Geometry-Aware Particle Swarm Sampling

and connections to transformers

Hong Ye Tan[*†], Stanley Osher[†], Wuchen Li[‡]

[*]University of Cambridge, [†]UCLA, [‡]University of South Carolina

Level Set Meeting

Sep 29 2025

Markov Chain Monte Carlo methods (MCMC)

We wish to sample from

$$\pi(x) \propto \exp(-\beta V(x))$$

- $V(x)$ is a $\mathcal{C}^1$ potential function
- $\beta > 0$ a temperature parameter

Applications/related tasks:

- Uncertainty quantification,
- generative modelling, score matching,
- Bayesian inverse problems, Bayesian parameter estimation...

Langevin methods

Based on discretizations of the SDE (where W is a Wiener process)

$$\mathrm{d}X = -\nabla V(X)\mathrm{d}t + \sqrt{2\beta^{-1}}\mathrm{dW}$$

(Ex.) Euler–Maruyama $\rightarrow$ Unadjusted Langevin algorithm

$$X_{k+1} = X_k - \eta\nabla V(X_k) + \sqrt{2\beta^{-1}\eta}Z_k \qquad \text{(ULA)}$$

for step-size $\eta > 0$, where Z_k i.i.d. Gaussians.

- ULA converges to a biased stationary distribution for $\eta > 0$
- Adding Metropolis–Hastings correction step $\rightarrow$ Metropolis-adjusted Langevin algorithm (MALA)
  - Correction step ensures the correct stationary distribution
- Convergence from ergodic theory, using e.g., Poincaré or log-Sobolev inequality

Liouville equation

The density of the SDE (overdamped Langevin dynamics)

$$\mathrm{d}X = -\nabla V(X)\mathrm{d}t + \sqrt{2\beta^{-1}}\mathrm{dW} \qquad \text{(SDE)}$$

corresponds to the Fokker–Planck equation

$$\frac{\partial \rho}{\partial t} = \nabla \cdot (\nabla V(x)\rho) + \beta^{-1}\Delta\rho \qquad \text{(FP-ODE)}$$

which induces a deterministic particle evolution

$$\frac{\mathrm{d}X}{\mathrm{d}t} = -\nabla V(X) - \beta^{-1}\nabla \log \rho(X) \qquad \text{(Liouville)}$$

Challenge: what is the particle density at time t?

- Kernel density estimation
- Learned scores (e.g. diffusion models)

This work: based on regularized Wasserstein proximal operators (to be defined)

Liouville equation

The density of the SDE (overdamped Langevin dynamics)

$$\mathrm{d}X = -\nabla V(X)\mathrm{d}t + \sqrt{2\beta^{-1}}\mathrm{dW} \qquad \text{(SDE)}$$

corresponds to the Fokker–Planck equation

$$\frac{\partial \rho}{\partial t} = \nabla \cdot (\nabla V(x)\rho) + \beta^{-1}\Delta\rho \qquad \text{(FP-ODE)}$$

which induces a deterministic particle evolution

$$\frac{\mathrm{d}X}{\mathrm{d}t} = -\nabla V(X) - \beta^{-1}\nabla \log \rho(X) \qquad \text{(Liouville)}$$

Challenge: what is the particle density at time t?

- Kernel density estimation
- Learned scores (e.g. diffusion models)

This work: based on regularized Wasserstein proximal operators (to be defined)

Talk structure

1. **Application and numerics**
   - Sampling algorithm
   - Connections to transformers
   - Convergence rate
   - Examples
2. **Derivation**
   - Regularized Wasserstein proximal defined as coupled PDEs
   - Discretizing the Liouville equation from a modified Fokker–Planck equation

Transforming the ODE

$$\frac{\mathrm{d}X}{\mathrm{d}t} = -\nabla V(X) - \beta^{-1} \nabla \log \rho(X)$$

Replace $\nabla \log \rho(X)$ with the "regularized Wasserstein proximal" $\nabla \log \mathrm{WProx}\, \rho(X)$ defined later:

$$\frac{\mathrm{d}X}{\mathrm{d}t} = -\nabla V(X) - \beta^{-1} \nabla \log \mathrm{WProx}\, \rho(X) \tag{1}$$

Transforming the ODE

$$\frac{\mathrm{d}X}{\mathrm{d}t} = -\nabla V(X) - \beta^{-1} \nabla \log \rho(X)$$

Replace $\nabla \log \rho(X)$ with the "regularized Wasserstein proximal" $\nabla \log \mathrm{WProx}\, \rho(X)$ defined later:

$$\frac{\mathrm{d}X}{\mathrm{d}t} = -\nabla V(X) - \beta^{-1} \nabla \log \mathrm{WProx}\, \rho(X) \tag{1}$$

Facts: (derive later)

- $\mathrm{WProx}\, \rho$ admits an approximable *kernel formula*;
- Liouville equation (1) arises as the continuous limit of the kernel formula.

After applying a particular semi-implicit time discretization, we obtain...

Preconditioned BRWP algorithm

BRWP[1]: "Backwards Regularized Wasserstein Proximal".
Sampling $\propto \exp(-\beta V(x))$ for collection of particles $\mathbf{X} = \begin{bmatrix}\mathbf{x}_1 & \dots & \mathbf{x}_N\end{bmatrix} \in \mathbb{R}^{d\times N}$:

$$\mathbf{X}^{(k+1)} = \mathbf{X}^{(k)} - \frac{\eta}{2} M \underbrace{\nabla V(\mathbf{X}^{(k)})}_{\text{dynamics}} + \frac{\eta}{2T} \underbrace{\left(\mathbf{X}^{(k)} - \mathbf{X}^{(k)}\mathrm{softmax}(W^{(k)})^\top\right)}_{\text{diffusion}}$$

where interaction matrix

$$W_{ij} = -\beta \frac{\|\mathbf{x}_i - \mathbf{x}_j\|_M^2}{4T} - \log \int_{\mathbb{R}^d} e^{-\frac{\beta}{2}(V(z) + \frac{\|z-\mathbf{x}_j\|_M^2}{2T})} \, \mathrm{d}z \,.$$

and $M \in \mathrm{Sym}_{++}(\mathbb{R}^d)$ is some preconditioning matrix.

[1] Tan, Osher, L. *Noise-free sampling algorithms via regularized Wasserstein proximals*.

