NEW AFFINE INVARIANT ENSEMBLE SAMPLERS AND THEIR DIMENSIONAL SCALING

YIFAN CHEN

Abstract. We introduce some new affine invariant ensemble samplers that are easy to construct and improve upon existing widely used algorithms, especially for high-dimensional problems. Specifically, we propose a derivative-free ensemble side move sampler that performs favorably compared to popular samplers in the `emcee` package. Additionally, we develop a class of derivative-based ensemble Hamiltonian Monte Carlo (HMC) samplers with affine invariance, which outperform standard HMC without affine invariance when sampling highly skewed distributions. We provide asymptotic scaling analysis for high-dimensional Gaussian targets to further elucidate the properties of these affine invariant ensemble samplers. In particular, with derivative information, the affine invariant ensemble HMC can scale much better with dimension compared to derivative-free ensemble samplers.

Contents

1. Introduction

The concept of affine invariance [26] has played an important role in the development of efficient Markov chain Monte Carlo (MCMC) samplers as it provides robustness against anisotropy. Samplers are called affine invariant if they behave consistently across all coordinate systems related through affine transformations. In MCMC, affine invariance is typically achieved through ensemble samplers.

Affine invariant ensemble samplers have proven effective for routine Bayesian inference applications [20], but they may perform poorly in high-dimensional problems [30, 7]. This particularly applies to derivative-free samplers such as the stretch move sampler [26], widely used via the `emcee` package [20]. Indeed, compelling arguments

Courant Institute, New York University, NY, USA
E-mail address: `yifan.chen@nyu.edu`.

suggest that "ensemble methods are doomed to fail in high dimensions" [7], based on insightful observations about typical sets: high-dimensional distributions typically concentrate on thin shells, and the interpolation or extrapolation between two points in the stretch move—as well as in many other ensemble samplers—is unlikely to fall within this shell. As a result, small step sizes must be used, and the samplers effectively "devolve into random walks with poorly biased directional choices."

In this paper, we introduce new affine invariant ensemble samplers and analyze their scaling behavior with dimension. We show that ensemble methods can, in fact, achieve the same high-dimensional scaling as their single-chain counterparts, for both derivative-free and derivative-based samplers. In particular, derivative-based ensemble samplers can break the random walk behavior and scale more favorably with dimension. Thanks to affine invariance, the hidden constants in the scaling are also insensitive to the problem's condition number.

1.1. **This work.**

1.1.1. *Derivative-free ensemble samplers.* We first propose a derivative-free side move sampler that improves upon the stretch move by adopting a more favorable directional proposal in high dimensions. For Gaussian targets, we analyze the acceptance probability, which suggests that the optimal step size parameters (denoted $a-1$ for the stretch move and σ for the side move) should scale as $d^{-1/2}$. This scaling matches that of the single-chain random walk [23, 51]. We further show that the expected squared jumping distance [41, 2, 46, 51] under the optimal step size is larger for the side move than for the stretch move. Our numerical experiments show that the autocorrelation times of both samplers scale linearly with d, and that the side move outperforms the stretch move by a factor of two or more across various $\sim$100-dimensional examples.

1.1.2. *Derivative-based ensemble samplers.* Beyond derivative-free samplers, it is well known that incorporating derivative information allows MCMC samplers to scale more favorably with dimension [45, 3]. In particular, the Hamiltonian Monte Carlo (HMC) sampler [39] can overcome random walk behavior and achieves state-of-the-art $d^{-1/4}$ scaling for the step size.

Whether it is possible to develop an efficient, affine invariant ensemble HMC sampler has remained an open question. We propose a new class of samplers in this vein, one of which achieves $d^{-1/4}$ scaling for step size on high-dimensional Gaussian targets. Importantly, our method employs a specific antisymmetric preconditioning strategy based on complementary ensembles to enable efficient parallelism — rather than relying on traditional mass matrices for preconditioning, as in standard HMC.

Our experiments demonstrate that affine invariant HMC samplers can outperform derivative-free samplers by an order of magnitude and achieve 10- to 100-fold reductions in autocorrelation time compared to standard HMC (without elaborate tuning) when sampling from $\sim$100-dimensional synthetic anisotropic distributions and distributions arising from stochastic PDEs, which are typically ill-conditioned.

Consequently, while ensemble samplers based on derivative-free interpolation or extrapolation inevitably devolve into random walks and may struggle in high dimensions, ensemble samplers that incorporate HMC's derivative information avoid this limitation and are therefore not doomed to fail in high-dimensional settings.

1.2. **Related work on affine invariant samplers.** The concept of affine invariance, introduced to MCMC samplers by [26], draws inspiration from the empirical success of the Nelder-Mead simplex algorithm [40] in optimization. This idea has been extended to many other sampling and related areas, including data assimilation [42], annealed importance sampling [11], and variational inference [12], to list a few.

The stretch move [26], which is related to the scaling-invariant t-walk move proposed in [15], remains one of the most widely used affine invariant ensemble samplers. Popularized through the `emcee` package [20], it has been extensively adopted in the astrophysics community. The stretch move is most effective in moderate dimensions (e.g., $d \leq 20$) but must be used cautiously for high-dimensional problems [30]. To address high-dimensional or infinite-dimensional problems, particularly in PDEs and inverse problems, researchers have developed hybrid approaches that combine function space MCMC [16] with ensemble samplers [17, 19].

The stretch move is closely related to the snooker algorithm [24, 44], which uses ensembles to enable adaptive directional sampling. The ensemble covariance has also been used to design derivative-free affine invariant samplers, such as the walk move [26]. Another well-known ensemble sampler is differential evolution MCMC [6, 50], which is inspired by the differential evolution optimization algorithm [49]. The affine invariant side move we introduce in this paper is connected to both the walk move and differential evolution MCMC, as we detail in Section 3.

Beyond derivative-free samplers, derivative-based affine invariant samplers have been developed, including those based on Riemannian geometry [25] and Newton-type directions [38, 48, 18, 14]. Affine invariance can also be achieved by combining ensemble covariance preconditioning with gradient-based directions [27, 33, 21, 22, 35, 12]. These methods establish mathematical links to gradient flows and ensemble Kalman filters via derivative-free approximations [21, 22, 12]. However, most of these approaches do not incorporate Metropolis adjustments and are therefore biased. Our focus in this paper is on adjusted, unbiased MCMC samplers.

Among unbiased derivative-based MCMC samplers, HMC achieves the state-of-the-art scaling with dimension [3]. However, HMC often struggles with anisotropic or ill-conditioned target distributions and typically requires adaptation of the mass matrix to correct scale imbalances, as it is not inherently affine invariant. Metrics based on Riemannian manifolds and Hessians have been used to make HMC affine invariant [25, 32, 31], but these approaches generally require costly second-order information. We believe that systematically developing affine invariant ensemble HMC methods — based solely on first-order information — represents a promising algorithmic opportunity to advance the state of ensemble samplers.

1.3. **Organization.** We review the affine invariance concept in Section 2. We introduce the side move sampler in Section 3 and provide scaling analysis in Section 3.5. Section 4 discusses the affine invariant HMC sampler, followed by scaling analysis in Section 4.5. We present numerical experiments in Section 5 and conclude in Section 6. Appendix A includes pseudocode for all algorithms, while Appendices B and C contain technical proofs.

2. Affine Invariance

We review the mathematical concept of affine invariance. Following the seminal work of Goodman and Weare [26], we express a general MCMC sampler as

$$\mathbf{x}(m+1) = R(\mathbf{x}(m), \pi)$$

where R denotes a mapping at iteration $m \in \mathbb{N}$ that is random. This randomness typically depends on the random numbers generated during the proposal step and in the accept-reject procedure. The sampler is affine invariant if, for any invertible affine transformation $\mathbf{y} = \phi(\mathbf{x}) = A\mathbf{x} + \mathbf{b}$, the following equality holds for a given realization of the mapping R (i.e., by fixing the random number seeds):

$$\mathbf{y}(m+1) = R(\mathbf{y}(m), \phi\#\pi)\,. \tag{2.1}$$

Here, $\phi\#\pi$ denotes the push-forward density of π under ϕ, defined by

$$(\phi\#\pi)(\mathbf{y}) = |\det A|^{-1}\pi(\phi^{-1}(\mathbf{y}))\,.$$

Equation (2.1) indicates that the transformed sequence $\{\mathbf{y}(m)\}$ coincides with what would be obtained by applying the same MCMC algorithm to the transformed density $\phi\#\pi$ with initial value $\mathbf{y}(0)$. Consequently, the convergence behavior of $\mathbf{x}(m)$ toward sampling π matches that of $\mathbf{y}(m)$ toward sampling $\phi\#\pi$. In particular, the sampler inherits the convergence efficiency of the optimally preconditioned coordinate system that minimizes anisotropy through affine transformation.

Affine invariance in MCMC has been widely achieved through ensemble samplers. The concept extends naturally to the ensemble setting. We denote the positions of an ensemble of particles, or walkers, by $(\mathbf{x}_1(m), \mathbf{x}_2(m), \ldots, \mathbf{x}_N(m))$ at discrete time step $m \in \mathbb{N}$, with $m = 0$ representing the initial configuration. We can view the ensemble sampler as operating in the extended space $\mathbb{R}^{dN}$, targeting the product distribution π^N, which is the product of N independent copies of π. Similarly, we denote a general ensemble MCMC sampler as

$$(\mathbf{x}_1(m+1), \mathbf{x}_2(m+1), \ldots, \mathbf{x}_N(m+1)) = R(\mathbf{x}_1(m), \mathbf{x}_2(m), \ldots, \mathbf{x}_N(m), \pi)\,.$$

This sampler is affine invariant if, for any invertible affine transformation $\mathbf{y} = \phi(\mathbf{x}) = A\mathbf{x} + \mathbf{b}$ such that

$$(\mathbf{x}_1, \ldots, \mathbf{x}_N) \xrightarrow{\phi} (\mathbf{y}_1, \ldots, \mathbf{y}_N) = (A\mathbf{x}_1 + \mathbf{b}, \ldots, A\mathbf{x}_N + \mathbf{b})\,,$$

it holds that

$$(\mathbf{y}_1(m+1), \mathbf{y}_2(m+1), \ldots, \mathbf{y}_N(m+1)) = R(\mathbf{y}_1(m), \mathbf{y}_2(m), \ldots, \mathbf{y}_N(m), \phi\#\pi)\,.$$

3. Derivative-free Side Move Sampler

3.1. **Basic side move.** We denote the positions of an ensemble at discrete time step $m \in \mathbb{N}$ as $(\mathbf{x}_1(m), \mathbf{x}_2(m), ..., \mathbf{x}_N(m))$, with $m = 0$ representing the initial configuration. We require $N > d$ and that the N particles span the full space $\mathbb{R}^d$.

At each time step, our ensemble *side move* sampler randomly selects one particle $\mathbf{x}_i(m)$ and two distinct particles $\mathbf{x}_j(m)$ and $\mathbf{x}_k(m)$ from the ensemble which are different

from $\mathbf{x}_i(m)$. The sampler proposes the following *side move* (see an illustration in Figure 1) for the i-th particle:

$$\tilde{\mathbf{x}}_i(m+1) = \mathbf{x}_i(m) + \sigma(\mathbf{x}_j(m) - \mathbf{x}_k(m))\xi\,, \tag{3.1}$$

where σ is a user-specified scalar parameter, and $\xi \sim \mathcal{N}(0,1)$ is drawn from a normal distribution. We will discuss connections to walk move [26] and differential evolution [49, 6, 50] in Sections 3.3 and 3.4. In Section 3.5, we study the high dimensional scaling behavior, which suggests a choice of $\sigma = 1.687d^{-1/2}$.

The above proposal for the i-th particle is accepted or rejected according to the standard Metropolis criterion. Since for fixed j and k, the proposal is symmetric for the i-th particle, we get a simple acceptance probability:

$$\text{prob} = \min\{1, \frac{\pi(\tilde{\mathbf{x}}_i(m+1))}{\pi(\mathbf{x}_i(m))}\}\,. \tag{3.2}$$

With this probability, we update $\mathbf{x}_i(m+1) = \tilde{\mathbf{x}}_i(m+1)$, otherwise the proposal is rejected and we set $\mathbf{x}_i(m+1) = \mathbf{x}_i(m)$. All other particles (those other than the i-th) remain unchanged during this step.

The update preserves π^N as an invariant distribution. In fact, this update can be viewed as a Metropolis-within-Gibbs approach to sample from π^N: at each step, we propose an update that satisfies detailed balance for the conditional distribution of one particular particle, given the positions of all other particles in the ensemble.

The affine invariance property of the algorithm is apparent since it is based on relative positions of the particles. We include a derivation here for completeness. Given an invertible affine transformation $\phi(\mathbf{x}) = A\mathbf{x} + \mathbf{b}$, we transform the initial ensemble

$$(\mathbf{x}_1, ..., \mathbf{x}_N) \overset{\phi}{\to} (\mathbf{y}_1, ..., \mathbf{y}_N) = (A\mathbf{x}_1 + \mathbf{b}, ..., A\mathbf{x}_N + \mathbf{b})\,. \tag{3.3}$$

Using the side move algorithm to sample $\phi\#\pi$ with the transformed initial ensemble, we get the proposal (assuming the same random number generator and seeds)

$$\tilde{\mathbf{y}}_i(m+1) = \mathbf{y}_i(m) + \sigma(\mathbf{y}_j(m) - \mathbf{y}_k(m))\xi = A\tilde{\mathbf{x}}_i(m+1) + \mathbf{b}\,.$$

Moreover, by the change of variables, the acceptance ratio remains the same as the untransformed case:

$$\frac{\pi(\tilde{\mathbf{x}}_i(m+1))}{\pi(\mathbf{x}_i(m))} = \frac{(\phi\#\pi)(\tilde{\mathbf{y}}_i(m+1))}{(\phi\#\pi)(\mathbf{y}_i(m))}\,.$$

Therefore, we obtain that for any $m \in \mathbb{N}$, it holds

$$(\mathbf{y}_1(m), ..., \mathbf{y}_N(m)) = (A\mathbf{x}_1(m) + \mathbf{b}, ..., A\mathbf{x}_N(m) + \mathbf{b})\,,$$

which justifies the affine invariance property.

3.2. **Parallel side move sampler.** The side move algorithm can be efficiently parallelized. However, this needs to be done carefully to avoid violating detailed balance. Our approach is similar to that used in the `emcee` package [20]. Specifically, we divide the ensemble into two groups:

$$S^{(0)} = \{\mathbf{x}_1, ..., \mathbf{x}_{N/2}\}, \quad S^{(1)} = \{\mathbf{x}_{N/2+1}, ..., \mathbf{x}_N\}\,. \tag{3.4}$$

At each time step, for each particle in $S^{(0)}$, we randomly select two particles from the complementary set $S^{(1)}$ and perform the side move, applying the Metropolis accept-reject criterion. Then, we follow the same procedure for particles in $S^{(1)}$, selecting particles from the complementary set $S^{(0)}$ to form the side moves and perform Metropolis. The key insight is that we never use particles from the same group as "sides" when updating particles within that group, and when we update particles in one group, the particles in another complementary group are treated fixed. This approach preserves the correct detailed balance condition so the parallel version of the sampler keeps π^N invariant. We consistently adopt this parallel approach in this paper.

