and eigenvectors for orientations in the case of multivariate normal distributions. These properties can be demonstrated using a bivariate normal distribution with a potential function specified by

$$U(x_1, x_2) = \frac{1}{2}(x_1 \ x_2) \begin{pmatrix} 1 & .7 \\ .7 & 1 \end{pmatrix}^{-1} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}. \tag{10}$$

Inverting the Hessian matrix (i.e., the potential function's second order derivative function) yields the covariance matrix

$$\left(\frac{d^2U}{dx^2}\right)^{-1} = \begin{pmatrix} 1 & 0.7 \\ 0.7 & 1 \end{pmatrix}. \tag{11}$$

The spans and orientations in Figure 1 are shown in two arbitrary positions. Standard deviations, which are the square roots of the eigenvalues, are used to illustrate the span of principal components. The first principal components, denoted by orange arrows, correspond to the largest standard deviation. For bivariate normal distributions, there are two principal components; for N-dimensional normal distributions, there are N components. Hessian matrices are constant matrices that generate the same span and orientation for all positions. Thus, principal components can be used to summarize the general characteristics of multivariate normal distributions.



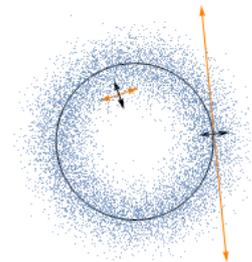
**Figure 1.** Principal components of a bivariate normal distribution

The same term can also be used with other probability models, such as a ring-shaped distribution with the potential function

$$U(x_1, x_2) = \frac{(\sqrt{x_1^2+x_2^2}-r)^2}{2\sigma^2}. \tag{12}$$

The Hessian of the potential functions is no longer constant but fluctuates point by point. Each position has a local coordinate associated with it. Normalized eigenvectors define the coordinate axes as functions of spatial position, while squared roots of eigenvalues define the unit scales.

Figure 2 illustrates two local coordinates and the scales associated with their relative placements. The greater the largest eigenvalue, the closer the angular axis (black circle) is. One scale approaches infinity on the angular axis. Because the Hessian of probability distributions might be negative definite, absolute eigenvalue values are utilized here. While the concept of principal components may be not appropriate for all multivariate distributions, this has no bearing on the subsequent arguments. As a result, the concepts of multivariate normal distributions apply to all distributions. The axis of local coordinates with the largest scale, as seen by orange arrows in Figure 2, is analogous to the first principal component, and so on. Because the local coordinates of multivariate normal distributions correspond to the global coordinates, the notion of principal components can also be generalized.



**Figure 2.** “Principal components” of a ring-shaped distribution

### 2.4 A Family of Kinetic Energies

The primary difference between HMC and RMHMC is in their kinetic energy, which can be generally defined as

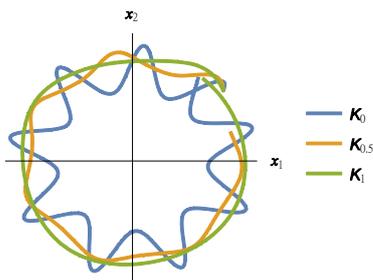
$$K_r(p, q) = \frac{1}{2}p^T \left(\frac{d^2U}{dq^2}\right)^{-r} p. \tag{13}$$

The parameter  $r$  is used to adjust the kinetic energy. Taking  $r = 0$  gives the kinetic energy of HMC, which is solely a function of momentum  $p$ . We obtain a similar kinetic energy of RMHMC when  $r = 1$ , which is a function of both momentum  $p$  and position  $q$ . When  $r$  is in the unit interval, a family of functions can be created with the HMC and RMMHC kinetic energies as the special cases.

For the most extreme case, physical energy  $K_0$  is determined solely by momenta. The other energies are non-physical in essence. Because Hamiltonian dynamics does not specify the form of kinetic energy, any kinetic function is theoretically feasible. Equation (13) mimics the negative logarithm of a multivariate normal distribution, except for the normalizing factors. For kinetic energy, the normalization

derived from the multivariate normal distribution may be superfluous. It is because, by symmetry, the initial momentum of a particle is uniformly picked from a spherical surface, followed by changing the magnitude according to the kinetic energy. Different kinetic functions bring about different distributions of momenta. In other words, kinetic energy can only determine magnitudes, whereas the directions of momenta are equally probable. Thus, in principle, the kinetic function need not be the negative logarithm of multivariate normal probability density functions.

Discretizing Hamiltonian equations via Euler integration generates simulation trajectories for  $q$  and  $p$ , although we are primarily concerned with the spatial variable  $q$ . By examining the trajectories in Figure 3, we can see that the form is determined by the parameter  $r$ , which takes on values of 0, 0.5, or 1. The blue trajectory formed by  $K_0$  proceeds preferentially along the polar coordinate's radial directions. On the other hand,  $K_1$ 's green trajectory revolves primarily around the angular coordinate. When  $r$  is between 0 and 1, the related trajectory takes on an intermediate form. A third type of trajectory appears unneeded in this case, as bivariate probability models have only two "principal components" that may be traversed using the  $K_0$  and  $K_1$  trajectories. However, for components with a range of scales, other types of trajectories may be necessary.



**Figure 3.** Simulation trajectories for the ring-shaped potential energy

To get the decimal power of the Hessian matrix, Eigen decomposition can be used:

$$\frac{d^2U}{dq^2} = V \cdot \Lambda \cdot V^T. \tag{14}$$

Thus, computing the Hessian to the power of  $r$  entails computing the eigenvalues to the power of  $r$

$$\left(\frac{d^2U}{dq^2}\right)^r = V \cdot \Lambda^r \cdot V^T. \tag{15}$$

The Hessian matrix must be positive definite here since only positive eigenvalues are meaningful for decimal  $r$ . The Hessian can be non-positive definite for potential energies of general probability density functions because it can change according to positions. Thus, we must generalize the kinetic energies.

Assume that the eigenvalues are nonzero. The kinetic family can be derived using quadratic potential energies, with the Hessian being positive, negative, or indefinite. Positive quadratic potential energies are idealized and simplified models with a positive definite Hessian, for example (10). Because the potential function is analogous to physical

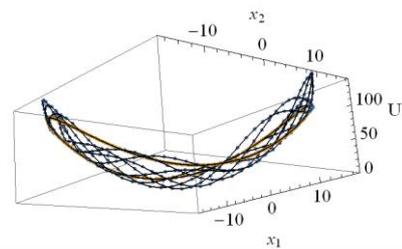
potential energy, it can be used in conjunction with physical kinetic energy

$$K_0(p, q) = \frac{1}{2}p^T p, \tag{16}$$

as well as the non-physical kinetic energy described below:

$$K_1(p, q) = \frac{1}{2}p^T \left(\frac{d^2U}{dq^2}\right)^{-1} p. \tag{17}$$

The total energies are also positive since both the kinetic and potential energies are positive. During simulations, the energies are converted into one another while preserving constant total energy. As illustrated in Figure 4, the potential surface can be traversed using trajectories created by Hamiltonian simulations utilizing two kinetic energies, where  $x_1$  and  $x_2$  signify spatial variables and  $U$  is the potential energy. Due to the fact that the trajectories of  $K_0$  and  $K_1$  move preferentially in virtually orthogonal directions, respectively, alternating two trajectories and sampling can be used to approximate target distributions. Furthermore, because negative kinetic energies result in divergence, not all kinetic energies are valid. As a result, kinetic and potential energy need to be matched.



