Understanding and Accelerating Particle-Based Variational Inference

Chang Liu [1] **Jingwei Zhuo** [1] **Pengyu Cheng** [2] **Ruiyi Zhang** [2] **Jun Zhu** [1] **Lawrence Carin** [2]

Abstract

Particle-based variational inference methods (ParVIs) have gained attention in the Bayesian inference literature, for their capacity to yield flexible and accurate approximations. We explore ParVIs from the perspective of Wasserstein gradient flows, and make both theoretical and practical contributions. We unify various finite-particle approximations that existing ParVIs use, and recognize that the approximation is essentially a compulsory smoothing treatment, in either of two equivalent forms. This novel understanding reveals the assumptions and relations of existing ParVIs, and also inspires new ParVIs. We propose an acceleration framework and a principled bandwidth-selection method for general ParVIs; these are based on the developed theory and leverage the geometry of the Wasserstein space. Experimental results show the improved convergence by the acceleration framework and enhanced sample accuracy by the bandwidth-selection method.

1. Introduction

Bayesian inference provides powerful tools for modeling and reasoning with uncertainty. In Bayesian learning one seeks access to the posterior distribution p on the support space $\mathcal{X}$ (*i.e.*, the latent space) given data. As p is intractable in general, various approximations have been developed. Variational inference methods (VIs) seek to approximate p within a certain distribution family by minimizing typically the Kullback-Leibler (KL) divergence to p. The approximating distribution is commonly chosen to be a member of a parametric family (Wainwright et al., 2008; Hoffman et al., 2013), but often it poses a restrictive assumption and the closeness to p suffers. Markov chain Monte Carlo (MCMC)

[1]Dept. of Comp. Sci. & Tech., Institute for AI, BNRist Center, Tsinghua-Fuzhou Inst. for Data Tech., THBI Lab, Tsinghua University, Beijing, 100084, China [2]Dept. of Elec. & Comp. Engineering, Duke University, NC, USA. Correspondence to: Jun Zhu <dcszj@tsinghua.edu.cn>, Lawrence Carin <lcarin@duke.edu>.

methods (Geman & Geman, 1987; Neal et al., 2011; Welling & Teh, 2011; Ding et al., 2014) aim to directly draw samples from p. Although asymptotically accurate, they often converge slowly in practice due to undesirable autocorrelation between samples. A relatively large sample size is needed (Liu & Wang, 2016) for a good result, increasing the cost for downstream tasks.

Recently, particle-based variational inference methods (ParVIs) have been proposed. They use a set of samples, or particles, to represent the approximating distribution (like MCMC) and deterministically update particles by minimizing the KL-divergence to p (like VIs). ParVIs have greater non-parametric flexibility than classical VIs, and are also more particle-efficient than MCMCs, since they make full use of a finite number of particles by taking particle interaction into account. The availability of optimization-based update rules also makes them converge faster. Stein Variational Gradient Descent (SVGD) (Liu & Wang, 2016) is a representative method of this type; it updates particles by leveraging a proper vector field that minimizes the KL-divergence. Its unique benefits make SVGD popular, with many variants (Liu & Zhu, 2018; Zhuo et al., 2018; Chen et al., 2018b; Futami et al., 2018) and applications (Wang & Liu, 2016; Pu et al., 2017; Liu et al., 2017a; Haarnoja et al., 2017).

SVGD was later understood as simulating the steepest descending curves, or *gradient flows*, of the KL-divergence on a certain kernel-related distribution space $\mathcal{P}_\mathcal{H}(\mathcal{X})$ (Liu, 2017). Inspired by this, more ParVIs have been developed by simulating the gradient flow on the Wasserstein space $\mathcal{P}_2(\mathcal{X})$ (Ambrosio et al., 2008; Villani, 2008) with a finite set of particles. The particle optimization method (PO) (Chen & Zhang, 2017) and w-SGLD method (Chen et al., 2018a) explore the minimizing movement scheme (MMS) (Jordan et al. (1998); Ambrosio et al. (2008), Def. 2.0.6) of the gradient flow and make approximations for tractability with a finite set of particles. The Blob method (originally called w-SGLD-B) (Chen et al., 2018a) uses the vector field form of the gradient flow and approximates the update direction with a finite set of particles. Empirical comparisons of these ParVIs have been conducted, but theoretical understanding on their finite-particle approximations for gradient flow simulation remains unknown, particularly for the assumption and relation of these approximations. We also

note that, from the optimization point of view, all existing ParVIs simulate the gradient flow, but no ParVI yet exploits the geometry of $\mathcal{P}_2(\mathcal{X})$ and uses the more appealing accelerated first-order methods on the manifold $\mathcal{P}_2(\mathcal{X})$. Moreover, the smoothing kernel bandwidth of ParVIs is found crucial for performance (Zhuo et al., 2018) and a more principled bandwidth selection method is needed, relative to the current heuristic median method (Liu & Wang, 2016).

In this work, we examine the $\mathcal{P}_2(\mathcal{X})$ gradient flow perspective to address these problems and demands, and contribute to the ParVI field a unified theory on the finite-particle approximations, and two practical techniques: an acceleration framework and a principled bandwidth selection method. The theory discovers that various finite-particle approximations of ParVIs are essentially a smoothing operation, in the form of either smoothing the density or smoothing functions. We reveal the two forms of smoothing, and draw a connection among ParVIs by discovering their equivalence. Furthermore, we recognize that ParVIs actually and necessarily make an assumption on the approximation family. The theory also establishes a principle for developing ParVIs, and we use this principle to conceive two novel models. The acceleration framework follows the Riemannian version (Liu et al., 2017b; Zhang & Sra, 2018) of Nesterov's acceleration method (Nesterov, 1983), which enjoys a proved convergence improvement over direct gradient flow simulation. In developing the framework, we make novel use of the Riemannian structure of $\mathcal{P}_2(\mathcal{X})$, particularly the inverse exponential map and parallel transport. We emphasize that the direct application of Nesterov's acceleration on every particle in $\mathcal{X}$ is unsound theoretically, since the KL-divergence is not minimized on $\mathcal{X}$ but on $\mathcal{P}_2(\mathcal{X})$, and each single particle is not optimizing a function. For the bandwidth method, we elaborate on the goal of smoothing and develop a principled approach for setting the bandwidth parameter. Experimental results show the improved sample quality by the principled bandwidth method over the median method (Liu & Wang, 2016), and improved convergence of the acceleration framework on both supervised and unsupervised tasks.

Related work On understanding SVGD, Liu (2017) first views it as a gradient flow on $\mathcal{P}_\mathcal{H}(\mathcal{X})$. Chen et al. (2018a) then formulate it as a gradient flow on $\mathcal{P}_2(\mathcal{X})$, but they find this implausible. In this work, we find SVGD approximates the $\mathcal{P}_2(\mathcal{X})$ gradient flow by smoothing functions, achieving an understanding and improvement of all ParVIs within a universal perspective. The $\mathcal{P}_\mathcal{H}(\mathcal{X})$ viewpoint is difficult to apply in general and it lacks appealing properties.

On accelerating ParVIs, the particle optimization method (PO) (Chen & Zhang, 2017) resembles the Polyak's momentum (Polyak, 1964) version of SVGD. However, it was not derived originally for acceleration and is thus not theoreti-

cally sound for such; we observe that PO is less stable than our acceleration framework, similar to the discussion by Sutskever et al. (2013). More recently, Taghvaei & Mehta (2018) also considered accelerating the $\mathcal{P}_2(\mathcal{X})$ gradient flow. They use the variational formulation of optimization methods (Wibisono et al., 2016) and define components in the formulation for $\mathcal{P}_2(\mathcal{X})$, while we leverage the geometry of $\mathcal{P}_2(\mathcal{X})$ and apply Riemannian acceleration methods. Algorithmically, Taghvaei & Mehta (2018) use a set of momentums and we use auxiliary particles. However, both approaches face the problem of a finite-particle approximation, solved here systematically using our theory. Their implementation is recognized, in our theory, as smoothing the density. Moreover, our novel perspective on $\mathcal{P}_2(\mathcal{X})$ and corresponding techniques provide a general tool that enables other Riemannian optimization techniques for ParVIs.

On incorporating Riemannian structure with ParVIs, Liu & Zhu (2018) consider the case for which the support space $\mathcal{X}$ is a Riemannian manifold, either specified by task or taken as the manifold in information geometry (Amari, 2016). We summarize that they utilize the geometry of the Riemannian support space $\mathcal{X}$, while we leverage deeper knowledge on the Riemannian structure of $\mathcal{P}_2(\mathcal{X})$ itself. Algorithmically, our acceleration is model-agnostic and computationally cheaper. The work of Detommaso et al. (2018) considers second-order information of the KL-divergence on $\mathcal{P}_\mathcal{H}(\mathcal{X})$. Our acceleration remains first-order, and with our theory we consider the Wasserstein space $\mathcal{P}_2(\mathcal{X})$ so that the acceleration is applicable for all ParVIs.

2. Preliminaries

We first introduce the Wasserstein space $\mathcal{P}_2(\mathcal{X})$ and the gradient flow on it, and review related ParVIs. We only consider Euclidean support $\mathcal{X} = \mathbb{R}^D$ to reduce unnecessary sophistication, and highlight our key contributions.

We denote $\mathcal{C}_c^\infty$ as the set of compactly supported $\mathbb{R}^D$-valued smooth functions on $\mathcal{X}$, and C_c^∞ for scalar-valued functions. Denote $\mathcal{L}_q^2$ as the Hilbert space of $\mathbb{R}^D$-valued functions $\{u : \mathbb{R}^D \to \mathbb{R}^D \text{ s.t. } \int \|u(x)\|_2^2 \, \mathrm{d}q < \infty\}$ with inner product $\langle u, v \rangle_{\mathcal{L}_q^2} := \int u(x) \cdot v(x) \, \mathrm{d}q$, and L_q^2 for scalar-valued functions. The Lebesgue measure is taken if q is not specified. We define the pushforward of a distribution q under a measurable transformation $\mathcal{T} : \mathcal{X} \to \mathcal{X}$ as the distribution of the $\mathcal{T}$-transformed random variable of q, denoted as $\mathcal{T}_\# q$.

2.1. The Riemannian Structure of the Wasserstein Space $\mathcal{P}_2(\mathcal{X})$

Figure 1 illustrates the related concepts discussed here. Consider distributions on a support space $\mathcal{X}$ with distance $d(\cdot, \cdot)$, and denote $\mathcal{P}(\mathcal{X})$ as the set of all such distributions. The Wasserstein space is the metric space $\mathcal{P}_2(\mathcal{X}) := \{q \in \mathcal{P}(\mathcal{X}) : \exists x_0 \in \mathcal{X} \text{ s.t. } \mathbb{E}_q[d(x_0, x)^2] < +\infty\}$ equipped with the well-known Wasserstein distance W_2 (Villani (2008),
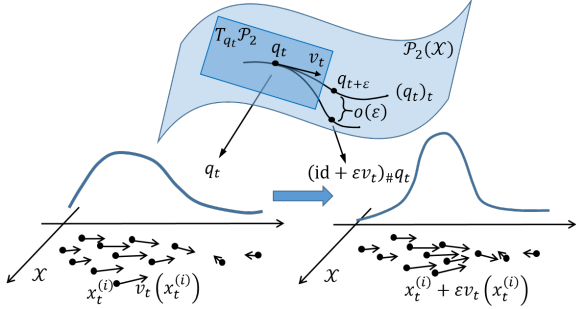
Figure 1. Illustration of the Wasserstein space $\mathcal{P}_2(\mathcal{X})$ and related concepts.