3.3. **Comparison to other affine invariant moves.** One of the most popular affine invariant sampler is based on *stretch move* [26]. In the basic stretch move, at step m, one proposes

$$\tilde{\mathbf{x}}_i(m) = \mathbf{x}_j(m) + Z(\mathbf{x}_i(m) - \mathbf{x}_j(m))\,,$$

where the density of the scaling variable Z satisfies $g(1/z) = zg(z)$, with a commonly used example

$$g(z) \propto \begin{cases} \frac{1}{\sqrt{z}} & \text{if } z \in \left[\frac{1}{a}, a\right] \\ 0 & \text{otherwise} \end{cases} \tag{3.5}$$

where a is the stretch parameter with a recommended default value $a = 2$ (see [20]). To maintain detailed balance, the proposal is accepted with probability

$$\text{prob} = \min\{1, Z^{d-1}\frac{\pi(\tilde{\mathbf{x}}_i(m+1))}{\pi(\mathbf{x}_i(m))}\}\,.$$

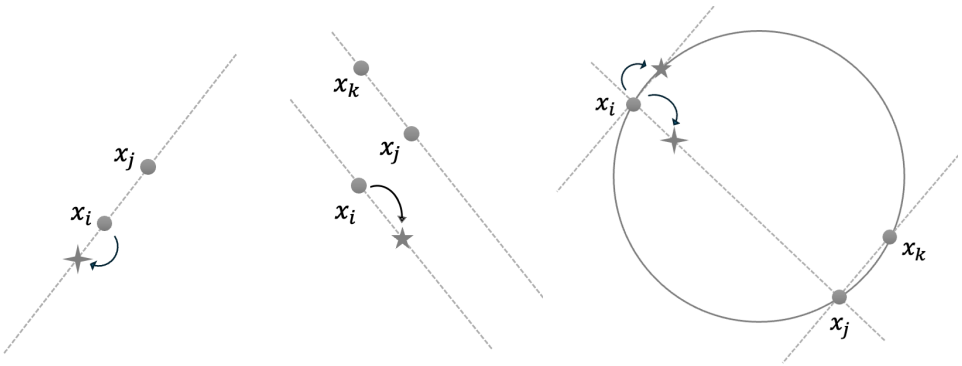

FIGURE 1. Demonstration of stretch move and side move. Left: stretch move to the four-pointed star; middle: side move to the five-pointed star; right: both moves when points are on a circle

We illustrate the stretch move and side move concepts in Figure 1. For the stretch move, the new point is positioned along the line formed by $\mathbf{x}_i$ and $\mathbf{x}_j$. In the side move, two distinct particles $\mathbf{x}_j$ and $\mathbf{x}_k$ form a reference side, and $\mathbf{x}_i$ moves in a direction parallel to this side. The effectiveness of each approach depends on the underlying distribution. In the right panel of Figure 1, we illustrate a scenario where points lie on

a circle. Here, for the given selection of points, the side move approximately follows the tangential direction, while the stretch move proposes points far from the circle.

In Section 3.5, we provide quantitative high-dimensional scaling analysis for an isotropic Gaussian target. It shows that the side move leads to improved jumped distance compared to the stretch move. The insight is that high-dimensional isotropic Gaussians concentrate on a thin shell, and the inner product between the side direction $\mathbf{x}_j(m)-\mathbf{x}_k(m)$ and $\mathbf{x}_i(m)$ is small with high probability, so the side move proposal typically goes in the favorable tangential direction, in a similar spirit as the right panel of Figure 1.

We note that other affine invariant moves based on empirical covariance exist, such as the *walk move* [26]. In the walk move, we select a subset S of particles which are different from $\mathbf{x}_i(m)$. Denote the empirical mean by $\mathbf{m}_S$. The proposal is

$$\tilde{\mathbf{x}}_i(m+1) = \mathbf{x}_i(m) + \frac{1}{\sqrt{|S|}}\sum_{j\in S}(\mathbf{x}_j(m) - \mathbf{m}_S)\xi_j\,, \tag{3.6}$$

where $\xi_j \in \mathcal{N}(0,1)$ are independent normal random variables. This corresponds to a Gaussian proposal with covariance equal to the empirical covariance of particles in S.

The empirical covariance accounts for global features of the distribution if $|S|$ is large. As discussed in [33, 43], local features could be more representative for certain distributions, which may explain the wider popularity of the stretch move in the literature [20]. We note that when $|S| = 2$, the walk move can be shown to be equivalent to the side move with a specific step size. This can be demonstrated by noting:

$$\frac{1}{\sqrt{|S|}}\sum_{j\in S}(\mathbf{x}_j(m) - \mathbf{m}_S)\xi_j = \frac{1}{2\sqrt{2}}(\mathbf{x}_j(m) - \mathbf{x}_k(m))(\xi_j - \xi_k) \tag{3.7}$$

for $S = \{\mathbf{x}_j, \mathbf{x}_k\}$ and $\xi_j - \xi_k \sim \mathcal{N}(0,2)$.

3.4. **Connection to differential evolution.** Affine invariant ensemble MCMC [26] was motivated by optimization algorithms. The side move is also connected to methods in the optimization literature, specifically the differential evolution algorithm [49], where particle differences are used to guide the exploration of the function. Differential evolution MCMC [6, 50] has been developed and has found numerous successful applications. In detail, differential evolution MCMC employs the proposal

$$\tilde{\mathbf{x}}_i(m+1) = \mathbf{x}_i(m) + \gamma(\mathbf{x}_j(m) - \mathbf{x}_k(m)) + \sigma\xi\,,$$

where γ, σ are scalars and $\xi \sim \mathcal{N}(0, I_{d\times d})$. The recommended choice for γ is $\gamma = \frac{2.38}{\sqrt{2d}}$. We note that differential evolution MCMC is not generally affine invariant.

3.5. **Analysis of high dimensional scaling.** We study the high dimensional scaling of the algorithm for Gaussians at the stationary phase. Since the algorithm is affine invariant, we consider isotropic Gaussians without loss of generality. The analysis suggests σ (in side move) and $a-1$ (in stretch move) to scale with $d^{-1/2}$ in high dimensions. We also study the limit of expected squared jumped distance [41, 2, 46, 51], which has been used to derive optimal parameters of MCMC. The proof of this proposition can be found in Appendix B.

Proposition 3.1. *Consider an isotropic Gaussian in d dimensions*

$$\pi(\mathbf{x}) \propto \exp(-\frac{1}{2}\mathbf{x}^T\mathbf{x}),$$

where $\mathbf{x} \in \mathbb{R}^d$. Under the ideal assumption that $\mathbf{x}_i(m), \mathbf{x}_j(m), \mathbf{x}_k(m)$ are all independent draws from this target distribution, the following holds almost surely.

- *For the side move, $\tilde{\mathbf{x}}_i(m+1) = \mathbf{x}_i(m) + \sigma(\mathbf{x}_j(m) - \mathbf{x}_k(m))\xi$, if $\sigma = \frac{\alpha}{\sqrt{d}}$, then we have the following limit of the expected acceptance probability*

$$\lim_{d\to\infty} \mathbb{E}[\min\{1, \frac{\pi(\tilde{\mathbf{x}}_i(m+1))}{\pi(\mathbf{x}_i(m))}\}] = \mathbb{E}[\min\{1, \exp(-\alpha^2\xi^2 - \sqrt{2}\alpha\xi z)\}],$$

and the expected squared jumped distance

$$\lim_{d\to\infty} \mathbb{E}[\|\mathbf{x}_i(m+1) - \mathbf{x}_i(m)\|_2^2] = 2\alpha^2\mathbb{E}[\xi^2 \min\{1, \exp(-\alpha^2\xi^2 - \sqrt{2}\alpha\xi z\}],$$

where $\xi \sim \mathcal{N}(0,1)$ is independent of $z \sim \mathcal{N}(0,1)$.

- *For the stretch move, $\tilde{\mathbf{x}}_i(m+1) = \mathbf{x}_j(m) + Z(\mathbf{x}_i(m) - \mathbf{x}_j(m))$, if $a = 1 + \frac{\beta}{\sqrt{d}}$ in (3.5), then we have the following limit of the expected acceptance probability*

$$\lim_{d\to\infty} \mathbb{E}[\min\{1, Z^{d-1}\frac{\pi(\tilde{\mathbf{x}}_i(m+1))}{\pi(\mathbf{x}_i(m))}\}] = \mathbb{E}[\min\{1, \exp(-\frac{3}{2}\beta^2U^2 - \sqrt{3}\beta U z)\}],$$

and the expected squared jumped distance

$$\lim_{d\to\infty} \mathbb{E}[\|\mathbf{x}_i(m+1) - \mathbf{x}_i(m)\|_2^2] = 2\beta^2\mathbb{E}[U^2 \min\{1, \exp(-\frac{3}{2}\beta^2U^2 - \sqrt{3}\beta U z)\}],$$

where $U \sim \text{Unif}[-1,1]$ is independent of $z \sim \mathcal{N}(0,1)$.

The scaling of $\sigma \sim d^{-1/2}$ for the side move is natural since it resembles a basic random walk, aligning with existing scaling results for non-affine-invariant random walk Metropolis [23, 51]. The above result applies to arbitrary Gaussians given the affine invariance property of the side move. The scaling $a - 1 \sim d^{-1/2}$ for the stretch move is perhaps less obvious. The presence of the Z^{d-1} factor is the key element leading to such scaling.

The proposition demonstrates that affine invariant ensemble samplers lead to the same high-dimensional scaling as single-chain random walk (at least for Gaussian targets). However, the ensemble methods maintain consistent performance for highly anisotropic distributions, while basic random walk without affine invariance does not.

With Proposition 3.1, we can perform simulations to find the optimal α and β that lead to the largest expected squared jumped distance. The results are shown in Figure 2. In three decimal numbers, we found that the optimal α is approximately 1.687 with a squared jumped distance of 0.744, and the optimal β is approximately 2.151 with a squared jumped distance of 0.584. The corresponding acceptance probabilities are 0.443 and 0.416, respectively. With these parameters, the side move can achieve approximately $0.744/0.584 \approx 1.27$ times the squared jumped distance compared to the stretch move.

The random walk scaling shows that the sampler requires $O(d)$ steps to traverse the distribution's support. We conduct numerical experiments in Section 5.2 for Gaussian targets using the optimal parameters described above. These experiments demonstrate

that the autocorrelation time scales as $O(d)$. The side move achieves a smaller constant in the $O(d)$ scaling of autocorrelation time compared to the stretch move.

Remark 3.2. We note that using a different distribution for ξ in the side move results in different optimal parameters. For example, if we choose $\xi \sim \text{Unif}[-1, 1]$, by examining the formula, we find that the optimal α and β are related by $\alpha = \sqrt{\frac{3}{2}}\beta$. With this choice, the side move achieves exactly 1.5 times the expected squared jumped distance compared to the stretch move. ◊

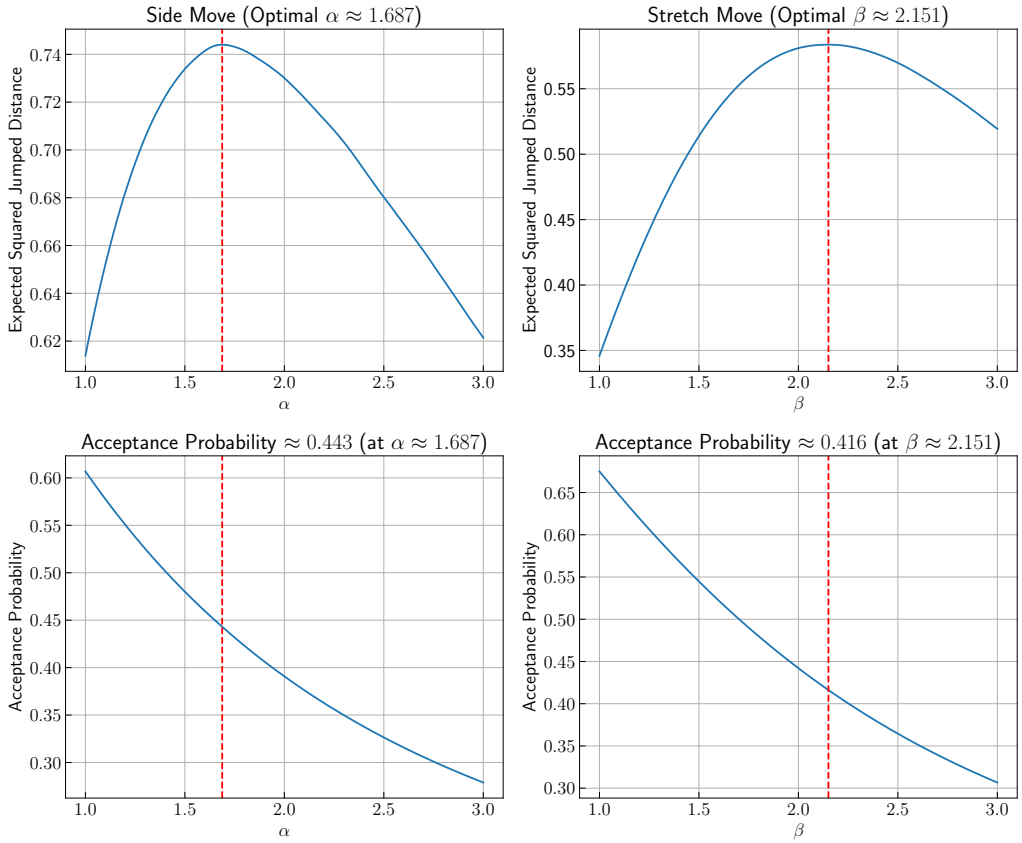

FIGURE 2. Side move versus stretch move: expected acceptance rate and squared expected jumped distance. Optimal α and β in terms of the squared expected jumped distance are marked.

4. Derivative-based Affine Invariant HMC Samplers

The samplers in the previous section are derivative-free, making them simple, convenient, and widely used in applications. However, when derivative information is available, derivative-based samplers such as Langevin [47] and Hamiltonian Monte Carlo (HMC) [39] have been shown to scale better with dimension [45, 3].

There has been considerable interest in developing affine invariant Langevin samplers based on Riemannian geometry and empirical covariance preconditioning [25, 27, 33, 21, 22]. To the best of our knowledge, it remains unknown whether an adjusted ensemble HMC that is affine invariant can be efficiently developed. Since HMC is one of the state-of-the-art samplers that overcomes random walk-like behavior and scales favorably with dimension, developing an affine invariant ensemble HMC is important for further understanding and advancement of affine invariant samplers. This section aims to discuss the design principles and propose new samplers of this kind.

4.1. **Hamiltonian Monte Carlo.** Let $\pi \propto \exp(-V)$ where $V : \mathbb{R}^d \to \mathbb{R}$ is a potential function. In HMC, we consider a probability density $\exp(-V(\mathbf{x}) - \frac{1}{2}\mathbf{p}^T M^{-1}\mathbf{p})$ over an extended state space $(\mathbf{x}, \mathbf{p})$ where $\mathbf{x}, \mathbf{p} \in \mathbb{R}^d$ and $\mathbf{p}$ is often referred to as the momentum vector; M is the called the mass matrix which often plays the role of preconditioning.

The following Hamiltonian dynamics keep the joint distribution invariant

$$\frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t} = M^{-1}\mathbf{p}, \quad \frac{\mathrm{d}\mathbf{p}}{\mathrm{d}t} = -\nabla V(\mathbf{x})\,.$$

The standard HMC sampler alternates between two operations: Gibbs sampling to refresh the momentum $\mathbf{p} \sim \mathcal{N}(0, M)$ and running approximate Hamiltonian dynamics followed by momentum negation with Metropolis correction. Since both operations individually preserve the invariant distribution, their composition does as well.