**Figure 4.** Potential energy trajectory with positive definite Hessian; orange:  $K_1$ , blue:  $K_0$

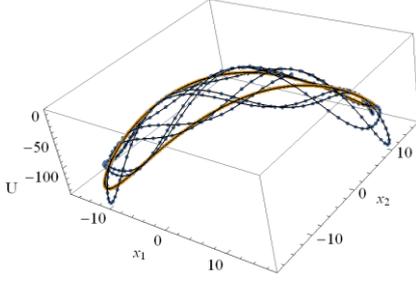
We can also examine potential functions that have a negative definite Hessian, such as

$$U(x_1, x_2) = -\frac{1}{2} \begin{pmatrix} x_1 & x_2 \end{pmatrix} \begin{pmatrix} 1 & .7 \\ .7 & 1 \end{pmatrix}^{-1} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}. \tag{18}$$

If we have negative potential energies, negative kinetic energies are required to maintain a stable simulation, as positive kinetic energies will result in divergence. Only  $K_0$  is modified, as  $K_1$  is negative due to the negative definite Hessian. Thus, to avoid divergence, a negative quadratic function is used

$$K_0(p, q) = -\frac{1}{2}p^T p. \tag{19}$$

Because of the negative potential and kinetic energies, the total energy in simulations is both negative and conserved. The fact that both  $K_0$  and  $K_1$  have stable trajectories is illustrated in Figure 5. Each trajectory follows a unique path around the concave quadratic surface, which has an effect on subsequent sampling. As a result of the concave shape, sampling algorithms have a tendency to drop into nearby regions of low potential energy. As a result, borders can be drawn around the zone.



**Figure 5.** Potential energy trajectory with negative definite Hessian; orange:  $K_1$ , blue:  $K_0$

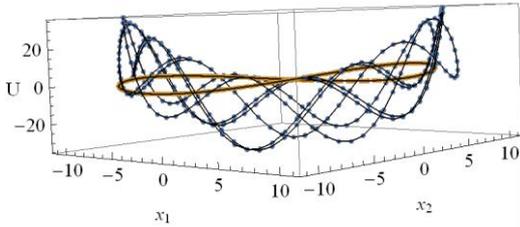
To be comprehensive, we also take into account potential energies with indefinite Hessian. For instance,

$$U(x_1, x_2) = \frac{1}{2}(x_1 \ x_2) \begin{pmatrix} .7 & 1 \\ 1 & .7 \end{pmatrix}^{-1} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}. \quad (20)$$

The eigenvalues are either positive or negative.  $K_1$  is identical to the preceding two instances. However, if the above (positive or negative) kinetic energies  $K_0$  are used, the simulations will diverge. Using the method described below, we find that the right form is

$$K_0(p, q) = p_1 p_2. \quad (21)$$

As illustrated in Figure 6, the potential energy surface is saddle-shaped and does not have a maximum or a minimum. As is customary, the trajectory of  $K_0$  and  $K_1$  traverses the surface in some way. It is difficult to access regions of high or low potential energy for  $K_1$  trajectories.



**Figure 6.** Potential energy trajectory with indefinite Hessian; orange:  $K_1$ , blue:  $K_0$

In summary, we infer that the eigenvalues' power is required to preserve the sign. As a result, Hessian to the power of  $r$  can be implemented by computing the absolute eigenvalues to the power of  $r$  and then multiplying elementwisely by its sign, which is 1, -1, or 0, depending on whether the value is positive, negative, or zero

$$\left(\frac{d^2U}{dq^2}\right)^r = V \cdot (|A|^r \odot \text{sign}(A)) \cdot V^T. \quad (22)$$

It allows for the application of the new kinetic energies to general multivariate probability models. Despite possible inaccuracies or divergences, physical kinetic energy  $K_0$  and non-physical kinetic energy  $K_1$  were initially utilized for this purpose. Now, in order to employ  $K_r$  for general multivariate models, the power of Hessian could be computed appropriately. The only criterion for models is that they have a non-singular Hessian, or, more precisely, that they have non-zero eigenvalues for the Hessian.

## 2.5 The ‘‘Optimal’’ Kinetic Energy

According to the above quadratic examples, the trajectories of  $K_1$  does not favor any specific principal components and therefore the methods based on it should be accurate. However, it is known from the forthcoming experiments that these methods tend to underestimate the scale of latter principal components. Thus, we need to analyze and fix the problem.

Considering the case of positive definite, the Hessian in the  $K_1$  kinetic energy can be substituted with its Eigen decomposition

$$K(p, q) = \frac{1}{2}p^T V \Lambda^{-1} V^T p. \quad (23)$$

According to Hamiltonian dynamics, the time derivative of the position equals the partial derivative of kinetic energy with respect to the momentum

$$\frac{dq}{dt} = \frac{\partial H}{\partial p} = \frac{\partial K}{\partial p} = V \Lambda^{-1} V^T p. \quad (24)$$

Therefore,  $\Lambda^{-1}$  can be treated as the updating magnitudes of principal components. Since the squared root of the inverse of eigenvalue matrix (i.e.  $\Lambda^{-1/2}$ ) corresponds to the scales (i.e. standard deviations) of principal components, the updating magnitudes and scales are mismatched. To tackle the problem, we can rescale the updating magnitudes of principal components with the squared root of the eigenvalue matrix

$$\frac{\partial K}{\partial p} = V \Lambda^{\frac{1}{2}} \Lambda^{-1} V^T p = V \Lambda^{-\frac{1}{2}} V^T p. \quad (25)$$

Thus, the kinetic energy needs to have the following form

$$K(p, q) = \frac{1}{2}p^T V \Lambda^{-\frac{1}{2}} V^T p. \quad (26)$$

Considering also the cases of non-positive-definite, the required kinetic energy is obtained

$$K_{0.5}(p, q) = \frac{1}{2}p^T V (|A|^{-0.5} \odot \text{sign}(A)) V^T p. \quad (27)$$

The kinetic energy is a kind of  $K_r$  and is the only function that can impartially traverse all principal components of multivariate normal distributions. Therefore,  $K_{0.5}$  is arguably one of the feasible kinetic energy for sampling algorithms.

