New Affine Invariant Ensemble Samplers

and their dimensional scaling

Yifan Chen

Courant Institute, NYU

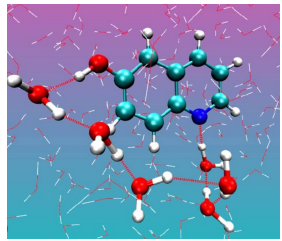
May 2025

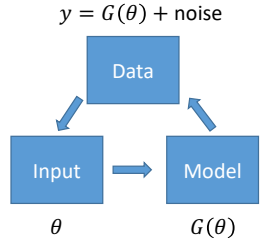
Sampling and MCMC

Sampling from probability distributions $\pi \propto \exp(-V)$ is a classical and fundamental challenge in scientific computing and statistics

Wide applications in

- Statistical physics
- Bayes inverse problems
- Uncertainty quantification
- Filtering
- ...





Markov chain Monte Carlo (MCMC) has been the workhorse for sampling problems

at a discrete m-step: $\mathbf{x}(m+1) = R(\mathbf{x}(m), \pi)$

where R is some random mapping. As $m \to \infty, \mathbf{x}(m)$ approximately draws from π

Anisotropy and affine invariance [Goodman, Weare 2010]

Probability distributions can be highly anisotropic or ill-conditioned, such as due to hierarchical Bayes or multiple scales in physical fields

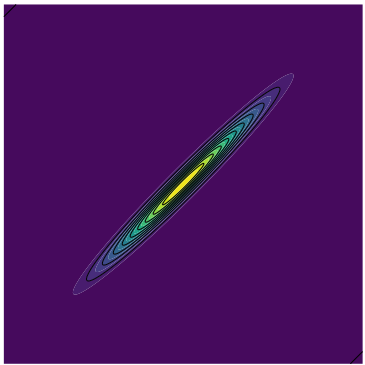


Figure: $\exp\left(-\frac{(x_1-x_2)^2}{2\epsilon} - \frac{(x_1+x_2)^2}{2}\right)$

- Ordinary MCMC may suffer from anisotropy
- **Affine invariance**: for any invertible affine transformation $\mathbf{y} = \phi(\mathbf{x}) = A\mathbf{x} + \mathbf{b}$

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

  Convergence behavior of $\mathbf{x}(m)$ toward π matches that of $\mathbf{y}(m)$ toward $\phi\#\pi$
- Concept motivated by affine invariance of optimization algorithms: Nelder-Mead simplex [Nelder, Mead 1965] and Newton

Ensemble affine invariant sampler [Goodman, Weare 2010]

Affine invariance in MCMC is widely achieved by ensemble samplers

Ensemble samplers: at a discrete m-step

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

Target distribution is π^N in the product space

Affine invariance: for any invertible affine transformation $\mathbf{y} = \phi(\mathbf{x}) = A\mathbf{x} + \mathbf{b}$

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

it holds that

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

Ensemble affine invariant sampler [Goodman, Weare 2010]

Affine invariance in MCMC is widely achieved by ensemble samplers

Stretch move sampler

- Ensemble at step $m \in \mathbb{N}$ $(\mathbf{x}_1(m), \mathbf{x}_2(m), ..., \mathbf{x}_N(m))$
- Stretch move randomly selects two $\mathbf{x}_i(m)$ and $\mathbf{x}_j(m)$

$$\tilde{\mathbf{x}}_i(m+1) = \mathbf{x}_j(m) + Z(\mathbf{x}_i(m) - \mathbf{x}_j(m))$$

  where density of Z satisfies $g(z) \propto \frac{1}{\sqrt{z}}$ for $z \in [a^{-1}, a]$; recommended $a = 2$
- Accept this proposal with probability

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

- Affine invariant since based on relative locations
- Used routinely in Bayesian applications: `emcee` package (>10k citations)

  [Foreman-Mackey, Hogg, Lang, Goodman 2013]

Challenges in high dimensions

Affine-invariant ensemble samplers are reported to behave well for moderate dimensions but suffer from higher dimensions (e.g. $d \geq 50$) [Huijser, Goodman, Brewer 2015]

"Ensemble methods are doomed to fail in high dimensions" [Carpenter 2017 (blog)]

- high-dimensional distributions typically concentrate on thin shells
- 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
- small stepsizes must be used!
- the samplers effectively "devolve into random walks with poorly biased directional choices"