Preconditioned BRWP algorithm

PBRWP: "Preconditioned BRWP".
Sampling $\propto \exp(-\beta V(x))$ for collection of particles $\mathbf{X} = \begin{bmatrix} \mathbf{x}_1 & \dots & \mathbf{x}_N \end{bmatrix} \in \mathbb{R}^{d \times N}$:

$$\mathbf{X}^{(k+1)} = \mathbf{X}^{(k)} - \frac{\eta}{2} M \underbrace{\nabla V(\mathbf{X}^{(k)})}_{\text{dynamics}} + \frac{\eta}{2T} \underbrace{\left(\mathbf{X}^{(k)} - \mathbf{X}^{(k)} \mathrm{softmax}(W^{(k)})^\top\right)}_{\text{diffusion}}$$

where interaction matrix

$$W_{ij} = -\beta \frac{\|\mathbf{x}_i - \mathbf{x}_j\|_M^2}{4T} - \log \int_{\mathbb{R}^d} e^{-\frac{\beta}{2}(V(z) + \frac{\|z - \mathbf{x}_j\|_M^2}{2T})} \, \mathrm{d}z \,.$$

and $M \in \mathrm{Sym}_{++}(\mathbb{R}^d)$ is some preconditioning matrix.

Preconditioned BRWP algorithm

PBRWP: "Preconditioned BRWP".
Sampling $\propto \exp(-\beta V(x))$ for collection of particles $\mathbf{X} = \begin{bmatrix} \mathbf{x}_1 & \dots & \mathbf{x}_N \end{bmatrix} \in \mathbb{R}^{d \times N}$:

$$\mathbf{X}^{(k+1)} = \mathbf{X}^{(k)} - \frac{\eta}{2} M \underbrace{\nabla V(\mathbf{X}^{(k)})}_{\text{dynamics}} + \frac{\eta}{2T} \underbrace{\left(\mathbf{X}^{(k)} - \mathbf{X}^{(k)} \mathrm{softmax}(W^{(k)})^\top\right)}_{\text{diffusion}}$$

where interaction matrix

$$W_{ij} = -\beta \frac{\|\mathbf{x}_i - \mathbf{x}_j\|_M^2}{4T} - \log \int_{\mathbb{R}^d} e^{-\frac{\beta}{2}(V(z) + \frac{\|z - \mathbf{x}_j\|_M^2}{2T})} \, \mathrm{d}z \,.$$

and $M \in \mathrm{Sym}_{++}(\mathbb{R}^d)$ is some preconditioning matrix.

- Preconditioning happens **inside** the diffusion.

Comparison: Mirror Langevin Algorithm (MLA)

$$X_{k+1} = X_k - \eta \nabla V(X_k) + \sqrt{2\beta^{-1}\eta} Z_k \qquad \text{(ULA)}$$

ULA can also be preconditioned to obtain MLA

$$X_{k+1} = X_k - \eta M \nabla V(X_k) + \sqrt{2\beta^{-1}\eta} \sqrt{M} Z_k \qquad \text{(MLA)}$$

Difference in diffusion:

MLA	$\mathcal{N}(0, 2\beta^{-1}\eta M)$
PBRWP	$\mathbf{X}^{(k)} - \mathbf{X}^{(k)} \mathrm{softmax}(W^{(k)})^\top$

The preconditioner affects the inter-particle diffusion weights.

Comparison: Mirror Langevin Algorithm (MLA)

$$X_{k+1} = X_k - \eta \nabla V(X_k) + \sqrt{2\beta^{-1}\eta} Z_k \qquad \text{(ULA)}$$

ULA can also be preconditioned to obtain MLA

$$X_{k+1} = X_k - \eta M \nabla V(X_k) + \sqrt{2\beta^{-1}\eta}\sqrt{M} Z_k \qquad \text{(MLA)}$$

Difference in diffusion:

MLA	$\mathcal{N}(0, 2\beta^{-1}\eta M)$
PBRWP	$\mathbf{X}^{(k)} - \mathbf{X}^{(k)}\mathrm{softmax}(W^{(k)})^\top$

The preconditioner affects the inter-particle diffusion weights.
Where does the softmax come from?

Comparison: Mirror Langevin Algorithm (MLA)

$$X_{k+1} = X_k - \eta \nabla V(X_k) + \sqrt{2\beta^{-1}\eta} Z_k \qquad \text{(ULA)}$$

ULA can also be preconditioned to obtain MLA

$$X_{k+1} = X_k - \eta M \nabla V(X_k) + \sqrt{2\beta^{-1}\eta} \sqrt{M} Z_k \qquad \text{(MLA)}$$

Difference in diffusion:

MLA	$\mathcal{N}(0, 2\beta^{-1}\eta M)$
PBRWP	$\mathbf{X}^{(k)} - \mathbf{X}^{(k)} \mathrm{softmax}(W^{(k)})^\top$

The preconditioner affects the inter-particle diffusion weights.
Where does the softmax come from? Exactly $\log \mathrm{WProx}\ \rho$.

Relation with kernel methods

Consider the Gaussian kernel $k(x,y) = \exp\bigl(-\|x-y\|^2/2T\bigr)$ with bandwidth T. For points $\{\mathbf{x}_i\}_{i=1}^N$,

$$\rho_{\mathsf{KDE}}(\mathbf{x}_i) = \frac{1}{N}\sum_{j=1}^{N} \frac{\exp\left[-\frac{\|\mathbf{x}_i-\mathbf{x}_j\|^2}{2T}\right]}{(2\pi T)^{d/2}}$$

$$\rho_{\mathsf{RWPO}}(\mathbf{x}_i) = \frac{1}{N}\sum_{j=1}^{N} \frac{\exp\left[-\frac{\beta}{2}\left(V(\mathbf{x}_i) + \frac{\|\mathbf{x}_i-\mathbf{x}_j\|_M^2}{2T}\right)\right]}{\mathcal{Z}(\mathbf{x}_j)}$$

Recall approximate Liouville equation:

$$\frac{\mathrm{d}X}{\mathrm{d}t} = -\nabla V(X) - \beta^{-1}\nabla \log \rho_{\mathsf{approx}}(X)$$

Relation with kernel methods

Consider the Gaussian kernel $k(x,y) = \exp\left(-\|x-y\|^2/2T\right)$ with bandwidth T. For points $\{\mathbf{x}_i\}_{i=1}^N$,

$$\rho_{\mathsf{KDE}}(\mathbf{x}_i) = \frac{1}{N}\sum_{j=1}^{N} \frac{\exp\left[-\frac{\|\mathbf{x}_i-\mathbf{x}_j\|^2}{2T}\right]}{(2\pi T)^{d/2}}$$

$$\rho_{\mathsf{RWPO}}(\mathbf{x}_i) = \frac{1}{N}\sum_{j=1}^{N} \frac{\exp\left[-\frac{\beta}{2}\left(V(\mathbf{x}_i) + \frac{\|\mathbf{x}_i-\mathbf{x}_j\|_M^2}{2T}\right)\right]}{\mathcal{Z}(\mathbf{x}_j)}$$

Main differences:

- Usage of V inside kernel
- Normalizing constant $\mathcal{Z}$

N.B. Both can be written as a transformer structure

Transformer Attention Diffusion

Transformer structure (up to scaling):

$$\mathrm{Attn}(Q;K,V) = V\mathrm{softmax}\left(Q^\top K\right)^\top.$$

Self attention: $\mathbf{X} \mapsto \mathrm{Attn}(Q(\mathbf{X});K(\mathbf{X}),V(\mathbf{X}))$

$$\mathbf{X}^{(k+1)} = \mathbf{X}^{(k)} - \frac{\eta}{2}M\nabla V(\mathbf{X}^{(k)}) + \frac{\eta}{2T}\left(\mathbf{X}^{(k)} - \mathbf{X}^{(k)}\mathrm{softmax}(W^{(k)})^\top\right)$$

$$W_{ij}^{(k)} = -\beta\frac{\|\mathbf{x}_i - \mathbf{x}_j\|_M^2}{4T} - \underbrace{\log\int_{\mathbb{R}^d} e^{-\frac{\beta}{2}(V(z)+\frac{\|z-\mathbf{x}_j\|_M^2}{2T})}\,\mathrm{d}z}_{=:\log\mathcal{Z}(\mathbf{x}_j)}.$$

Diffusion rewritten as masked-attention structure:

$$\text{(Red)} = \mathrm{softmax}(Q^\top K - \mathbf{1}\mathbf{z}^\top)^\top,$$

$$Q^\top K = -\frac{\beta}{2T}\mathbf{X}^\top M^{-1}\mathbf{X},\ \mathbf{z}_j = \log\mathcal{Z}(\mathbf{x}_j) + \beta\frac{\|\mathbf{x}_j\|_M^2}{4T},\ V = \mathbf{X}.$$

Accelerated diffusion

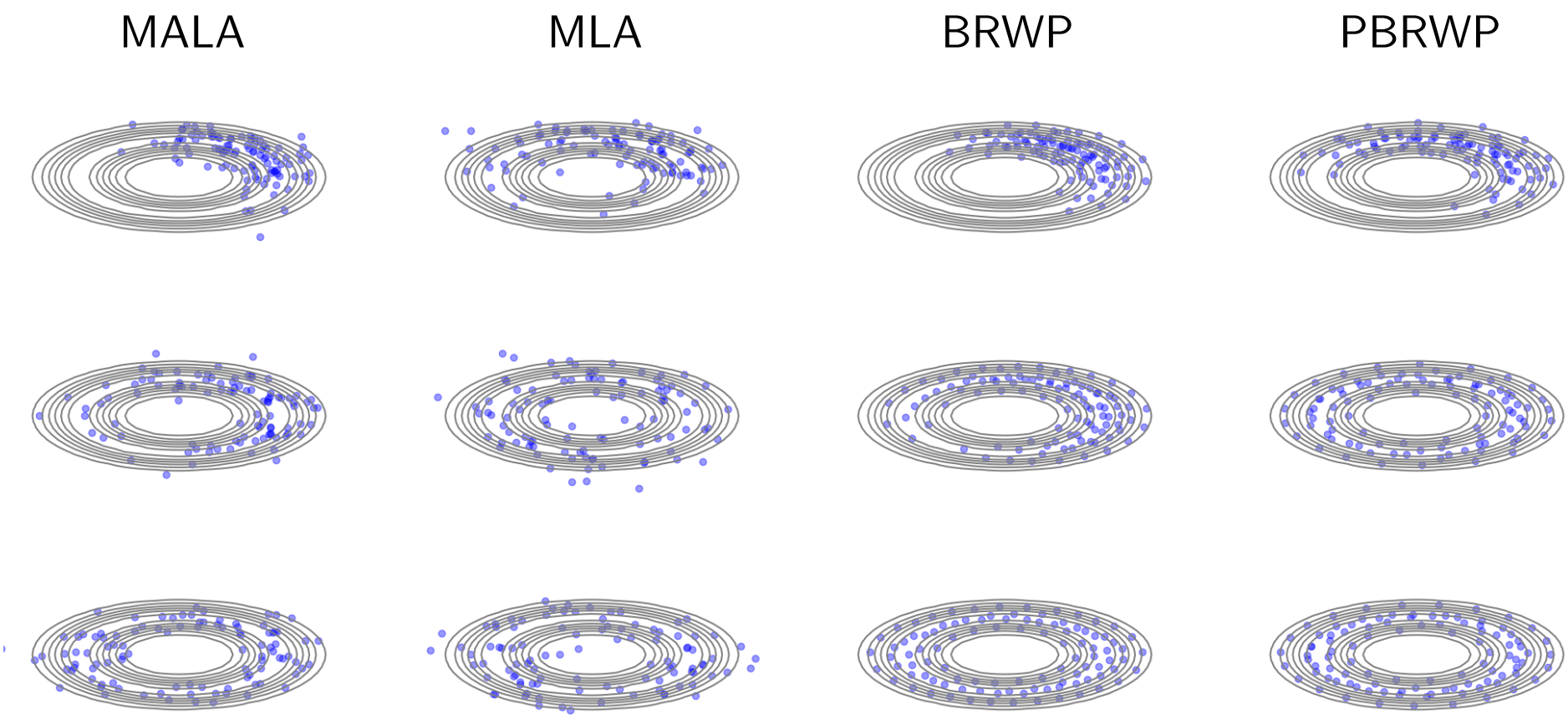


Figure: Evolution of the various methods for the stretched annulus at iterations 10, 50, and 200.