Def. 6.4). Its Riemannian structure is then discovered (Otto, 2001; Benamou & Brenier, 2000), enabling explicit expression of quantities of interest, like gradient. The first step is the recognition of tangent vectors and tangent spaces on it. For any smooth curve $(q_t)_t$ on $\mathcal{P}_2(\mathcal{X})$, there exists an a.e.-unique time-dependent vector field $v_t(x)$ on $\mathcal{X}$ such that for a.e. $t \in \mathbb{R}$, $\partial_t q_t + \nabla \cdot (v_t q_t) = 0$ and $v_t \in \overline{\{\nabla\varphi : \varphi \in C_c^\infty\}}^{\mathcal{L}^2_{q_t}}$ where the overline means closure (Villani (2008), Thm. 13.8; Ambrosio et al. (2008), Thm. 8.3.1, Prop. 8.4.5). The unique existence of such a v_t allows us to recognize v_t as the tangent vector of the curve at q_t, and the mentioned closure as the tangent space at q_t, denoted by $T_{q_t}\mathcal{P}_2$ (Ambrosio et al. (2008), Def. 8.4.1). The inherited inner product in $T_{q_t}\mathcal{P}_2$ from $\mathcal{L}^2_{q_t}$ defines a Riemannian structure on $\mathcal{P}_2(\mathcal{X})$, and it is consistent with the Wasserstein distance W_2 due to the Benamou-Brenier formula (Benamou & Brenier, 2000). Restricted to a parametric family as a submanifold of $\mathcal{P}_2(\mathcal{X})$, this structure gives a metric in the parameter space that differs from the Fisher-Rao metric (Amari, 2016), and it has been used in classical VIs (Chen & Li, 2018). Finally, the vector field representation provides a convenient means of simulating the distribution curve $(q_t)_t$: it is known that $(\mathrm{id} + \varepsilon v_t)_\# q_t$ is a first-order approximation of $q_{t+\varepsilon}$ in terms of W_2 (Ambrosio et al. (2008), Prop. 8.4.6). This means that given a set of samples $\{x^{(i)}\}_i$ of q_t, $\{x^{(i)} + \varepsilon v_t(x^{(i)})\}_i$ is approximately a set of samples of $q_{t+\varepsilon}$ for small ε.

2.2. Gradient Flows on $\mathcal{P}_2(\mathcal{X})$

Gradient flow of a function F is roughly the family of *steepest descending curves* $\{(q_t)_t\}$ for F. It has various technical definitions on metric spaces (Ambrosio et al. (2008), Def. 11.1.1; Villani (2008), Def. 23.7), *e.g.*, as the limit of the minimizing movement scheme (MMS; Ambrosio et al. (2008), Def. 2.0.6):

$$q_{t+\varepsilon} = \operatorname*{argmin}_{q \in \mathcal{P}_2(\mathcal{X})} F(q) + \frac{1}{2\varepsilon} W_2^2(q, q_t), \qquad (1)$$

and these definitions all coincide when a Riemannian structure is endowed (Villani (2008), Prop. 23.1, Rem. 23.4; Ambrosio et al. (2008), Thm. 11.1.6; Erbar et al. (2010), Lem. 2.7). In this case the gradient flow $(q_t)_t$ has its tangent

vector at any t being the gradient of F at q_t, defined as:

$$\operatorname{grad} F(q_t) := \max \cdot \operatorname*{argmax}_{v:\, \|v\|_{T_{q_t}\mathcal{P}_2}=1} \frac{\mathrm{d}}{\mathrm{d}\varepsilon} F\big((\mathrm{id}+\varepsilon v)_\# q_t\big)\Big|_{\varepsilon=0}, \quad (2)$$

where "$\max \cdot \operatorname{argmax}$" denotes the scalar multiplication of the maximum and the maximizer. For Bayesian inference tasks, given an absolutely continuous target distribution p, we aim to minimize the KL-divergence (a.k.a. relative entropy) $\mathrm{KL}_p(q) := \int_{\mathcal{X}} \log(q/p)\,\mathrm{d}q$. Its gradient flow $(q_t)_t$ has its tangent vector at any t being:

$$v_t^{\mathrm{GF}} := -\operatorname{grad}\mathrm{KL}_p(q_t) = \nabla\log p - \nabla\log q_t, \quad (3)$$

whenever q_t is absolutely continuous (Villani (2008), Thm. 23.18; Ambrosio et al. (2008), Example 11.1.2). When KL_p is geodesically μ-convex on $\mathcal{P}_2(\mathcal{X})$,[1] the gradient flow $(q_t)_t$ enjoys exponential convergence: $W_2(q_t, p) \leq e^{-\mu t} W_2(q_0, p)$ (Villani (2008), Thm. 23.25, Thm. 24.7; Ambrosio et al. (2008), Thm. 11.1.4), as expected.

Remark 1. The Langevin dynamics $\mathrm{d}x = \nabla\log p(x)\,\mathrm{d}t + \sqrt{2}\,\mathrm{d}B_t(x)$ (B_t is Brownian motion) is also known to produce the gradient flow of KL_p on $\mathcal{P}_2(\mathcal{X})$ (*e.g.*, Jordan et al. (1998) from the MMS perspective). It produces the same curve $(q_t)_t$ as the deterministic dynamics $\mathrm{d}x = v_t^{\mathrm{GF}}(x)\,\mathrm{d}t$.

2.3. Particle-Based Variational Inference Methods

Stein Variational Gradient Descent (SVGD) (Liu & Wang, 2016) uses a vector field v to update particles: $x_{k+1}^{(i)} = x_k^{(i)} + \varepsilon v(x_k^{(i)})$, and v is selected to maximize the decreasing rate: $-\frac{\mathrm{d}}{\mathrm{d}\varepsilon}\mathrm{KL}_p\big((\mathrm{id}+\varepsilon v)_\# q\big)\big|_{\varepsilon=0}$, where q is the distribution that $\{x^{(i)}\}_i$ obeys. A closed-form solution can be derived when v is optimized over the vector-valued reproducing kernel Hilbert space (RKHS) $\mathcal{H}^D$ of a kernel K (Steinwart & Christmann (2008), Def. 4.18):

$$v^{\mathrm{SVGD}}(\cdot) := \mathbb{E}_{q(x)}[K(x,\cdot)\nabla\log p(x) + \nabla_x K(x,\cdot)]. \quad (4)$$

Noting that the optimization problem fits the form of Eq. (2), Liu (2017) interprets SVGD as the gradient flow on $\mathcal{P}_{\mathcal{H}}$, a distribution manifold that takes $\mathcal{H}^D$ as its tangent space. Equation (4) can be estimated by a finite set of particles, equivalently taking $q(x)$ as the empirical distribution $\hat{q}(x) := \frac{1}{N}\sum_{i=1}^N \delta_{x^{(i)}}(x)$.

Other methods have been developed to simulate the $\mathcal{P}_2(\mathcal{X})$ gradient flow. The Blob method (Chen et al., 2018a) estimates Eq. (3) with a finite set of particles. It reformulates the intractable part $u^{\mathrm{GF}} := -\nabla\log q$ from the perspective of variation: $u^{\mathrm{GF}} = \nabla\big(-\frac{\delta}{\delta q}\mathbb{E}_q[\log q]\big)$, and then partly smooths the density q by convolving with a kernel K:

$$u^{\mathrm{Blob}} = \nabla\big(-\frac{\delta}{\delta q}\mathbb{E}_q[\log(q*K)]\big)$$

$$= -\nabla\log\tilde{q} - \nabla\big((q/\tilde{q})*K\big), \text{ where } \tilde{q} := q*K \text{ and "}*\text{"}$$

denotes convolution. This form enables the usage of $\hat{q}$.

---

[1] *E.g.*, p is μ-log-concave on $\mathcal{X}$ (Villani (2008), Thm. 17.15).

The particle optimization method (PO) (Chen & Zhang, 2017) simulates the $\mathcal{P}_2(\mathcal{X})$ gradient flow by MMS (1), where the Wasserstein distance W_2 is estimated by solving a dual optimal transport problem that optimizes over quadratic functions. The resulting update rule $x_k^{(i)} =$

$$x_{k-1}^{(i)} + \varepsilon(v^{\text{SVGD}}(x_{k-1}^{(i)}) + \mathcal{N}(0, \sigma^2 I)) + \mu(x_{k-1}^{(i)} - x_{k-2}^{(i)})$$

(with parameters ε, σ, μ) comes in the form of the Polyak's momentum (Polyak, 1964) version of SVGD. The w-SGLD method (Chen et al., 2018a) estimates W_2 by solving the primal problem with entropy regularization. The algorithm is similar to PO.

3. ParVIs as Approximations to $\mathcal{P}_2(\mathcal{X})$ Gradient Flow

This part of the paper constitutes our main theoretical contributions. We find that ParVIs approximate the $\mathcal{P}_2(\mathcal{X})$ gradient flow by smoothing, either smoothing the density or smoothing functions. We make recognition of existing ParVIs, analyze equivalence and the necessity of smoothing, and develop two novel ParVIs based on the theory.

3.1. SVGD Approximates $\mathcal{P}_2(\mathcal{X})$ Gradient Flow

Currently SVGD is only known to simulate the gradient flow on the distribution space $\mathcal{P}_\mathcal{H}(\mathcal{X})$. We first interpret SVGD as a simulation of the gradient flow on the Wasserstein space $\mathcal{P}_2(\mathcal{X})$ with a finite set of particles, so that all existing ParVIs can be analyzed from a common perspective. Noting that v^{GF} is an element of the Hilbert space $\mathcal{L}_q^2$, we can identify it by:

$$v^{\text{GF}} = \max \cdot \operatorname*{argmax}_{v \in \mathcal{L}_q^2, \|v\|_{\mathcal{L}_q^2} = 1} \langle v^{\text{GF}}, v \rangle_{\mathcal{L}_q^2}. \tag{5}$$

We then find that by changing the optimization domain from $\mathcal{L}_q^2$ to the vector-valued RKHS $\mathcal{H}^D$ of a kernel K, the problem can be solved in closed-form and the solution coincides with v^{SVGD}. This connects the two notions:

Theorem 2 (v^{SVGD} approximates v^{GF})**.** *The SVGD vector field v^{SVGD} defined in Eq. (4) approximates the vector field v^{GF} of the gradient flow on $\mathcal{P}_2(\mathcal{X})$, in the following sense:*

$$v^{\text{SVGD}} = \max \cdot \operatorname*{argmax}_{v \in \mathcal{H}^D, \|v\|_{\mathcal{H}^D} = 1} \langle v^{\text{GF}}, v \rangle_{\mathcal{L}_q^2}.$$

The proof is provided in Appendix A.1. We will see that $\mathcal{H}^D$ is roughly a subspace of $\mathcal{L}_q^2$, so v^{SVGD} can be seen as the projection of v^{GF} on $\mathcal{H}^D$.