This work: New affine invariant ensemble samplers and their dimensional scaling

1. **An ensemble side move sampler with better proposal in high dimensions**
   (derivative-free affine invariant samplers)
   - random walk scaling of stepsize $d^{-1/2}$ for Gaussian targets
   - outperform stretch move by a factor of two or more in autocorrelation time

2. **Affine invariant ensemble Hamiltonian Monte Carlo**
   (derivative-based affine invariant samplers)
   - better scaling of stepsize $d^{-1/4}$ for Gaussian targets
   - can outperform derivative-free ensemble samplers and HMC by order of magnitude

Basic ensemble side move sampler (parallel version in later slides)

Side move sampler

- Ensemble at step $m \in \mathbb{N}$: $(\mathbf{x}_1(m), \mathbf{x}_2(m), ..., \mathbf{x}_N(m))$
- Side move randomly selects one $\mathbf{x}_i(m)$ and two distinct $\mathbf{x}_j(m), \mathbf{x}_k(m) \neq \mathbf{x}_i(m)$

$$\tilde{\mathbf{x}}_i(m+1) = \mathbf{x}_i(m) + \sigma(\mathbf{x}_j(m) - \mathbf{x}_k(m))\xi, \quad \xi \sim \mathcal{N}(0,1)$$

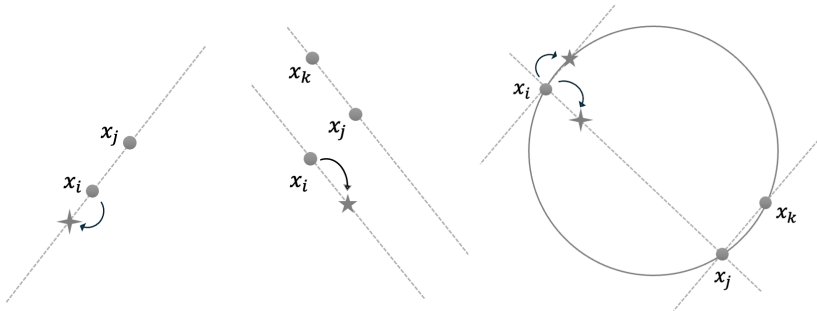
- Accept this proposal with probability

$$\min\{1, \frac{\pi(\tilde{\mathbf{x}}_i(m+1))}{\pi(\mathbf{x}_i(m))}\}$$

Connected to existing samplers: differential evolution and walk move (later slides)

Illustrations of stretch and side moves

1. Left: stretch move to a four-pointed star
2. Middle: side move to a five-pointed star



3. Right: one demo for both moves, for a ring-shaped distribution

Intuition: Side move may align better with the tangential directions

Connection to existing ensemble samplers

1. **Connection to differential evolution MCMC** [Ter Braak 2006]

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

where γ, σ are scalars and $\xi \sim \mathcal{N}(0, I_{d\times d})$

2. **Connection to walk move** [Goodman, Weare 2010]: select a subset S of particles (with mean $\mathbf{m}_S$) different from $\mathbf{x}_i(m)$. With i.i.d. $\xi_j \in \mathcal{N}(0,1)$, the proposal is

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

When $|S| = 2$, walk move is equivalent to side move with a specific step size since

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

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

Analysis of dimensional scaling for Gaussian targets

Proposition

Assume that $\mathbf{x}_i, \mathbf{x}_j, \mathbf{x}_k$ are independent draws from $\pi(\mathbf{x}) \propto \exp\left(-\frac{1}{2}\mathbf{x}^T\mathbf{x}\right), \mathbf{x} \in \mathbb{R}^d$:

- For 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

$$\lim_{d\to\infty} \mathbb{E}[\text{acceptance}] = \mathbb{E}[\min\{1, \exp\left(-\alpha^2\xi^2 - \sqrt{2}\alpha\xi z\right)\}] > 0$$