Quantitative evidence

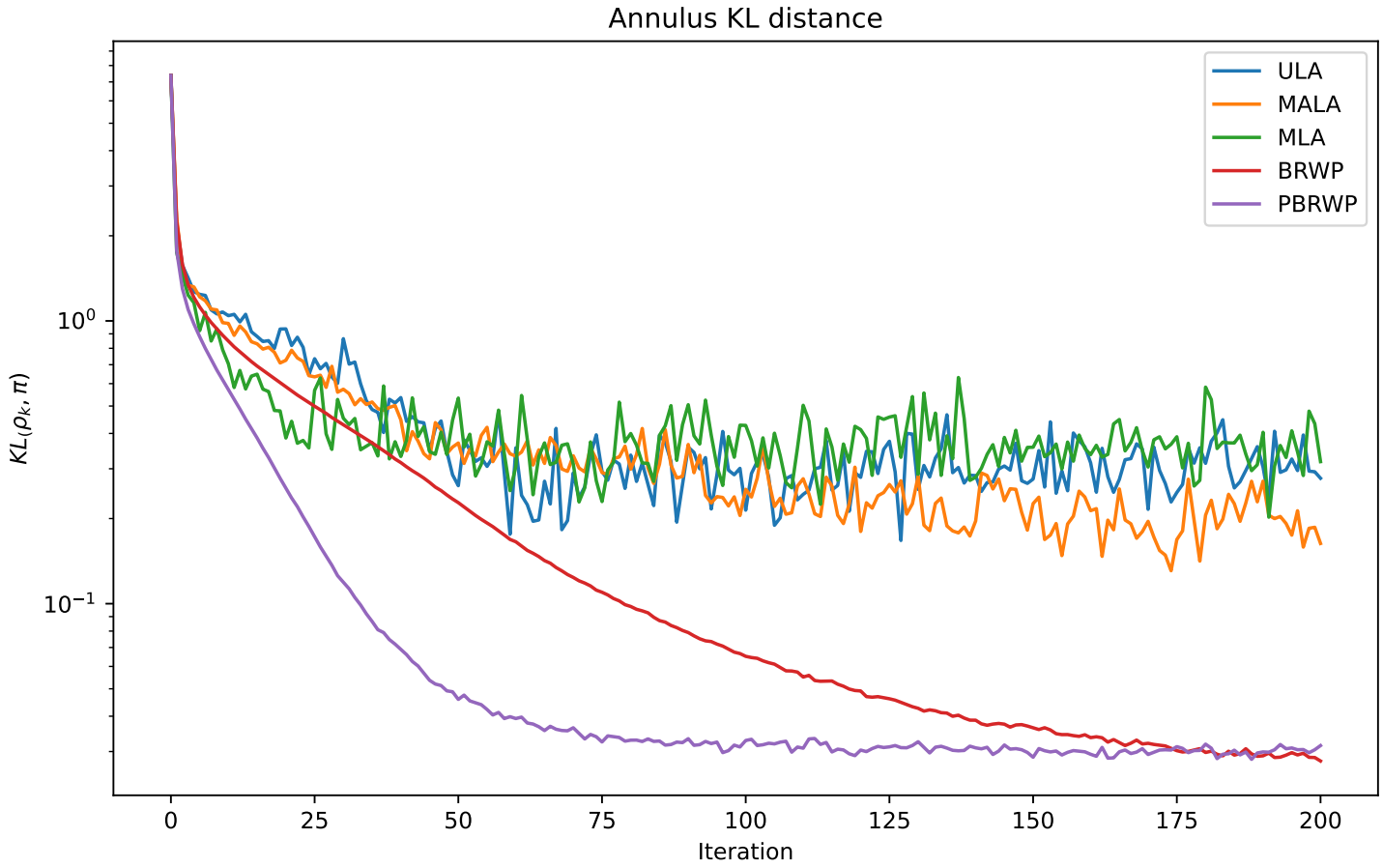
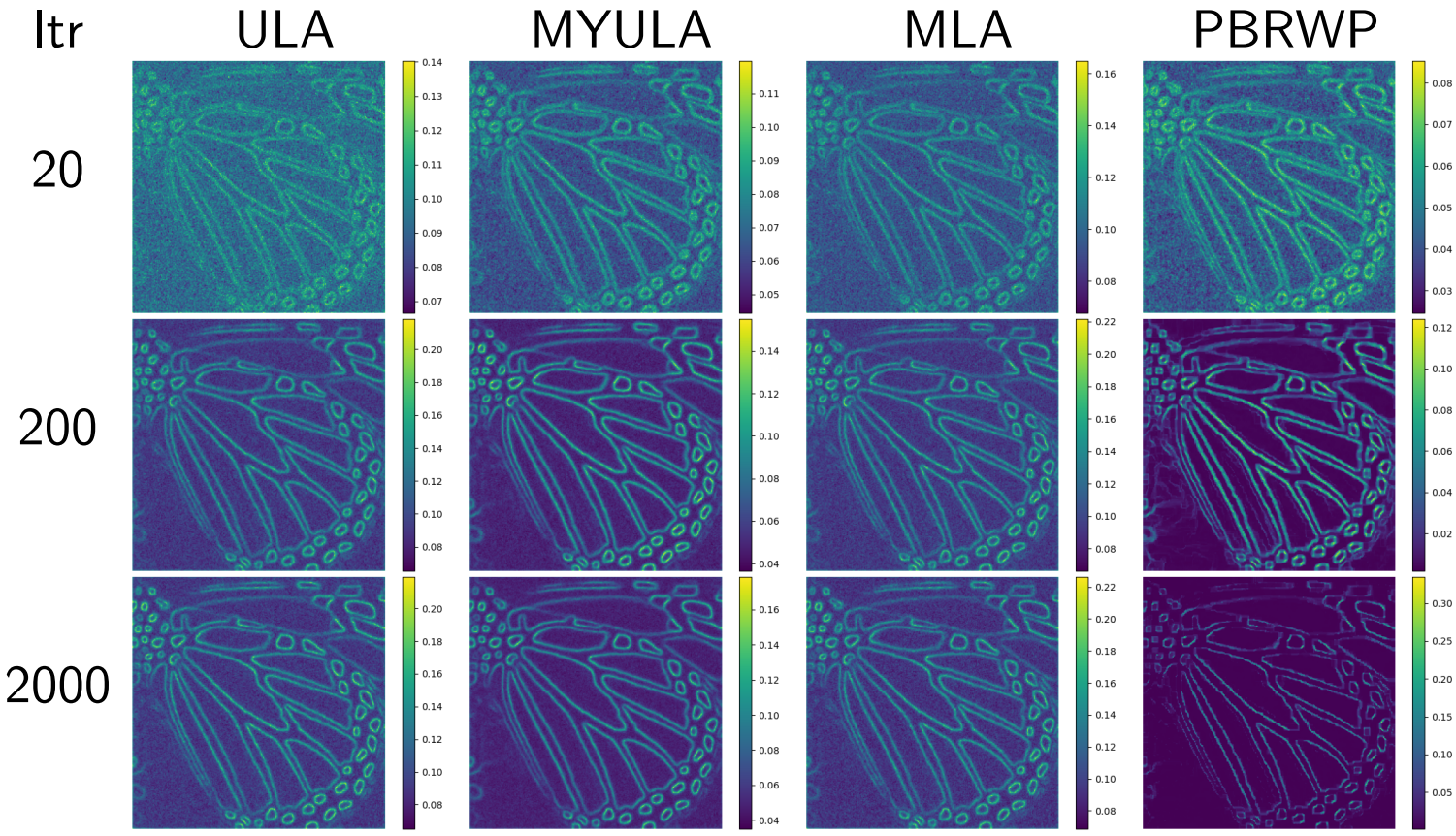


Figure: The KL distance to the ground truth converges faster (using Gaussian bandwidth estimator)

High-dimensional deconvolution

$$V(x) = \frac{1}{2}\|Ax - y\|^2 + \lambda\,\mathrm{TV}(x)$$

where A is a convolution operator and y is a corrupted image.
Precondition with A^*A. Std for 40 particles:



Discrete-time convergence of PBRWP

Fact: For quadratic potentials, Gaussians stay Gaussian under PBRWP.