The current $\mathcal{P}_\mathcal{H}(\mathcal{X})$ gradient flow interpretation of SVGD (Liu, 2017; Chen et al., 2018a) is not fully satisfying. We point out that $\mathcal{P}_\mathcal{H}$ is not yet a well defined Riemannian manifold. It is characterized by taking $\mathcal{H}^D$ as its tangent space, but it is only known that the tangent space of a manifold is determined by the manifold's topology (Do Carmo, 1992), and it is unknown if there uniquely exists a manifold with a specified tangent space. Particularly, the tangent vector of any smooth curve should uniquely exist in the tangent space. Wasserstein space $\mathcal{P}_2(\mathcal{X})$ satisfies this (Villani (2008), Thm. 13.8; Ambrosio et al. (2008), Thm. 8.3.1, Prop. 8.4.5), but it remains unknown for $\mathcal{P}_\mathcal{H}$. The manifold $\mathcal{P}_\mathcal{H}$ also lacks appealing properties like an explicit expression of the distance (Chen et al., 2018a). SVGD has also been formulated as a Vlasov process (Liu, 2017; Chen et al., 2018a); this shows that SVGD keeps p invariant, but does not provide much knowledge on its convergence behavior.

3.2. ParVIs Approximate $\mathcal{P}_2(\mathcal{X})$ Gradient Flow by Smoothing

With the above knowledge, all ParVIs approximate the $\mathcal{P}_2(\mathcal{X})$ gradient flow. We then find that the approximation is made by a compulsory smoothing treatment, either smoothing the density or smoothing functions.

Smoothing the Density We note that the Blob method approximates the $\mathcal{P}_2(\mathcal{X})$ gradient flow by replacing q with a smoothed density $\tilde{q} := \hat{q} * K$ in the variational formulation of the gradient flow. In the w-SGLD method (Chen et al., 2018a), an entropy regularization is introduced to the primal optimal transport problem. This term avoids the density solution being highly concentrated, and thus effectively poses a smoothing requirement on densities.

Smoothing Functions We note in Theorem 2 that SVGD approximates the $\mathcal{P}_2(\mathcal{X})$ gradient flow by replacing the function family $\mathcal{L}_q^2$ with $\mathcal{H}^D$ in an optimization formulation of the gradient flow. We then reveal the fact that a function in $\mathcal{H}^D$ is roughly a kernel smoothed function in $\mathcal{L}_q^2$, as stated formally in the following theorem:

Theorem 3 ($\mathcal{H}^D$ smooths $\mathcal{L}_q^2$)**.** *For $\mathcal{X} = \mathbb{R}^D$, a Gaussian kernel K on $\mathcal{X}$ and an absolutely continuous q, the vector-valued RKHS $\mathcal{H}^D$ of K is isometrically isomorphic to the closure $\mathcal{G} := \overline{\{\phi * K : \phi \in \mathcal{C}_c^\infty\}}^{\mathcal{L}_q^2}$.*

The proof is presented in Appendix A.2. Noting that $\mathcal{C}_c^\infty$ is roughly $\mathcal{L}_q^2$ in the sense $\overline{\mathcal{C}_c^\infty}^{\mathcal{L}_q^2} = \mathcal{L}_q^2$ (Kováčik & Rákosník (1991), Thm. 2.11), the closure $\mathcal{G}$ is roughly the set of kernel smoothed functions in $\mathcal{L}_q^2$, and it is roughly $\mathcal{H}^D$.

As mentioned in Section 2.3, the particle optimization method (PO) (Chen & Zhang, 2017) restricts the optimization domain to be quadratic functions when solving the dual optimal transport problem. Since quadratic functions have restricted sharpness (no change in second-order derivatives), this treatment effectively smooths the function family.

Equivalence The above analysis draws more importance when we note the equivalence between the two smoothing approaches. Recall that the objective in the optimization problem that SVGD uses, Eq. (5), is $\langle v^{\text{GF}}, v \rangle_{\mathcal{L}_q^2} = \mathbb{E}_q[v^{\text{GF}} \cdot v]$. We generalize this objective in the form $\mathbb{E}_q[L(v)]$ with a

linear map $L : \mathcal{L}_q^2 \to L_q^2$. Due to the interchangeability of the integral and the linearity of L, we have the relation:
$$\mathbb{E}_{\tilde{q}}[L(v)] = \mathbb{E}_{q*K}[L(v)] = \mathbb{E}_q[L(v) * K] = \mathbb{E}_q[L(v * K)],$$
which indicates that smoothing the density $q * K$ is equivalent to smoothing functions $v * K$. This equivalence bridges the two types of ParVIs, making the analysis and techniques (*e.g.*, our acceleration and bandwidth methods in Secs. 4 and 5, respectively) for one type also applicable to the other.

Necessity and ParVI Assumption We stress that this smoothing endeavor is essentially required by a well-defined gradient flow of the KL-divergence $\mathrm{KL}_p(q)$. It returns infinity when q is not absolutely continuous, as is the case $q = \hat{q}$, thus the gradient flow cannot be reasonably defined. So we recognize that ParVIs have to make the assumption that q is smooth, and this can be implemented by either smoothing the density or smoothing functions.

This claim seems straightforward for smoothing density methods, but is perhaps obscure for smoothing function methods. We now make a direct analysis for SVGD, that neither smoothing the density (take $q = \hat{q}$) nor smoothing functions (optimize over $\mathcal{L}_p^{2\ 2}$) leads to an unreasonable result.

Theorem 4 (Necessity of smoothing for SVGD). *For $q = \hat{q}$ and $v \in \mathcal{L}_p^2$, problem (5) has no optimal solution. In fact the supremum of the objective is infinite, indicating that a maximizing sequence of v tends to be ill-posed.*

The proof is given in Appendix A.3. SVGD claims no assumption on the form of the approximating density q as long as its samples are known, but we find it actually transmits the restriction on q to the functions v. The choice for v in $\mathcal{H}^D$ is not just for a tractable solution, but more importantly, for guaranteeing a valid vector field. We see that there is no free lunch in making the smoothing assumption. ParVIs have to assume a smoothed density or functions.

3.3. New ParVIs with Smoothing

The theoretical understanding of smoothing for the finite-particle approximation constructs a principle for developing new ParVIs. We conceive two new instances based on the smoothing-density and smoothing-function formulations.

GFSD We directly approximate q with smoothed density $\tilde{q} := \hat{q} * K$ and adopt the vector field form of gradient flow (Eq. (3)): $u^{\mathrm{GFSD}} := -\nabla \log \tilde{q}$. We call the corresponding method Gradient Flow with Smoothed Density (GFSD).

GFSF We discover another novel optimization formulation to identify u^{GF}, which could build a new ParVI by smoothing functions. We reform the intractable component $u^{\mathrm{GF}} := -\nabla \log q$, as $q u^{\mathrm{GF}} + \nabla q = 0$, and treat it as an equality that

holds in the weak sense. This means $\mathbb{E}_q[\phi \cdot u - \nabla \cdot \phi] = 0, \forall \phi \in \mathcal{C}_c^\infty,$[3] or equivalently,
$$u^{\mathrm{GF}} = \operatorname*{argmin}_{u \in \mathcal{L}^2} \max_{\substack{\phi \in \mathcal{C}_c^\infty, \\ \|\phi\|_{\mathcal{L}_q^2} = 1}} \left(\mathbb{E}_q[\phi \cdot u - \nabla \cdot \phi] \right)^2.$$
We take $q = \hat{q}$ and smooth functions $\phi \in \mathcal{C}_c^\infty$ with kernel K, which is equivalent to taking ϕ from the vector-valued RKHS $\mathcal{H}^D$ according to Theorem 3:
$$u^{\mathrm{GFSF}} := \operatorname*{argmin}_{u \in \mathcal{L}^2} \max_{\substack{\phi \in \mathcal{H}^D, \\ \|\phi\|_{\mathcal{H}^D} = 1}} \left(\mathbb{E}_{\hat{q}}[\phi \cdot u - \nabla \cdot \phi] \right)^2. \quad (6)$$

The closed-form solution is $\hat{u}^{\mathrm{GFSF}} = \hat{K}' \hat{K}^{-1}$ in matrix form, where $\hat{u}_{:,i}^{\mathrm{GFSF}} := u^{\mathrm{GFSF}}(x^{(i)})$, $\hat{K}_{ij} := K(x^{(i)}, x^{(j)})$, and $\hat{K}'_{:,i} := \sum_j \nabla_{x^{(j)}} K(x^{(j)}, x^{(i)})$ (see Appendix B.1). We call this method the Gradient Flow with Smoothed test Functions (GFSF). Note that the above objective fits the form $\mathbb{E}_q[L(\phi)]$ with L linear, indicating the equivalence to smoothing the density, as discussed in Section 3.2. An interesting relation between the matrix-form expression of GFSF and SVGD is that $\hat{v}^{\mathrm{GFSF}} = \hat{g} + \hat{K}' \hat{K}^{-1}$ while $\hat{v}^{\mathrm{SVGD}} = \hat{g} \hat{K} + \hat{K}'$, where $\hat{g}_{:,i} := \nabla \log p(x^{(i)})$. We also note that the GFSF estimate of $-\nabla \log q$ coincides with the method of (Li & Turner, 2017), which is derived by Stein's identity and approximating the ℓ^2 space with RKHS $\mathcal{H}$.

Due to Remark 1, all these ParVIs aim to simulate the same path on $\mathcal{P}_2(\mathcal{X})$ as the Langevin dynamics (LD) (Roberts & Stramer (2002)). They directly utilize the particle interaction via the smoothing kernel, so every particle is aware of others and they could be more particle-efficient (Liu & Wang, 2016) than the vanilla LD simulation. To scale to large datasets, LD has employed stochastic gradient in simulation (Welling & Teh, 2011), which is appropriate (Chen et al., 2015). Due to the connection to LD, ParVIs can also adopt stochastic gradient for scalability. Finally, our theory could utilize more techniques for developing ParVIs, *e.g.*, implicit distribution gradient estimation (Shi et al., 2018).

4. Accelerated First-Order Methods on $\mathcal{P}_2(\mathcal{X})$

We have developed a unified understanding on ParVIs under the $\mathcal{P}_2(\mathcal{X})$ gradient flow perspective, which corresponds to the gradient descent method on $\mathcal{P}_2(\mathcal{X})$. It is well-known that the Nesterov's acceleration method (Nesterov, 1983) can give a faster convergence rate, and its Riemannian variants have been developed recently, such as Riemannian Accelerated Gradient (RAG) (Liu et al., 2017b) and Riemannian Nesterov's method (RNes) (Zhang & Sra, 2018). We aim to employ ParVIs with these methods. However, this requires more knowledge on the geometry of $\mathcal{P}_2(\mathcal{X})$.

---

[2] Why not $\mathcal{L}_q^2$: $v \in \mathcal{L}_p^2$ is required by the condition of Stein's identity (Liu, 2017), on which SVGD is based. Also, $\hat{q}$ is not absolutely continuous so $\mathcal{L}_{\hat{q}}^2$ is not a proper Hilbert space of functions.

