

# Accelerated Stein Variational Gradient Flow

Viktor Stein, Wuchen Li

**Goal:** sample density  $\pi \propto e^{-V}$  with potential  $V: \mathbb{R}^d \rightarrow \mathbb{R}$ , e.g., for Bayesian inference and uncertainty quantification.

**Contribution.** Novel interacting particle algorithm converging faster than SVGD with deterministic behavior, without need for score estimation, e.g. by kernel density estimation (KDE).

## Stein Variational Gradient Descent (SVGD)

Liu & Wang, NeurIPS'16

$K: \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}$  symmetric, positive definite, differentiable "kernel".

SVGD update in the  $n$ -th iteration with step size  $\tau > 0$ :

$$x_i^{n+1} \leftarrow x_i^n - \frac{\tau}{N} \left( \underbrace{\sum_{j=1}^N K(x_i^n, x_j^n) \nabla V(x_j^n)}_{\text{attraction}} - \underbrace{\sum_{j=1}^N \nabla_2 K(x_i^n, x_j^n)}_{\text{repulsion}} \right), \quad i \in \{1, \dots, N\}.$$

SVGD is gradient descent of  $\text{KL}(\cdot \mid \pi)$  with respect to a "Stein geometry" (induced by  $K$ ) when replacing  $\rho$  by a sample average, where  $\text{KL}(\rho \mid \pi) := \int_{\mathbb{R}^d} \rho(x) \log \left( \frac{\rho(x)}{\pi(x)} \right) dx$ .

We focus on: generalized bilinear kernel:  $K(x, y) = x^T A y + 1$ ,  $A \in \text{Sym}_+(d)$ , Gauss kernel  $K(x, y) := \exp(-\frac{1}{2\sigma^2} \|x - y\|_2^2)$ ,  $\sigma > 0$ .

## Nesterov's accelerated gradient descent

Nesterov, 1983

To find minimizer  $x^*$  of convex, differentiable function  $f: \mathbb{R}^d \rightarrow \mathbb{R}$  with  $L$ -Lipschitz gradient use gradient descent  $x^{k+1} \leftarrow x^k - \tau \nabla f(x^k)$  with step size  $\tau \in (0, \frac{2}{L})$

converges linearly:  $f(x^k) - f(x^*) \in O(k^{-1})$ .

**Nesterov's accelerated gradient descent:**

$$\begin{cases} x^{k+1} = y^k - \tau \nabla f(y^k) & \text{gradient step} \\ y^{k+1} = x^{k+1} + \alpha_{k+1}(x^{k+1} - x^k) & \text{momentum step} \end{cases}$$

converges quadratically:  $f(x^k) - f(x^*) \in O(k^{-2})$ .

**Damping:**  $\alpha_k = \frac{k-1}{k+2}$  or  $\alpha_k = \frac{\sqrt{L}-\sqrt{\beta}}{\sqrt{L}+\sqrt{\beta}}$  if  $f$  is  $\beta$ -strongly convex.

Then,  $f(x^k) - f(x^*) \in O\left(\left(1 - \sqrt{\frac{\beta}{L}}\right)^k\right)$  vs.  $O\left(\left(1 - \frac{\beta}{L}\right)^k\right)$  for gradient descent.

## The density manifold

Lafferty, 1988; Otto, 2001

The set of smooth positive probability densities

$$\tilde{\mathcal{P}}(\Omega) := \left\{ \rho \in \mathcal{C}^\infty(\Omega) : \rho(x) > 0 \forall x \in \Omega, \int_\Omega \rho(x) dx = 1 \right\}.$$

forms an infinite-dimensional  $C^\infty$ -Fréchet manifold (under regularity assumptions on  $\Omega$ ) with (co)tangent spaces  $T_\rho \tilde{\mathcal{P}}(\Omega) := \{\sigma \in \mathcal{C}^\infty(\Omega) : \int_\Omega \sigma(x) dx = 0\}$  and  $T_\rho^* \tilde{\mathcal{P}}(\Omega) := \mathcal{C}^\infty(\Omega)/\mathbb{R}$ .

**Metric tensor field** on  $\tilde{\mathcal{P}}(\Omega)$  = smooth map  $G: \tilde{\mathcal{P}}(\Omega) \ni \rho \mapsto G_\rho: T_\rho \tilde{\mathcal{P}}(\Omega) \leftrightarrow T_\rho^* \tilde{\mathcal{P}}(\Omega)$ .