More specifically, let L_h be the deterministic map that performs one leapfrog step for the Hamiltonian dynamics with time step size h. That is, $(\mathbf{x}_h, \mathbf{p}_h) = \mathsf{L}_h(\mathbf{x}, \mathbf{p})$ satisfies

$$\mathbf{p}_{h/2} = \mathbf{p} - \frac{h}{2}\nabla V(\mathbf{x}), \quad \mathbf{x}_h = \mathbf{x} + hM^{-1}\mathbf{p}_{h/2}, \quad \mathbf{p}_h = \mathbf{p}_{h/2} - \frac{h}{2}\nabla V(\mathbf{x}_h)\,. \tag{4.1}$$

We denote P as the momentum flip operator such that $\mathsf{P}(\mathbf{x}, \mathbf{p}) = (\mathbf{x}, -\mathbf{p})$. Then, each single iteration of HMC goes as follows:

- Sample a momentum $\mathbf{p} \sim \mathcal{N}(0, M)$.
- Propose an update $(\tilde{\mathbf{x}}, \tilde{\mathbf{p}}) = \mathsf{PL}_h^n(\mathbf{x}, \mathbf{p})$ where n is the number of leapfrog steps taken; such proposal is accepted with probability

$$\text{prob} = \min\left\{1, \exp(-V(\tilde{\mathbf{x}}) - \frac{1}{2}\tilde{\mathbf{p}}^T M^{-1}\tilde{\mathbf{p}} + V(\mathbf{x}) + \frac{1}{2}\mathbf{p}^T M^{-1}\mathbf{p})\right\}.$$

The key to the derivation of HMC is the property that $\mathsf{PL}_h^n = (\mathsf{PL}_h^n)^{-1}$ so the detailed balance for the second step is valid. Modern development of HMC focuses on tuning the stepsize and integration time [28, 8], among many others.

4.2. **Covariance preconditioning and its challenge.** How shall we develop an ensemble HMC sampler that is affine invariant? The simplest idea is to use empirical covariance as the inverse mass matrix M for preconditioning. To implement this idea, let us follow the ensemble splitting strategy in Section 3.2 that is easy to parallelize and works better with detailed balance. Consider an ensemble of N particles $\mathbf{x}_1, ..., \mathbf{x}_N$ split into two groups

$$S^{(0)} = \{\mathbf{x}_1, ..., \mathbf{x}_{N/2}\}, \quad S^{(1)} = \{\mathbf{x}_{N/2+1}, ..., \mathbf{x}_N\}\,, \tag{4.2}$$

and the joint distribution

$$\exp\left(-\sum_{i=1}^{N/2}\left(V(\mathbf{x}_i)+\frac{1}{2}\mathbf{p}_i^T\mathrm{Cov}_{S^{(1)}}\mathbf{p}_i\right)-\sum_{i=N/2+1}^{N}\left(V(\mathbf{x}_i)+\frac{1}{2}\mathbf{p}_i^T\mathrm{Cov}_{S^{(0)}}\mathbf{p}_i\right)\right),$$

where $\mathbf{p}_i\in\mathbb{R}^d$ is the associated momentum vector with $\mathbf{x}_i$. Here $\mathrm{Cov}_{S^{(0)}}$ and $\mathrm{Cov}_{S^{(1)}}$ are the empirical covariance matrices of particles in $S^{(0)}$ and $S^{(1)}$. The formula is

$$\mathrm{Cov}_{S^{(0)}}=\frac{2}{N}\sum_{i=1}^{N/2}(\mathbf{x}_i-\mathbf{m}_{S^{(0)}})(\mathbf{x}_i-\mathbf{m}_{S^{(0)}})^T$$

where $\mathbf{m}_{S^{(0)}}$ is the empirical mean for particles in $S^{(0)}$.

We note that when marginalized over all $\mathbf{p}_i$, the above joint distribution does not give the correct marginal distribution for $\mathbf{x}_i$ due to the existence of normalization constants. In fact, we need to additionally add $-\frac{N}{4}\log\det\mathrm{Cov}_{S^{(0)}}-\frac{N}{4}\log\det\mathrm{Cov}_{S^{(1)}}$ to the potential to address this issue. The modified final joint distribution is

$$\pi_\star^N(\mathbf{x}_1,...,\mathbf{x}_N,\mathbf{p}_1,...\mathbf{p}_N)\propto\exp(-V_\star(\mathbf{x}_1,...,\mathbf{x}_N,\mathbf{p}_1,...\mathbf{p}_N)) \tag{4.3}$$

where

$$\begin{aligned}V_\star(\mathbf{x}_1,...,\mathbf{x}_N,\mathbf{p}_1,...\mathbf{p}_N)=&\sum_{i=1}^{N/2}\left(V(\mathbf{x}_i)+\frac{1}{2}\mathbf{p}_i^T\mathrm{Cov}_{S^{(1)}}\mathbf{p}_i\right)\\&+\sum_{i=N/2+1}^{N}\left(V(\mathbf{x}_i)+\frac{1}{2}\mathbf{p}_i^T\mathrm{Cov}_{S^{(0)}}\mathbf{p}_i\right)\\&-\frac{N}{4}\log\det\mathrm{Cov}_{S^{(0)}}-\frac{N}{4}\log\det\mathrm{Cov}_{S^{(1)}}.\end{aligned} \tag{4.4}$$

A direct calculation shows that the gradient of the added potential is

$$\nabla_{\mathbf{x}_i}\log\det\mathrm{Cov}_{S^{(0)}}=\mathrm{Cov}_{S^{(0)}}^{-1}\nabla_{\mathbf{x}_i}\mathrm{Cov}_{S^{(0)}}=\frac{4}{N}\mathrm{Cov}_{S^{(0)}}^{-1}(\mathbf{x}_i-\mathbf{m}_{S^{(0)}}).$$

Thus, using the HMC pipeline, we obtain the following algorithm:

- We fix particles in $S^{(1)}$, and update particles in $S^{(0)}$: for each $1\le i\le N/2$, we sample $\mathbf{p}_i\sim\mathcal{N}(0,\mathrm{Cov}_{S^{(1)}}^{-1})$, run leapfrog approximation of the dynamics

$$\frac{\mathrm{d}\mathbf{x}_i}{\mathrm{d}t}=\mathrm{Cov}_{S^{(1)}}\mathbf{p}_i,\quad\frac{\mathrm{d}\mathbf{p}_i}{\mathrm{d}t}=-\nabla V(\mathbf{x}_i)+\mathrm{Cov}_{S^{(0)}}^{-1}(\mathbf{x}_i-\mathbf{m}_{S^{(0)}}),$$

  and apply the Metropolis accept-reject criterion.
- Then we fix particles in $S^{(0)}$, and update particles in $S^{(1)}$ in a similar fashion.
- Iterate the above two steps.

We can show the algorithm is affine invariant, thanks to the covariance preconditioning. In fact, given an invertible affine transformation $\phi(\mathbf{x})=A\mathbf{x}+\mathbf{b}$, we can transform

$$\begin{aligned}&(\mathbf{x}_1,...,\mathbf{x}_N)\to(\mathbf{y}_1,...,\mathbf{y}_N)=(A\mathbf{x}_1+\mathbf{b},...,A\mathbf{x}_N+\mathbf{b})\\&(\mathbf{p}_1,...,\mathbf{p}_N)\to(\mathbf{r}_1,...,\mathbf{r}_N)=(A^{-T}\mathbf{p}_1,...,A^{-T}\mathbf{p}_N).\end{aligned} \tag{4.5}$$

By the change of variables, we get (for $1 \le i \le N/2$)

$$\frac{\mathrm{d}\mathbf{y}_i}{\mathrm{d}t} = \mathrm{Cov}_{S_{\mathbf{y}}^{(1)}}\mathbf{r}_i, \quad \frac{\mathrm{d}\mathbf{r}_i}{\mathrm{d}t} = -\nabla V^{\phi}(\mathbf{y}_i) + \mathrm{Cov}^{-1}_{S_{\mathbf{y}}^{(0)}}(\mathbf{y}_i - \mathbf{m}_{S_{\mathbf{y}}^{(0)}}),$$

where $V^{\phi}(\mathbf{y}) = V(\phi^{-1}(\mathbf{y}))$ corresponds to the potential of the transformed density $\phi\#\pi$, and $S_{\mathbf{y}}^{(0)}$ is the first group of transformed particles. One can further show that the above correspondence holds for the leapfrog scheme, and the final acceptance ratio remains the same for the original and transformed dynamics. This demonstrates the affine invariance property.

However, we note that here the $N/2$ particles in a group will need to be accepted or rejected *at the same time*, as they are coupled through the log det term. The potential simultaneous rejection can be wasteful and inefficient. A different approach is therefore needed to make an efficient ensemble HMC with affine invariance.

4.3. **Hamiltonian walk move sampler.** Our discussion in the previous subsection suggests it is preferable to decouple the momentum for different particles in the joint distribution. Now consider the simple joint distribution with identity mass matrices:

$$\exp\left(-\sum_{i=1}^{N/2}\left(V(\mathbf{x}_i) + \frac{1}{2}\mathbf{p}_i^T\mathbf{p}_i\right) - \sum_{i=N/2+1}^{N}\left(V(\mathbf{x}_i) + \frac{1}{2}\mathbf{p}_i^T\mathbf{p}_i\right)\right),$$

where again, the particles are split into two groups

$$S^{(0)} = \{\mathbf{x}_1, ..., \mathbf{x}_{N/2}\}, \quad S^{(1)} = \{\mathbf{x}_{N/2+1}, ..., \mathbf{x}_N\}. \tag{4.6}$$

Instead of running the standard Hamiltonian dynamics, we note that with a preconditioning matrix $B \in \mathbb{R}^{d\times d}$, the preconditioned dynamics

$$\frac{\mathrm{d}\mathbf{x}_i}{\mathrm{d}t} = B\mathbf{p}_i, \quad \frac{\mathrm{d}\mathbf{p}_i}{\mathrm{d}t} = -B^T\nabla V(\mathbf{x}_i)$$

will also preserve the joint distribution. A similar antisymmetric preconditioning has been explored in [33] for underdamped Langevin dynamics.

Our goal is to select an appropriate B that depends on the ensembles $S^{(0)}$ and $S^{(1)}$ to make the preconditioned Hamiltonian dynamics affine invariant. We must choose B in a way that enables efficient calculation of the acceptance rate. Another important observation is that the dimension of $\mathbf{p}_i$ need not match the dimension of $\mathbf{x}_i$; in such cases, the matrix B is chosen to be rectangular to ensure dimensional consistency.

Based on the above insights, we propose the following algorithm.

- We fix particles in $S^{(1)}$, and update particles in $S^{(0)}$: for each $1 \le i \le N/2$, we sample $\mathbf{p}_i \sim \mathcal{N}(0, I_{N/2\times N/2})$, and run n steps of leapfrog approximation of the dynamics
$$\frac{\mathrm{d}\mathbf{x}_i}{\mathrm{d}t} = B_{S^{(1)}}\mathbf{p}_i, \quad \frac{\mathrm{d}\mathbf{p}_i}{\mathrm{d}t} = -B_{S^{(1)}}^T\nabla V(\mathbf{x}_i),$$
where
$$B_{S^{(1)}} = \frac{1}{\sqrt{N/2}}[\mathbf{x}_{N/2+1} - \mathbf{m}_{S^{(1)}}, ..., \mathbf{x}_N - \mathbf{m}_{S^{(1)}}] \in \mathbb{R}^{d\times N/2} \tag{4.7}$$
is called a normalized centered ensemble for particles in group $S^{(1)}$. Here $\mathbf{m}_{S^{(1)}}$ is the mean of all particles in $S^{(1)}$.

More precisely, denote $\mathsf{L}_{h,B_{S^{(1)}}}$ as the corresponding leapfrog operator. That is, $(\mathbf{x}_h, \mathbf{p}_h) = \mathsf{L}_{h,B_{S^{(1)}}}(\mathbf{x}, \mathbf{p})$ satisfies

$$\mathbf{p}_{h/2} = \mathbf{p} - \frac{h}{2} B_{S^{(1)}}^T \nabla V(\mathbf{x}), \quad \mathbf{x}_h = \mathbf{x} + h B_{S^{(1)}} \mathbf{p}_{h/2}, \quad \mathbf{p}_h = \mathbf{p}_{h/2} - \frac{h}{2} B_{S^{(1)}}^T \nabla V(\mathbf{x}_h)\,. \tag{4.8}$$

We propose $(\tilde{\mathbf{x}}_i, \tilde{\mathbf{p}}_i) = \mathsf{PL}^n_{h,B_{S^{(1)}}}(\mathbf{x}_i, \mathbf{p}_i)$, and we accept this proposal for each $1 \le i \le N/2$ with probability

$$\text{prob}_i = \min\left\{1, \exp(-V(\tilde{\mathbf{x}}_i) - \frac{1}{2}\tilde{\mathbf{p}}_i^T \tilde{\mathbf{p}}_i + V(\mathbf{x}_i) + \frac{1}{2}\mathbf{p}_i^T \mathbf{p}_i)\right\}.$$

- Then, we fix particles in $S^{(0)}$, and update particles in $S^{(1)}$ in a similar fashion. Here the dynamics for particles $N/2+1 \le i \le N$ are

$$\frac{\mathrm{d}\mathbf{x}_i}{\mathrm{d}t} = B_{S^{(0)}} \mathbf{p}_i, \quad \frac{\mathrm{d}\mathbf{p}_i}{\mathrm{d}t} = -B_{S^{(0)}}^T \nabla V(\mathbf{x}_i)\,,$$

with

$$B_{S^{(0)}} = \frac{1}{\sqrt{N/2}}[\mathbf{x}_1 - \mathbf{m}_{S^{(0)}}, ..., \mathbf{x}_{N/2} - \mathbf{m}_{S^{(0)}}] \in \mathbb{R}^{d \times N/2}\,. \tag{4.9}$$

- Iterate the above two steps.

The algorithm is affine invariant. Indeed, given an invertible affine transformation $\phi(\mathbf{x}) = A\mathbf{x} + \mathbf{b}$, we can transform

$$(\mathbf{x}_1, ..., \mathbf{x}_N) \xrightarrow{\phi} (\mathbf{y}_1, ..., \mathbf{y}_N) = (A\mathbf{x}_1 + \mathbf{b}, ..., A\mathbf{x}_N + \mathbf{b}) \tag{4.10}$$

while keeping $\mathbf{p}_i$ untransformed. This differs from the previous subsection where both $\mathbf{x}$ and $\mathbf{p}$ are transformed. However, this difference does not matter because our ultimate goal is to sample the distribution on $\mathbf{x}$, not $\mathbf{p}$. At the continuous level, we obtain

$$\frac{\mathrm{d}\mathbf{y}_i}{\mathrm{d}t} = B_{S^{(1)}_{\mathbf{y}}} \mathbf{p}_i, \quad \frac{\mathrm{d}\mathbf{p}_i}{\mathrm{d}t} = -B_{S^{(1)}_{\mathbf{y}}}^T \nabla V^\phi(\mathbf{y}_i)\,,$$

which represents the preconditioned Hamiltonian dynamics applied to the transformed density $\phi\#\pi$. This relationship extends to the leapfrog discretization of the dynamics. Furthermore, the acceptance ratio remains identical for both the transformed and untransformed cases, thus confirming the method's affine invariance.

We note that the centered ensembles here play a role similar to the Cholesky factor of the empirical covariance matrix. While it is possible to directly use the Cholesky factor for preconditioning, which would result in a momentum vector with the same dimension as the state, using centered ensembles avoids the computational cost of factorization. Similar use of centered ensembles to avoid Cholesky has been considered in [22] for covariance preconditioned Langevin dynamics.

The algorithm may be interpreted as a gradient-enhanced walk move. In one form of the walk move [26], the proposal is $\mathbf{x}_i + B_{S^{(1)}}\mathbf{z}$ with $\mathbf{z} \sim \mathcal{N}(0, I_{N/2 \times N/2})$. In our algorithm, the derivative of $\mathbf{x}_i$ is similarly a linear combination of the columns of $B_{S^{(1)}}$. When the integration time is small, the leading order term follows the same direction as $B_{S^{(1)}}\mathbf{z}$. With longer integration times, our approach adaptively determines the move strength using Hamiltonian-type dynamics that incorporate derivative information of the potential. For this reason, we term the method the *Hamiltonian walk move* sampler.