Theorem

Consider the potential $V = x^\top \Sigma^{-1} x/2$, corresponding to target stationary distribution $\pi \sim \mathcal{N}(0, \Sigma)$. Suppose the preconditioner satisfies $cM \preceq \Sigma \preceq CM$, and let $T \in (0, c)$. Then:

Discrete-time convergence of PBRWP

Fact: For quadratic potentials, Gaussians stay Gaussian under PBRWP.

Theorem

Consider the potential $V = x^\top \Sigma^{-1} x/2$, corresponding to target stationary distribution $\pi \sim \mathcal{N}(0, \Sigma)$. Suppose the preconditioner satisfies $cM \preceq \Sigma \preceq CM$, and let $T \in (0, c)$. Then:

1. *The invariant distribution $\hat{\pi}$ of PBRWP satisfies $\mathrm{WProx}\, \hat{\pi} = \pi$,*

- Bias is characterized by inverting the regularized Wasserstein proximal operator

Discrete-time convergence of PBRWP

Fact: For quadratic potentials, Gaussians stay Gaussian under PBRWP.

Theorem

Consider the potential $V = x^\top \Sigma^{-1} x/2$, *corresponding to target stationary distribution* $\pi \sim \mathcal{N}(0, \Sigma)$. *Suppose the preconditioner satisfies* $cM \preceq \Sigma \preceq CM$, *and let* $T \in (0, c)$. *Then:*

1. *The invariant distribution* $\hat{\pi}$ *of PBRWP satisfies* $\mathrm{WProx}\, \hat{\pi} = \pi$,
2. *For sufficiently small step-size* $\eta > 0$ *(closed form), the PBRWP iterations converge as follows, where* $\tilde{\rho}_k = \mathrm{WProx}\, \rho(X_k)$,

$$\begin{aligned} &\mathrm{D}_{\mathrm{KL}}(\tilde{\rho}_{k+1} \| \pi) - \mathrm{D}_{\mathrm{KL}}(\tilde{\rho}_k \| \pi) \\ &\leq -\frac{\eta}{2C[\beta + 2T(1 + TC^{-1})^{-1}(1 + Tc^{-1})^2 \lambda^{-1}]} \mathrm{D}_{\mathrm{KL}}(\tilde{\rho}_k \| \pi). \end{aligned} \quad (2)$$

- Bias is characterized by inverting the regularized Wasserstein proximal operator

Verifying the bias

Problem: sampling from a 2D standard Gaussian

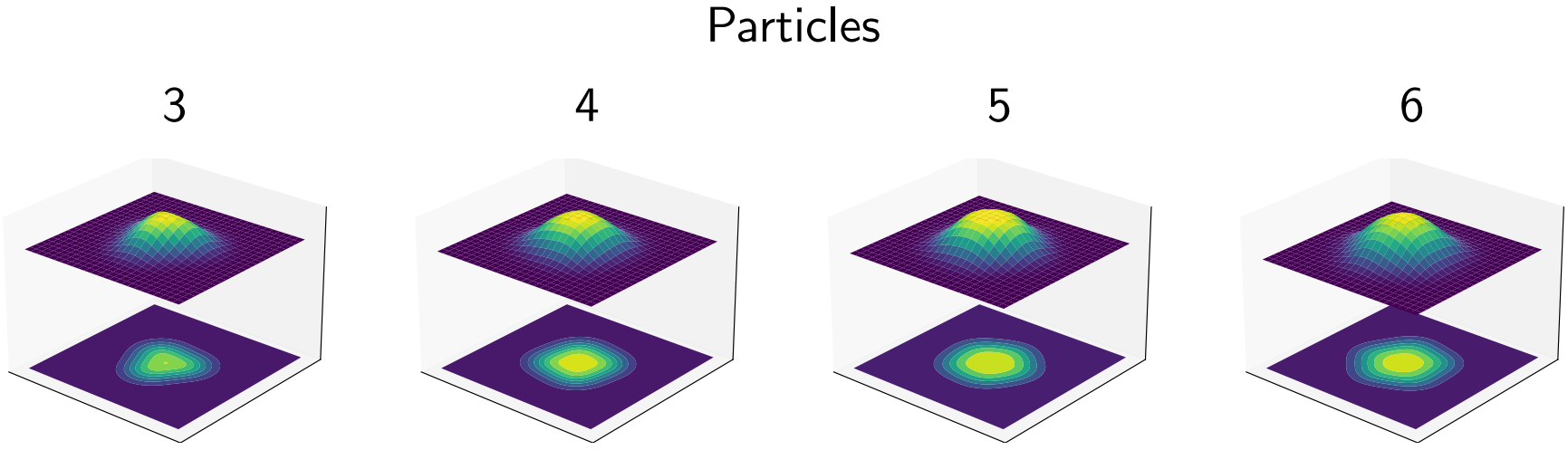


Figure: Densities of the regularized Wasserstein proximal $\mathrm{WProx}^{I}_{0.2I}$ for the 2-dimensional standard Gaussian at iteration 100, done with $n \in \{3, 4, 5, 6\}$ particles. Density of the Wasserstein proximal gradually becomes more spherical and Gaussian-like.

Observation: The regularized Wasserstein proximal of the empirical distribution approaches the standard Gaussian.

Defining the Regularized Wasserstein Proximal

What is WProx? Adding Laplacian regularization to the Benamou–Brenier formulation:

$$\begin{cases} \partial_t \rho(t,x) + \nabla_x \cdot (\rho(t,x) \nabla_x \Phi(t,x)) = \beta^{-1} \Delta_x \rho(t,x), \\ \partial_t \Phi(t,x) + \frac{1}{2} \|\nabla_x \Phi(t,x)\|^2 = -\beta^{-1} \Delta_x \Phi(t,x), \\ \rho(0,x) = \rho_0(x), \quad \Phi(T,x) = -V(x). \end{cases}$$

The terminal solution yields a kernel representation, denoted $\mathrm{WProx}_{T,V}$

$$\mathrm{WProx}_{T,V}\, \rho(x) := \rho(T,x) = \int_{\mathbb{R}^d} K(x,y) \rho(y)\, \mathrm{d}y,$$

$$K(x,y) = \frac{\exp\left(-\frac{\beta}{2}\left(V(x) + \frac{\|x-y\|^2}{2T}\right)\right)}{\int_{\mathbb{R}^d} \exp\left(-\frac{\beta}{2}\left(V(z) + \frac{\|z-y\|^2}{2T}\right)\right) \mathrm{d}z}.$$

K is convolution with a *heat kernel*.