## 2.6 The Derivative of Kinetic Energies with Respect to $q$

To implement an accurate Hamiltonian simulation, the derivative of energies with respect to phase variables should be determined. Due to the fact that the potential energies are simply a function of  $q$ , only three derivatives are required:  $dU/dq$ ,  $\partial K/\partial p$ , and  $\partial K/\partial q$ . Because the first two derivatives are trivial, we discuss only  $\partial K/\partial q$ . Kinetic energy modifies eigenvalues, which are dependent on the second derivatives of potential energy in terms of  $q$ . As a result, in order to compute the derivative of kinetic energies with respect to  $q$ , we need first compute the derivatives of kinetic energies with regard to eigenvalues and eigenvectors, followed by the derivatives of eigenvalues and eigenvectors with respect to  $q$ . As a result, the

problem is recast as computing the derivatives of eigenvalues and eigenvectors in terms of a scalar variable  $t$ . The approach may be directly generalized to a  $D$ -dimensional vector  $q$ . The appendix discusses the derivatives of eigenvalues and eigenvectors with respect to a scalar variable, which is used to determine the  $\partial K/\partial q$ .

To begin, let us decompose the Hessian matrix into its eigenvectors and eigenvalues

$$\frac{d^2U}{dq^2} \cdot v_i = \lambda_i v_i. \tag{28}$$

The generated kinetic energy is calculated using eigenvalues and eigenvectors

$$\begin{aligned} K_r(p, q) &= \frac{1}{2} p^T \cdot V \cdot (|A|^r \odot \text{sign}(A)) \cdot V^T \cdot p \\ &= \frac{1}{2} p^T \cdot V \cdot f(A) \cdot V^T \cdot p, \end{aligned} \tag{29}$$

in which the eigenvalue transforming function is defined as

$$f(A) = |A|^r \odot \text{sign}(A). \tag{30}$$

In computing  $\partial K/\partial q$ , the momentum  $p$  can be considered as a constant.  $K_r$  transforms eigenvalues rather than eigenvectors. The derivative of the kinetic energy with respect to  $q_j$  is made of two parts according to the step-by-step derivation

$$\begin{aligned} \frac{dK_r}{dq_j} &= p^T \cdot \frac{dV}{dq_j} \cdot f(A) \cdot V^T \cdot p + \frac{1}{2} p^T \cdot V \cdot \frac{df(A)}{dq_j} \cdot V^T \cdot p = p^T \cdot \\ &\frac{dV}{dq_j} \cdot f(A) \cdot V^T \cdot p + \frac{1}{2} p^T \cdot V \cdot (\dot{f}(A) \odot \frac{dA}{dq_j}) \cdot V^T \cdot p. \end{aligned} \tag{31}$$

in which the transforming function's derivative is

$$\dot{f}(A) = -r|A|^{r-1}. \tag{32}$$

Thus, the derivative of kinetic energy in terms of  $q$  is turned into the derivatives of eigenvalues and eigenvectors in terms of  $q$ , which is transformed into the derivative of the Hessian matrix in terms of  $q$

$$\frac{d\lambda_i}{dq_j} = v_i^T \frac{d^3U}{dq^2 dq_j} v_i. \tag{33}$$

Similarly, the eigenvector's derivative with respect to  $q$  is

$$\frac{dv_i}{dq_j} = \sum_{k \neq i} \frac{1}{\lambda_i - \lambda_k} (v_k^T \frac{d^3U}{dq^2 dq_j} v_i) v_k \tag{34}$$

According to (33) and (34), the derivative of kinetic energies requires the third-order derivative of potential energies with respect to  $q$ , and the computation is an order of magnitude higher than the Hessian matrix. The approximate approach stated behind may be more appropriate for high-dimensional models, i.e.  $\partial K/\partial q$  can be implemented via numerical differences.

The effect, on the other hand, appears to be connected to  $\delta$ , implying that sampling errors may be caused by numerical differences. As illustrated in Figure 7, for the ring-shaped distribution with  $\sigma=0.0001$  and  $r=10$ , a larger  $\delta$  results in a larger error, which causes the radial samples to deviate to the

right. Due to the floating-point limit,  $\delta$  cannot be infinitely tiny, and as a result, certain errors will occur. The finite difference approach is similar to the accurate method in terms of computing and can be utilized in contexts that lack the facility of third-order derivatives.

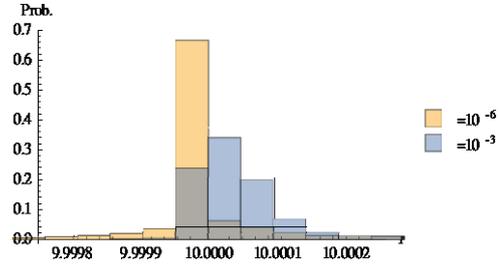


Figure 7. Influence of  $\delta$  in numerical difference

## 2.7 The Effect of Derivatives of Energy

A particle's total energy is the sum of its kinetic and potential energies. While potential energies are entirely dependent on location, kinetic energy can depend on both position and momentum

$$H(q, p) = K(p, q) + U(q). \tag{35}$$

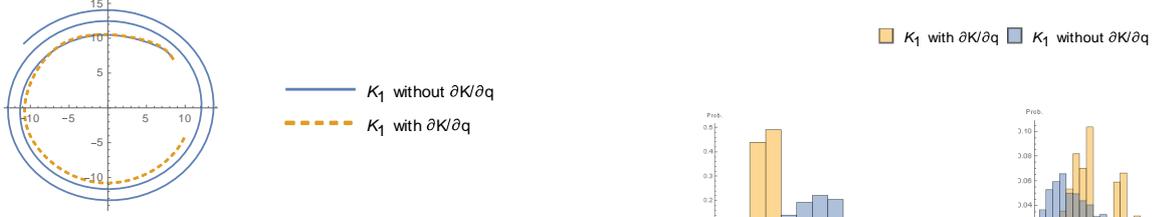
We obtain the rate of change for momentum and position variables by substituting kinetic and potential energy in Hamiltonian equations

$$\frac{dp}{dt} = -\frac{\partial H}{\partial q} = -\frac{dU}{dq} - \frac{\partial K}{\partial q}, \tag{36}$$

$$\frac{dq}{dt} = \frac{\partial H}{\partial p} = \frac{\partial K}{\partial p}. \tag{37}$$

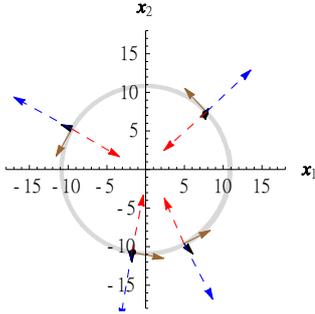
A sometimes neglected point is that kinetic energy can also be functions of position in general. Thus, the final component on the right of (36) is required in theory. Due to the fact that kinetic energies are computed using the second order derivatives of potential energies with respect to  $q$ , determining  $\partial K/\partial q$  requires the third order derivatives of potential energies with respect to  $q$ , necessitating additional computation. As a result,  $\partial K/\partial q$  might be omitted or simplified in practice [13]. Ignoring the term, on the other hand, may result in systematic errors for single-trajectory algorithms that utilize non-physical kinetic energy [7].