4.4. **Hamiltonian side move sampler.** In the last subsection, the use of empirical covariance or centered ensembles makes the algorithm affine invariant. This approach accounts for global statistics through covariance. Alternatively, we can use the local side move direction to derive an affine invariant algorithm as follows.

- We fix particles in $S^{(1)}$, and update particles in $S^{(0)}$: for each $1 \le i \le N/2$, we sample $p_i \sim \mathcal{N}(0,1)$ and two particles $\mathbf{x}_j$ and $\mathbf{x}_k$ from $S^{(1)}$. We run n steps of leapfrog approximation of the dynamics

$$\frac{\mathrm{d}\mathbf{x}_i}{\mathrm{d}t} = \frac{1}{\sqrt{2d}}(\mathbf{x}_j - \mathbf{x}_k)p_i, \quad \frac{\mathrm{d}p_i}{\mathrm{d}t} = -\frac{1}{\sqrt{2d}}(\mathbf{x}_j - \mathbf{x}_k)^T \nabla V(\mathbf{x}_i)\,.$$

  More precisely, for the i-th particle, denote $\mathsf{L}_{h,(\mathbf{x}_j-\mathbf{x}_k)/\sqrt{2d}}$ as the corresponding leapfrog operator. That is, $(\mathbf{x}_h, p_h) = \mathsf{L}_{h,(\mathbf{x}_j-\mathbf{x}_k)/\sqrt{2d}}(\mathbf{x}, p)$ satisfies

$$\begin{aligned} p_{h/2} &= p - \frac{h}{2}\left(\frac{\mathbf{x}_j - \mathbf{x}_k}{\sqrt{2d}}\right)^T \nabla V(\mathbf{x}), \\ \mathbf{x}_h &= \mathbf{x} + h\left(\frac{\mathbf{x}_j - \mathbf{x}_k}{\sqrt{2d}}\right) p_{h/2}, \\ p_h &= p_{h/2} - \frac{h}{2}\left(\frac{\mathbf{x}_j - \mathbf{x}_k}{\sqrt{2d}}\right)^T \nabla V(\mathbf{x}_h)\,. \end{aligned} \tag{4.11}$$

  We propose $(\tilde{\mathbf{x}}_i, \tilde{p}_i) = \mathsf{PL}^n_{h,(\mathbf{x}_j-\mathbf{x}_k)/\sqrt{2d}}(\mathbf{x}_i, p_i)$, and we accept the proposal with probability

$$\mathrm{prob}_i = \min\left\{1, \exp(-V(\tilde{\mathbf{x}}_i) - \frac{1}{2}\tilde{p}_i^2 + V(\mathbf{x}_i) + \frac{1}{2}p_i^2)\right\}.$$

- Then, we fix particles in $S^{(0)}$, and update particles in $S^{(1)}$ in a similar fashion. Here for each $N/2+1 \le i \le N$, the corresponding $\mathbf{x}_j$ and $\mathbf{x}_k$ are randomly drawn from $S^{(0)}$.
- Iterate the above two steps.

This approach only needs to calculate directional gradients rather than full gradients. We refer to the algorithm as the *Hamiltonian side move* sampler since it uses precisely the same direction as the standard side move, with the move length adaptively determined by Hamiltonian-type dynamics. Like our previous methods, this approach maintains affine invariance for the same underlying reasons.

4.5. **Analysis of high dimensional scaling.** We study the stepsize scaling in high dimensions for our proposed affine invariant HMC algorithms. As in Section 3.5, we focus on isotropic Gaussian distributions and the stationary phase of the algorithms. We denote by T a fixed time horizon for the Hamiltonian dynamics. All statements below pertain to a single particle $\mathbf{x}_i$ during one iteration of the algorithm. Following our convention from previous sections, we denote $\mathbf{x}_i(m)$ as the state at step m, and $\mathbf{x}_i(m+1)$ as the state in the subsequent iteration after applying the preconditioned Hamiltonian dynamics and the accept-reject mechanism. For notational simplicity, we omit the indices m and $m+1$ when the context is clear. The proof of the proposition can be found in Appendix C.

Proposition 4.1. *Consider an isotropic Gaussian in d dimensions*

$$\pi(\mathbf{x}) \propto \exp(-\frac{1}{2}\mathbf{x}^T\mathbf{x}),$$

where $\mathbf{x} \in \mathbb{R}^d$. Under the ideal assumption that all $\mathbf{x}_i$ are independent draws from this target distribution, the following holds almost surely.

- *For the Hamiltonian walk move, if we take $h = \alpha d^{-1/4}, n = T/h$ and assume $\lim_{d\to\infty} \frac{d}{N(d)/2} = \rho \in [0,1)$, then as $d \to \infty$, the acceptance probability converges to*

$$\mathbb{E}[\min\{1, \exp(\mathcal{N}(\alpha^4\mu_\rho, \alpha^4\sigma_\rho))\}],$$

where $\mathcal{N}(\alpha^4\mu_\rho, \alpha^4\sigma_\rho)$ is a Gaussian distribution with mean and variance

$$\mu_\rho = -\frac{1}{32}\int \lambda^4 \sin^2(\sqrt{\lambda}T)\mathrm{d}\nu_\rho(\lambda), \quad \sigma_\rho = \frac{1}{16}\int \lambda^6 \sin^2(\sqrt{\lambda}T)\mathrm{d}\nu_\rho(\lambda).$$

For $\rho \in [0,1)$, $\mathrm{d}\nu_\rho(\lambda) = \frac{1}{2\pi\rho\lambda}\sqrt{(c-\lambda)(\lambda-b)}\chi_{[b,c]}(\lambda)\mathrm{d}\lambda$, where $b = (1-\sqrt{\rho})^2, c = (1+\sqrt{\rho})^2$ and $\chi_{[b,c]}$ is the characteristic function of $[b,c]$. When $\rho = 0$, $\mathrm{d}\nu_\rho(\lambda)$ concentrates on the Dirac mass at $\lambda = 1$. Moreover, the expected squared jumped distance satisfies

$$\lim_{d\to\infty} \frac{1}{d}\mathbb{E}[\|\mathbf{x}_i(m+1) - \mathbf{x}_i(m)\|_2^2] = 4\int \sin^2(\frac{\sqrt{\lambda}T}{2})\mathrm{d}\nu_\rho(\lambda)\mathbb{E}[\min\{1, \exp(\mathcal{N}(\alpha^4\mu_\rho, \alpha^4\sigma_\rho))\}].$$

- *For the Hamiltonian side move, we can take $h = \alpha < 2$ to be a constant that does not depend on d. Let n be the number of leapfrog steps. As $d \to \infty$, the acceptance probability converges to a nonzero limit $\mathbb{E}[\min\{1, \exp(P(z_1, z_2, n, \alpha))\}]$ where*

$$P(z_1, z_2, n, \alpha) = \frac{\alpha^2}{8}\left(\sin^2(n\phi)(z_1^2 - z_2^2) - \sin(2n\phi)z_1z_2\right).$$

Here $z_1 \sim \mathcal{N}(0,1), z_2 \sim \mathcal{N}(0, \frac{1}{1-\alpha^2/4})$ are independent, and $\phi \in [0,\pi]$ satisfies $\cos\phi = 1 - \frac{\alpha^2}{2}$. Moreover, the expected squared jumped distance satisfies

$$\lim_{d\to\infty} \mathbb{E}[\|\mathbf{x}_i(m+1) - \mathbf{x}_i(m)\|_2^2] = \mathbb{E}[Q(z_1, z_2, n, \alpha)\min\{1, \exp(P(z_1, z_2, n, \alpha))\}],$$

where we define

$$Q(z_1, z_2, n, \alpha) = (\cos(n\phi) - 1)^2 z_1^2 + \sin^2(n\phi)z_2^2 + 2(\cos(n\phi) - 1)\sin(n\phi)z_1z_2.$$

The proposition demonstrates that in d dimensions, the expected squared distance traveled in one iteration of the Hamiltonian walk move is $O(d)$, whereas it is only $O(1)$ for the Hamiltonian side move since the latter restricts movement along a single line in each iteration.

Overall, the Hamiltonian walk move requires $O(d^{1/4})$ leapfrog steps, or gradient and function evaluations, to traverse the support of the target distribution. This is much more efficient than the $O(d)$ evaluations needed for the Hamiltonian side move and the previously developed derivative-free stretch and side moves.

5. Numerical Experiments

5.1. **Evaluation criterion.** We investigate the efficiency of affine invariant ensemble samplers through numerical experiments. Our main evaluation criterion is the autocorrelation time at the stationary phase, following [26]. These samplers generate sequences $(\mathbf{x}_1(m), ..., \mathbf{x}_N(m))$ for $1 \le m \le M$, where M represents the length of the ensemble chain. We use these ensembles to estimate the observable

$$A = \mathbb{E}^{\mathbf{x}\sim\pi}[f(\mathbf{x})] = \int f(\mathbf{x})\pi(\mathbf{x})\mathrm{d}\mathbf{x}\,,$$

via the approximation

$$\hat{A}_e = \frac{1}{M}\sum_{m=1}^{M} F(\mathbf{x}_1(m), ..., \mathbf{x}_N(m)) = \frac{1}{M}\sum_{m=1}^{M}\left(\frac{1}{N}\sum_{i=1}^{N} f(\mathbf{x}_i(m))\right).$$

At the stationary phase, for large M, the variance of the estimator satisfies

$$\mathrm{Var}(\hat{A}_e) \approx \frac{\tau_e}{M}\mathrm{Var}^{\mathbf{x}_1,...,\mathbf{x}_N\sim\pi^N}[F(\mathbf{x}_1, ..., \mathbf{x}_N)] = \frac{\tau_e}{NM}\mathrm{Var}^{\mathbf{x}\sim\pi}[f(\mathbf{x})]\,,$$

where τ_e is the integrated autocorrelation time for the ensemble method defined as $\tau_e = \sum_{m=-\infty}^{+\infty}\frac{C_e(m)}{C_e(0)}$ with the autocovariance function defined as

$$C_e(m) = \lim_{m'\to\infty}\mathrm{Cov}[F(\mathbf{x}_1(m'), ..., \mathbf{x}_N(m')), F(\mathbf{x}_1(m+m'), ..., \mathbf{x}_N(m+m'))]\,.$$

The autocorrelation function at lag m is the ratio $\frac{C_e(m)}{C_e(0)}$.

In contrast, when using a single chain MCMC algorithm such as HMC, we obtain a single sequence $\mathbf{x}(m)$ for $1 \le m \le M$. The estimator becomes

$$\hat{A}_s = \frac{1}{M}\sum_{m=1}^{M} f(\mathbf{x}(m))\,,$$

with variance

$$\mathrm{Var}(\hat{A}_s) \approx \frac{\tau_s}{M}\mathrm{Var}^{\mathbf{x}\sim\pi}[f(\mathbf{x})]\,,$$

where $\tau_s = \sum_{m=-\infty}^{+\infty}\frac{C_s(m)}{C_s(0)}$ and $C_s(m) = \lim_{m'\to\infty}\mathrm{Cov}[f(\mathbf{x}(m')), f(\mathbf{x}(m+m'))]$. As noted by [26], a natural criterion to compare performance is the values of τ_e and τ_s. The estimation procedure of the autocorrelation time follows [26, Section 5].

Code is available at `https://github.com/yifanc96/AffineInvariantSamplers`.

5.2. **Synthetic: Gaussian.** As our first example, we consider Gaussian distributions. Given a dimension d and condition number κ, we generate d eigenvalues equi-distributed between 10^{-1} and $10^{-1}\kappa$. These form a diagonal matrix Σ with diagonal entries equal to the eigenvalues, which we use as the precision matrix of the Gaussian distribution.

For all ensemble samplers, we use $N = 2d$ walkers. For the stretch move, we use parameter $a = 1 + 2.151d^{-1/2}$ as suggested in Section 3.5. For the side move, we set $\sigma = 1.687d^{-1/2}$. For HMC, we fix the total time horizon at $T = 1$ and use $n = 2$ or $n = 10$ leapfrog steps with corresponding step size $h = T/n$. The same parameters apply to the Hamiltonian walk and side moves.

We run 2×10^5 iterations of these samplers as burn-in, followed by 10^6 steps treated as the stationary phase. Due to memory constraints, we thin the samples by a factor of

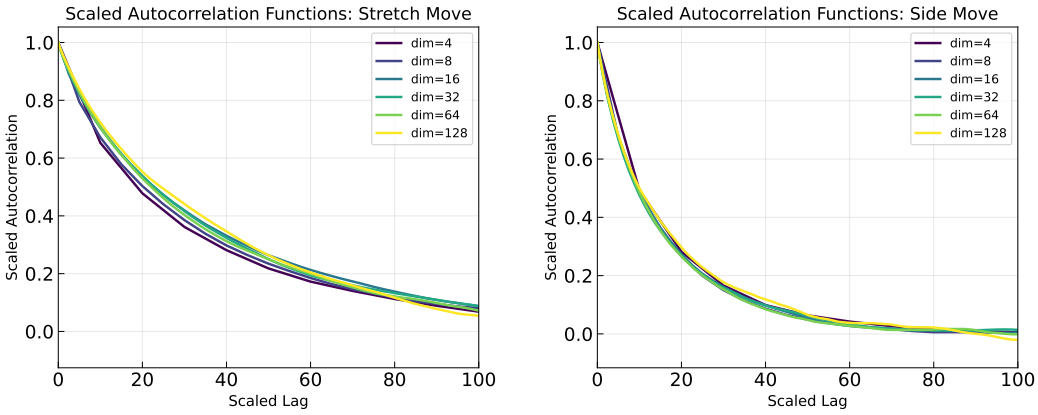

FIGURE 3. Scaled autocorrelation functions for sampling anisotropic Gaussian targets with condition number $\kappa = 1000$. Left: stretch move; right: side move. Scaled lag = original lag/dim$\times$4.

10, compute the autocorrelation function and time for these thinned samples, and then scale the results to approximate the autocorrelation function and time of the original samples. The autocorrelation function is computed for the observable $f(\mathbf{x}) = x_1$, the first component.

In Figure 3, we show a scaled version of the autocorrelation functions for the stretch and side moves. We scale the lag by dividing the original lag by the dimension and then multiplying by 4. With this scaling, the curve represents the unscaled autocorrelation function for $d = 4$. We observe that this rescaling produces similar curves for the scaled autocorrelation function, which confirms the high-dimensional linear scaling behavior of both samplers. Notably, the side move leads to faster decay of autocorrelation.

For $d = 128$, we report the detailed results to one decimal place in Table 1. We observe that the side move achieves approximately $1/2$ of the autocorrelation time of the stretch move. Moreover, with derivative information, samplers become more scalable in high dimensions, as demonstrated by the substantial decrease in autocorrelation time for HMC. We note that when $n = 2$, the step size is too large for HMC, causing negligible acceptance rates. However, this large step size works for the affine invariant Hamiltonian walk and side moves. Notably, the Hamiltonian walk move with $n = 2$ yields a small autocorrelation time of 12.7. Compared to HMC, the computational cost is approximately $1/3$, resulting in a speedup of roughly $67.8/12.7 \times 3 \approx 16$ times. If gradient and function evaluations have comparable costs, then the Hamiltonian walk move with $n = 2$ achieves $1000.1/12.7 \times 1/4 \approx 19.7$ times acceleration compared to the side move.

Since the Hamiltonian side move employs a side direction (supported on a line) in each iteration, we can reasonably compare it with the stretch and side moves that also move on a line. The Hamiltonian side move leads to shorter autocorrelation time than side move, but it needs more cost (in terms of gradient evaluations) per iteration.