Cole–Hopf transform

The regularized Benamou–Brenier formulation arises from coupled heat equations:

$$\begin{cases} \partial_t \rho(t,x) + \nabla_x \cdot (\rho(t,x) \nabla_x \Phi(t,x)) = \beta^{-1} \Delta_x \rho(t,x), \\ \partial_t \Phi(t,x) + \frac{1}{2} \|\nabla_x \Phi(t,x)\|^2 = -\beta^{-1} \Delta_x \Phi(t,x), \\ \rho(0,x) = \rho_0(x), \quad \Phi(T,x) = -V(x) \end{cases}$$

$$\Updownarrow$$

$$\begin{cases} \partial_t \hat{\eta}(t,x) = \beta^{-1} \Delta \hat{\eta}(t,x), \\ \partial_t \eta(t,x) = -\beta^{-1} \Delta \eta(t,x), \\ \eta(0,x)\hat{\eta}(0,x) = \rho_0(x), \quad \eta(T,x) = e^{-\beta V(x)/2}. \end{cases}$$

The coupled heat equations give rise to the kernel formulation.

Preconditioning

Goal: we want a different norm in the kernel.
Question: What is the corresponding PDE system?
To use a different kernel, $M \in \mathbb{R}^{d\times d}$ symmetric +ve def,

$$K_M(x,y) = \frac{\exp\left(-\frac{1}{2\beta}(V(x) + \frac{\|x-y\|_M^2}{2T})\right)}{\int_{\mathbb{R}^d} \exp\left(-\frac{1}{2\beta}(V(z) + \frac{\|z-y\|_M^2}{2T})\right) \mathrm{d}z}.$$

Anisotropic heat kernel K_M is Green's function for anisotropic heat eq.

$$\partial_t u = \nabla \cdot (M \nabla u)$$

Derivation

By using Cole–Hopf transform on coupled *anisotropic* heat equations

$$\begin{cases} \partial_t \hat{\eta}(t,x) = \beta^{-1} \nabla \cdot (M \nabla \hat{\eta}(t,x)) \\ \partial_t \eta(t,x) = -\beta^{-1} \nabla \cdot (M \nabla \eta(t,x)), \\ \eta(0,x)\hat{\eta}(0,x) = \rho_0(x), \quad \eta(T,x) = e^{-\beta V(x)/2} \end{cases}$$

$$\Downarrow$$

$$\begin{cases} \partial_t \rho(t,x) + \nabla \cdot (\rho(t,x) \nabla \Phi(t, M^{-1}x)) = \beta^{-1} \nabla \cdot (M \nabla \rho)(t,x) \\ \partial_t \Phi(t, M^{-1}x) + \frac{1}{2} \|\nabla \Phi(t, M^{-1}x)\|_M^2 = -\beta^{-1} \operatorname{Tr}\big(M^{-1} (\nabla^2 \Phi)(t, M^{-1}x)\big) \\ \rho(0,x) = \rho_0(x), \quad \Phi(T, M^{-1}x) = -V(x) \end{cases}$$

- Changing the norm $\Leftrightarrow$ changing the PDE regularization.
- Admits a kernel formula. **Our score approximator is computable.**

Time discretization

The first equation is a modified Fokker–Planck equation:

$$\partial_t \rho(t,x) + \nabla \cdot (\rho(t,x) \nabla \Phi(t, M^{-1}x)) = \beta^{-1} \nabla \cdot (M \nabla \rho)(t,x) \quad (3)$$

which corresponds to the particle evolution

$$\frac{\mathrm{d}X}{\mathrm{d}t} = \nabla \Phi(t, M^{-1}X) - \beta^{-1} M \nabla \log \rho(t,X). \quad (4)$$

Use:

Time discretization

The first equation is a modified Fokker–Planck equation:

$$\partial_t \rho(t,x) + \nabla \cdot (\rho(t,x) \nabla \Phi(t, M^{-1}x)) = \beta^{-1} \nabla \cdot (M \nabla \rho)(t,x) \tag{3}$$

which corresponds to the particle evolution

$$\frac{\mathrm{d}X}{\mathrm{d}t} = \nabla \Phi(t, M^{-1}X) - \beta^{-1} M \nabla \log \rho(t,X). \tag{4}$$

Use:

1. Boundary condition $\nabla \Phi(T, M^{-1}X) = -M \nabla V(x)$

Time discretization

The first equation is a modified Fokker–Planck equation:

$$\partial_t \rho(t,x) + \nabla \cdot (\rho(t,x) \nabla \Phi(t, M^{-1}x)) = \beta^{-1} \nabla \cdot (M \nabla \rho)(t,x) \quad (3)$$

which corresponds to the particle evolution

$$\frac{\mathrm{d}X}{\mathrm{d}t} = \nabla \Phi(t, M^{-1}X) - \beta^{-1} M \nabla \log \rho(t,X). \quad (4)$$

Use:

1. Boundary condition $\nabla \Phi(T, M^{-1}X) = -M \nabla V(x)$
2. Solution $\rho(T,X) = \mathrm{WProx}_T \, \rho_0(X)$ (*kernel formula*)

Then using a semi-implicit discretization, the particle evolution is

$$X_{k+1} = X_k + \eta \left(-M \nabla V(X_k) - \beta^{-1} M \nabla \log \mathrm{WProx}^M_{T,V} \, \rho_k(X_k) \right) \quad (5)$$

Getting something computable

$$X_{k+1} = X_k + \eta \left(-M \nabla V(X_k) - \beta^{-1} M \nabla \log \mathrm{WProx}^{M}_{T,V}\, \rho_k(X_k) \right)$$

$$\mathbf{X}^{(k+1)} = \mathbf{X}^{(k)} - \frac{\eta}{2} M \nabla V(\mathbf{X}^{(k)}) + \frac{\eta}{2T} \left(\mathbf{X}^{(k)} - \mathbf{X}^{(k)} \mathrm{softmax}(W^{(k)})^{\top} \right).$$

Getting something computable

$$X_{k+1} = X_k + \eta \left(-M \nabla V(X_k) - \beta^{-1} M \nabla \log \mathrm{WProx}^M_{T,V}\, \rho_k(X_k) \right)$$

using the kernel formula

$$\mathrm{WProx}^M_{T,V}\, \rho(x) = \int_{\mathbb{R}^d} K_M(x,y) \rho(y)\, \mathrm{d}y.$$

$$\mathbf{X}^{(k+1)} = \mathbf{X}^{(k)} - \frac{\eta}{2} M \nabla V(\mathbf{X}^{(k)}) + \frac{\eta}{2T} \left(\mathbf{X}^{(k)} - \mathbf{X}^{(k)} \mathrm{softmax}(W^{(k)})^\top \right).$$