As illustrated in Figure 8, disregarding  $\partial K/\partial q$  violates the rule of conservation of energy, causing the trajectory formed by the  $K_1$  energy for the ring-shaped potential energy to be inclined outward. The trajectory's outward inclination results in a sample bias toward the outer side. The physical kinetic energy  $K_0$  is purely dependent on  $p$  and is unaffected by the factor, as its  $\partial K/\partial q$  is equal to zero. By alternating various trajectories, the systematic error can be reduced. Because the trajectories of  $K_1$  incline outward, there is a possibility that the trajectories of  $K_0$  incline inward, thus eliminating partial inaccuracy.



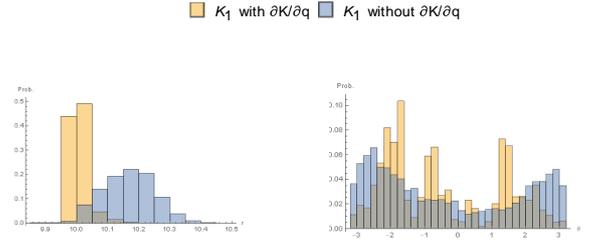
**Figure 8.** The influence of ignoring  $\partial K/\partial q$  for trajectories

In Figure 9, the light grey curve depicts the simulation trajectory for the ring-shaped potential energy. The brown arrows represent the vectors of  $\partial K/\partial p$  at their current locations, which represent the incremental changes in spatial positions along the ring's tangent. The blue arrows indicate  $dU/dq$ , while the red arrows indicate  $\partial K/\partial q$ . Thus, considering solely  $dU/dq$  will result in outer deviations of the trajectories. The black arrows indicate the total of  $dU/dq$  and  $\partial K/\partial q$ , or the incremental changes in momenta. Due to the fact that  $dU/dq$  and  $\partial K/\partial q$  have nearly opposite orientations, the two are largely cancelled, leaving only minor incremental changes for momenta, which in turn results in smaller incremental changes for spatial positions. Thus, we confront a contradiction between accuracy and exploration, as precise methods with  $\partial K/\partial q$  may be inferior in terms of exploration, but approximate approaches without  $\partial K/\partial q$  may result in a superior sampling effect as a result of increased exploration.



**Figure 9.** Energy derivatives on the simulation trajectory; brown:  $\partial K/\partial p$ , blue arrow:  $dU/dq$ , red arrow:  $\partial K/\partial q$ , black arrow:  $dU/dq + \partial K/\partial q$

On the left side of Figure 10, the histogram of samples for the ring-shaped distribution on the radial coordinate using  $K_1$  kinetic energy with  $\sigma=0.1$  and  $\rho=10$  is shown. The approximate method's samples are typically larger than 10, indicating systematic error. The accurate method employing  $\partial K/\partial q$  distributes the samples more evenly about 10, thereby reducing the corresponding systematic error. For the precise method, the radial distribution may deviate from the normal distribution, which could be due to a variety of factors (such as the characteristics of  $K_1$ ). While the technique based on  $\partial K/\partial q$  is more precise in the radial component, it has a large error in the angle component, indicating inferior exploration. The right picture demonstrates that the samples generated by the approach that ignores  $\partial K/\partial q$  are more uniformly distributed, exhibiting stronger exploration.



**Figure 10.** Samples for the ring-shaped distribution (left: radial, right: angular)

## 2.8 Algorithm Description

The suggested single or multi-trajectory algorithm is analogous to the alternating two-trajectory algorithm [7]. By incorporating  $N$  distinct types of trajectories, both  $H$  and  $\delta$  will be  $N$ -dimensional arrays, as each trajectory needs adjust its own total energy and step size. The  $N$ -dimensional array  $R$  is predefined with the kinetic energy parameter  $r$  as its elements. If  $R$  consists of only zero or one, we obtain the modified HMC or RMHMC algorithms.  $R$  will contain both 0 and 1 for the bi-trajectory algorithm. More intermediate decimals between 0 and 1 can be considered when sampling general probability models. Line 4 specifies the kinetic type that will be used to calculate the total kinetic energy of all particles in line 9. The expected total kinetic energy of the system is determined in line 12 using the potential energy and total energy of the system. Following that, the magnitudes of all particles' momenta are rescaled to preserve energy throughout the system. Line 17 starts the simulation and sampling. The leapfrog algorithm is defined in line 18. Due to the fact that just variable  $q$  is sampled, the leapfrog final update can be ignored. Lines 20 to 21 make use of the step size associated with the present kinetic energy. Line 32 contains the adjustment for the total energy and step size. The following kinetic energy is selected sequentially and cyclically at line 33.

---

### Algorithm 1. The proposed method

---

```

1 initialize  $H$  and  $\delta$ 
2  $\rho=1$ 
3 for  $i = 1$  to  $N$ 
4    $r=R_\rho$ 
5    $U_{\text{total}}=0$ 
6    $K_{\text{total}}=0$ 
   //  $M$  particles
7   for  $j=1$  to  $M$ 
   //equiprobable direction
8      $p_j = \text{NormalRandom}(0,1)$ 
9      $K_{\text{total}}=K_{\text{total}} + K_r(p_j, q_j)$ 
10     $U_{\text{total}}=U_{\text{total}}+U(q_j)$ 
11  end
12   $K_d=H_\rho-U_{\text{total}}$ 
   // total energy conservation for the system
13  for  $j=1$  to  $M$ 
14     $p_j = p_j \sqrt{|K_d/K_{\text{total}}|}$ 
15  end
16  for  $j=1$  to  $M$ 
17     $q_0=q_j$ 
   // leapfrog
18     $p_j = p_j - \frac{1}{2} \delta_\rho \left( \frac{dU}{dq_j} + \frac{\partial K_r}{\partial q_j} \right)$ 

```

```

19   for s = 1 to S
20        $q_j = q_j + \delta_\rho \frac{\partial K_r}{\partial p_j}$ 
21        $p_j = p_j - \delta_\rho \left( \frac{dU}{dq_j} + \frac{\partial K_r}{\partial q_j} \right)$ 
22       save  $U(q_j)$ 
23   end
24   // Metropolis
25    $\alpha = e^{U(q_0) - U(q)}$ 
26    $u = \text{UniformRandom}(0,1)$ 
27   if  $\alpha < u$ 
28        $q_j = q_0$ 
29   end
30   save  $q_j$ 
31   save  $\alpha$ 
32   end
33   tune  $H_\rho$  and  $\delta_\rho$ 
34   if  $\rho == \text{size of R}$ 
35        $\rho = 1$ 
36   else
37        $\rho = \rho + 1$ 
38   end

```

Algorithm 2 and Algorithm 3 establishes a definition for kinetic energy and its derivatives. To obtain the Hessian's eigenvalues and eigenvectors, an Eigen decomposition is performed. Following the transformation of the eigenvalues matrix, we reconstruct the  $W$  matrix whose inverse can be accomplished by solving a linear equation.