In Figure 4, we further show the autocorrelation time of these samplers. The figure clearly demonstrates the $O(d)$ scaling of stretch and side moves and their Hamiltonian

	acceptance rate	autocorrelation time τ_e or τ_s	func eval per iter	grad eval per iter
Stretch move	0.45	2043.6	1	0
Side move	0.45	1000.1	1	0
HMC: $n = 10$	0.57	67.8	1	11
HMC: $n = 2$	0.00	—	1	3
Hamiltonian walk move: $n = 10$	0.98	10.5	1	11
Hamiltonian walk move: $n = 2$	0.61	12.7	1	3
Hamiltonian side move: $n = 10$	1.00	898.2	1	11
Hamiltonian side move: $n = 2$	0.98	732.3	1	3

TABLE 1. Performance for anisotropic Gaussian targets with condition number $\kappa = 1000$. Dimension $d = 128$. For HMC and affine invariant HMC (Hamiltonian walk and side moves), the total integration time is $T = 1$ and n is the number of leapfrog iterations.

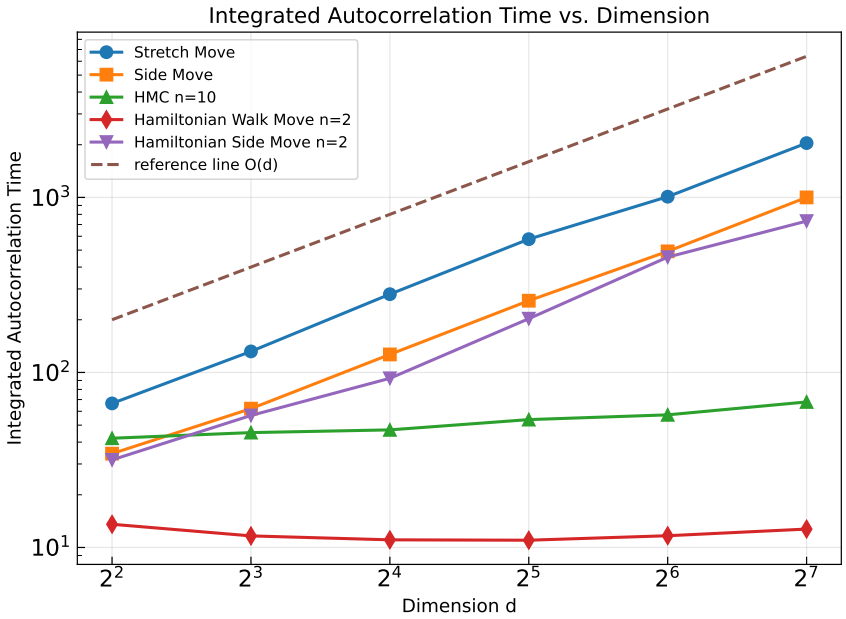

FIGURE 4. Autocorrelation time versus dimension for anisotropic Gaussian targets with condition number $\kappa = 1000$. For HMC and affine invariant HMC (Hamiltonian walk and side moves), the total integration time is $T = 1$ and n is the number of leapfrog iterations.

variants. The HMC and Hamiltonian walk move scale much better with dimension, which aligns with the theoretical insights provided by Proposition 4.1.

5.3. **Synthetic: Rings.** We then consider a ring shaped distribution (motivated by Figure 1 and discussions therein) with density

$$\pi(\mathbf{x}) \propto \exp\left(-\frac{(\|\mathbf{x}\|_2^2 - 1)^2}{l^2}\right),$$

where we choose $d = 50$ and $l = 0.25$. We use the same experimental set-up as in the Gaussian case. In Table 2, we present the results. Similar phenomenon is observed. The reduction of autocorrelation time of side move compared to stretch move becomes more apparent: the reduction is around 6.8 times. For this example, standard HMC with $n = 10$ performs well, since there is no strong anisotropy among coordinates. In such case, our affine invariant Hamiltonian walk move still leads to a smaller autocorrelation time.

	acceptance rate	autocorrelation time τ_e or τ_s	func eval per iter	grad eval per iter
Stretch move	0.29	2435.4	1	0
Side move	0.45	355.4	1	0
HMC: $n = 10$	0.69	20.8	1	11
HMC: $n = 2$	0.00	—	1	3
Hamiltonian walk move: $n = 10$	0.99	10.7	1	11
Hamiltonian walk move: $n = 2$	0.72	11.9	1	3
Hamiltonian side move: $n = 10$	1.00	354.8	1	11
Hamiltonian side move: $n = 2$	0.98	309.7	1	3

TABLE 2. Performance for ring shaped distributions: dimension $d = 50$. For HMC and affine invariant HMC (Hamiltonian walk and side moves), the total integration time is $T = 1$ and n is the number of leapfrog iterations.

5.4. **Invariant distribution to stochastic PDEs.** In this illustration, we generate samples from an infinite-dimensional probability measure defined over continuous functions on the unit interval $[0, 1]$; see [26]. The measure is formally

$$\exp\left(-\int_0^1 \frac{1}{2}(\partial_x u(x))^2 + V(u(x))\mathrm{d}x\right), \tag{5.1}$$

with V denoting a double-well potential function:

$$V(u) = (1 - u^2)^2.$$

This probability measure is the stationary distribution for the stochastic Allen-Cahn dynamics, which is a stochastic PDE (SPDE):

$$\partial_t u = \partial_{xx} u - V'(u) + \sqrt{2}\,\eta, \tag{5.2}$$

subject to natural boundary conditions at both endpoints $x = 0$ and $x = 1$. Here, η denotes space-time white noise. Realizations from this distribution typically exhibit

rough, approximately constant profiles near either 1 or -1; thus this is a bimodal distribution. We discretize the derivatives using finite difference with equidistributed points. Denote the total number of points by d, which leads to a d dimensional distribution.

We adopt the same setup as before. The autocorrelation function is computed for the path integral observable $f(u) = \int_0^1 u(x)\mathrm{d}x$ discretized using composite trapezoid rules. In Figure 5, we show the scaling of the autocorrelation time. We observe similar $O(d)$ scaling for stretch and side moves. The HMC scales worse than in the Gaussian example, since the condition number of the SPDE example deteriorates as dimension grows and we use a fixed number of leapfrog steps. For the affine invariant Hamiltonian walk move, we observe a nearly constant autocorrelation time as before, which demonstrates its superior performance in sampling such ill-conditioned distributions. In particular, for $d = 64$, Hamiltonian walk move with $n = 2$ leapfrog steps leads to $100\times$ reduction in autocorrelation time compared to HMC with $n = 10$ leapfrog steps[1].

We also report detailed experimental results in Table 3 for $d = 128$, a case not included in Figure 5. For this dimension, standard HMC with $n = 10$ fails due to negligible acceptance rates, but the Hamiltonian walk move performs consistently well.

We note that in this test example, the target distribution is absolutely continuous with respect to a Gaussian measure; accordingly, function space MCMC [16] is also expected to achieve dimension-robust convergence. Combining affine invariant samplers with function space MCMC may further enhance performance [17, 19].

	acceptance rate	autocorrelation time τ_e or τ_s	func eval per iter	grad eval per iter
Stretch move	0.44	3021.3	1	0
Side move	0.44	1398.3	1	0
HMC: $n = 10$	0.00	—	1	11
HMC: $n = 2$	0.00	—	1	3
Hamiltonian walk move: $n = 10$	0.98	11.2	1	11
Hamiltonian walk move: $n = 2$	0.59	14.9	1	3
Hamiltonian side move: $n = 10$	1.00	902.8	1	11
Hamiltonian side move: $n = 2$	0.98	770.8	1	3

TABLE 3. Performance for the bimodal distribution which is the invariant distribution to stochastic PDEs: dimension $d = 128$. For HMC and affine invariant HMC (Hamiltonian walk and side moves), the total integration time is $T = 1$ and n is the number of leapfrog iterations.

6. Discussions and Conclusions

In this paper, we propose a derivative-free, affine invariant side move sampler that improves upon the popular stretch move sampler in high dimensions. We also propose a class of derivative-based, affine invariant HMC samplers, particularly the Hamiltonian

[1]In this example with $d = 64$, the acceptance rate for HMC is 0.12, while for the Hamiltonian walk move it is 0.70.

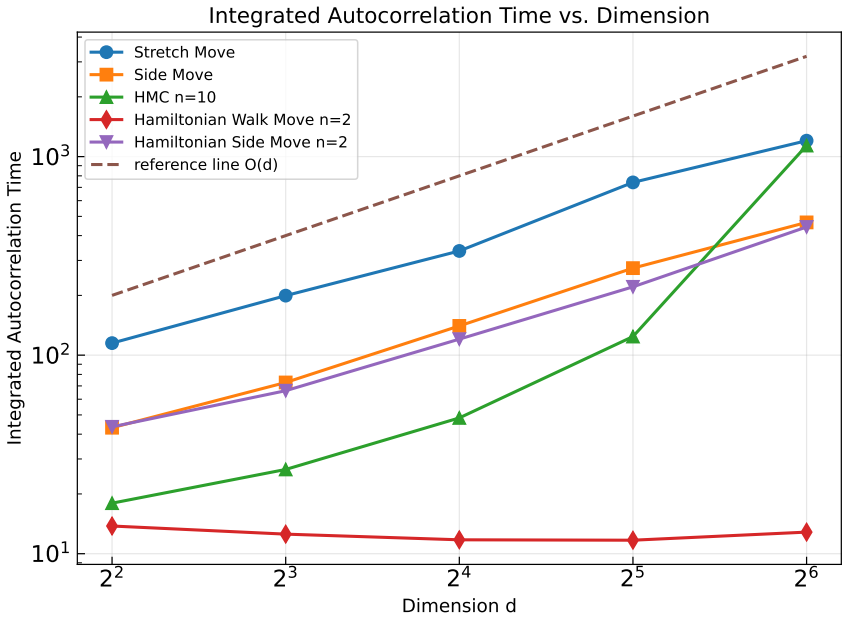

FIGURE 5. Autocorrelation time versus dimension for the bimodal SPDE targets. For HMC and affine invariant HMC (Hamiltonian walk and side moves), the total integration time is $T = 1$ and n is the number of leapfrog iterations.

walk move, which outperform HMC for sampling from highly anisotropic distributions. We show the dimensional scaling of these samplers for Gaussian targets at the stationary phase, identifying the step size scaling and expected squared jumped distance.

These new affine invariant samplers are shown to be highly efficient on several test examples. Notably, the Hamiltonian walk move sampler achieves a 10 to 100-fold reduction in autocorrelation time compared to vanilla HMC. This highlights the importance of incorporating affine invariance in designing practical samplers. We anticipate the application of these affine invariant samplers to various scientific problems.

Algorithmically, adapting the step size and, in the case of affine invariant HMC, the number of leapfrog steps could further improve performance, according to the wide success of the No-U-Turn Sampler (NUTS) [28, 8]. In our experiments, we observe that the Hamiltonian side move leads to smaller autocorrelation time compared to the standard side move. In each iteration, they move in the same direction, suggesting potential for further adaptation of the move length of the derivative-free side move. A recent relevant development is the No-Underrun Sampler (NURS) [5], which focuses on adapting move lengths along random line directions without using derivatives.

Theoretically, establishing the ergodicity of these affine invariant ensemble samplers remains an important challenge. To the best of our knowledge, the only work in this direction is [22] for covariance-preconditioned Langevin dynamics. Still, their analysis applies to continuous dynamics rather than discrete algorithms. Moreover, in addition

to the autocorrelation time at the stationary phase, understanding the mixing time or convergence during the burn-in period is also of interest. In this direction, existing results [21, 9] primarily focus on Gaussian target distributions in continuous-time covariance-preconditioned Langevin dynamics. We also note that affine invariant algorithms projected onto the Gaussian family (in the sense of variational inference) has been shown to achieve exponential improvement in condition number dependence when the target is a one-dimensional log-concave distribution [12, Theorem 5.7].

Finally, beyond affine invariance, there have been developments leveraging diffeomorphism invariant dynamics for sampling, particularly through the Fisher-Rao gradient flow [12]. The numerical approximation of diffeomorphism-invariant dynamics is subtle, and several algorithms can be interpreted as implementations of the Fisher-Rao gradient flow; these include methods based on birth-death processes [36, 37], Kalman methodology and variational inference [29, 13, 10], and ensemble chains [34]. Typically, at most affine invariance is preserved in the numerical implementation and the samplers are often approximate and have bias. It is an interesting direction to integrate these advances to develop exact or approximate samplers with broader invariance properties.

Acknowledgments YC is grateful to Jonathan Goodman for many intellectual discussions and shared insights on MCMC that inspired this work. YC also thanks Andrew Stuart and Jonathon Weare for valuable feedback on an earlier version of this article.

References

[1] Simon Apers, Sander Gribling, and Dániel Szilágyi. Hamiltonian monte carlo for efficient gaussian sampling: long and random steps. *Journal of Machine Learning Research*, 25(348):1–30, 2024.

[2] Yves F Atchadé, Gareth O Roberts, and Jeffrey S Rosenthal. Towards optimal scaling of metropolis-coupled markov chain monte carlo. *Statistics and Computing*, 21:555–568, 2011.

[3] Alexandros Beskos, Natesh Pillai, Gareth Roberts, Jesus-Maria Sanz-Serna, and Andrew Stuart. Optimal tuning of the hybrid Monte Carlo algorithm. *Bernoulli*, 19(5A):1501–1534, 2013.

[4] Patrick Billingsley. *Convergence of probability measures*. John Wiley & Sons, 2013.

[5] Nawaf Bou-Rabee, Bob Carpenter, Sifan Liu, and Stefan Oberdörster. The no-underrun sampler: A locally-adaptive, gradient-free mcmc method. *arXiv preprint arXiv:2501.18548*, 2025.

[6] Cajo JF Ter Braak. A Markov chain Monte Carlo version of the genetic algorithm Differential Evolution: easy Bayesian computing for real parameter spaces. *Statistics and Computing*, 16:239–249, 2006.

[7] Bob Carpenter. Ensemble methods are doomed to fail in high dimensions. Statistical Modeling, Causal Inference, and Social Science (blog), March 2017. Accessed on May 2, 2025.

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

[9] José A Carrillo and Urbain Vaes. Wasserstein stability estimates for covariance-preconditioned fokker–planck equations. *Nonlinearity*, 34(4):2275, 2021.

[10] Baojun Che, Yifan Chen, Zhenghao Huan, Daniel Zhengyu Huang, and Weijie Wang. Stable derivative free Gaussian mixture variational inference for Bayesian inverse problems. *arXiv preprint arXiv:2501.04259*, 2025.

[11] Haoxuan Chen and Lexing Ying. Ensemble-based annealed importance sampling. *arXiv preprint arXiv:2401.15645*, 2024.

[12] Yifan Chen, Daniel Zhengyu Huang, Jiaoyang Huang, Sebastian Reich, and Andrew M Stuart. Sampling via gradient flows in the space of probability measures. *arXiv preprint arXiv:2310.03597*, 2023.

[13] Yifan Chen, Daniel Zhengyu Huang, Jiaoyang Huang, Sebastian Reich, and Andrew M Stuart. Efficient, multimodal, and derivative-free Bayesian inference with Fisher–Rao gradient flows. *Inverse Problems*, 40(12):125001, 2024.

[14] Sinho Chewi, Thibaut Le Gouic, Chen Lu, Tyler Maunu, Philippe Rigollet, and Austin Stromme. Exponential ergodicity of mirror-langevin diffusions. *Advances in Neural Information Processing Systems*, 33:19573–19585, 2020.

[15] J Christen. A general purpose scale-independent MCMC algorithm. *technical report I-07-16, CIMAT*, 2007.