Getting something computable

$$X_{k+1} = X_k + \eta \left(-M \nabla V(X_k) - \beta^{-1} M \nabla \log \mathrm{WProx}^M_{T,V}\, \rho_k(X_k) \right)$$

using the kernel formula

$$\mathrm{WProx}^M_{T,V}\, \rho(x) = \int_{\mathbb{R}^d} K_M(x,y) \rho(y)\, \mathrm{d}y.$$

For some particles $\{\mathbf{x}_1, ..., \mathbf{x}_N\}$, empirical dist. $\rho = \frac{1}{N} \sum_{l=1}^N \delta(\mathbf{x}_l)$,

$$\begin{aligned} \mathrm{WProx}^M_{T,V}\, \rho(\mathbf{x}_i) &= \frac{1}{N} \sum_{j=1}^N K_M(\mathbf{x}_i, \mathbf{x}_j) \\ &= \frac{1}{N} \sum_{j=1}^N \exp\left(-\frac{1}{2\beta} \left(V(\mathbf{x}_i) + \frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2_M}{2T} \right) - \log \mathcal{Z}(\mathbf{x}_j) \right) \end{aligned}$$

$$\mathbf{X}^{(k+1)} = \mathbf{X}^{(k)} - \frac{\eta}{2} M \nabla V(\mathbf{X}^{(k)}) + \frac{\eta}{2T} \left(\mathbf{X}^{(k)} - \mathbf{X}^{(k)} \mathrm{softmax}(W^{(k)})^\top \right).$$

Getting something computable

$$X_{k+1} = X_k + \eta \left(-M \nabla V(X_k) - \beta^{-1} M \nabla \log \mathrm{WProx}^M_{T,V}\, \rho_k(X_k) \right)$$

using the kernel formula

$$\mathrm{WProx}^M_{T,V}\, \rho(x) = \int_{\mathbb{R}^d} K_M(x,y) \rho(y)\, \mathrm{d}y.$$

For some particles $\{\mathbf{x}_1, ..., \mathbf{x}_N\}$, empirical dist. $\rho = \frac{1}{N} \sum_{l=1}^N \delta(\mathbf{x}_l)$,

$$\begin{aligned} \mathrm{WProx}^M_{T,V}\, \rho(\mathbf{x}_i) &= \frac{1}{N} \sum_{j=1}^N K_M(\mathbf{x}_i, \mathbf{x}_j) \\ &= \frac{1}{N} \sum_{j=1}^N \exp\left(-\frac{1}{2\beta} \left(V(\mathbf{x}_i) + \frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2_M}{2T} \right) - \log \mathcal{Z}(\mathbf{x}_j) \right) \end{aligned}$$

Differentiate w.r.t. $\mathbf{x}_i$ to yield the iterations

$$\mathbf{X}^{(k+1)} = \mathbf{X}^{(k)} - \frac{\eta}{2} M \nabla V(\mathbf{X}^{(k)}) + \frac{\eta}{2T} \left(\mathbf{X}^{(k)} - \mathbf{X}^{(k)} \mathrm{softmax}(W^{(k)})^\top \right).$$

Summary

- We present a *principled density estimator* based on regularized Wasserstein proximal
- The diffusive term is a self-attention block
- Preconditioning the kernel corresponds to modified second-order regularization
  - Derived using a Cole–Hopf transform
  - Accelerated convergence
- Discrete-time convergence for quadratic potential

Future work:

- Discrete particle dynamics - explaining the structure
- Convergence for more general distributions
- Position-dependent preconditioning?

High-dimensional modifications

As in transformers, which take

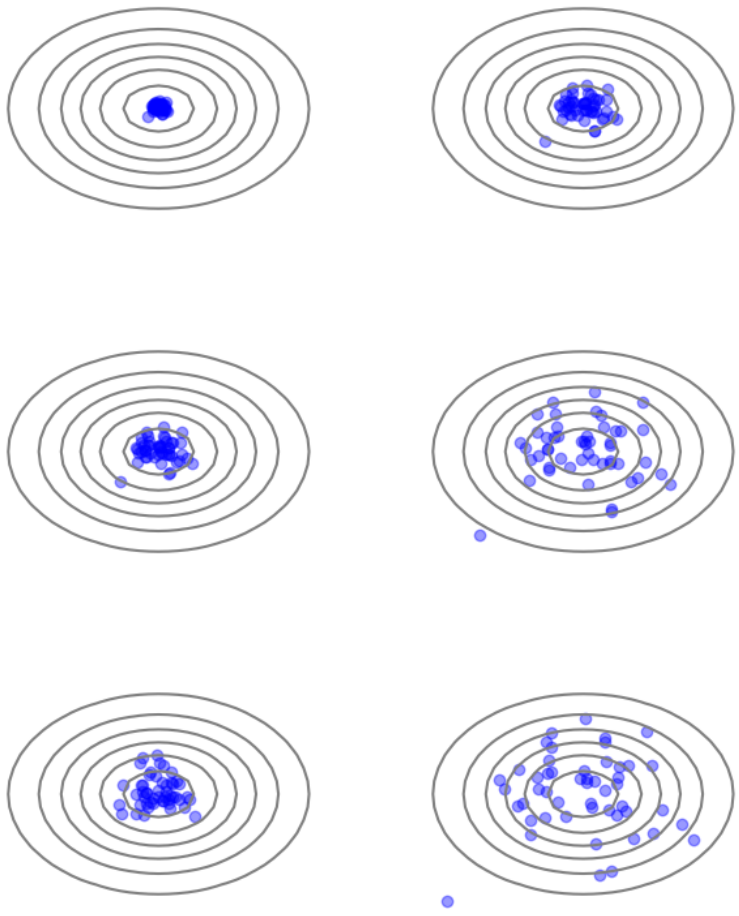
$$\mathrm{Attn}(Q;K,V) = V\mathrm{softmax}\left(Q^\top K/\sqrt{d}\right)^\top, \tag{6}$$

we may directly take the diffusion $\beta = \sqrt{d}$ to prevent mode collapse.

Ill-conditioned Gaussian

50 dimensions, condition number 50.

$\beta = 1$	$\beta = d^{-1/2}$



Bayesian neural networks

We can empirically use variable preconditioners.

Table: Test root-mean-square-error (RMSE) on test datasets on various Bayesian neural network tasks. Bold indicates smallest in row. We observe that the adaptive Fisher preconditioned BRWP uniformly outperforms BRWP on each of the BNN tasks. Adam and the noise-free methods both generally exhibit high variance in this setting, which may be due to the relatively small neural network architecture and sensitivity to initialization. We may further interpret the high variance of these methods as being able to find better trained models.