[3] We also consider scalar-valued functions $\varphi \in \mathcal{C}_c^\infty$ smoothed in $\mathcal{H}$, which gives the same result, as shown in Appendix B.2.
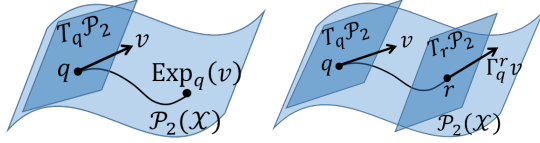
Figure 2. Illustration of the concepts exponential map (left) and parallel transport (right)

4.1. Leveraging the Riemannian Structure of $\mathcal{P}_2(\mathcal{X})$

RAG and RNes require the *exponential map* (and its inverse) and *parallel transport* on $\mathcal{P}_2(\mathcal{X})$. As depicted in Fig. 2, the exponential map $\mathrm{Exp}_q : T_q\mathcal{P}_2(\mathcal{X}) \to \mathcal{P}_2(\mathcal{X})$ is the displacement from position q to a new position along the geodesic (a "straight line" on a manifold) with a given direction, and the parallel transport $\Gamma_q^r : T_q\mathcal{P}_2(\mathcal{X}) \to T_r\mathcal{P}_2(\mathcal{X})$ is the transformation of a tangent vector at q to another one at r in a certain sense of parallel along the geodesic from q to r. We now examine these components and find practical estimation with a finite set of particles.

Exponential map on $\mathcal{P}_2(\mathcal{X})$ For an absolutely continuous measure q, $\mathrm{Exp}_q(v) = (\mathrm{id} + v)_{\#}q$ (Villani (2008), Coro. 7.22; Ambrosio et al. (2008), Prop. 8.4.6; Erbar et al. (2010), Prop. 2.1). Finite-particle estimation is just an application of the map $x \mapsto x + v(x)$ on each particle.

Inverse of the exponential map With details in Appendix A.4, we find that the inverse exponential map $\mathrm{Exp}_q^{-1}(r)$ for $q, r \in \mathcal{P}_2(\mathcal{X})$ can be expressed by the optimal transport map from q to r. We can approximate the map by a discrete one between particles $\{x^{(i)}\}_{i=1}^N$ of q and $\{y^{(i)}\}_{i=1}^N$ of r. However, this is still a costly task (Pele & Werman, 2009). Faster approaches like the Sinkhorn method (Cuturi, 2013; Xie et al., 2018) still require empirically $O(N^2)$ time, and it appears unacceptably unstable in our experiments. We consider an approximation when $\{x^{(i)}\}_i$ and $\{y^{(i)}\}_i$ are pairwise close: $d(x^{(i)}, y^{(i)}) \ll \min\left\{\min_{j\neq i} d(x^{(i)}, x^{(j)}), \min_{j\neq i} d(y^{(i)}, y^{(j)})\right\}$. In this case we have the result (deduction in Appendix A.4):

Proposition 5 (Inverse exponential map). *For pairwise close samples $\{x^{(i)}\}_{i=1}^N$ of q and $\{y^{(i)}\}_{i=1}^N$ of r, we have $\left(\mathrm{Exp}_q^{-1}(r)\right)(x^{(i)}) \approx y^{(i)} - x^{(i)}$.*

Parallel Transport on $\mathcal{P}_2(\mathcal{X})$ There has been formal research on the parallel transport on $\mathcal{P}_2(\mathcal{X})$ (Lott, 2008; 2017), but the result is expensive to estimate with a finite set of particles. Here we utilize Schild's ladder method (Ehlers et al., 1972; Kheyfets et al., 2000), a first-order approximation of parallel transport that only requires the exponential map. We derive the following estimation with a finite set of particles (deduction in Appendix A.5):

Proposition 6 (Parallel transport). *For pairwise close samples $\{x^{(i)}\}_{i=1}^N$ of q and $\{y^{(i)}\}_{i=1}^N$ of r, we have $\left(\Gamma_q^r(v)\right)(y^{(i)}) \approx v(x^{(i)}), \forall v \in T_q\mathcal{P}_2(\mathcal{X})$.*

Both results may not seem surprising. This is because the

Algorithm 1 The acceleration framework with Wasserstein Accelerated Gradient (WAG) and Wasserstein Nesterov's method (WNes)

1: WAG: select acceleration factor $\alpha > 3$;
    WNes: select or calculate $c_1, c_2 \in \mathbb{R}^+$ (Appendix C.2);
2: Initialize $\{x_0^{(i)}\}_{i=1}^N$ distinctly; let $y_0^{(i)} = x_0^{(i)}$;
3: **for** $k = 1, 2, \cdots, k_{\max}$, **do**
4:   **for** $i = 1, \cdots, N$, **do**
5:     Find $v(y_{k-1}^{(i)})$ by SVGD/Blob/GFSD/GFSF;
6:     $x_k^{(i)} = y_{k-1}^{(i)} + \varepsilon v(y_{k-1}^{(i)})$;
7:     $y_k^{(i)} = x_k^{(i)} +$
    $\begin{cases} \text{WAG: } \frac{k-1}{k}(y_{k-1}^{(i)} - x_{k-1}^{(i)}) + \frac{k+\alpha-2}{k}\varepsilon v(y_{k-1}^{(i)}); \\ \text{WNes: } c_1(c_2-1)(x_k^{(i)} - x_{k-1}^{(i)}); \end{cases}$
8:   **end for**
9: **end for**
10: Return $\{x_{k_{\max}}^{(i)}\}_{i=1}^N$.

---

geometry of $\mathcal{P}_2(\mathcal{X})$ is determined by that of $\mathcal{X}$. We consider Euclidean $\mathcal{X}$, so $\mathcal{P}_2(\mathcal{X})$ also appears flat. Extension to non-flat Riemannian $\mathcal{X}$ can be done with the same procedure.

4.2. Acceleration Framework for ParVIs

Now we apply RAG and RNes to the Wasserstein space $\mathcal{P}_2(\mathcal{X})$ and construct an accelerated sequence $\{q_k\}_k$ minimizing KL_p. Both methods introduce an auxiliary variable $r_k \in \mathcal{P}_2(\mathcal{X})$, on which the gradient is evaluated: $v_k := -\mathrm{grad}\,\mathrm{KL}(r_k)$. RAG (Liu et al., 2017b) needs to solve a nonlinear equation in each step. We simplify it with moderate approximations to give an explicit update rule: $q_k = \mathrm{Exp}_{r_{k-1}}(\varepsilon v_{k-1}), r_k = \mathrm{Exp}_{q_k}\left[-\Gamma_{r_{k-1}}^{q_k}\left(\frac{k-1}{k}\mathrm{Exp}_{r_{k-1}}^{-1}(q_{k-1}) - \frac{k+\alpha-2}{k}\varepsilon v_{k-1}\right)\right]$ (details in Appendix C.1). RNes (Zhang & Sra, 2018) involves an additional variable. We collapse the variable and reformulate RNes as: $q_k = \mathrm{Exp}_{r_{k-1}}(\varepsilon v_{k-1}), r_k = \mathrm{Exp}_{q_k}\left\{c_1 \mathrm{Exp}_{q_k}^{-1}\left[\mathrm{Exp}_{r_{k-1}}\left((1-c_2)\mathrm{Exp}_{r_{k-1}}^{-1}(q_{k-1}) + c_2\mathrm{Exp}_{r_{k-1}}^{-1}(q_k)\right)\right]\right\}$ (details in Appendix C.2). To implement both methods with a finite set of particles, we leverage the geometric calculations in the previous subsection and estimate v_k with ParVIs. The resulting algorithms are called Wasserstein Accelerated Gradient (WAG) and Wasserstein Nesterov's method (WNes), and are presented in Alg. 1 (deduction details in Appendix C.3). They form an acceleration framework for ParVIs. In the deduction, the pairwise-close condition is satisfied, so the usage of Propositions 5 and 6 is appropriate.

In theory, the acceleration framework inherits the proved improvement on the convergence rate from RAG and RNes, and it can be applied to all ParVIs, since our theory has recognized the equivalence of ParVIs. In practice, the framework imposes a computational cost linear in the particle

size N, which is not a significant overhead. Moreover, we emphasize that we cannot directly apply the vanilla Nesterov's acceleration method (Nesterov, 1983) in $\mathcal{X}$ on every particle, since a single particle is not optimizing a certain function. Finally, the developed knowledge on the geometry of $\mathcal{P}_2(\mathcal{M})$ (Propositions 5, 6) also makes it possible to apply other optimization techniques on Riemannian manifolds to benefit ParVIs, *e.g.*, Riemannian BFGS (Gabay, 1982; Qi et al., 2010; Yuan et al., 2016) and Riemannian stochastic variance reduction gradient (Zhang et al., 2016).

5. Bandwidth Selection via Heat Equation

Our theory points out that all ParVIs need smoothing, and this can be done with a kernel. Thus it is an essential problem to choose the bandwidth of the kernel. SVGD uses the median method (Liu & Wang, 2016) based on a heuristic for numerical stability. We work towards a more principled method. We first analyze the goal of smoothing, then build a practical algorithm.

As noted in Remark 1, the deterministic dynamics $\mathrm{d}x = v^{\mathrm{GF}}(x)\,\mathrm{d}t$ and the Langevin dynamics produce the same rule of density evolution. In particular, the deterministic dynamics $\mathrm{d}x = -\nabla \log q_t(x)\,\mathrm{d}t$ and the Brownian motion $\mathrm{d}x = \sqrt{2}\,\mathrm{d}B_t(x)$ produce the same evolution rule specified by the heat equation (HE): $\partial_t q_t(x) = \Delta q_t(x)$. So a good smoothing kernel should let the evolving density under the approximated dynamics match the rule of HE. This is the principle for selecting a kernel bandwidth.

We implement this principle for GFSD, which estimates q_t by the kernel smoothed density $\tilde{q}(x) = \tilde{q}(x; \{x^{(i)}\}_i) = \frac{1}{N}\sum_{i=1}^{N} K(x, x^{(i)})$. The approximate dynamics $\mathrm{d}x = -\nabla \log \tilde{q}(x)\,\mathrm{d}t$ moves particles $\{x^{(i)}\}_i$ of q_t to $\{x^{(i)} - \varepsilon\nabla \log \tilde{q}(x^{(i)})\}_i$, which approximates $q_{t+\varepsilon}$. On the other hand, according to the evolution rule of the HE, the new density $q_{t+\varepsilon}$ should be approximated by $q_t + \varepsilon\partial_t q_t \approx \tilde{q} + \varepsilon\Delta\tilde{q}$. The two approximations should match, which means $\tilde{q}(x; \{x^{(i)} - \varepsilon\nabla \log \tilde{q}(x^{(i)})\}_i)$ should be close to $\tilde{q} + \varepsilon\Delta\tilde{q}$. Expanding up to first order in ε, this requirement translates to imposing that the function $\lambda(x) := \Delta\tilde{q}(x; \{x^{(i)}\}_i) + \sum_j \nabla_{x^{(j)}}\tilde{q}(x; \{x^{(i)}\}_i) \cdot \nabla \log \tilde{q}(x^{(j)}; \{x^{(i)}\}_i)$ should be close to zero. To achieve this practically, we propose to minimize $\frac{N}{h^{D+2}}\mathbb{E}_{q(x)}[\lambda(x)^2] \approx \frac{1}{h^{D+2}}\sum_k \lambda(x^{(k)})^2$ w.r.t. the bandwidth h. We introduce the factor $\frac{1}{h^{D+2}}$ (note that x^2/h is dimensionless) to make the final objective dimensionless.