[16] SL Cotter, GO Roberts, AM Stuart, and D White. Mcmc methods for functions: Modifying old algorithms to make them faster. *Statistical Science*, 28(3):424–446, 2013.

[17] Jeremie Coullon and Robert J Webber. Ensemble sampler for infinite-dimensional inverse problems. *Statistics and Computing*, 31:1–9, 2021.

[18] Gianluca Detommaso, Tiangang Cui, Youssef Marzouk, Alessio Spantini, and Robert Scheichl. A stein variational newton method. *Advances in Neural Information Processing Systems*, 31, 2018.

[19] Matthew M Dunlop and Georg Stadler. A gradient-free subspace-adjusting ensemble sampler for infinite-dimensional bayesian inverse problems. *arXiv preprint arXiv:2202.11088*, 2022.

[20] Daniel Foreman-Mackey, David W Hogg, Dustin Lang, and Jonathan Goodman. emcee: the MCMC hammer. *Publications of the Astronomical Society of the Pacific*, 125(925):306, 2013.

[21] Alfredo Garbuno-Inigo, Franca Hoffmann, Wuchen Li, and Andrew M Stuart. Interacting Langevin diffusions: Gradient structure and ensemble Kalman sampler. *SIAM Journal on Applied Dynamical Systems*, 19(1):412–441, 2020.

[22] Alfredo Garbuno-Inigo, Nikolas Nüsken, and Sebastian Reich. Affine invariant interacting Langevin dynamics for Bayesian inference. *SIAM Journal on Applied Dynamical Systems*, 19(3):1633–1658, 2020.

[23] Andrew Gelman, Walter R Gilks, and Gareth O Roberts. Weak convergence and optimal scaling of random walk Metropolis algorithms. *The annals of applied probability*, 7(1):110–120, 1997.

[24] Walter R Gilks, Gareth O Roberts, and Edward I George. Adaptive direction sampling. *Journal of the Royal Statistical Society: Series D (The Statistician)*, 43(1):179–189, 1994.

[25] Mark Girolami and Ben Calderhead. Riemann manifold Langevin and Hamiltonian Monte Carlo methods. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 73(2):123–214, 2011.

[26] Jonathan Goodman and Jonathan Weare. Ensemble samplers with affine invariance. *Communications in Applied Mathematics and Computational Science*, 5(1):65–80, 2010.

[27] Philip Greengard. An ensemblized Metropolized Langevin sampler. *Master's thesis, Courant Institute, New York University*, 2015.

[28] Matthew D Hoffman, Andrew Gelman, et al. The No-U-Turn sampler: adaptively setting path lengths in Hamiltonian Monte Carlo. *J. Mach. Learn. Res.*, 15(1):1593–1623, 2014.

[29] Daniel Zhengyu Huang, Jiaoyang Huang, Sebastian Reich, and Andrew M Stuart. Efficient derivative-free Bayesian inference for large-scale inverse problems. *Inverse Problems*, 38(12):125006, 2022.

[30] David Huijser, Jesse Goodman, and Brendon J Brewer. Properties of the affine invariant ensemble sampler in high dimensions. *arXiv preprint arXiv:1509.02230*, 2015.

[31] Yunbum Kook, Yin-Tat Lee, Ruoqi Shen, and Santosh Vempala. Sampling with riemannian hamiltonian monte carlo in a constrained space. *Advances in Neural Information Processing Systems*, 35:31684–31696, 2022.

[32] Yin Tat Lee and Santosh S Vempala. Convergence rate of riemannian hamiltonian monte carlo and faster polytope volume computation. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, pages 1115–1121, 2018.

[33] Benedict Leimkuhler, Charles Matthews, and Jonathan Weare. Ensemble preconditioning for Markov chain Monte Carlo simulation. *Statistics and Computing*, 28:277–290, 2018.

[34] Michael Lindsey, Jonathan Weare, and Anna Zhang. Ensemble Markov chain Monte Carlo with teleporting walkers. *SIAM/ASA Journal on Uncertainty Quantification*, 10(3):860–885, 2022.
[35] Ziming Liu, Andrew M Stuart, and Yixuan Wang. Second order ensemble Langevin method for sampling and inverse problems. *arXiv preprint arXiv:2208.04506*, 2022.
[36] Yulong Lu, Jianfeng Lu, and James Nolen. Accelerating Langevin sampling with birth-death. *arXiv preprint arXiv:1905.09863*, 2019.
[37] Yulong Lu, Dejan Slepčev, and Lihan Wang. Birth–death dynamics for sampling: global convergence, approximations and their asymptotics. *Nonlinearity*, 36(11):5731, 2023.
[38] James Martin, Lucas C Wilcox, Carsten Burstedde, and Omar Ghattas. A stochastic newton mcmc method for large-scale statistical inverse problems with application to seismic inversion. *SIAM Journal on Scientific Computing*, 34(3):A1460–A1487, 2012.
[39] Radford M Neal. MCMC using Hamiltonian dynamics. *Handbook of markov chain monte carlo*, 2(11):2, 2011.
[40] John A Nelder and Roger Mead. A simplex method for function minimization. *The computer journal*, 7(4):308–313, 1965.
[41] Cristian Pasarica and Andrew Gelman. Adaptively scaling the metropolis algorithm using expected squared jumped distance. *Statistica Sinica*, pages 343–364, 2010.
[42] Sebastian Reich and Colin Cotter. *Probabilistic forecasting and Bayesian data assimilation*. Cambridge University Press, 2015.
[43] Sebastian Reich and Simon Weissmann. Fokker–Planck particle systems for Bayesian inference: Computational approaches. *SIAM/ASA Journal on Uncertainty Quantification*, 9(2):446–482, 2021.
[44] Gareth O Roberts and Walter R Gilks. Convergence of adaptive direction sampling. *Journal of multivariate analysis*, 49(2):287–298, 1994.
[45] Gareth O Roberts and Jeffrey S Rosenthal. Optimal scaling of discrete approximations to Langevin diffusions. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 60(1):255–268, 1998.
[46] Gareth O Roberts and Jeffrey S Rosenthal. Minimising mcmc variance via diffusion limits, with an application to simulated tempering. *The Annals of Applied Probability*, 24(1):131–149, 2014.
[47] Gareth O Roberts and Richard L Tweedie. Exponential convergence of Langevin distributions and their discrete approximations. *Bernoulli*, 2(3):341–363, 1996.
[48] Umut Simsekli, Roland Badeau, Taylan Cemgil, and Gaël Richard. Stochastic quasi-newton langevin monte carlo. In *International Conference on Machine Learning*, pages 642–651. PMLR, 2016.
[49] Rainer Storn and Kenneth Price. Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. *Journal of global optimization*, 11:341–359, 1997.
[50] Jasper A Vrugt, Cajo JF Ter Braak, Cees GH Diks, Bruce A Robinson, James M Hyman, and Dave Higdon. Accelerating Markov chain Monte Carlo simulation by differential evolution with self-adaptive randomized subspace sampling. *International Journal of Nonlinear Sciences and Numerical Simulation*, 10(3):273–290, 2009.
[51] Jun Yang, Gareth O Roberts, and Jeffrey S Rosenthal. Optimal scaling of random-walk metropolis algorithms on general target distributions. *Stochastic Processes and their Applications*, 130(10):6094–6132, 2020.

Appendix A. Pseudocode for All the Affine Invariant Samplers

Here we present the pseudocode for the affine invariant samplers used in our experiments:

- Parallel stretch move (Algorithm 1)
- Parallel side move (Algorithm 2)
- Parallel Hamiltonian walk move (Algorithm 3)
- Parallel Hamiltonian side move (Algorithm 4)

The pseudocode describes the iteration step from m to $m+1$. For all the algorithms, we have N walkers in the ensemble and we split the walkers into two groups:

$$S^{(0)}(m) = \{\mathbf{x}_1(m), ..., \mathbf{x}_{N/2}(m)\}, \quad S^{(1)} = \{\mathbf{x}_{N/2+1}(m), ..., \mathbf{x}_N(m)\}\,. \tag{A.1}$$

We use the notation $S^{(-s)}$ to represent the complementary ensemble of $S^{(s)}$ for $s \in \{0, 1\}$. Specifically, $S^{(-0)} = S^{(1)}$ and $S^{(-1)} = S^{(0)}$.

Algorithm 1 The parallel stretch move at step m (adapted from [20])

Require: Ensemble $S^{(s)}(m)$ for $s \in \{0, 1\}$, number of walkers N, unnormalized probability density $\pi(\cdot)$, parameter a
Ensure: Updated ensemble $S^{(s)}(m+1)$ for $s \in \{0, 1\}$

for $s \in \{0, 1\}$ **do**
  for $i = 1 + sN/2, \ldots, N/2 + sN/2$ **do**
    // This loop can be done in parallel for all i
    Draw a walker $\mathbf{x}_j(m)$ at random from the complementary ensemble $S^{(-s)}(m)$
    $z \leftarrow Z \sim g$ where g is defined in (3.5), with the given parameter a
    $\tilde{\mathbf{x}}_i(m+1) \leftarrow \mathbf{x}_j(m) + z[\mathbf{x}_i(m) - \mathbf{x}_j(m)]$
    $q \leftarrow z^{d-1}\pi(\tilde{\mathbf{x}}_i(m+1))/\pi(\mathbf{x}_i(m))$
    $r \leftarrow U \sim \mathrm{Unif}[0, 1]$
    if $r \le q$ **then**
      $\mathbf{x}_i(m+1) \leftarrow \tilde{\mathbf{x}}_i(m+1)$
    else
      $\mathbf{x}_i(m+1) \leftarrow \mathbf{x}_i(m)$
    end if
  end for
end for

Appendix B. Proof for the Dimension Scaling of Side and Stretch Moves

Proof for Proposition 3.1. For notational convenience, we omit the dependence on m.

B.1. **Scaling of side move.** For the side move $\tilde{\mathbf{x}}_i = \mathbf{x}_i + \sigma(\mathbf{x}_j - \mathbf{x}_k)\xi$, we have

$$\begin{aligned}\log \frac{\pi(\tilde{\mathbf{x}}_i)}{\pi(\mathbf{x}_i)} &= -\frac{1}{2}\tilde{\mathbf{x}}_i^T\tilde{\mathbf{x}}_i + \frac{1}{2}\mathbf{x}_i^T\mathbf{x}_i \\ &= -\frac{1}{2}\sigma^2\xi^2(\mathbf{x}_j - \mathbf{x}_k)^T(\mathbf{x}_j - \mathbf{x}_k) - \sigma\mathbf{x}_i \cdot (\mathbf{x}_j - \mathbf{x}_k)\xi\,.\end{aligned}$$

Let $\sigma = \frac{\alpha}{\sqrt{d}}$. By the law of large numbers and central limit theorem, it holds that as $d \to \infty$,

$$\log \frac{\pi(\tilde{\mathbf{x}}_i)}{\pi(\mathbf{x}_i)} \xrightarrow{\mathcal{D}} -\alpha^2\xi^2 - \sqrt{2}\alpha\xi z\,,$$

where $z \sim \mathcal{N}(0, 1)$ and the convergence is in distribution.

Therefore, the limit of the acceptance probability is

$$\lim_{d\to\infty} \mathbb{E}[\min\{1, \frac{\pi(\tilde{\mathbf{x}}_i)}{\pi(\mathbf{x}_i)}\}] = \mathbb{E}[\min\{1, \exp(-\alpha^2\xi^2 - \alpha\sqrt{2}\xi z)\}]\,.$$

Algorithm 2 The parallel side move at step m

Require: Ensemble $S^{(s)}(m)$ for $s \in \{0,1\}$, number of walkers N, unnormalized probability density $\pi(\cdot)$, parameter σ
Ensure: Updated ensemble $S^{(s)}(m+1)$ for $s \in \{0,1\}$
for $s \in \{0,1\}$ **do**
  for $i = 1 + sN/2, \ldots, N/2 + sN/2$ **do**
    // This loop can be done in parallel for all i
    Draw two walker $\mathbf{x}_j(m), \mathbf{x}_k(m)$ at random from the complementary $S^{(-s)}(m)$
    $z \leftarrow Z \sim \mathcal{N}(0,1)$
    $\tilde{\mathbf{x}}_i(m+1) \leftarrow \mathbf{x}_i(m) + \sigma z[\mathbf{x}_j(m) - \mathbf{x}_k(m)]$
    $q \leftarrow \pi(\tilde{\mathbf{x}}_i(m+1))/\pi(\mathbf{x}_i(m))$
    $r \leftarrow U \sim \mathrm{Unif}[0,1]$
    if $r \leq q$ **then**
      $\mathbf{x}_i(m+1) \leftarrow \tilde{\mathbf{x}}_i(m+1)$
    else
      $\mathbf{x}_i(m+1) \leftarrow \mathbf{x}_i(m)$
    end if
  end for
end for

Algorithm 3 The parallel Hamiltonian walk move at step m

Require: Ensemble $S^{(s)}(m)$ for $s \in \{0,1\}$, number of walkers N, unnormalized probability density $\pi(\cdot)$, leapfrog stepsize h and steps n
Ensure: Updated ensemble $S^{(s)}(m+1)$ for $s \in \{0,1\}$
for $s \in \{0,1\}$ **do**
  for $i = 1 + sN/2, \ldots, N/2 + sN/2$ **do**
    // This loop can be done in parallel for all i
    Draw $\mathbf{p}_i \sim \mathcal{N}(0, I_{N/2 \times N/2})$
    Form the centered B for the complementary $S^{(-s)}(m)$ (see (4.7) (4.9))
    Run n steps leapfrog $(\tilde{\mathbf{x}}_i(m+1), \tilde{\mathbf{p}}_i) = \mathsf{PL}^n_{h,B}(\mathbf{x}_i(m), \mathbf{p}_i)$ as in (4.8)
    $q \leftarrow \exp(-V(\tilde{\mathbf{x}}_i(m+1)) - \frac{1}{2}\tilde{\mathbf{p}}_i^T\tilde{\mathbf{p}}_i + V(\mathbf{x}_i(m)) + \frac{1}{2}\mathbf{p}_i^T\mathbf{p}_i)$
    $r \leftarrow U \sim \mathrm{Unif}[0,1]$
    if $r \leq q$ **then**
      $\mathbf{x}_i(m+1) \leftarrow \tilde{\mathbf{x}}_i(m+1)$
    else
      $\mathbf{x}_i(m+1) \leftarrow \mathbf{x}_i(m)$
    end if
  end for
end for

This is because the function $t \to \min\{1, \exp(t)\}$ is a bounded continuous function, and we have used the fact that convergence in distribution implies convergence of all bounded continuous observables. The convergence of the expected squared jumped distance follows similarly.