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

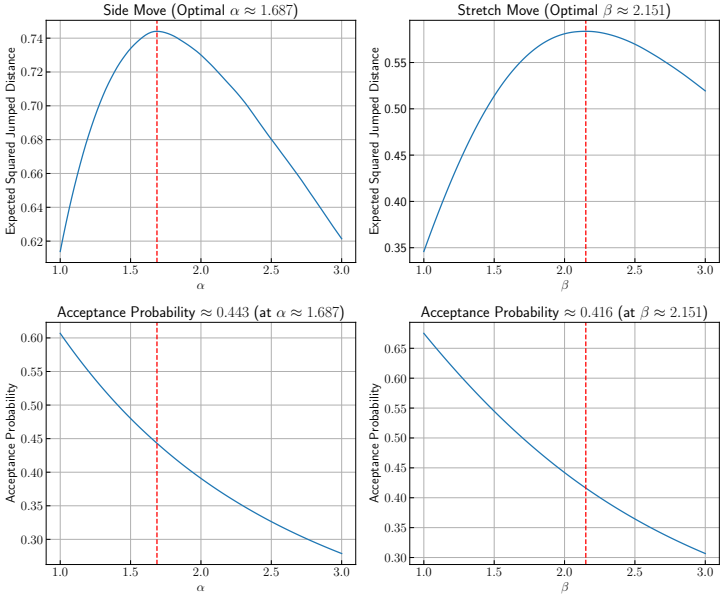
- For stretch move, $\tilde{\mathbf{x}}_i(m+1) = \mathbf{x}_j(m) + Z(\mathbf{x}_i(m) - \mathbf{x}_j(m))$, if $a = 1 + \frac{\beta}{\sqrt{d}}$, then

$$\lim_{d\to\infty} \mathbb{E}[\text{acceptance}] = \mathbb{E}[\min\{1, \exp\left(-\frac{3}{2}\beta^2U^2 - \sqrt{3}\beta U z\right)\}] > 0$$

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

- Random walk type scaling of stepsize: σ and $a - 1 \sim d^{-1/2}$
  Celebrated results for single-chain MCMC [Gelman, Gilks, Roberts 1997]

Optimize the expected squared jump distance (ESJD)



ESJD in one iteration:

[Pasarica, Gelman 2010]

$$\mathbb{E}[\|\mathbf{x}_i(m+1) - \mathbf{x}_i(m)\|_2^2]$$

Optimize over stepsize:

- For side move

$$\sigma^\star \approx \frac{1.687}{\sqrt{d}}$$

- For stretch move

$$a^\star - 1 \approx \frac{2.151}{\sqrt{d}}$$

Optimal ESJD:

side move > stretch move

Implementation: Parallel ensemble side move

Split ensemble approach [Foreman-Mackey, Hogg, Lang, Goodman 2013]

- Divide the ensemble into two groups:

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

- 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
- Iterate the above two steps

This approach preserves the correct detailed balance condition

Test criterion: integrated autocorrelation time in the stationary phase

Ensemble samplers generate sequences $(\mathbf{x}_1(m), ..., \mathbf{x}_N(m))$ for $1 \leq m \leq M$

- We 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

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

- 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 $\tau_e = \sum_{m=-\infty}^{+\infty}\frac{C_e(m)}{C_e(0)}$ 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 $\frac{C_e(m)}{C_e(0)}$

Numerical experiments on high dimensional Gaussians

Target: Gaussian distributions in d dims

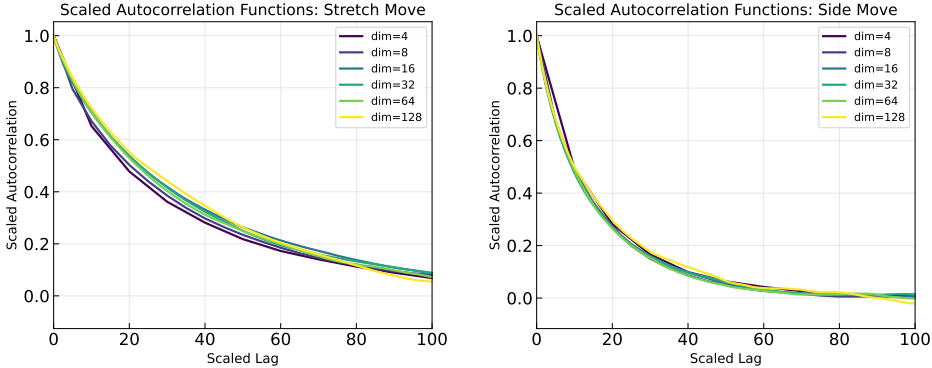


Figure: Scaled autocorrelation function for 1d position observable; Scaled lag $=$ original lag$/$dim$\times 4$

This implies the $O(d)$ scaling of autocorrelation function and autocorrelation time