We call this the HE method. Although the derivation is based on GFSD, the resulting algorithm can be applied to all kernel-based ParVIs, like SVGD, Blob and GFSF, due to the equivalence of smoothing the density and functions from our theory. See Appendix D for further details.

6. Experiments

Detailed experimental settings and parameters are provided in Appendix E, and codes are available at https://github.com/chang-ml-thu/AWGF. Corresponding to WAG and WNes, we call the vanilla gradient flow simulation of ParVIs as Wasserstein Gradient Descent (WGD).

6.1. Toy Experiments

We first investigate the benefit of the HE method for selecting the bandwidth, with comparison to the median method. Figure 3 shows 200 particles produced by four ParVIs using both the HE and median methods. We find that the median method makes the particles collapse to the modes, since the numerical heuristic cannot guarantee the effect of smoothing. The HE method achieves an attractive effect: the particles align neatly and distribute almost uniformly along the contour, building a more representative approximation. SVGD gives diverse particles also with the median method, which may be due to the averaging of gradients for each particle.
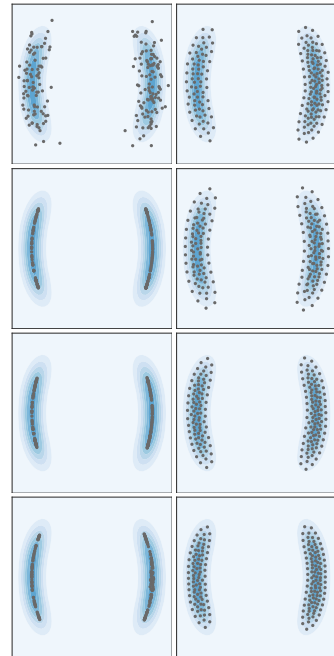

Figure 3. Comparison of HE (right column) with the median method (left column) for bandwidth selection. Rows correspond to SVGD, Blob, GFSD and GFSF, respectively.

6.2. Bayesian Logistic Regression (BLR)

We show the accelerated convergence of the proposed WAG and WNes methods (Alg. 1) for various ParVIs on BLR. Although PO is not developed for acceleration, we treat it as an empirical acceleration. We follow the same settings as Liu & Wang (2016) and Chen et al. (2018a), except that results are averaged over 10 random trials. Results are evaluated by test accuracy (Fig. 4) and log-likelihood (Fig. 8 in Appendix F.1). For all four ParVIs, WAG and WNes notably improve the convergence over WGD and PO. Moreover, WNes gets better results than WAG, especially at the early stage, and it is also more stable w.r.t. hyperparameters. The performance of the PO method is roughly the same as WGD, matching the observation by Chen & Zhang (2017). We also note that the four ParVIs have a similar performance, which is natural since they approximate the same gradient flow with equivalent smoothing treatments.
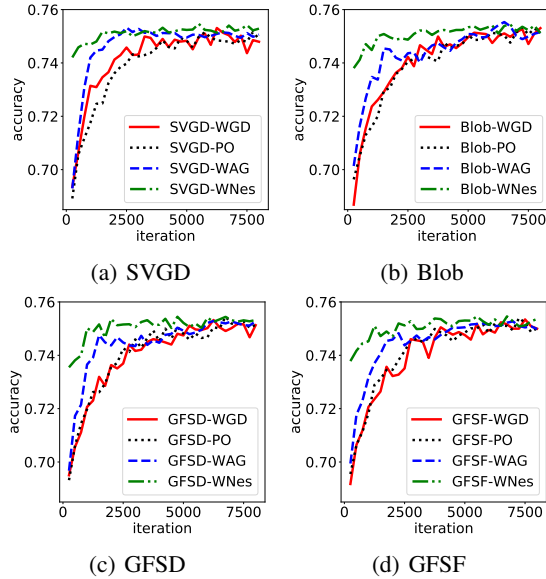
Figure 4. Acceleration effect of WAG and WNes on BLR on the Covertype dataset. Curves are averaged over 10 runs.

Figure 5. Acceleration effect of WAG and WNes on LDA. Curves are averaged over 10 runs.

6.3. Bayesian Neural Networks (BNNs)

We test all methods on BNNs for a fixed number of iterations, following the settings of Liu & Wang (2016), and present results in Table 1. We observe that WAG and WNes acceleration methods outperform the WGD and PO for all the four ParVIs. The PO method also improves the performance, but not to the extent of WAG.

6.4. Latent Dirichlet Allocation (LDA)

We show the improved performance by the acceleration methods on an unsupervised task: posterior inference of LDA (Blei et al., 2003). We follow the same settings as Ding et al. (2014), including the ICML dataset[4] and the Expanded-Natural parameterization (Patterson & Teh, 2013). The particle size is fixed at 20. Inference results are evaluated by the conventional hold-out perplexity (the lower the better).
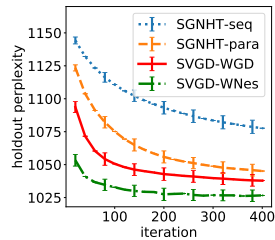

Figure 6. Comparison of SVGD and SGNHT on LDA, as representatives of ParVIs and MCMCs. Average over 10 runs.

The acceleration effect is shown in Fig. 5. We see again that WAG and WNes improve the convergence rate over WGD. PO has a comparable empirical acceleration performance. We note that WAG is sensitive to its parameter α and exhibits minor oscillation, while WNes is more stable. We also compare ParVIs with stochastic gradient Nosé-Hoover thermostats (SGNHT) (Ding et al., 2014), an advanced MCMC
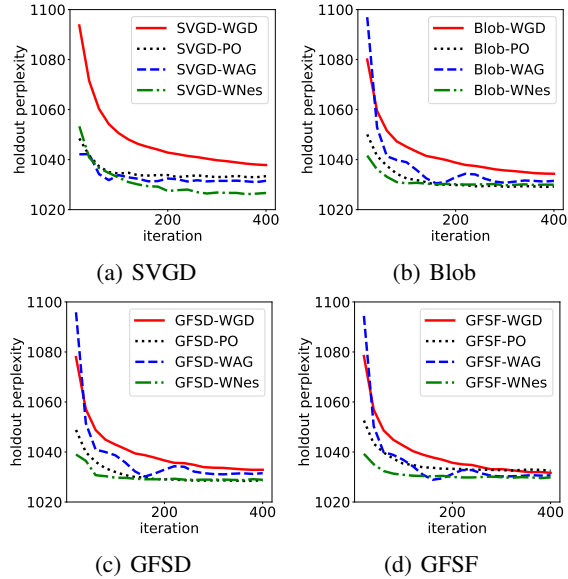
method. Its samples are taken as either the last 20 samples from one chain (-seq), or the very last sample from 20 parallel chains (-para). From Figure 6, we observe the faster convergence of ParVIs over the MCMC method.

7. Conclusions

By exploiting the $\mathcal{P}_2(\mathcal{X})$ gradient flow perspective of ParVIs, we establish a unified theory on the finite-particle approximations of ParVIs, and propose an acceleration framework and a principled bandwidth-selection method to improve ParVIs. The theory recognizes various approximations as a smoothing treatment, by either smoothing the density or smoothing functions. The equivalence of the two smoothing forms connects existing ParVIs, and their necessity reveals the assumptions of ParVIs. Algorithm acceleration is developed via a deep exploration on the geometry of $\mathcal{P}_2(\mathcal{X})$ and the bandwidth method is based on a principle of smoothing. Experiments show more representative particles by the principled bandwidth method, and the speed-up of ParVIs by the acceleration framework.

---

[4]https://cse.buffalo.edu/~changyou/code/SGNHT.zip

Table 1. Results on BNN on the Kin8nm dataset (one of the UCI datasets (Asuncion & Newman, 2007)). Results averaged over 20 runs.

Method	Avg. Test RMSE ($\times 10^{-2}$)				Avg. Test LL			
	SVGD	Blob	GFSD	GFSF	SVGD	Blob	GFSD	GFSF
WGD	8.4±0.2	8.2±0.2	8.0±0.3	8.3±0.2	1.042±0.016	1.079±0.021	1.087±0.029	1.044±0.016
PO	7.8±0.2	8.1±0.2	8.1±0.2	8.0±0.2	1.114±0.022	1.070±0.020	1.067±0.017	1.073±0.016
WAG	7.0±0.2	**7.0±0.2**	7.1±0.1	7.0±0.1	1.167±0.015	**1.169±0.015**	1.167±0.017	1.190±0.014
WNes	**6.9±0.1**	7.0±0.2	**6.9±0.1**	**6.8±0.1**	**1.171±0.014**	1.168±0.014	**1.173±0.016**	**1.193±0.014**

References

Amari, S.-I. *Information geometry and its applications*. Springer, 2016.

Ambrosio, L., Gigli, N., and Savaré, G. *Gradient flows: in metric spaces and in the space of probability measures*. Springer Science & Business Media, 2008.

Asuncion, A. and Newman, D. Uci machine learning repository, 2007.

Benamou, J.-D. and Brenier, Y. A computational fluid mechanics solution to the monge-kantorovich mass transfer problem. *Numerische Mathematik*, 84(3):375–393, 2000.

Blei, D. M., Ng, A. Y., and Jordan, M. I. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan): 993–1022, 2003.

Chen, C. and Zhang, R. Particle optimization in stochastic gradient mcmc. *arXiv preprint arXiv:1711.10927*, 2017.

Chen, C., Ding, N., and Carin, L. On the convergence of stochastic gradient mcmc algorithms with high-order integrators. In *Advances in Neural Information Processing Systems*, pp. 2278–2286, 2015.

Chen, C., Zhang, R., Wang, W., Li, B., and Chen, L. A unified particle-optimization framework for scalable bayesian sampling. *arXiv preprint arXiv:1805.11659*, 2018a.

Chen, W. Y., Mackey, L., Gorham, J., Briol, F.-X., and Oates, C. J. Stein points. *arXiv preprint arXiv:1803.10161*, 2018b.

Chen, Y. and Li, W. Natural gradient in wasserstein statistical manifold. *arXiv preprint arXiv:1805.08380*, 2018.

Cuturi, M. Sinkhorn distances: Lightspeed computation of optimal transport. In *Advances in neural information processing systems*, pp. 2292–2300, 2013.

Detommaso, G., Cui, T., Marzouk, Y., Spantini, A., and Scheichl, R. A stein variational newton method. In *Advances in Neural Information Processing Systems*, pp. 9187–9197, 2018.

Ding, N., Fang, Y., Babbush, R., Chen, C., Skeel, R. D., and Neven, H. Bayesian sampling using stochastic gradient thermostats. In *Advances in neural information processing systems*, pp. 3203–3211, 2014.

Do Carmo, M. P. *Riemannian Geometry*. 1992.

Ehlers, J., Pirani, F., and Schild, A. The geometry of free fall and light propagation, in the book general relativity(papers in honour of jl synge), 63–84, 1972.