Algorithm 4 The parallel Hamiltonian side walk move at step m

Require: Ensemble $S^{(s)}(m)$ for $s \in \{0,1\}$, number of walkers N, unnormalized probability density $\pi(\cdot)$, leapfrog stepsize h and steps n
Ensure: Updated ensemble $S^{(s)}(m+1)$ for $s \in \{0,1\}$

for $s \in \{0,1\}$ **do**
  for $i = 1 + sN/2, \dots, N/2 + sN/2$ **do**
    // This loop can be done in parallel for all i
    Draw $p_i \sim \mathcal{N}(0,1)$
    Draw two walker $\mathbf{x}_j(m), \mathbf{x}_k(m)$ at random from the complementary $S^{(-s)}(m)$
    Run n steps leapfrog $(\tilde{\mathbf{x}}_i(m+1), \tilde{p}_i) = \mathsf{PL}^n_{h,(\mathbf{x}_j - \mathbf{x}_k)/\sqrt{2d}}(\mathbf{x}_i(m), p_i)$ as in (4.11)
    $q \leftarrow \exp(-V(\tilde{\mathbf{x}}_i(m+1)) - \frac{1}{2}\tilde{p}_i^T\tilde{p}_i + V(\mathbf{x}_i(m)) + \frac{1}{2}p_i^T p_i)$
    $r \leftarrow U \sim \mathrm{Unif}[0,1]$
  end for
  if $r \le q$ **then**
    $\mathbf{x}_i(m+1) \leftarrow \tilde{\mathbf{x}}_i(m+1)$
  else
    $\mathbf{x}_i(m+1) \leftarrow \mathbf{x}_i(m)$
  end if
end for

B.2. **Scaling of stretch move.** For the stretch move, $\tilde{\mathbf{x}}_i = \mathbf{x}_j + Z(\mathbf{x}_i - \mathbf{x}_j)$, we have

$$\begin{aligned}
\log\left(Z^{d-1}\frac{\pi(\tilde{\mathbf{x}}_i)}{\pi(\mathbf{x}_i)}\right) &= (d-1)\log Z - \frac{1}{2}\tilde{\mathbf{x}}_i^T\tilde{\mathbf{x}}_i + \frac{1}{2}\mathbf{x}_i^T\mathbf{x}_i \\
&= (d-1)\log Z - \frac{1}{2}\left((Z^2-1)\mathbf{x}_i^T\mathbf{x}_i + (1-Z)^2\mathbf{x}_j^T\mathbf{x}_j + 2Z(1-Z)\mathbf{x}_i^T\mathbf{x}_j\right) \\
&= d(\log Z - Z + 1 + \frac{1}{2}(Z-1)^2) - \log Z - \frac{1}{2}d(Z-1)^2 \\
&\quad - (Z-1)(\frac{Z+1}{2}\mathbf{x}_i^T\mathbf{x}_i - Z\mathbf{x}_i^T\mathbf{x}_j - d) - \frac{1}{2}(Z-1)^2\mathbf{x}_j^T\mathbf{x}_j \\
&= d(\log Z - Z + 1 + \frac{1}{2}(Z-1)^2) - \log Z - \frac{1}{2}d(Z-1)^2 \\
&\quad - (Z-1)(\mathbf{x}_i^T\mathbf{x}_i - \mathbf{x}_i^T\mathbf{x}_j - d) - \frac{1}{2}(Z-1)^2\mathbf{x}_j^T\mathbf{x}_j \\
&\quad - \frac{(Z-1)^2}{2}\mathbf{x}_i^T\mathbf{x}_i + (Z-1)^2\mathbf{x}_i^T\mathbf{x}_j\,,
\end{aligned}$$

where we organize the terms in a way that is convenient to apply limit theorems later.

Here, the Z has the density

$$g(z) \propto \begin{cases} \frac{1}{\sqrt{z}} & \text{if } z \in \left[\frac{1}{a}, a\right] \\ 0 & \text{otherwise}\,, \end{cases}$$

which implies that we can write $Z = \left(\frac{(\sqrt{a}-\sqrt{a^{-1}})U + \sqrt{a} + \sqrt{a^{-1}}}{2}\right)^2$ for $U \sim \mathrm{Unif}[-1,1]$. Using the formula, we have $\sqrt{d}(Z-1) \xrightarrow{a.s.} \beta U$ for $a = 1 + \frac{\beta}{\sqrt{d}}$.

By direct calculation, as $d \to \infty$, we have

- $d(\log Z - Z + 1 + \frac{1}{2}(Z-1)^2) \xrightarrow{a.s.} 0$.
- $\log Z \xrightarrow{a.s.} 0$.

These show that several parts in the formula of $\log\left(Z^{d-1}\frac{\pi(\tilde{\mathbf{x}}_i)}{\pi(\mathbf{x}_i)}\right)$ will converge to zero almost surely. For the remaining part in the formula of $\log\left(Z^{d-1}\frac{\pi(\tilde{\mathbf{x}}_i)}{\pi(\mathbf{x}_i)}\right)$, we note that the random vector

$$(\sqrt{d}(Z-1), \frac{1}{\sqrt{d}}(\mathbf{x}_i^T\mathbf{x}_i - d), \frac{1}{\sqrt{d}}\mathbf{x}_i^T\mathbf{x}_j, \frac{1}{d}\mathbf{x}_i^T\mathbf{x}_i, \frac{1}{d}\mathbf{x}_i^T\mathbf{x}_j)$$

converges in distribution to $(\beta U, z_1, z_2, \delta_1, \delta_0)$ (for convergence in distribution in the product space, see [4, Section 2]). Here $U \sim \text{Unif}[-1,1], z_1 \sim \mathcal{N}(0,3), z_2 \sim \mathcal{N}(0,1)$ are independent, and δ_x is a Diracs point pass at x. The convergence is obtained using the law of large numbers and the central limit theorem as well as the fact that Z and $\mathbf{x}_i, \mathbf{x}_j$ are independent. We also used the fact that convergence almost surely implies convergence in distribution.

Then, the remaining part in $\log\left(Z^{d-1}\frac{\pi(\tilde{\mathbf{x}}_i)}{\pi(\mathbf{x}_i)}\right)$ can be seen as a continuous function of the above random vector. Using the property of convergence in distribution, we get that

$$\log\left(Z^{d-1}\frac{\pi(\tilde{\mathbf{x}}_i)}{\pi(\mathbf{x}_i)}\right) \xrightarrow{\mathcal{D}} -\frac{3}{2}\beta^2U^2 - \sqrt{3}\beta U z\,, \tag{B.1}$$

where $U \sim \text{Unif}[-1,1]$ is independent of $z \sim \mathcal{N}(0,1)$. This then implies the convergence of the acceptance probability and the squared jumped distance as desired. □

Appendix C. Proof for the Dimension Scaling of Affine Invariant HMC

Proof for Proposition 4.1. We focus on one particle $\mathbf{x}_i$. For simplicity of notation, we omit the subindex i when there is no confusion. We analyze one iteration in the affine invariant HMC: we use superscript $\mathbf{x}^{(l)}$ to denote the result from the leapfrog scheme, and we use B for a general preconditioner which can either be the one in the Hamiltonian walk move or the side move. By definition,

$$\begin{aligned}
\mathbf{p}^{(l+1/2)} &= \mathbf{p}^{(l)} - \frac{h}{2}B^T\nabla V(\mathbf{x}) \\
\mathbf{x}^{(l+1)} &= \mathbf{x}^{(l)} + hB\mathbf{p}^{(l+1/2)} \\
\mathbf{p}^{(l+1)} &= \mathbf{p}^{(l+1/2)} - \frac{h}{2}B^T\nabla V(\mathbf{x}^{(l+1)})\,.
\end{aligned}$$

The above leads to the iteration

$$\begin{bmatrix} B^T\mathbf{x}^{(l+1)} \\ \mathbf{p}^{(l+1)} \end{bmatrix} = \begin{bmatrix} I - \frac{h^2}{2}B^TB & hB^TB \\ -h(I - \frac{h^2}{4}B^TB) & I - \frac{h^2}{2}B^TB \end{bmatrix} \begin{bmatrix} B^T\mathbf{x}^{(l)} \\ \mathbf{p}^{(l)} \end{bmatrix},$$

where $B^T\mathbf{x}^{(l)}$ and $\mathbf{p}^{(l)}$ are of the same dimension; we denote the dimension by D. We will use the notation $\mathbf{q}^{(l)} = B^T\mathbf{x}^{(l)}$.

C.1. **A simple formula for the acceptance probability.** First, we show that the acceptance probability can be computed using the coordinates $\mathbf{q}^{(n)}$ and $\mathbf{p}^{(n)}$. To do so, note that we have the relation

$$\mathbf{x}^{(n)} - \mathbf{x}^{(0)} = B(B^TB)^+(\mathbf{q}^{(n)} - \mathbf{q}^{(0)})\,,$$

where $(B^TB)^+$ is the Moore–Penrose inverse of the matrix $B^TB \in \mathbb{R}^{D\times D}$.

This implies that

$$\begin{aligned}\frac{|\mathbf{x}^{(n)}|_2^2}{2} - \frac{|\mathbf{x}^{(0)}|_2^2}{2} &= \frac{1}{2}|\mathbf{x}^{(0)} + B(B^TB)^+(\mathbf{q}^{(n)} - \mathbf{q}^{(0)})|_2^2 - \frac{1}{2}|\mathbf{x}^{(0)}|_2^2\\ &= (\mathbf{x}^{(0)})^TB(B^TB)^+(\mathbf{q}^{(n)} - \mathbf{q}^{(0)}) + \frac{1}{2}(\mathbf{q}^{(n)} - \mathbf{q}^{(0)})^T(B^TB)^+(\mathbf{q}^{(n)} - \mathbf{q}^{(0)})\\ &= (\mathbf{q}^{(0)})^T(B^TB)^+(\mathbf{q}^{(n)} - \mathbf{q}^{(0)}) + \frac{1}{2}(\mathbf{q}^{(n)} - \mathbf{q}^{(0)})^T(B^TB)^+(\mathbf{q}^{(n)} - \mathbf{q}^{(0)})\\ &= \frac{1}{2}(\mathbf{q}^{(n)})^T(B^TB)^+\mathbf{q}^{(n)} - \frac{1}{2}(\mathbf{q}^{(0)})^T(B^TB)^+\mathbf{q}^{(0)}\,.\end{aligned}$$

Thus the acceptance probability can be written in the $\mathbf{q}^{(n)}$ and $\mathbf{p}^{(n)}$ coordinates:

$$\min\{1, \exp(-\frac{(\mathbf{q}^{(n)})^T(B^TB)^+\mathbf{q}^{(n)} + (\mathbf{p}^{(n)})^T\mathbf{p}^{(n)}}{2} + \frac{(\mathbf{q}^{(0)})^T(B^TB)^+\mathbf{q}^{(0)} + (\mathbf{p}^{(0)})^T\mathbf{p}^{(0)}}{2})\}\,.$$

Let $B^TB = U\Sigma U^T \in \mathbb{R}^{D\times D}$ be the eigenvalue decomposition where U is an orthogonal matrix and $\Sigma = \mathrm{diag}(\lambda_1^2, ..., \lambda_r^2, \lambda_{r+1}^2, ..., \lambda_D^2)$. Here we assume the rank of B^TB to be r and $\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_r > 0 = \lambda_{r+1} = \cdots = \lambda_D$. Let $\bar{\mathbf{q}}^{(l)} = U^T\mathbf{q}^{(l)}$ and $\bar{\mathbf{p}}^{(l)} = U^T\mathbf{p}^{(l)}$. Then the acceptance probability is

$$\min\{1, \exp(-\frac{(\bar{\mathbf{q}}^{(n)})^T\Sigma^+\bar{\mathbf{q}}^{(n)} + (\bar{\mathbf{p}}^{(n)})^T\bar{\mathbf{p}}^{(n)}}{2} + \frac{(\bar{\mathbf{q}}^{(0)})^T\Sigma^+\bar{\mathbf{q}}^{(0)} + (\bar{\mathbf{p}}^{(0)})^T\bar{\mathbf{p}}^{(0)}}{2})\}\,.$$

Moreover, we have the iteration

$$\begin{bmatrix}\bar{\mathbf{q}}^{(l+1)}\\ \bar{\mathbf{p}}^{(l+1)}\end{bmatrix} = \begin{bmatrix}I - \frac{h^2}{2}\Sigma & h\Sigma\\ -h(I - \frac{h^2}{4}\Sigma) & I - \frac{h^2}{2}\Sigma\end{bmatrix}\begin{bmatrix}\bar{\mathbf{q}}^{(l)}\\ \bar{\mathbf{p}}^{(l)}\end{bmatrix},$$

Denote $\bar{\mathbf{q}}^{(l)} = (\bar{q}_1^{(l)}, ..., \bar{q}_D^{(l)})$ and $\bar{\mathbf{p}}^{(l)} = (\bar{p}_1^{(l)}, ..., \bar{p}_D^{(l)})$. For the i-th coordinate, we have

$$\begin{bmatrix}\bar{q}_i^{(l+1)}\\ \bar{p}_i^{(l+1)}\end{bmatrix} = \begin{bmatrix}1 - \frac{h^2}{2}\lambda_i^2 & h\lambda_i^2\\ -h(1 - \frac{h^2}{4}\lambda_i^2) & 1 - \frac{h^2}{2}\lambda_i^2\end{bmatrix}\begin{bmatrix}\bar{q}_i^{(l)}\\ \bar{p}_i^{(l)}\end{bmatrix}\,.$$

We can solve the above system explicitly. First, for $r+1 \leq i \leq D$, we get $\bar{q}_i^{(l+1)} = \bar{q}_i^{(l)}$ and $\bar{p}_i^{(l+1)} = -h\bar{q}_i^{(l)} + \bar{p}_i^{(l)}$. Recall $\bar{\mathbf{q}}^{(0)} = U^TB^T\mathbf{x}^{(0)} \sim \mathcal{N}(0, U^TB^TBU) = \mathcal{N}(0, \Sigma)$. This means that $\bar{q}_i^{(0)} = 0$ for $r+1 \leq i \leq D$. Thus $\bar{q}_i^{(l)} = 0$ for all l and $\bar{p}_i^{(l+1)} = \bar{p}_i^{(l)}$ for all l. Therefore the term that depends on $r+1 \leq i \leq D$ in the acceptance probability satisfies

$$-\frac{|\bar{p}_i^{(n)}|^2}{2} + \frac{|\bar{p}_i^{(0)}|^2}{2} = 0\,.$$

For $1 \le i \le r$, let us assume $h \le 2/\lambda_i$. The eigenvalues of the matrix are

$$\mu = 1 - \frac{h^2}{2}\lambda_i^2 \pm \mathsf{j}\left(h\lambda_i\sqrt{1 - \frac{1}{4}h^2\lambda_i^2}\right)$$

where j is the imaginary number. Denote $\phi_i \in [0, \pi]$ such that $\cos\phi_i = 1 - \frac{h^2\lambda_i^2}{2}, \sin\phi_i = h\lambda_i\sqrt{1 - \frac{1}{4}h^2\lambda_i^2}$. We get

$$\begin{bmatrix} \bar{q}_i^{(l+1)} \\ \bar{p}_i^{(l+1)} \end{bmatrix} = \begin{bmatrix} \cos\phi_i & \hat{\lambda}_i \sin\phi_i \\ -\frac{1}{\hat{\lambda}_i}\sin\phi_i & \cos\phi_i \end{bmatrix} \begin{bmatrix} \bar{q}_i^{(l)} \\ \bar{p}_i^{(l)} \end{bmatrix}, \tag{C.1}$$

where $\hat{\lambda}_i = \frac{\lambda_i}{\sqrt{1 - h^2\lambda_i^2/4}}$. Iterating the equation leads to the solution

$$\begin{bmatrix} \bar{q}_i^{(n)} \\ \bar{p}_i^{(n)} \end{bmatrix} = \begin{bmatrix} \cos(n\phi_i) & \hat{\lambda}_i \sin(n\phi_i) \\ -\frac{1}{\hat{\lambda}_i}\sin(n\phi_i) & \cos(n\phi_i) \end{bmatrix} \begin{bmatrix} \bar{q}_i^{(0)} \\ \bar{p}_i^{(0)} \end{bmatrix}.$$

On the other hand, consider the Hamiltonian dynamics

$$\frac{\mathrm{d}\bar{q}(t)}{\mathrm{d}t} = \hat{\lambda}_i^2 \bar{p}(t), \quad \frac{\mathrm{d}\bar{p}(t)}{\mathrm{d}t} = -\bar{q}(t)\,,$$

which has an explicit solution

$$\begin{bmatrix} \bar{q}(t) \\ \bar{p}(t) \end{bmatrix} = \begin{bmatrix} \cos(\hat{\lambda}_i t) & \hat{\lambda}_i \sin(\hat{\lambda}_i t) \\ -\frac{1}{\hat{\lambda}_i}\sin(\hat{\lambda}_i t) & \cos(\hat{\lambda}_i t) \end{bmatrix} \begin{bmatrix} \bar{q}(0) \\ \bar{p}(0) \end{bmatrix}. \tag{C.2}$$

By taking the same initial conditions for (C.1) and (C.2), and $\hat{\lambda}_i t = n\phi_i$, we get the same solution for the discrete and continuous systems.