Dataset	Adam	PBRWP	BRWP	AIG	WGF	SVGD
Boston	$3.350_{\pm 8.33e-1}$	$2.866_{\pm 5.94e-1}$	$3.309_{\pm 5.31e-1}$	$2.871_{\pm 3.41e-3}$	$3.077_{\pm 5.52e-3}$	$\mathbf{2.775_{\pm 3.78e-3}}$
Combined	$3.971_{\pm 1.79e-1}$	$\mathbf{3.925_{\pm 1.52e-1}}$	$3.975_{\pm 3.94e-2}$	$4.067_{\pm 9.27e-1}$	$4.077_{\pm 3.85e-4}$	$4.070_{\pm 2.02e-4}$
Concrete	$4.698_{\pm 4.85e-1}$	$\mathbf{4.387_{\pm 4.88e-1}}$	$4.478_{\pm 2.05e-1}$	$4.440_{\pm 1.34e-1}$	$4.883_{\pm 1.93e-1}$	$4.888_{\pm 1.39e-1}$
Kin8nm	$0.089_{\pm 2.72e-3}$	$\mathbf{0.087_{\pm 2.67e-3}}$	$0.089_{\pm 6.06e-6}$	$0.094_{\pm 5.56e-6}$	$0.096_{\pm 3.36e-5}$	$0.095_{\pm 1.32e-5}$
Wine	$0.629_{\pm 4.01e-2}$	$0.612_{\pm 4.17e-2}$	$0.623_{\pm 1.35e-3}$	$0.606_{\pm 1.40e-5}$	$0.614_{\pm 3.48e-4}$	$\mathbf{0.604_{\pm 9.89e-5}}$

Adam preconditioner

Algorithm Adam-based Preconditioner

Data: Objective function f, exponential decay rates $\beta_2 = 0.999$, point sequence $(x^{(l)})_{l \geq 1}$, epsilon $\epsilon = 0.001$.

Result: Preconditioners $M^{(k)}$, where $M^{(k)} = M^{(k)}(x^{(1)}, ..., x^{(k)})$.

$v_0 \leftarrow 0$; `// initialize second moment vector`

for $k = 1, ..., K$ **do**

$g_k \leftarrow \nabla f(x^{(k)})$; `// compute gradient`

$v_k \leftarrow \beta_2 v_{k-1} + (1 - \beta_2) g_k^2$; `// update second moment estimate`

$\hat{v}_k \leftarrow v_k / (1 - \beta_2^k)$; `// bias correction`

$M^{(k)} = \mathbf{diag}(1/(\sqrt{\hat{v}_k} + \epsilon))$; `// construct preconditioning matrix`

end

return $(M^{(k)})_{k=1}^{K}$.

Other works

Application of regularized Wasserstein proximal to accelerated density ODEs:

$$\begin{cases} \mathrm{d}X = P\mathrm{d}t, \\ \mathrm{d}P = -\alpha P\mathrm{d}t - \nabla V(X)\mathrm{d}t - \nabla \log \rho(X)\mathrm{d}t \end{cases}$$

Wasserstein proximal/Benamou–Brenier

The Wasserstein proximal with linear energy and potential V is

$$\mathrm{WProx}_{T,V}(\rho_0) := \operatorname*{argmin}_{q \in \mathcal{P}_2(\mathbb{R}^d)} \int_{\mathbb{R}^d} V(x) q(x) \, \mathrm{d}x + \frac{\mathcal{W}(\rho_0, q)^2}{2T}. \tag{7}$$

This equivalently is

$$\begin{cases} \partial_t \rho(t,x) + \nabla_x \cdot (\rho(t,x) \nabla_x \Phi(t,x)) = 0 \\ \partial_t \Phi(t,x) + \frac{1}{2} \|\nabla_x \Phi(t,x)\|^2 = 0 \\ \rho(0,x) = \rho_0(x), \quad \Phi(T,x) = -V(x). \end{cases} \tag{8}$$

a combination of a (forward time) Fokker–Planck equation and (backward time) HJ equation.

Cole–Hopf transform (1)

The kernel

$$G_{t,M}(x,y) := \frac{1}{(4\pi\beta^{-1}t)^{d/2}|M|^{1/2}} e^{-\beta\frac{(x-y)^\top M^{-1}(x-y)}{4t}}. \tag{9}$$

is a Green's function for

$$\begin{cases} \partial_t u - \beta^{-1}\nabla\cdot(M\nabla u) = 0, \\ u(0,x) = \delta(y). \end{cases}$$

We consider the coupled forward-backward anisotropic

$$\begin{cases} \partial_t \hat{\eta}(t,x) = \beta^{-1}\nabla\cdot(M\nabla\hat{\eta}(t,x)), \\ \partial_t \eta(t,x) = -\beta^{-1}\nabla\cdot(M\nabla\eta(t,x)), \\ \eta(0,x)\hat{\eta}(0,x) = \rho_0(x), \quad \eta(T,x) = e^{\beta\Phi(T,x)/2} = e^{-\beta V(x)/2}. \end{cases} \tag{10}$$

From B.C., $\eta(0)$ has a kernel formula, so $\hat{\eta}(T)$ has a kernel formula.

Cole–Hopf transform (2)

The Cole–Hopf transform is

$$\begin{cases} \eta(t,x) = e^{\beta\Phi(t,M^{-1}x)/2} \\ \hat{\eta}(t,x) = \rho(t,x)e^{-\beta\Phi(t,M^{-1}x)/2} \end{cases} \Leftrightarrow \begin{cases} \Phi(t,x) = 2\beta^{-1}\log\eta(t,Mx) \\ \rho(t,x) = \eta(t,x)\hat{\eta}(t,x) \end{cases}. \tag{11}$$

Under this, we have

$$\partial_t\eta(t,x) = -\beta^{-1}\nabla\cdot(M\nabla\eta(t,x))$$
$$\Downarrow$$
$$\partial_t\Phi(t,M^{-1}x) + \frac{1}{2}\|\nabla\Phi(t,M^{-1}x)\|_M^2 = -\beta^{-1}\operatorname{Tr}\big(M^{-1}(\nabla^2\Phi)(t,M^{-1}x)\big),$$

$$\partial_t\hat{\eta}(t,x) = \beta^{-1}\nabla\cdot(M\nabla\hat{\eta}(t,x))$$
$$\Downarrow$$
$$\partial_t\rho + \nabla\cdot(\rho(t,x)\nabla\Phi(t,M^{-1}x)) = \beta^{-1}\nabla\cdot(M\nabla\rho)$$