Erbar, M. et al. The heat equation on manifolds as a gradient flow in the wasserstein space. In *Annales de l'Institut Henri Poincaré, Probabilités et Statistiques*, volume 46, pp. 1–23. Institut Henri Poincaré, 2010.

Futami, F., Cui, Z., Sato, I., and Sugiyama, M. Frank-wolfe stein sampling. *arXiv preprint arXiv:1805.07912*, 2018.

Gabay, D. Minimizing a differentiable function over a differential manifold. *Journal of Optimization Theory and Applications*, 37(2):177–219, 1982.

Geman, S. and Geman, D. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. In *Readings in Computer Vision*, pp. 564–584. Elsevier, 1987.

Haarnoja, T., Tang, H., Abbeel, P., and Levine, S. Reinforcement learning with deep energy-based policies. *arXiv preprint arXiv:1702.08165*, 2017.

Hoffman, M. D., Blei, D. M., Wang, C., and Paisley, J. Stochastic variational inference. *The Journal of Machine Learning Research*, 14(1):1303–1347, 2013.

Jordan, R., Kinderlehrer, D., and Otto, F. The variational formulation of the fokker–planck equation. *SIAM journal on mathematical analysis*, 29(1):1–17, 1998.

Kheyfets, A., Miller, W. A., and Newton, G. A. Schild's ladder parallel transport procedure for an arbitrary connection. *International Journal of Theoretical Physics*, 39 (12):2891–2898, 2000.

Kováčik, O. and Rákosník, J. On spaces lp̂(x) and wk̂, p(x). *Czechoslovak Mathematical Journal*, 41(4):592–618, 1991.

Li, Y. and Turner, R. E. Gradient estimators for implicit models. *arXiv preprint arXiv:1705.07107*, 2017.

Liu, C. and Zhu, J. Riemannian stein variational gradient descent for bayesian inference. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

Liu, Q. Stein variational gradient descent as gradient flow. In *Advances in neural information processing systems*, pp. 3118–3126, 2017.

Liu, Q. and Wang, D. Stein variational gradient descent: A general purpose bayesian inference algorithm. In *Advances In Neural Information Processing Systems*, pp. 2378–2386, 2016.

Liu, Y., Ramachandran, P., Liu, Q., and Peng, J. Stein variational policy gradient. *arXiv preprint arXiv:1704.02399*, 2017a.

Liu, Y., Shang, F., Cheng, J., Cheng, H., and Jiao, L. Accelerated first-order methods for geodesically convex optimization on riemannian manifolds. In *Advances in Neural Information Processing Systems*, pp. 4875–4884, 2017b.

Lott, J. Some geometric calculations on wasserstein space. *Communications in Mathematical Physics*, 277(2):423–437, 2008.

Lott, J. An intrinsic parallel transport in wasserstein space. *Proceedings of the American Mathematical Society*, 145 (12):5329–5340, 2017.

Neal, R. M. et al. Mcmc using hamiltonian dynamics. *Handbook of Markov Chain Monte Carlo*, 2(11), 2011.

Nesterov, Y. A method of solving a convex programming problem with convergence rate o (1/k2). In *Soviet Mathematics Doklady*, volume 27, pp. 372–376, 1983.

Otto, F. The geometry of dissipative evolution equations: the porous medium equation. 2001.

Patterson, S. and Teh, Y. W. Stochastic gradient riemannian langevin dynamics on the probability simplex. In *Advances in Neural Information Processing Systems*, pp. 3102–3110, 2013.

Pele, O. and Werman, M. Fast and robust earth mover's distances. In *ICCV*, volume 9, pp. 460–467, 2009.

Polyak, B. T. Some methods of speeding up the convergence of iteration methods. *USSR Computational Mathematics and Mathematical Physics*, 4(5):1–17, 1964.

Pu, Y., Gan, Z., Henao, R., Li, C., Han, S., and Carin, L. Vae learning via stein variational gradient descent. In *Advances in Neural Information Processing Systems*, pp. 4239–4248, 2017.

Qi, C., Gallivan, K. A., and Absil, P.-A. Riemannian bfgs algorithm with applications. In *Recent advances in optimization and its applications in engineering*, pp. 183–192. Springer, 2010.

Rezende, D. J. and Mohamed, S. Variational inference with normalizing flows. *arXiv preprint arXiv:1505.05770*, 2015.

Roberts, G. O. and Stramer, O. Langevin diffusions and metropolis-hastings algorithms. *Methodology and computing in applied probability*, 4(4):337–357, 2002.

Shi, J., Sun, S., and Zhu, J. A spectral approach to gradient estimation for implicit distributions. *arXiv preprint arXiv:1806.02925*, 2018.

Steinwart, I. and Christmann, A. *Support vector machines*. Springer Science & Business Media, 2008.

Sutskever, I., Martens, J., Dahl, G., and Hinton, G. On the importance of initialization and momentum in deep learning. In *International conference on machine learning*, pp. 1139–1147, 2013.

Taghvaei, A. and Mehta, P. G. Accelerated gradient flow for probability distributions. *ICLR 2019 submission*, 2018. URL https://openreview.net/forum?id=HkxCEhAqtQ&noteId=H1lvDj2FCX.

Villani, C. *Optimal transport: old and new*, volume 338. Springer Science & Business Media, 2008.

Wainwright, M. J., Jordan, M. I., et al. Graphical models, exponential families, and variational inference. *Foundations and Trends® in Machine Learning*, 1(1–2):1–305, 2008.

Wang, D. and Liu, Q. Learning to draw samples: With application to amortized mle for generative adversarial learning. *arXiv preprint arXiv:1611.01722*, 2016.

Welling, M. and Teh, Y. W. Bayesian learning via stochastic gradient langevin dynamics. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pp. 681–688, 2011.

Wibisono, A., Wilson, A. C., and Jordan, M. I. A variational perspective on accelerated methods in optimization. *Proceedings of the National Academy of Sciences*, 113(47): E7351–E7358, 2016.

Xie, Y., Wang, X., Wang, R., and Zha, H. A fast proximal point method for computing wasserstein distance. *arXiv preprint arXiv:1802.04307*, 2018.

Yuan, X., Huang, W., Absil, P.-A., and Gallivan, K. A. A riemannian limited-memory bfgs algorithm for computing the matrix geometric mean. *Procedia Computer Science*, 80:2147–2157, 2016.

Zhang, H. and Sra, S. An estimate sequence for geodesically convex optimization. In *Conference On Learning Theory*, pp. 1703–1723, 2018.

Zhang, H., Reddi, S. J., and Sra, S. Riemannian svrg: fast stochastic optimization on riemannian manifolds. In *Advances in Neural Information Processing Systems*, pp. 4592–4600, 2016.

Zhou, D.-X. Derivative reproducing properties for kernel methods in learning theory. *Journal of computational and Applied Mathematics*, 220(1):456–463, 2008.

Zhuo, J., Liu, C., Shi, J., Zhu, J., Chen, N., and Zhang, B. Message passing stein variational gradient descent. In *Proceedings of the 35th International Conference on Machine Learning (ICML-18)*, pp. 6013–6022, 2018.

Appendix

A. Proofs

A.1: PROOF OF THEOREM 2

For $v \in \mathcal{H}^D$, the objective can be expressed as:

$$\langle v^{\text{GF}}, v \rangle_{\mathcal{L}_q^2}$$
$$= \mathbb{E}_q[(\nabla \log p - \nabla \log q) \cdot v]$$
$$= \mathbb{E}_q[\nabla \log p \cdot v] - \int_{\mathcal{X}} \nabla q \cdot v \, \mathrm{d}x$$
$$\overset{(*)}{=} \mathbb{E}_q[\nabla \log p \cdot v] + \int_{\mathcal{X}} q \nabla \cdot v \, \mathrm{d}x$$
$$= \mathbb{E}_{q(x)} \left[\sum_{\alpha=1}^{D} \left(\partial_\alpha \log p(x) v_\alpha(x) + \partial_\alpha v_\alpha(x) \right) \right]$$
$$\overset{(\#)}{=} \mathbb{E}_{q(x)} \left[\sum_{\alpha=1}^{D} \left(\partial_\alpha \log p(x) \langle K(x, \cdot), v_\alpha(\cdot) \rangle_{\mathcal{H}} \right. \right.$$
$$\left. \left. + \langle \partial_\alpha K(x, \cdot), v_\alpha(\cdot) \rangle_{\mathcal{H}} \right) \right]$$
$$= \mathbb{E}_{q(x)} \left[\langle K(x, \cdot) \nabla \log p(x), v(\cdot) \rangle_{\mathcal{H}^D} + \langle \nabla K(x, \cdot), v(\cdot) \rangle_{\mathcal{H}^D} \right]$$
$$= \mathbb{E}_{q(x)} \left[\langle K(x, \cdot) \nabla \log p(x) + \nabla K(x, \cdot), v(\cdot) \rangle_{\mathcal{H}^D} \right]$$
$$= \left\langle \mathbb{E}_{q(x)} \left[K(x, \cdot) \nabla \log p(x) + \nabla K(x, \cdot) \right], v(\cdot) \right\rangle_{\mathcal{H}^D}$$
$$= \langle v^{\text{SVGD}}, v \rangle_{\mathcal{H}^D},$$

where the equality $(*)$ holds due to the definition of weak derivative of distributions, and equality $(\#)$ holds due to the reproducing property for any function f in the reproducing kernel Hilbert space $\mathcal{H}$ of kernel K: $\langle K(x, \cdot), f(\cdot) \rangle_{\mathcal{H}} = f(x)$ (Steinwart & Christmann (2008), Chapter 4), and $\langle \partial_{x_\alpha} K(x, \cdot), f(\cdot) \rangle_{\mathcal{H}} = \partial_{x_\alpha} f(x)$ (Zhou, 2008).

A.2: PROOF OF THEOREM 3

When q is absolutely continuous with respect to the Lebesgue measure of $\mathcal{X} = \mathbb{R}^D$, $\mathcal{L}_q^2$ has the same topological properties as $\mathcal{L}^2$, so conclusions we cite below can be adapted from $\mathcal{L}^2$ to $\mathcal{L}_q^2$. Note that the map $\phi \mapsto \phi * K, \mathcal{L}^2 \to \mathcal{L}^2$ is continuous, so $\mathcal{G} := \overline{\{\phi * K : \phi \in \mathcal{C}_c^\infty\}}^{\mathcal{L}_q^2} = \{\phi * K : \phi \in \overline{\mathcal{C}_c^\infty}^{\mathcal{L}^2}\} = \{\phi * K : \phi \in \mathcal{L}^2\} = \{\phi * K : \phi \in L^2\}^D$, where the second last equality holds due to e.g., Theorem 2.11 of (Kováčik & Rákosník, 1991). On the other hand, due to Proposition 4.46 and Theorem 4.47 of (Steinwart & Christmann, 2008), the map $\phi \mapsto \phi * K$ is an isometric isomorphism between $\{\phi * K : \phi \in L^2\}$ and $\mathcal{H}$, the reproducing kernel Hilbert space of K. This indicates that $\mathcal{G}$ is isometrically isomorphic to $\mathcal{H}^D$.