---

**Algorithm 2.** Kinetic energy

```

1   function  $K_r(p, q)$ 
2   // eigenvalue decomposition
3    $\Lambda, V = \text{eigen} \left( \frac{d^2U}{dq^2} \right)$ 
4    $W = V \cdot (|\Lambda|^r \odot \text{sign}(\Lambda)) \cdot V^T$ 
5   return  $1/2p^T W^{-1}p$ 
6   end

```

---

**Algorithm 3.** The derivative with respect to  $p$ 

```

1   function  $\partial K_r / \partial p(p, q)$ 
2    $\Lambda, V = \text{eigen} \left( \frac{d^2U}{dq^2} \right)$ 
3    $W = V \cdot (|\Lambda|^r \odot \text{sign}(\Lambda)) \cdot V^T$ 
4   return  $W^{-1}p$ 
5   end

```

---

Algorithm 4 illustrates how to compute  $\partial K / \partial q$ . From lines 7 to 15, the derivative of eigenvectors with respect to  $q$  is determined, where the variable  $u$  in line 11 is a  $D \times D \times D$ -dimensional tensor. In line 22, the derivative of the eigenvalues with respect to  $q$  is determined. Line 17 accumulates the derivative due to eigenvectors, and line 22 accumulates the derivative due to eigenvalues.

---

**Algorithm 4.** The derivative with respect to  $q$ 

```

function  $\partial K_r / \partial q(p, q)$ 
1    $D = \text{dimension}(p)$ 
2    $\Lambda, V = \text{eigen} \left( \frac{d^2U}{dq^2} \right)$ 

```

```

3    $V = \frac{V}{\|V\|}$ 
4    $f = -r|\Lambda|^{-r-1}$ 
5    $f = |\Lambda|^r \odot \text{sign}(\Lambda)$ 
6    $y = V \cdot p$ 
7   for  $k=1$  to  $D$ 
8       for  $i=1$  to  $D$ 
9           for  $j=1$  to  $D$ 
10              if  $i \neq j$ 
11                   $u_{k,i,j} = u_{k,i,j} + \frac{v_j}{\lambda_i - \lambda_j} \left( v_j^T \cdot \frac{d^3U}{dq^2 dq_k} \cdot v_i \right)$ 
12              end
13          end
14      end
15  end
16  // due to  $V$ 
17  for  $i=1$  to  $D$ 
18       $\frac{dK}{dq_i} = p^T \cdot u_i \cdot f \cdot y$ 
19  end
20   $g = f \odot y^2$ 
21  // due to  $\Lambda$ 
22  for  $j=1$  to  $D$ 
23      for  $i=1$  to  $D$ 
24           $\frac{dK}{dq_j} = \frac{dK}{dq_j} + \left( \frac{1}{2} v_i^T \cdot \frac{d^3U}{dq^2 dq_j} \cdot v_i \right) g_i$ 
25      end
26  end
27  return  $\frac{dK}{dq}$ 
28  end

```

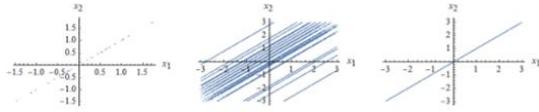
---

## 3 Experiments

### 3.1 Bivariate Potential Energies with Positive Definite, Negative Definite, or Indefinite Hessian

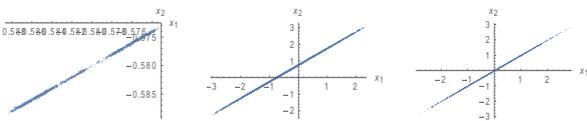
The program uses 30 particles in the experiment to reveal differences between single-particle and multi-particle samples. Over the range of -3 to 3, the prior distribution follows a uniform distribution. All chains are initiated within the region. The acceptance probability is zero whenever a simulation ends outside.

The parameter estimation of weakly correlated multivariate normal distributions is trivial for most methods. To investigate the effect of kinetic energies on sampling, a bivariate normal distribution with a correlation coefficient of 0.99999999 is used. Figure 11 depicts scatter plots of all particle samples for the (positive definite) potential energy. The samples based on the  $K_0$  method are scattered on the diagonal line on the left. Because the trajectories move antidiagonally only from initial positions, the samples of  $K_1$  are distributed on many diagonal lines, as shown in the middle. The samples generated by using  $K_0$  and  $K_1$  alternately resemble the target distribution on the right. The reason for the disparity in results is that the trajectories of  $K_0$  primarily progress along the last components, whereas the trajectories of  $K_1$  can align with the first principal components. Two components can be traversed at the same time by using two kinetic energies alternately.



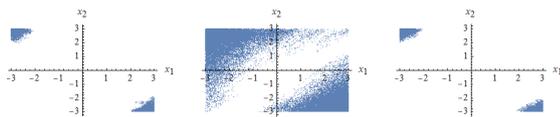
**Figure 11.** Scatter plots of all particles for potential energy with a positive definite Hessian; left:  $K_0$ , middle:  $K_1$ , and right:  $K_0, K_1$

The scatter plots for a single particle are shown in Figure 12.  $K_0$  samples are concentrated in a tiny area. On the other hand, the samples of  $K_1$  extend only anti-diagonally, with poor coverage along diagonals due to a lack of movement in that direction. The reason for this is that the sampling of the trajectories is heavily influenced by their form. On the right, alternating between two trajectories creates a more desirable result.