Discussions

Both ensemble stretch and side moves are affine invariant and derivative free

- Optimal scaling of stepsize parameters $\sim d^{-1/2}$
- Side move outperforms stretch move in ESJD, and autocorrelation time in practice (more numerical examples at the end)
- Still, bottleneck of $O(d)$ autocorrelation time: intrinsic to random walk scaling

Discussions

Both ensemble stretch and side moves are affine invariant and derivative free

- Optimal scaling of stepsize parameters $\sim d^{-1/2}$
- Side move outperforms stretch move in ESJD, and autocorrelation time in practice (more numerical examples at the end)
- Still, bottleneck of $O(d)$ autocorrelation time: intrinsic to random walk scaling

Derivative-based samplers have been shown to scale better with dimension

- Metropolized Langevin stepsize $\sim d^{-1/3}$ [Roberts, Rosenthal 1998]
- Hamiltonian Monte Carlo stepsize $\sim d^{-1/4}$ [Beskos, Pillai, Roberts, Sanz-Serna, Stuart 2013]

Discussions

Both ensemble stretch and side moves are affine invariant and derivative free

- Optimal scaling of stepsize parameters $\sim d^{-1/2}$
- Side move outperforms stretch move in ESJD, and autocorrelation time in practice (more numerical examples at the end)
- Still, bottleneck of $O(d)$ autocorrelation time: intrinsic to random walk scaling

Derivative-based samplers have been shown to scale better with dimension

- Metropolized Langevin stepsize $\sim d^{-1/3}$ [Roberts, Rosenthal 1998]
- Hamiltonian Monte Carlo stepsize $\sim d^{-1/4}$ [Beskos, Pillai, Roberts, Sanz-Serna, Stuart 2013]

Many existing work to adapt these derivative-based samplers to anisotropy [Girolami, Calderhead 2011], [Martin, Wilcox, Burstedde, Ghattas 2012], [Greengard 2015], [Simsekli, Badeau, Cemgil, Richard 2016], [Leimkuhler, Matthews, Weare 2018], [Kleppe 2019], [Garbuno-Inigo, Hoffmann, Li, Stuart 2020], [Garbuno-Inigo, Nüsken, Reich 2020], [Hoffman, Sountsov 2022], [Tran, Kleppe 2024], etc.

This work: New affine invariant ensemble samplers and their dimensional scaling

1. An ensemble side move sampler with better proposal in high dimensions
   (derivative-free affine invariant samplers)
   - random walk scaling of stepsize $d^{-1/2}$ for Gaussian targets
   - outperform stretch move by a factor of two or more in autocorrelation time

2. Affine invariant ensemble Hamiltonian Monte Carlo
   (derivative-based affine invariant samplers)
   - better scaling $d^{-1/4}$ of stepsize for Gaussian targets
   - can outperform derivative-free ensemble samplers and HMC by order of magnitude

Hamiltonian Monte Carlo (HMC)

For $\pi \propto \exp(-V)$, HMC augments it to $\tilde{\pi} \propto \exp\big(-V(\mathbf{x}) - \frac{1}{2}\mathbf{p}^T M^{-1}\mathbf{p}\big)$

- Hamiltonian dynamics $\frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t} = M^{-1}\mathbf{p}, \quad \frac{\mathrm{d}\mathbf{p}}{\mathrm{d}t} = -\nabla V(\mathbf{x})$ keep $\tilde{\pi}$ invariant
- Discretization: L_h is one leapfrog with step size h, i.e., $(\mathbf{x}_h, \mathbf{p}_h) = \mathsf{L}_h(\mathbf{x}, \mathbf{p})$ with

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

- P is the momentum flip operator such that $\mathsf{P}(\mathbf{x}, \mathbf{p}) = (\mathbf{x}, -\mathbf{p})$

HMC algorithm: alternating between two steps

1. Sample a momentum $\mathbf{p} \sim \mathcal{N}(0, M)$
2. Propose an update $(\tilde{\mathbf{x}}, \tilde{\mathbf{p}}) = \mathsf{PL}_h^n(\mathbf{x}, \mathbf{p})$ and accept with probability

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

Importance of tuning of M, h, n: NUTS [Hoffman, Gelman 2014], `Stan` [Carpenter et al. 2017], etc.