A.3: PROOF OF THEOREM 4

We will redefine some notations in this proof. According to the deduction in Appendix A.1, the objective of the optimization problem Eq. (5) $\langle v^{\text{GF}}, v \rangle_{\mathcal{L}_q^2}$ can be cast as $\mathbb{E}_q[\nabla \log p \cdot v + \nabla \cdot v]$. With $q = \hat{q}$ and $v \in \mathcal{L}_p^2$, we write the optimization problem as:

$$\sup_{v \in \mathcal{L}_p^2, \|v\|=1} \sum_{i=1}^{N} \left(\nabla \log p(x^{(i)}) \cdot v(x^{(i)}) + \nabla \cdot v(x^{(i)}) \right), \tag{7}$$

We will find a sequence of functions $\{v_n\}$ satisfying conditions in Eq. (7) while the objective goes to infinity.

We assume that there exists $r_0 > 0$ such that $p(x) > 0$ for any $\left\| x - x^{(i)} \right\|_\infty < r_0$, $i = 1, 2, \cdots, N$, which is reasonable because it is almost impossible to sample $x^{(i)}$ with $p(x)$ vanishes in every neighborhood of $x^{(i)}$.

Denoting $v(x) = (v_1(x), \cdots, v_D(x))^\top$ for any D-dimensional vector function v and $\nabla f(x) = (\partial_1 f(x), \cdots, \partial_D f(x))^\top$ for any real-valued function f, the objective can be written as:

$$\mathcal{L}_v = \sum_{i=1}^{N} \left(\nabla \log p(x^{(i)}) \cdot v(x^{(i)}) + \nabla \cdot v(x^{(i)}) \right)$$
$$= \sum_{i=1}^{N} \left(\sum_{\alpha=1}^{D} \partial_\alpha [\log p(x^{(i)})] v_\alpha(x^{(i)}) + \sum_{\alpha=1}^{D} \partial_\alpha [v_\alpha(x^{(i)})] \right)$$
$$= \sum_{\alpha=1}^{D} \sum_{i=1}^{N} \left(\partial_\alpha [\log p(x^{(i)})] v_\alpha(x^{(i)}) + \partial_\alpha [v_\alpha(x^{(i)})] \right). \tag{8}$$

For every $v \in \mathcal{L}_p^2, \|v\| = 1$, we can define a function $\phi = (\phi_1, \cdots, \phi_D)^\top \in \mathcal{L}^2$ correspondingly, such that $\phi(x) = p(x)^{\frac{1}{2}} v(x)$, which means $\phi_\alpha(x) = p(x)^{\frac{1}{2}} v_\alpha(x)$, and

$$\|\phi\|_2^2 = \int_{\mathbb{R}^D} \phi^2 \, \mathrm{d}x = \int_{\mathbb{R}^D} \sum_{\alpha=1}^{D} (\phi_\alpha(x))^2 \, \mathrm{d}x$$
$$= \int_{\mathbb{R}^D} \sum_{\alpha=1}^{D} (v_\alpha(x))^2 p(x) \, \mathrm{d}x = \|v\|^2 = 1.$$

Rewrite Eq. (8) in term of ϕ, we have:

$$\mathcal{L}_\phi = \sum_{\alpha=1}^{D} \sum_{i=1}^{N} \left(\partial_\alpha [\log p(x^{(i)})] v_\alpha(x^{(i)}) + \partial_\alpha [v_\alpha(x^{(i)})] \right)$$
$$= \sum_{\alpha=1}^{D} \sum_{i=1}^{N} \left(\partial_\alpha [\log p(x^{(i)})] \phi_\alpha(x^{(i)}) p(x^{(i)})^{-\frac{1}{2}} \right.$$
$$\left. + \partial_\alpha [\phi_\alpha(x^{(i)}) p(x^{(i)})^{-\frac{1}{2}}] \right)$$

$$= \sum_{\alpha=1}^{D} \sum_{i=1}^{N} \left(\frac{1}{2} p(x^{(i)})^{-\frac{3}{2}} \partial_\alpha [p(x^{(i)})] \phi_\alpha(x^{(i)}) \right.$$
$$\left. + p(x^{(i)})^{-\frac{1}{2}} \partial_\alpha [\phi_\alpha(x^{(i)})] \right)$$
$$= \sum_{\alpha=1}^{D} \sum_{i=1}^{N} \left(A_\alpha^{(i)} \phi_\alpha(x^{(i)}) + B^{(i)} \partial_\alpha [\phi_\alpha(x^{(i)})] \right),$$

where $A_\alpha^{(i)} = \frac{1}{2} p(x^{(i)})^{-\frac{3}{2}} \partial_\alpha [p(x^{(i)})]$ and $B^{(i)} = p(x^{(i)})^{-\frac{1}{2}} > 0$. We will now construct a sequence $\{\phi_n\}$ to show the following problem:

$$\inf_{\phi \in \mathcal{L}^2, \|\phi\|=1} \sum_{\alpha=1}^{D} \sum_{i=1}^{N} \left(A_\alpha^{(i)} \phi_\alpha(x^{(i)}) + B^{(i)} \partial_\alpha [\phi_\alpha(x^{(i)})] \right) \tag{9}$$

has no solution, then induce a sequence $\{v_n\}$ by $\{\phi_n\}$ for problem Eq. (7).

Define a sequence of functions

$$\chi_n(x) = \begin{cases} I_n^{-1/2} (1-x^2)^{n/2}, & \text{for } x \in [-1,1], \\ 0, & \text{otherwise.} \end{cases}$$

We have $\int_{\mathbb{R}} \chi_n(x)^2 \, dx = 1$ with $I_n = \int_{-1}^{1} (1-x^2)^n \, dx = \sqrt{\pi} \frac{\Gamma(n+1)}{\Gamma(n+3/2)}$, where $\Gamma(\cdot)$ is the Gamma function. Note that when $x = -1/\sqrt{n}$,

$$\chi_n'(x) = -n I_n^{-\frac{1}{2}} x (1-x^2)^{\frac{n-2}{2}}$$
$$= \pi^{-\frac{1}{4}} \sqrt{\frac{\Gamma(n+\frac{3}{2})}{\Gamma(n+1)}} \sqrt{n} (1-\frac{1}{n})^{\frac{n-2}{2}} \quad (x = -\frac{1}{\sqrt{n}})$$
$$> \pi^{-\frac{1}{4}} \sqrt{n} (1-\frac{1}{n})^{\frac{n-2}{2}}, \quad (\Gamma(n+\frac{3}{2}) > \Gamma(n+1))$$

therefore,

$$\lim_{n \to \infty} \chi_n'(-\frac{1}{\sqrt{n}})$$
$$> \lim_{n \to \infty} \pi^{-\frac{1}{4}} \sqrt{n} (1-\frac{1}{n})^{\frac{n-2}{2}} = \pi^{-\frac{1}{4}} e^{-\frac{1}{2}} \lim_{n \to \infty} \sqrt{n} = +\infty.$$

Denote $x^{(i)} = (x_1^{(i)}, x_2^{(i)}, \cdots, x_D^{(i)})^\top \in \mathbb{R}^D, i = 1, \cdots, N$ and

$$r_1 = \frac{1}{3} \min_{i \neq j} \left\| x^{(i)} - x^{(j)} \right\|_\infty = \frac{1}{3} \min_{\alpha \in \{1,\cdots,D\}, i \neq j} \left| x_\alpha^{(i)} - x_\alpha^{(j)} \right|.$$

We extend χ_n to $\mathbb{R}^D$ as ξ_n with support $\text{supp}(\xi_n) = [-r,r]^D$,

$$\xi_n(x_1, x_2, \cdots, x_D) = r^{-D/2} \prod_{\alpha=1}^{D} \chi_n(\frac{x_i}{r}),$$

where $r = \min\{r_0, r_1\}$. It is easy to show that $\int_{\mathbb{R}^D} \xi_n(x)^2 \, dx = 1$, and

$$\lim_{n \to \infty} \partial_\alpha \xi_n(-\epsilon_n) = +\infty, \quad \alpha = 1, 2, \cdots, D,$$

with $\epsilon_n = \frac{r}{\sqrt{n}} (1, 1, \cdots, 1)^\top$.

We choose $\phi_\alpha(x) = \frac{1}{ND} \sum_{i=1}^{N} \psi_\alpha^{(i)}$, where $\psi_\alpha^{(i)}$ is defined by:

$$\psi_\alpha^{(i)}(x) = \begin{cases} \xi_n(x - x^{(i)} - \epsilon_n), & \text{if } A_\alpha^{(i)} >= 0, \\ -\xi_n(x - x^{(i)} + \epsilon_n), & \text{if } A_\alpha^{(i)} < 0. \end{cases}$$

With $\int_{\mathbb{R}^D} \psi_\alpha^{(i)}(x) \psi_\alpha^{(j)}(x) \, dx = 0, \forall i \neq j$, we know ϕ_n satisfies conditions in Eq. (9). Noting that $\forall i, j$, $A_\alpha^{(i)} \psi_\alpha^{(j)}(x^{(i)}) \geq 0$, and

$$\partial_\alpha \psi_\alpha^{(j)}(x^{(i)}) = \begin{cases} +\infty, & \text{when } n \to \infty, \text{ if } i = j, \\ 0, & \text{if } i \neq j, \end{cases}$$

we see $\mathcal{L}_{\phi_n} \to +\infty$ in Eq. (**??**) when $n \to \infty$.

Since $\text{supp}(\phi_n) \subset \text{supp}(p)$, we can induce a sequence of $\{v_n\}$ from $\{\phi_n\}$ as $v_n = \phi_n / \sqrt{p(x)}$, which satisfies restrictions in Eq. (7) and the objective $\mathcal{L}_{v_n}$ will go to infinity when $n \to \infty$. Note that any element in $\mathcal{L}_p^2$, as a function, cannot take infinite value. So the infinite supremum of the objective in Eq. (7) cannot be obtained by any element in $\mathcal{L}_p^2$, thus no optimal solution for the optimization problem.

A.4: DEDUCTION OF PROPOSITION 5

We first derive the exact inverse exponential map on the Wasserstein space $\mathcal{P}_2(\mathcal{X})$, then develop finite-particle estimation for it. Given $q, r \in \mathcal{P}_2(\mathcal{X})$, $\text{Exp}_q^{-1}(r)$ is defined as the tangent vector at q of the geodesic curve $q_{t \in [0,1]}$ from q to r. When q is absolutely continuous, the optimal transport map $\mathcal{T}_q^r$ from q to r exists (Villani (2008), Thm. 10.38). This condition fits the case of ParVIs since as our theory indicates, ParVIs have to do a smoothing treatment in any way, which is equivalent to assume an absolutely continuous distribution q. Under this case, the geodesic is given by $q_t = \left((1-t) \, \text{id} + t\mathcal{T}_q^r \right)_{\#} q$ (Ambrosio et al. (2008), Thm. 7.2.2), and its tangent vector at q (*i.e.*, $t = 0$) can be characterized by $\text{Exp}_q^{-1}(r) = \lim_{t \to 0} \frac{1}{t} (\mathcal{T}_q^{q_t} - \text{id})$ (Ambrosio et al. (2008), Prop. 8.4.6). Due to the uniqueness of optimal transport map, we have $\mathcal{T}_q^{q_t} = (1-t) \, \text{id} + t\mathcal{T}_q^r$, so we finally get $\text{Exp}_q^{-1}(r) = \mathcal{T}_q^r - \text{id}$.