**Figure 12.** Scatter plots of a single particle for potential energy with a positive definite Hessian; left:  $K_0$ , middle:  $K_1$ , and right:  $K_0, K_1$

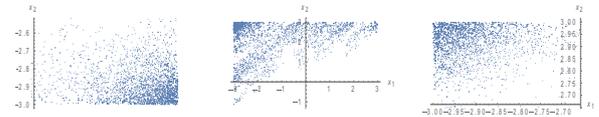
Despite the fact that the Hessian of potential energy specified in (18) is negative definite, it can still be sampled, as shown in Figure 13. Because the minimum potential energy is at infinity, a square boundary is used to avoid divergence. The potential energy is related to a non-normal distribution with approximately complementary samples at diagonal corners. The proposed kinetic energies can be used to overcome problems with convergence and the acceptance probability for potential energies with non-positive definite Hessian. The simulation trajectories can be automatically stabilized, setting the basis for sampling.  $K_0$  samples are found at diagonal corners on the left side of the picture.  $K_1$  samples are likewise generally divided into two disconnected sections by the anti-diagonal, as illustrated in the centre plots. The high energy region blocks all trajectories since no sample is situated on the anti-diagonal. The bi-trajectory approach, whose outcome is comparable to  $K_0$ , is shown on the right side of the picture. The significant disparity between  $K_0$  and  $K_1$  demonstrates how important trajectories are. Various trajectories may be required for generic probability models to visit the target regions thoroughly.



**Figure 13.** Scatter plots of all particles for potential energy with a negative definite Hessian; left:  $K_0$ , middle:  $K_1$ , and right:  $K_0, K_1$

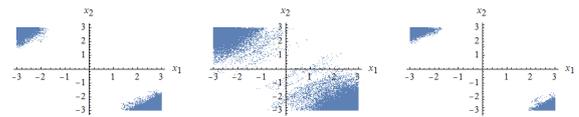
Figure 14 depicts samples from a single Markov chain. On the left, the samples with  $K_0$  are concentrated in the lower right corner, indicating that the trajectories cannot cross over to the opposite corner. The samples of  $K_1$  scatter widely in the centre of the picture, owing to the large scale movement of trajectories. On the right, the bi-trajectory algorithm's samples are dispersed in the upper left corner. It is evident from the

centre and right of the diagram that even  $K_1$  trajectories cannot pass through regions of large potential energy. To overcome energy barriers, it may be prudent to increase absolute kinetic energies by lowering the lower threshold of acceptance probability.



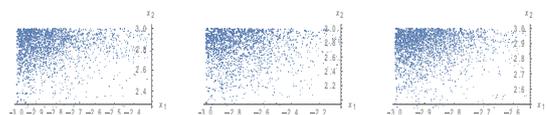
**Figure 14.** Scatter plots of one particle for the potential energy with a negative definite Hessian; left:  $K_0$ , middle:  $K_1$ , and right:  $K_0, K_1$

Figure 15 illustrates the sampling results for an indefinite Hessian. As the case of negative definite, we can rely on qualitative analysis to conclude that the method of  $K_1$  is less accurate than the approach of  $K_0$ , if we use the bi-trajectory result as a benchmark. The conclusion may come as a surprise, given the widespread belief that methods based on  $K_1$  (e.g. RMHMC) are superior to methods based on  $K_0$  (HMC). We recall from Figure 6 that the approach of  $K_1$  yields more restrictive trajectories, where sampling on the trajectories deviates from the desired distribution.



**Figure 15.** Scatter plots of all particles for potential energy with an indefinite Hessian; left:  $K_0$ , middle:  $K_1$ , and right:  $K_0, K_1$

By examining the samples from a single Markov chain, we can see in Figure 16 that  $K_0$  produces a result similar to alternating  $K_0$  and  $K_1$ . Because the Hessian matrix's absolute eigenvalues (1.7 and -0.3) are of the same order of magnitude, any kinetic energy within the family can be applied. If the absolute eigenvalues change by orders of magnitude, indicating a wide range of component scales, additional kinetic energies may be necessary to construct trajectories of diverse shapes.



**Figure 16.** T Scatter plots of one particle for the potential energy with an indefinite Hessian; left:  $K_0$ , middle:  $K_1$ , and right:  $K_0, K_1$

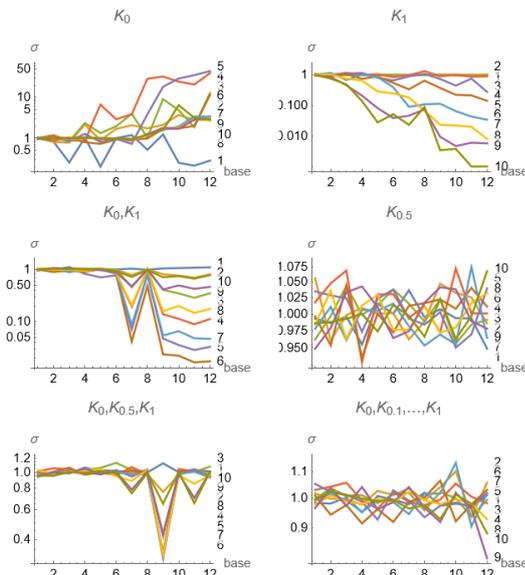
### 3.2 Multivariate Normal Distributions

To study the choice of kinetic energies, the methods are compared using a zero-mean and uncorrelated 10-dimensional normal distribution, whose standard deviation of the  $i$ th dimension (i.e. principal components) is set to  $\text{base}^{1-i}$ , with  $\text{base}=1\dots 12$ . When base equals 1, we get a 10-dimensional standard normal distribution. However, larger value for base will cause more difficulty for sampling. To assess sampling accuracies, the transformed samples will be a 10-dimensional

standard normal distribution, which means that their standard deviations will be approximately one

$$\Sigma^{-\frac{1}{2}}x \sim N(0,1). \tag{38}$$

As shown in the upper left corner of Figure 17, the method based on  $K_0$  tends to overestimates sample variances ( $\sigma^2$ ), which indicates the method (e.g. HMC) is inaccurate. The upper right corner shows that the  $K_1$  method (e.g. RMHMC) tends to increasingly underestimate sample variance of latter principal components. The middle left of Figure 17 suggests that alternating  $K_0$  and  $K_1$  is moderately helpful, as the curves are more concentrated toward to 1. The middle right of the figure reveals that the method based on  $K_{0.5}$  can accurately sample all components, since all curves are close to 1. More trajectories, instead, may lead to underperformance, exemplified by the lower left corner of Figure 17 that alternating  $K_0$ ,  $K_{0.5}$ , and  $K_1$ . The problem can be lessened by adopting more trajectories, such as 11 trajectories shown in the lower right corner of the figure. In summary, the sampling can be improved by adopting  $K_{0.5}$ .