First attempt: Covariance preconditioning using mass matrices

Attempt: The simplest idea is to set M^{-1} to be a covariance matrix

For the sake of parallel algorithm, we use split ensembles with two groups

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

The joint distribution ($\mathrm{Cov}_{S^{(0)}}, \mathrm{Cov}_{S^{(1)}}$: empirical covariance matrices of $S^{(0)}, S^{(1)}$)

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

does not preserve the correct marginal on $\mathbf{x}_i$

Resolution: need to add $-\frac{N}{4} \log\det \mathrm{Cov}_{S^{(0)}} - \frac{N}{4} \log\det \mathrm{Cov}_{S^{(1)}}$ to the potential

Issue: particles in each group must be accepted or rejected at the same time

Better idea: Antisymmetric preconditioning

Consider the simple joint distribution

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

Key observation: The following preconditioned Hamiltonian dynamics

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

leaves the distribution invariant

This is an antisymmetric preconditioning of gradient flow

$$\begin{bmatrix}\frac{\mathrm{d}\mathbf{x}_i}{\mathrm{d}t}\\ \frac{\mathrm{d}\mathbf{p}_i}{\mathrm{d}t}\end{bmatrix}=\underbrace{\begin{bmatrix}0 & B\\ -B^T & 0\end{bmatrix}}_{\text{antisymmetric}}\begin{bmatrix}\nabla V(\mathbf{x}_i)\\ \mathbf{p}_i\end{bmatrix}$$

Used in [Leimkuhler, Matthews, Weare 2018] for preconditioning underdamped Langevin

Affine invariant ensemble HMC

- **Fix particles in $S^{(1)}$, and update particles in $S^{(0)}$:** sample $\mathbf{p}_i \sim \mathcal{N}(0, I_{N/2\times N/2})$, run n steps of leapfrog of the dynamics and apply Metropolis

$$\frac{\mathrm{d}\mathbf{x}_i}{\mathrm{d}t} = B_{S^{(1)}}\mathbf{p}_i, \quad \frac{\mathrm{d}\mathbf{p}_i}{\mathrm{d}t} = -B_{S^{(1)}}^T \nabla V(\mathbf{x}_i), \quad 1 \le i \le N/2$$

with ($\mathbf{m}_{S^{(1)}}$ is the mean of all particles in $S^{(1)}$)

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

- **Fix particles in $S^{(0)}$, and update particles in $S^{(1)}$ similarly:** dynamic is now

$$\frac{\mathrm{d}\mathbf{x}_i}{\mathrm{d}t} = B_{S^{(0)}}\mathbf{p}_i, \quad \frac{\mathrm{d}\mathbf{p}_i}{\mathrm{d}t} = -B_{S^{(0)}}^T \nabla V(\mathbf{x}_i), \quad N/2+1 \le i \le N$$

with ($\mathbf{m}_{S^{(0)}}$ is the mean of all particles in $S^{(0)}$)

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

- Iterate the above two steps

Dimensional scaling analysis of the affine invariant ensemble HMC

Proposition: Assume that $\mathbf{x}_i$ are independent draws from $\pi(\mathbf{x}) \propto \exp\big(-\frac{1}{2}\mathbf{x}^T\mathbf{x}\big), \mathbf{x} \in \mathbb{R}^d$. If we take $h = \alpha d^{-1/4}, n = T/h$ and $\lim_{d\to\infty} \frac{d}{N/2} = \rho \in [0,1)$, then

$$\lim_{d\to\infty} \mathbb{E}[\text{acceptance}] = \mathbb{E}[\min\{1, \exp\big(\mathcal{N}(\alpha^4\mu_\rho, \alpha^4\sigma_\rho)\big)\}] > 0$$

where $\mathcal{N}(\alpha^4\mu_\rho, \alpha^4\sigma_\rho)$ is Gaussian with

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

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 $[b,c]$

Moreover, the **expected squared jumped distance (ESJD)** for one iteration satisfies

$\lim_{d\to\infty} \frac{1}{d}\text{ESJD} = c \in (0,\infty)$, in contrast to $\lim_{d\to\infty} \text{ESJD} = c' \in (0,\infty)$ for stretch/side moves

Variant: directional side move in ensemble HMC

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

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

- **Fix particles in $S^{(0)}$, and update particles in $S^{(1)}$ similarly**
- Iterate the above two steps

We call the algorithm Hamiltonian side move; the previous one Hamiltonian walk move

Variant: directional side move in ensemble HMC

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

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

- **Fix particles in $S^{(0)}$, and update particles in $S^{(1)}$ similarly**
- Iterate the above two steps

We call the algorithm Hamiltonian side move; the previous one Hamiltonian walk move