## Gradient flows in the density manifold

If it exists, the first functional derivative of  $E: \tilde{\mathcal{P}}(\Omega) \rightarrow \mathbb{R}$  is  $\delta E: \tilde{\mathcal{P}}(\Omega) \rightarrow \mathcal{C}^\infty(\Omega)/\mathbb{R}$  with

$$\langle \delta E(\rho), \phi \rangle_{L^2(\Omega)} = \partial_t \Big|_{t=0} E(\rho + t\phi), \quad \forall \phi \in \mathcal{C}^\infty(\Omega) : \rho + t\phi \in \tilde{\mathcal{P}}(\Omega), |t| \text{ small.}$$

A smooth curve  $\rho: [0, \infty) \rightarrow \tilde{\mathcal{P}}(\Omega)$ ,  $t \mapsto \rho_t$  is a  $(\tilde{\mathcal{P}}(\Omega), G)$ -gradient flow of  $E$  starting at  $\rho(0)$  if

$$\partial_t \rho_t = -G_{\rho_t}^{-1}[\delta E(\rho_t)], \quad \forall t > 0.$$

**Stein metric** defined via:

$$(G_\rho^{(K)})^{-1}(\phi) := \left( x \mapsto -\nabla_x \cdot \left( \rho(x) \int_\Omega K(x, y) \rho(y) \nabla \Phi(y) dy \right) \right).$$

The  $(\tilde{\mathcal{P}}(\Omega), G^{(K)})$ -gradient flow of  $E = \text{KL}(\cdot \mid Z^{-1}e^{-V})$  is

$$\partial_t \rho_t(x) = \nabla_x \cdot \left( \rho_t(x) \int_\Omega (K(x, y) \nabla V(y) - \nabla_2 K(x, y)) \rho_t(y) dy \right).$$

## Accelerated Stein variational gradient flow on densities

The Hamiltonian on  $\tilde{\mathcal{P}}(\Omega)$  is

$$H: T\tilde{\mathcal{P}}(\Omega) \rightarrow \mathbb{R} \cup \{\infty\}, \quad (\rho, \Phi) \mapsto \frac{1}{2} \int_\Omega \Phi(x) G_\rho^{-1}[\Phi](x) dx + E(\rho).$$

$\rho \triangleq$  position,  $\Phi \triangleq$  momentum.

SVGD is accelerated Stein variational gradient flow is

$$\begin{cases} \partial_t \rho_t + \nabla \cdot (\rho_t \int_\Omega K(\cdot, y) \rho_t(y) \nabla \Phi_t(y) dy) = 0, \\ \partial_t \Phi_t + \alpha_t \Phi_t + \int_\Omega K(y, \cdot) \langle \nabla \Phi_t(y), \nabla \Phi_t(\cdot) \rangle \rho_t(y) dy + \delta E(\rho_t) = 0. \end{cases}$$

## Discretization of ASVGD

To avoid score-estimation, use particle momentum  $Y: (0, \infty) \rightarrow \mathbb{R}^d$ ,  $t \mapsto \dot{Y}_t$

$$\dot{Y}_t = Y_t = \int_\Omega K(X_t, y) \nabla \Phi_t(y) \rho_t(y) dy.$$

Key idea of SVGD, using integration by parts to shift derivative from density  $\rho$  to the kernel  $K$ , can also be applied in the accelerated scheme:

**Lemma (ASVG flow with particles' momenta)**

The associated deterministic interacting particle system is

$$\begin{cases} \dot{X}_t = Y_t, \\ \dot{Y}_t = -\alpha_t Y_t + \int_\Omega (K(X_t, y) \nabla V(y) - \nabla_2 K(X_t, y)) \rho_t(y) dy + \int_{\Omega^2} \langle \nabla \Phi_t(z), \nabla \Phi_t(y) \rangle \\ \quad \cdot \left[ K(y, z) (\nabla_2 K)(X_t, y) + K(X_t, z) (\nabla_1 K)(X_t, y) - K(X_t, y) (\nabla_2 K)(z, y) \right] \rho_t(y) dy \rho_t(z) dz. \end{cases}$$

Then, replace expectation w.r.t.  $\rho_t$  by sample averages.