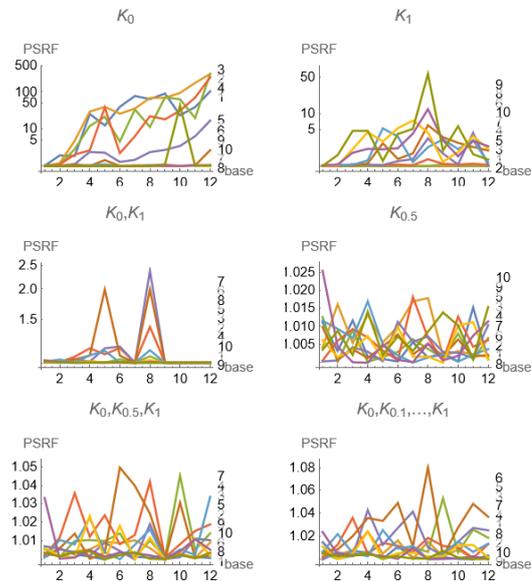
**Figure 17.** The standard deviations of transformed samples, the numbers on the right side indicate the order of principal components, the x-axis is base, and the y-axis is the standard deviations of transformed samples

To verify convergence, we also show the potential scale reduction factor (PSRF) whose value should be close to 1 if the chains have converged to the target posterior distribution [14-15]. As shown in the upper left corner of Figure 18, the method of  $K_0$  does not satisfy the convergence criterion in most cases. On the other hand, the convergence of  $K_1$  seems markedly improved despite the inflated values. Compared with the single trajectory methods, the convergence of bi-trajectory method of  $K_0$  and  $K_1$  is improved furthermore. The method of  $K_{0.5}$  converges for all components and cases, and other methods adopting  $K_{0.5}$  are also approximately converged. Or more exactly, the methods adopting  $K_{0.5}$  converge better.

The downside of multiple trajectories is the increased burn-in iterations, which is necessary, for each trajectories adapts its own energy and step size. Given a 10% change in each adjustment, each type of trajectory may require hundreds of iterations ( $n$ ) to span a suitably broad interval of  $[1.1^{-n}, 1.1^n]$ .

In other words, the number of burn-in iterations need to be multiplied by the number of kinetic energies. Because it is vital to ensure that all samples are located in high-density zones, or equivalently, have low potential energies, prior to start formal sampling.

### 3.3 Multimodal Probability

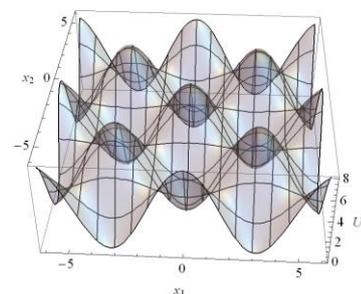


**Figure 18.** Potential scale reduction factor

Due to the presence of multiple modes, sampling multimodal probabilities may be more difficult, as illustrated by the potential energy of a two-dimensional cosine function

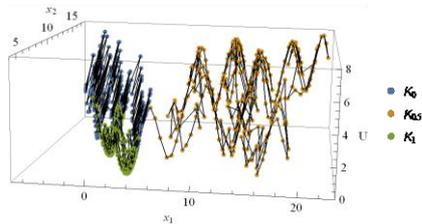
$$U(x_1, x_2) = 4(\cos(x_1)\cos(x_2) + 1). \tag{39}$$

As illustrated in Figure 19, the potential function has numerous extrema in the region, each with a distinct curvature, either positive or negative. Particles are attracted to locations with low potential energies and are thus more constrained to local regions when it comes to physical kinetic energy. By using the proposed kinetic energies, the particles can traverse between the extremes of high and low potential energies, continuously speeding toward the opposite extremum. The particles resemble rolling balls in bowls and inverted bowls and can continue their journey by passing across the boundaries of the two, allowing for more effective traversal of numerous extrema. On both sides of the boundary, the Hessian matrices of potential energy will be positive or negative definite, with similar trajectories to the quadratic positive or negative potential energies. As a result, once the trajectories pass through the boundaries, they accelerate ahead.



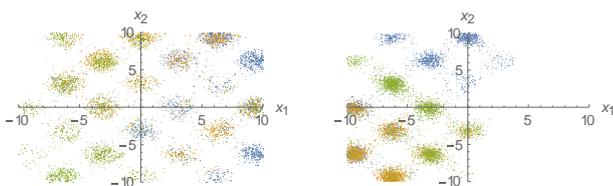
**Figure 19.** Two-dimensional cosine function

Trajectories can cover several extrema under the condition of total energy conservation during simulations, as shown in Figure 20 for three types of kinetic energies. For unconstrained domains, the trajectories will reach infinity. Setting sampling boundaries, with zero acceptance probability for samples obtained outside of the bounds, can solve the problem.



**Figure 20.** Simulation trajectories with the proposed kinetic energies

The scatter plots of three particles with distinct colours for the proposed  $K_0$  kinetic energy are shown in the left side of Figure 21. The burn-in period is 5000 iterations long, and 5000 samples are drawn for each particle in a square region. The colours of the particles are approximately mixed, indicating that they can freely traverse the region. The nearly uniform distribution of data across several modes indicates that the proposed technique is appropriate for multimodal probability. The right side of the figure reveals that when physical energy is used in the same settings, exploration is somewhat inhibited and samples concentrate in fewer modes, indicating a reduced capacity for border crossing. It is found that the proposed kinetic energies approach outperforms the physical kinetic energy method under various scenarios. Additionally, it is feasible to promote exploration and hence improve the sampling effect of multimodal probabilities by increasing kinetic energies, which can be accomplished by reducing the acceptance probability threshold.



**Figure 21.** Samples; left: the proposed kinetic energy  $K_0$ , right: the physical kinetic energy

## 4 Conclusion

Both HMC ( $K_0$ ) and RMHMC ( $K_1$ ) are subsets of a family of algorithms based on a general kinetic energy  $K_r$ . Although the trajectories generated from the kinetic energies are capable of traversing all components, the methods based on  $K_{0.5}$  are more accurate than others for multivariate normal distributions.

The HMC method utilizes only the gradient of potential energy and thus can be applied to high-dimensional models in principle. The drawback, however, is the inherent inaccuracy. It is found that more accurate methods are possible by adopting the Hessian matrix of potential energy, whose computational complexity is  $O(N^2)$  for  $N$ -dimensional models. As a result, the proposed methods are suitable only for low or medium-dimensional models. For high-dimensional models

(e.g. deep learning), more research is still needed for accurate methods.

## Acknowledgements

This work was partially supported by the National Natural Science Fund of China [Grant Nos. 61373063, 61872188 and 62172225]; and Fujian Provincial Key Laboratory of Information Processing and Intelligent Control, Minjiang University [Grant No. IIC1701].

This study was partially supported by the project of the Natural Science Foundation of the Education Department of Anhui Province (Grant No. KJ2020A0012), and the program of School Scientific Research of Anhui University of Finance and Economics (Grant No. ACKYC20049).