Proposition: Assume that $\mathbf{x}_i$ are independent draws from $\pi(\mathbf{x}) \propto \exp\left(-\frac{1}{2}\mathbf{x}^T\mathbf{x}\right), \mathbf{x} \in \mathbb{R}^d$. Let $h = \alpha < 2$ independent from d, then for one iteration:

$$\lim_{d\to\infty} \mathbb{E}[\text{acceptance}] > 0, \quad \lim_{d\to\infty} \text{ESJD} = c' \in (0,\infty)$$

Hamiltonian side move has the same scaling as the derivative-free side move

Numerical experiments: anisotropic Gaussian

Target: Gaussian distributions in d dims with condition number $\kappa = 10^3$

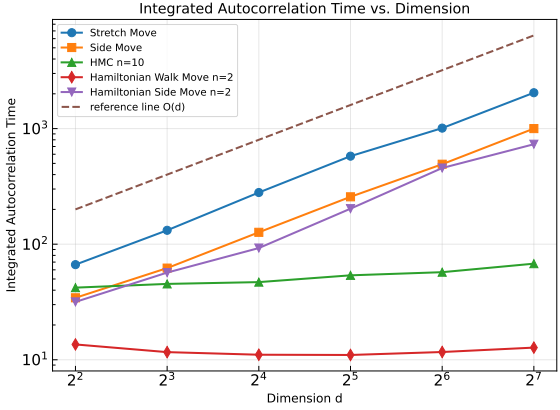


Figure: Autocorrelation time (1D position observable) versus dimension. For HMC and affine invariant HMC (Hamiltonian walk and side moves), the total integration time is $T = 1$ and n is the number of leapfrog steps

Numerical experiments: anisotropic Gaussian

Target: Gaussian distributions in $d = 128$ dims with condition number $\kappa = 10^3$

	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: 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 steps

Numerical experiments: physical fields (discretized by finite difference with d grid points)

$$\pi(u) \propto \exp\left(-\int_0^1 \tfrac{1}{2}(\partial_x u(x))^2 + V(u(x))\mathrm{d}x\right), \quad V(u) = (1-u^2)^2$$

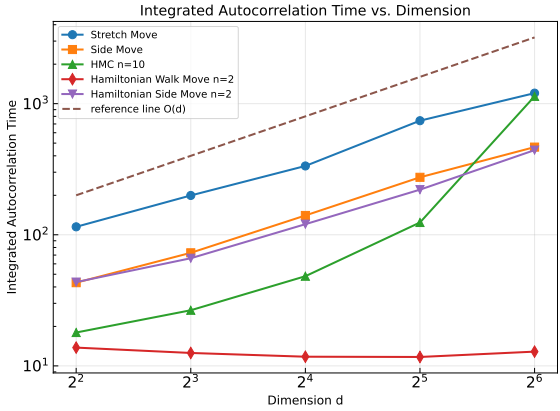


Figure: Autocorrelation time (integral observable $\int_0^1 u(x)\mathrm{d}x$) versus dimension. For HMC and affine invariant HMC (Hamiltonian walk and side moves), the total integration time is $T = 1$ and n is the number of leapfrog steps

Discussions

- **Side move**: more favorable high dimensional ensemble proposal than stretch move
  Nevertheless random walk scaling bottleneck applies
- **Affine invariant HMC**: derivative-based ensemble sampler
  Antisymmetric preconditioning and much better scaling for stepsize $h \sim d^{-1/4}$
  Further adaptation of stepsize and integration length?
- **Affine invariant Hamiltonian side move**: random walk scaling but slightly better than side move per iteration (further potential adaptation of derivative-free side moves?)
- Extension of analysis beyond Gaussians and stationary phases
- Broader invariance? Diffemorphism invariance of Fisher-Rao gradient flows and variational inference [Chentsov 1985], [Chen, Huang, Huang, Reich, Stuart 2023, 2024], etc.

Thank you!

Back-Up Slides

Numerical experiments: physical fields (discretized by finite difference with d grid points)

$$\pi(u) \propto \exp\left(-\int_0^1 \tfrac{1}{2}(\partial_x u(x))^2 + V(u(x))\mathrm{d}x\right), \quad V(u) = (1-u^2)^2$$

	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: 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 steps

Numerical experiments: ring-shaped distributions in d dimensions

$$\pi(\mathbf{x}) \propto \exp\left(-(\|\mathbf{x}\|_2^2 - 1)^2/l^2\right), l = 0.25, d = 50$$

	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: 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 steps