## Accelerated Stein variational gradient descent algorithm

**Data:** Number of particles  $N \in \mathbb{N}$ , number of steps  $T \in \mathbb{N}$ , step sizes  $\tau > 0$ , target score function  $\nabla V: \mathbb{R}^d \rightarrow \mathbb{R}^d$ . Either a symmetric positive definite matrix  $A \in \mathbb{R}^{n \times n}$  for bilinear kernel or a bandwidth  $\sigma^2 > 0$  for Gaussian kernel, Wasserstein regularization parameter  $\varepsilon \geq 0$ .

**Result:** Matrix  $X^{T_{\max}}$ , whose rows are particles that approximate the target distribution  $\pi \sim \exp(-V)$ .

**Step 0.** Initialize  $Y^0 = 0 \in \mathbb{R}^{N \times d}$ .

**for**  $k = 0, \dots, T_{\max} - 1$  **do**

**Step 1.** Update particle positions using particle momenta.

$$X^{k+1} \leftarrow X^k + \sqrt{\tau} Y^k.$$

**Step 2.** Form kernel matrix and update momentum in density space.

$$K^{k+1} = (K(X_i^{k+1}, X_j^{k+1}))_{i,j=1}^N, \quad M^{k+1} \leftarrow N(K^{k+1} + \varepsilon \text{id}_N)^{-1} Y^k.$$

**Step 3.** Update damping parameter  $\alpha^k$  using speed restart and/or gradient restart for each particle.

**Step 4.** Update momenta.

For bilinear kernel:

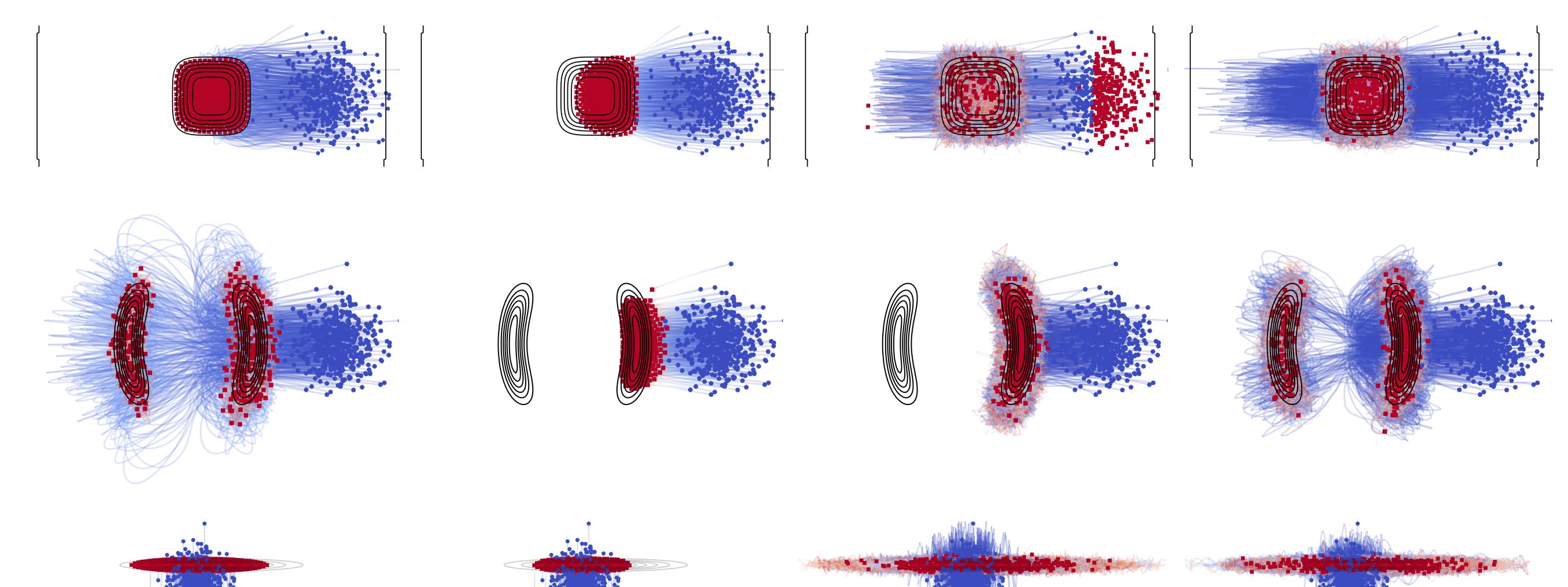
$$Y^{k+1} \leftarrow \alpha^k Y^k - \frac{\sqrt{\tau}}{N} K^{k+1} \nabla V(X^{k+1}) + \sqrt{\tau} \left( 1 + N^{-2} \text{tr}((M^{k+1})^T K^{k+1} M^k) \right) X^{k+1} A.$$