## References

- [1] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, E. Teller, Equation of State Calculations by Fast Computing Machines, *The Journal of Chemical Physics*, Vol. 21, No. 6, pp. 1087-1092, June, 1953.
- [2] Z. Hong, B. Dan, Visual Tracking Method based on Monte Carlo Compressed Sensing, *International Journal of Performability Engineering*, Vol. 16, No. 5, pp. 720-727, May, 2020.
- [3] S. Duane, A. Kennedy, B. J. Pendleton, D. Roweth, Hybrid Monte Carlo, *Physics Letters B*, Vol. 195, No. 2, pp. 216-222, September, 1987.
- [4] M. Kalaivani, R. Kannan, Reliability Assessment Model for Deterioration Systems Subject to Cumulative Damage, *International Journal of Performability Engineering*, Vol. 17, No. 12, pp. 981-988, December, 2021.
- [5] M. D. Homan, A. Gelman, The No-U-Turn Sampler: Adaptively Setting Path Lengths in Hamiltonian Monte Carlo, *The Journal of Machine Learning Research*, Vol. 15, No. 1, pp. 1593-1623, January, 2014.
- [6] K. Li, S. Jin, Z. He, Q. Yang, S. Lin, Critical Component Identification and Reliability Enhancement of AC Metro Traction Substation using FTA and Sequential Monte Carlo Simulation, *International Journal of Performability Engineering*, Vol. 16, No. 12, pp. 1862-1874, December, 2020.
- [7] X. Xu, C. Liu, H. Yang, Robust Inference Based On the Complementary Hamiltonian Monte Carlo, *IEEE Transactions on Reliability*, Vol. 71, No. 1, pp. 111-126, March, 2022.
- [8] M. Girolami, B. Calderhead, Riemann manifold Langevin and Hamiltonian Monte Carlo Methods, *Journal of the Royal Statistical Society Series B Statistical Methodology*, Vol. 73, No. 2, pp. 123-214, March, 2011.
- [9] M. Betancourt, A General Metric for Riemannian Manifold Hamiltonian Monte Carlo, *International Conference on Geometric Science of Information*, Paris, France, 2013, pp. 327-334.
- [10] T. S. Kleppe, Modified Cholesky Riemann Manifold Hamiltonian Monte Carlo: Exploiting Sparsity for Fast Sampling of High-dimensional Targets, *Statistics and Computing*, Vol. 28, No. 4, pp. 795-817, July, 2018.

[11] S. Lan, J. Streets, B. Shahbaba, Wormhole Hamiltonian Monte Carlo, *Proceedings of the AAAI Conference on Artificial Intelligence*, Québec City, Québec, Canada, 2014, pp. 1953-1959.

[12] W. Krauth, *Statistical Mechanics: Algorithms and Computations*, Oxford University Press, 2014.

[13] S. Byrne, M. Girolami, Geodesic Monte Carlo on Embedded Manifolds, *Scandinavian Journal of Statistics Theory and Applications*, Vol. 40, No. 4, pp. 825-845, December, 2013.

[14] A. Gelman, D. B. Rubin, Inference from Iterative Simulation using Multiple Sequences, *Statistical Science*, Vol. 7, No. 4, pp. 457-472, November, 1992.

[15] S. P. Brooks, A. Gelman, General Methods for Monitoring Convergence of Iterative Simulations, *Journal of Computational and Graphical Statistics*, Vol. 7, No. 4, pp. 434-455, 1998.

[16] L. Nicolaescu, Answer of Derivative of Eigenvectors of A Matrix with Respect to its Components, *MathOverflow*, 2016.

## Appendix Derivative of Eigenvectors and Eigenvalues

The next section discusses the derivatives of eigenvalues and eigenvectors with respect to a scalar variable, which is taken from an Internet forum dedicated to discussing the question [16].

If  $B$  depends on a single parameter  $t$ , then deriving with respect to  $t$  the equality

$$Bn_i = \lambda_i n_i, \tag{a.1}$$

we deduce

$$\dot{B}n_i + B\dot{n}_i = \dot{\lambda}_i n_i + \lambda_i \dot{n}_i. \tag{a.2}$$

Here we assume that  $\|n_i\| = 1$ . Hence  $\dot{n}_i \perp n_i, \forall i$ . Taking the inner product of the above equality with  $n_i$  and observing that

$$(B\dot{n}_i, n_i) = (\dot{n}_i, Bn_i) = \lambda_i (\dot{n}_i, n_i) = 0, \tag{a.3}$$

we deduce

$$\dot{\lambda}_i = (\dot{B}n_i, n_i). \tag{a.4}$$

This determines  $\dot{\lambda}_i$  in terms of  $\dot{B}$ .

Next, we take the inner product of (a.2) with  $n_j, j \neq i$ . Using the fact that  $B$  is symmetric we deduce

$$(\dot{B}n_i, n_j) + (\dot{n}_i, Bn_j) = \lambda_i (\dot{n}_i, n_j). \tag{a.5}$$

so that

$$(\dot{B}n_i, n_j) + \lambda_j (\dot{n}_i, n_j) = \lambda_i (\dot{n}_i, n_j). \tag{a.6}$$

This shows that

$$(\dot{n}_i, n_j) = \frac{1}{\lambda_i - \lambda_j} (\dot{B}n_i, n_j), \tag{a.7}$$

that is

$$\dot{n}_i = \sum_{j \neq i} \frac{1}{\lambda_i - \lambda_j} (\dot{B}n_i, n_j) n_j. \tag{a.8}$$

## Biographies



**Xiaopeng Xu** is a Ph.D candidate of the School of Computer Science and Technology, Nanjing University of Science and Technology. His research interests are the statistical inference of probability models, scientific computing, and formal language, computer algebra, etc.



**Chuancai Liu** is a Full Professor in the School of Computer Science and Engineering of Nanjing University of Science and Technology, China. He obtained his Ph.D. degree from China Ship Research Academy in 1997. His research interests include AI, pattern recognition and computer vision.



**Hongji Yang** received the B.Sc. and M.Sc. degrees in computer science from Jilin University, Changchun, China, in 1982 and 1985, respectively, and the Ph.D. degree in computer science from Durham University, Durham, U.K., in 1994. He is working at the School of Computing and Mathematica Sciences, University of Leicester. His main research interests include knowledge modeling and creative computing. He has published over 500 papers. He became a Golden Core Member of the IEEE Computer Society in 2010.



**Xiaochun Zhang** (Member, IEEE) received her Ph.D. degree from the School of Computer Science and Technology, Nanjing University of Science and Technology in 2014. She is currently working in the school of Management Science and Computer, Anhui University of Finance & Economics, China. Her research interests include the Computer Vision, Time Series Prediction, and Dialogue System.