Since (C.2) is the exact solution to a Hamiltonian dynamics, we have conservation of several quantities along the dynamics. In particular, for a specific energy, we get

$$\frac{|\bar{q}(t)|^2}{2\hat{\lambda}_i^2} + \frac{|\bar{p}(t)|^2}{2} = \frac{|\bar{q}(0)|^2}{2\hat{\lambda}_i^2} + \frac{|\bar{p}(0)|^2}{2}\,.$$

Due to the equivalence mentioned above, we thus get

$$\frac{|\bar{q}_i^{(n)}|^2}{2\hat{\lambda}_i^2} + \frac{|\bar{p}_i^{(n)}|^2}{2} = \frac{|\bar{q}_i^{(0)}|^2}{2\hat{\lambda}_i^2} + \frac{|\bar{p}_i^{(0)}|^2}{2}\,.$$

Therefore, the i-th term (for $1 \le i \le r$) in the acceptance probability satisfies

$$\begin{aligned} & -\frac{|\bar{q}_i^{(n)}|^2}{2\lambda_i^2} - \frac{|\bar{p}_i^{(n)}|^2}{2} + \frac{|\bar{q}_i^{(0)}|^2}{2\lambda_i^2} + \frac{|\bar{p}_i^{(0)}|^2}{2} \\ =& -\frac{|\bar{q}_i^{(n)}|^2}{2\lambda_i^2} + \frac{|\bar{q}_i^{(n)}|^2}{2\hat{\lambda}_i^2} + \frac{|\bar{q}_i^{(0)}|^2}{2\lambda_i^2} - \frac{|\bar{q}_i^{(0)}|^2}{2\hat{\lambda}_i^2} \\ =& \frac{h^2\lambda_i^4}{8}(-|\bar{q}_i^{(n)}|^2 + |\bar{q}_i^{(0)}|^2)\,. \end{aligned} \tag{C.3}$$

Then the overall acceptance rate is accordingly

$$\begin{aligned}&\min\{1,\exp(-\frac{(\bar{\mathbf{q}}^{(n)})^T\Sigma^+\bar{\mathbf{q}}^{(n)}+(\bar{\mathbf{p}}^{(n)})^T\bar{\mathbf{p}}^{(n)}}{2}+\frac{(\bar{\mathbf{q}}^{(0)})^T\Sigma^+\bar{\mathbf{q}}^{(0)}+(\bar{\mathbf{p}}^{(0)})^T\bar{\mathbf{p}}^{(0)}}{2})\}\\ =&\min\{1,\exp(\frac{h^2}{8}\sum_{i=1}^r\lambda_i^4(|\bar{q}_i^{(0)}|^2-|\bar{q}_i^{(n)}|^2))\}\,.\end{aligned}\tag{C.4}$$

This formula is similar to the one derived in [1] for standard Hamiltonian dynamics. We adapt the strategy to derive the formula for the above preconditioned Hamiltonian dynamics. Here, the additional B requires us to discuss the rank of the matrix B^TB.

C.2. **Large d limit of acceptance probability for the Hamiltonian walk move.** In this case, we have $B\in\mathbb{R}^{d\times N/2}$ and $D=N/2\geq d$. The rank of B^TB is thus $r=d$.

Now, for the i-th term, since $\bar{q}_i^{(n)}=\cos(n\phi_i)\bar{q}_i^{(0)}+\hat{\lambda}_i\sin(n\phi_i)\bar{p}_i^{(0)}$, we have

$$|\bar{q}_i^{(0)}|^2-|\bar{q}_i^{(n)}|^2=(1-\cos^2(n\phi_i))|\bar{q}_i^{(0)}|^2-\hat{\lambda}_i^2\sin^2(n\phi_i)|\bar{p}_i^{(0)}|^2-2\hat{\lambda}_i\cos(n\phi_i)\sin(n\phi_i)\bar{q}_i^{(0)}\bar{p}_i^{(0)}\,.\tag{C.5}$$

Recall our assumption: $\mathbf{x}^{(0)},\mathbf{p}^{(0)}$ are independent multivariate normal distributions. This implies that $\bar{q}_i^{(0)}\sim\mathcal{N}(0,\lambda_i^2)$. By a direct and tedious calculation, we get that for $h\leq\min_{1\leq i\leq d}\frac{2}{\lambda_i}$,

$$\begin{aligned}\mathbb{E}[\lambda_i^4(|\bar{q}_i^{(0)}|^2-|\bar{q}_i^{(n)}|^2)]&=-\frac{h^2\lambda_i^8}{4}\sin^2(n\phi_i)\frac{1}{1-h^2\lambda_i^2/4}\,,\\ \mathrm{Var}[\lambda_i^4(|\bar{q}_i^{(0)}|^2-|\bar{q}_i^{(n)}|^2)]&=\frac{4\lambda_i^{12}}{1-h^2\lambda_i^2/4}\sin^2(n\phi_i)+O(h^4)\,.\end{aligned}\tag{C.6}$$

Therefore

$$\mathbb{E}[\frac{h^2}{8}\sum_{i=1}^d\lambda_i^4(|\bar{q}_i^{(0)}|^2-|\bar{q}_i^{(n)}|^2)]=-\frac{h^4}{32}\sum_{i=1}^d\frac{\lambda_i^8}{1-h^2\lambda_i^2/4}\sin^2(n\phi_i)\,,\tag{C.7}$$

and

$$\mathrm{Var}[\frac{h^2}{8}\sum_{i=1}^d\lambda_i^4(|\bar{q}_i^{(0)}|^2-|\bar{q}_i^{(n)}|^2)]=(\frac{h^4}{16}+O(h^8))\sum_{i=1}^d\frac{\lambda_i^{12}}{1-h^2\lambda_i^2/4}\sin^2(n\phi_i)\,.\tag{C.8}$$

We also note that $\cos\phi_i=1-\frac{h^2\lambda_i^2}{2}$, so

$$\phi_i=\arccos(1-\frac{h^2\lambda_i^2}{2})=h\lambda_i+\frac{(h\lambda_i)^3}{24}+O(h^4)\,,$$

which implies that if we take $n=\lceil\frac{T}{h}\rceil$ with T fixed, then

$$\cos(n\phi_i)=\cos(\lambda_iT+O(h^2)),\quad \sin(n\phi_i)=\sin(\lambda_iT+O(h^2))\,.$$

We are now in the position to derive the $d\to\infty$ limit of the acceptance probability. By writing N as a function of d and assuming $\lim_{d\to\infty}\frac{d}{N(d)/2}=\rho<1$, we get that the spectral measure

$$\mu_d=\frac{1}{d}\sum_{i=1}^d\delta_{\lambda_i^2}$$

of the random matrix B^TB will converge almost surely to

$$\mathrm{d}\nu_\rho(\lambda) = \frac{1}{2\pi\rho\lambda}\sqrt{(c-\lambda)(\lambda-b)}\chi_{[b,c]}(\lambda)\mathrm{d}\lambda$$

where $b = (1-\sqrt{\rho})^2, c = (1+\sqrt{\rho})^2$ and $\chi_{[b,c]}$ is the characteristic function of $[b,c]$. This is due to the Marčenko–Pastur law. These limiting eigenvalues all belong to $[b,c]$ which is independent of d. Taking $h = \alpha d^{-1/4}, n = \lceil \frac{T}{h} \rceil$ and using the central limit theorem with (C.7) and (C.8), we get

$$\frac{h^2}{8}\sum_{i=1}^{d}\lambda_i^4(|\bar{q}_i^{(0)}|^2 - |\bar{q}_i^{(n)}|^2) \xrightarrow{\mathcal{D}} \mathcal{N}(\alpha^4\mu_\rho, \alpha^4\sigma_\rho) \tag{C.9}$$

where

$$\mu_\rho = -\frac{1}{32}\int \lambda^4 \sin^2(\sqrt{\lambda}T)\mathrm{d}\nu_\rho(\lambda)$$

and

$$\sigma_\rho = \frac{1}{16}\int \lambda^6 \sin^2(\sqrt{\lambda}T)\mathrm{d}\nu_\rho(\lambda)\,.$$

Thus, the acceptance probability converges to

$$\mathbb{E}[\min\{1, \exp(\mathcal{N}(\alpha^4\mu_\rho, \alpha^4\sigma_\rho))\}]\,.$$

C.3. **Large d limit of acceptance probability for the Hamiltonian side move.** In this case, we have $D = 1$ and $B \in \mathbb{R}^{d\times 1}$. Indeed, $B^TB = \frac{1}{2d}(\mathbf{x}_j - \mathbf{x}_k)^T(\mathbf{x}_j - \mathbf{x}_k) \to 1$ almost surely as $d \to \infty$. So the eigenvalue λ_1^2 of B^TB converges to 1, and the rank of B^TB is $r = 1$.

Based on (C.4), the acceptance rate has the following formula

$$\min\left\{1, \exp(\frac{h^2}{8}\lambda_1^4(|\bar{q}_1^{(0)}|^2 - |\bar{q}_1^{(n)}|^2)\right\},$$

where $\bar{q}_1^{(0)} \sim \mathcal{N}(0, \lambda_1^2)$, $\bar{q}_1^{(n)} = \cos(n\phi)\bar{q}_1^{(0)} + \hat{\lambda}_1\sin(n\phi)\bar{p}_1^{(0)}$ with $\bar{p}_1^{(0)} \sim \mathcal{N}(0,1)$ and $\hat{\lambda}_1 = \frac{\lambda_1}{\sqrt{1-h^2\lambda_1^2/4}}$.

Taking $h = \alpha < 2$, using the formula (C.5), we then get as $d \to \infty$,

$$\frac{h^2}{8}\lambda_1^4(|\bar{q}_1^{(0)}|^2 - |\bar{q}_1^{(n)}|^2) \xrightarrow{a.s.} \frac{\alpha^2}{8}\left(\sin^2(n\phi)(z_1^2 - z_2^2) - \sin(2n\phi)z_1z_2\right) \tag{C.10}$$

where $z_1 \sim \mathcal{N}(0,1), z_2 \sim \mathcal{N}(0, \frac{1}{1-\alpha^2/4})$ are independent, and $\phi \in [0,\pi]$ such that $\cos\phi = 1 - \frac{\alpha^2}{2}$. The acceptance probability converges to

$$\mathbb{E}[\min\{1, \exp(\frac{\alpha^2}{8}\left(\sin^2(n\phi)(z_1^2 - z_2^2) - \sin(2n\phi)z_1z_2\right))\}]\,.$$

C.4. **Expected squared jumped distance.** Next, we derive the asymptotic limit of the expected squared jumped distance. We consider the general setting and then take specific values of B to obtain results for Hamiltonian walk and side moves.

The squared jumped distance for each iteration is

$$\|\mathbf{x}^{(n)} - \mathbf{x}^{(0)}\|_2^2 = \|B(B^TB)^+(\mathbf{q}^{(n)} - \mathbf{q}^{(0)})\|_2^2 = \sum_{i=1}^{r} \frac{1}{\lambda_i^2}|\bar{q}_i^{(n)} - \bar{q}_i^{(0)}|^2\,,$$

where we used the fact that $\|B(B^TB)^+(\mathbf{q}^{(n)} - \mathbf{q}^{(0)})\|_2^2 = (\mathbf{q}^{(n)} - \mathbf{q}^{(0)})^T(B^TB)^+(\mathbf{q}^{(n)} - \mathbf{q}^{(0)})$ and $B^TB = U\Sigma U^T \in \mathbb{R}^{D\times D}$, $\bar{\mathbf{q}}^{(l)} = U^T\mathbf{q}^{(l)}$ with $\bar{\mathbf{q}}^{(l)} = (\bar{q}_1^{(l)}, ..., \bar{q}_D^{(l)})$. We note that r is the rank of B^TB. More specifically, $r = d$ in the Hamiltonian walk move and $r = 1$ in the Hamiltonian side move.

The expected squared jumped distance in each iteration then admits the formula

$$\mathbb{E}[\left(\sum_{i=1}^{r} \frac{1}{\lambda_i^2}|\bar{q}_i^{(n)} - \bar{q}_i^{(0)}|^2\right) \min\{1, \exp(\frac{h^2}{8}\sum_{i=1}^{r}\lambda_i^4(|\bar{q}_i^{(0)}|^2 - |\bar{q}_i^{(n)}|^2))\}]\,.$$

We recall that $\bar{q}_i^{(0)} \sim \mathcal{N}(0, \lambda_i^2)$ are independent Gaussian random variables for each i, and

$$\bar{q}_i^{(n)} = \cos(n\phi_i)\bar{q}_i^{(0)} + \hat{\lambda}_i \sin(n\phi_i)\bar{p}_i^{(0)}$$

where $\bar{p}_i^{(0)}$ are independent standard normal distributions for each i.

First, we consider the Hamiltonian walk move. Here $r = d$. For each i,

$$\frac{|\bar{q}_i^{(n)} - \bar{q}_i^{(0)}|^2}{\lambda_i^2} = \frac{(\cos(n\phi_i) - 1)^2}{\lambda_i^2}(\bar{q}_i^{(0)})^2 + \frac{\hat{\lambda}_i^2}{\lambda_i^2}\sin^2(n\phi_i)(\bar{p}_i^{(0)})^2 + \frac{2\hat{\lambda}_i}{\lambda_i^2}(\cos(n\phi_i) - 1)\sin(n\phi_i)\bar{q}_i^{(0)}\bar{p}_i^{(0)}\,.$$

By writing N as a function of d and assuming $\lim_{d\to\infty}\frac{d}{N(d)/2} = \rho < 1$, and using law of large numbers, we get

$$\begin{aligned}\lim_{N\to\infty}\frac{1}{d}\sum_{i=1}^{d}\frac{1}{\lambda_i^2}|\bar{q}_i^{(n)} - \bar{q}_i^{(0)}|^2 &= \int \left((\cos(\sqrt{\lambda}T) - 1)^2 + \sin^2(\sqrt{\lambda}T)\right) \mathrm{d}\nu_\rho(\lambda) \\ &= \int 4\sin^2(\frac{\sqrt{\lambda}T}{2})\mathrm{d}\nu_\rho(\lambda)\,.\end{aligned}$$

Second, for the Hamiltonian side move, we have as $d \to \infty$,

$$\frac{|\bar{q}_1^{(n)} - \bar{q}_1^{(0)}|^2}{\lambda_1^2} \xrightarrow{a.s.} (\cos(n\phi) - 1)^2 z_1^2 + \sin^2(n\phi)z_2^2 + 2(\cos(n\phi) - 1)\sin(n\phi)z_1z_2\,,$$

where again, $z_1 \sim \mathcal{N}(0, 1), z_2 \sim \mathcal{N}(0, \frac{1}{1-\alpha^2/4})$ are independent, and $\phi \in [0, \pi]$ such that $\cos\phi = 1 - \frac{\alpha^2}{2}$. Thus the expected squared jumped distance is

$$\mathbb{E}[Q(z_1, z_2, n, \alpha)\min\{1, \exp(\frac{\alpha^2}{8}\left(\sin^2(n\phi)(z_1^2 - z_2^2) - \sin(2n\phi)z_1z_2\right))\}]$$

where

$$Q(z_1, z_2, n, \alpha) = (\cos(n\phi) - 1)^2 z_1^2 + \sin^2(n\phi)z_2^2 + 2(\cos(n\phi) - 1)\sin(n\phi)z_1z_2\,.$$

□