For Gaussian kernel:

$$\begin{aligned} W^{k+1} &\leftarrow N K^{k+1} + K^{k+1} (M^{k+1} (M^{k+1})^T) \circ K^{k+1} - K^{k+1} \circ (K^{k+1} M^{k+1} (M^{k+1})^T), \\ Y^{k+1} &\leftarrow \alpha^k Y^k - \frac{\sqrt{\tau}}{N} K^{k+1} \nabla V(X^{k+1}) + \frac{\sqrt{\tau}}{2N^2 \sigma^2} (\text{diag}(W^{k+1} \mathbb{1}_N) - W^{k+1}) X^{k+1}. \end{aligned}$$

**end**

## Numerical Experiments



**Particle trajectories.** ASVGD, SVGD, with Gaussian kernel, MALA, and ULD (from left to right) for  $V(x, y) = \frac{1}{4}(x^4 + y^4)$  (convex, non-Lipschitz) (top), the double bananas target (middle) and an anisotropic Gaussian target (bottom).

Double bananas target: constant high damping  $\beta = 0.985$ . Other targets: speed restart and gradient restart.

Dataset	RMSE		Log-likelihood		time (seconds)	
	ASVGD	SVGD	ASVGD	SVGD	ASVGD	SVGD
Concrete	$5.536 \pm 0.060$	$7.349 \pm 0.067$	$-3.135 \pm 0.016$	$-3.439 \pm 0.010$	$14.867 \pm 0.040$	$14.001 \pm 0.044$
Energy	$0.899 \pm 0.057$	$1.950 \pm 0.028$	$-1.268 \pm 0.068$	$-2.088 \pm 0.016$	$14.870 \pm 0.051$	$13.942 \pm 0.035$
Housing	$2.346 \pm 0.077$	$2.386 \pm 0.048$	$-2.305 \pm 0.020$	$-2.343 \pm 0.014$	$18.278 \pm 0.045$	$17.214 \pm 0.077$
Kin8mn	$0.118 \pm 0.001$	$0.165 \pm 0.001$	$0.71 \pm 0.011$	$0.384 \pm 0.004$	$14.859 \pm 0.029$	$14.001 \pm 0.033$
Naval	$0.005 \pm 0.0$	$0.007 \pm 0.0$	$3.801 \pm 0.01$	$3.504 \pm 0.003$	$20.912 \pm 0.57$	$19.704 \pm 0.05$
power	$3.951 \pm 0.005$	$4.035 \pm 0.008$	$-2.799 \pm 0.001$	$-2.825 \pm 0.002$	$12.142 \pm 0.053$	$11.435 \pm 0.054$
protein	$4.777 \pm 0.007$	$4.987 \pm 0.009$	$-2.983 \pm 0.001$	$-3.026 \pm 0.002$	$15.616 \pm 0.04$	$14.752 \pm 0.047$
wine	$0.185 \pm 0.013$	$0.191 \pm 0.017$	$0.201 \pm 0.039$	$0.146 \pm 0.046$	$18.293 \pm 0.097$	$17.244 \pm 0.077$

Table 1. Bayesian neural network experiment: test root mean square error (RMSE, lower is better) and log-likelihood (higher is better) after 2000 iterations, without restarts and constant damping  $\beta = 0.95$ , with only 10 particles.

## Further work

Find best kernel parameters  $A$  and  $\sigma^2$  and best damping parameters (nearly done!).

Conformal symplectic discretization instead of explicit Euler (retains structure of continuous dynamics), investigate bias of the algorithm, incorporate annealing strategy, (finite particle) convergence guarantees.

## Acknowledgments

V. Stein thanks his advisor, Gabriele Steidl, for her support and guidance throughout. W. Li's work is supported by the AFOSR YIP award No. FA9550-23-10087, NSF RTG: 2038080, and NSF DMS-224509