To estimate it with a finite set of particles, we approximate the optimal transport map with the discrete one from particles $\{x^{(i)}\}_{i=1}^{N}$ of q to particles $\{y^{(i)}\}_{i=1}^{N}$ of r. As mentioned in the main context, it is a costly task, and the Sinkhorn approximations (for both the original version (Cuturi, 2013) and an improved version (Xie et al., 2018)) suffers from an
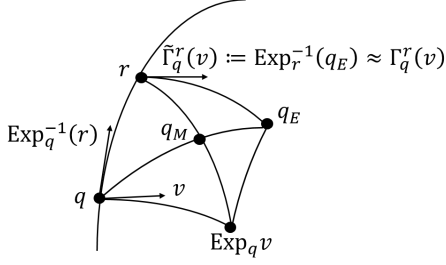
Figure 7. Illustration of the Schild's ladder method. Figure inspired by (Kheyfets et al., 2000).

unstable behavior in our experiments. We now utilize the pairwise-close condition and develop a light and stable approximation. The pairwise-close condition $d(x^{(i)}, y^{(i)}) \ll \min\{\min_{j \neq i} d(x^{(i)}, x^{(j)}), \min_{j \neq i} d(y^{(i)}, y^{(j)})\}$ indicates that $\frac{d(x^{(i)}, x^{(j)})}{d(x^{(j)}, y^{(j)})} \gg 1$, for any $i \neq j$. On the other hand, due to triangle inequality, we have $d(x^{(i)}, y^{(j)}) \geq |d(x^{(i)}, x^{(j)}) - d(x^{(j)}, y^{(j)})|$, or equivalently $\frac{d(x^{(i)}, y^{(j)})}{d(x^{(j)}, y^{(j)})} \geq \left|\frac{d(x^{(i)}, x^{(j)})}{d(x^{(j)}, y^{(j)})} - 1\right|$. Due to the above knowledge $\frac{d(x^{(i)}, x^{(j)})}{d(x^{(j)}, y^{(j)})} \gg 1$ by the pairwise-close condition, we have $\frac{d(x^{(i)}, y^{(j)})}{d(x^{(j)}, y^{(j)})} \gg 1$, or equivalently (by switching i and j) $d(x^{(i)}, y^{(i)}) \ll \min_{j \neq i} d(x^{(i)}, y^{(j)})$. This means that when transporting $\{x^{(i)}\}_i$ to $\{y^{(i)}\}_i$, the map $x^{(i)} \mapsto y^{(i)}$ for any i, has presumably the least cost. More formally, consider any amount of transportation from $x^{(i)}$ to $y^{(j)}$ other than $y^{(i)}$. It will introduce a change in the transportation cost that is proportional to $d(x^{(i)}, y^{(j)}) - d(x^{(i)}, y^{(i)}) + d(x^{(j)}, y^{(i)}) - d(x^{(j)}, y^{(j)})$, which is always positive due to our above recognition. Thus we can reasonably approximate the optimal transport map $\mathcal{T}_q^r$ by the discrete one $\mathcal{T}_q^r(x^{(i)}) \approx y^{(i)}$. With this approximation, we have $\left(\text{Exp}_q^{-1}(r)\right)(x^{(i)}) = \mathcal{T}_q^r(x^{(i)}) - x^{(i)} \approx y^{(i)} - x^{(i)}$.

A.5: DEDUCTION OF PROPOSITION 6

We derive the finite-particle estimation of the parallel transport on the Wasserstein space $\mathcal{P}_2(\mathcal{X})$. We follow the Schild's ladder method (Ehlers et al., 1972; Kheyfets et al., 2000) to parallel transport a tangent vector at q, $v \in T_q \mathcal{P}_2(\mathcal{X})$, to the tangent space at r, $T_r \mathcal{P}_2(\mathcal{X})$. As shown in Fig. 7, given q, r and $v \in T_q \mathcal{P}_2(\mathcal{X})$, the procedure to approximate $\Gamma_q^r(v)$ is

1. find the point $\text{Exp}_q(v)$;

2. find the midpoint of the geodesic from r to $\text{Exp}_q(v)$: $q_M := \text{Exp}_q(\frac{1}{2} \text{Exp}_q^{-1}(\text{Exp}_q(v)))$;

3. extrapolate the geodesic from q to q_M by doubling the length to find $q_E := \text{Exp}_q(2 \text{Exp}_q^{-1}(q_M))$;

4. take the approximator $\tilde{\Gamma}_q^r(v) := \text{Exp}_r^{-1}(q_E) \approx \Gamma_q^r(v)$.

Note that the Schild's ladder method only requires the exponential map and its inverse. It provides a tractable first-order approximation $\tilde{\Gamma}_q^r$ of the parallel transport Γ_q^r under Levi-Civita connection, as needed.

Assume q and r are close in the sense of the Wasserstein distance, so that the Schild's ladder finds a good first-order approximation. In the following we consider transporting εv for small $\varepsilon > 0$ for the sake of the pairwise-close condition, and the result can be recovered by noting the linearity of the parallel transport: $\Gamma_q^r(\varepsilon v) = \varepsilon \Gamma_q^r(v)$. Let $\{x^{(i)}\}_{i=1}^N$ and $\{y^{(i)}\}_{i=1}^N$ be the sets of samples of q and r, respectively, and assume that they are pairwise close.

Now we follow the procedure.

1. The measure $\text{Exp}_q(\varepsilon v)$ can be identified as $(\text{id} + \varepsilon v)_\# q$ due to the knowledge on the exponential map on $\mathcal{P}_2(\mathcal{X})$ explained in Section 4, thus $\{x^{(i)} + \varepsilon v(x^{(i)})\}_{i=1}^N$ is a set of samples of $\text{Exp}_q(\varepsilon v)$, and still pairwise close to $\{y^{(i)}\}_i$.

2. The optimal map $\mathcal{T}$ from r to $\text{Exp}_q(\varepsilon v)$ can be approximated by $\mathcal{T}(y^{(i)}) = x^{(i)} + \varepsilon v(x^{(i)})$ since the two sets of samples are pairwise close. According to Theorem 7.2.2 of (Ambrosio et al., 2008), the geodesic from r to $\text{Exp}_q(\varepsilon v)$ is $t \mapsto ((1 - t)\text{id} + t\mathcal{T})_\# r$. Thus a set of samples of q_M, *i.e.*, the midpoint of the geodesic, can be derived as $\{\frac{1}{2}(y^{(i)} + x^{(i)} + \varepsilon v(x^{(i)}))\}_i$.

3. Similarly, a set of samples of q_E is found as $\{(1 - t)x^{(i)} + \frac{1}{2}t(y^{(i)} + x^{(i)} + \varepsilon v(x^{(i)}))\}_i \big|_{t=2} = \{y^{(i)} + \varepsilon v(x^{(i)})\}_i$ and is pairwise close to $\{y^{(i)}\}_i$.

4. The approximated transported tangent vector $\text{Exp}_r^{-1}(q_E)$ satisfies $(\text{Exp}_r^{-1}(q_E))(y^{(i)}) = \varepsilon v(x^{(i)})$.

Finally, we get the approximation $(\Gamma_q^r(v))(y^{(i)}) \approx (\tilde{\Gamma}_q^r(v))(y^{(i)}) = v(x^{(i)})$.

B. Derivations of GFSF Vector Field $\hat{u}^{\text{GFSF}}$

B.1: DERIVATION WITH VECTOR-VALUED FUNCTIONS

The vector field $\hat{u}^{\text{GFSF}}$ is identified by the optimization problem (6):

$$\min_{u \in \mathcal{L}^2} \max_{\substack{\phi \in \mathcal{H}^D, \\ \|\phi\|_{\mathcal{H}^D} = 1}} \left(\sum_{i=1}^N \left(\phi(x^{(i)}) \cdot u^{(i)} - \nabla \cdot \phi(x^{(i)})\right)\right)^2,$$

where $u^{(i)} := u(x^{(i)})$. For ϕ in $\mathcal{H}^D$, by using the reproducing property $\langle \phi_\alpha(\cdot), K(x, \cdot)\rangle_{\mathcal{H}} = \phi_\alpha(x)$ and $\langle \phi_\alpha(\cdot), \partial_{x_\beta} K(x, \cdot)\rangle_{\mathcal{H}} = \partial_{x_\beta} \phi_\alpha(x)$ (Zhou, 2008), we can

write the objective function as:

$$\left(\sum_\alpha \sum_j \left(u_\alpha^{(j)} \phi_\alpha(x^{(j)}) - \partial_{x_\alpha^{(j)}} \phi_\alpha(x^{(j)}) \right) \right)^2$$

$$= \left(\sum_\alpha \left\langle \sum_j \left(u_\alpha^{(j)} K(x^{(j)}, \cdot) - \partial_{x_\alpha^{(j)}} K(x^{(j)}, \cdot) \right), \phi_\alpha(\cdot) \right\rangle_{\mathcal{H}} \right)^2$$

$$= \left\langle \sum_j \left(u^{(j)} K(x^{(j)}, \cdot) - \nabla_{x^{(j)}} K(x^{(j)}, \cdot) \right), \phi(\cdot) \right\rangle_{\mathcal{H}^D}^2.$$

We denote $\zeta := \sum_j \left(u^{(j)} K(x^{(j)}, \cdot) - \nabla_{x^{(j)}} K(x^{(j)}, \cdot) \right) \in \mathcal{H}^D$. Then the optimal value of the objective after maximizing out ϕ is $\|\zeta\|_{\mathcal{H}^D}^2 = \sum_{i,j} \left(u^{(i)} u^{(j)} K(x^{(i)}, x^{(j)}) - 2 u^{(i)} \nabla_{x^{(j)}} K(x^{(j)}, x^{(i)}) + \nabla_{x^{(i)}} \nabla_{x^{(j)}} K(x^{(i)}, x^{(j)}) \right) = \operatorname{tr}(\hat{u}\hat{K}\hat{u}^\top) - 2\operatorname{tr}(\hat{K}'\hat{u}^\top) + \text{const}$, where $\hat{u}_{:,i} := u^{(i)}$, and $\hat{K}$, $\hat{K}'$ are defined in the main text. To minimize this quadratic function with respect to $\hat{u}$, we further differentiate it with respect to $\hat{u}$ and solve for the stationary point. This finally gives the result $\hat{u}^{\text{GFSF}} = \hat{K}'\hat{K}^{-1}$.

B.2: DERIVATION WITH SCALAR-VALUED FUNCTIONS

We denote φ as scalar-valued functions on $\mathcal{X}$. For the equality $u(x) = -\nabla \log q(x)$, or $u(x)q(x) + \nabla q(x) = 0$, to hold in the weak sense with scalar-valued test function, we mean:

$$\mathbb{E}_{q(x)}[\varphi(x)u(x) - \nabla\varphi(x)] = 0, \forall \varphi \in C_c^\infty.$$

Let $\{x^{(j)}\}_j$ be a set of samples of $q(x)$. Then the above requirement on $u(x)$ is:

$$\sum_j \left(\varphi(x^{(j)})u^{(j)} - \nabla\varphi(x^{(j)}) \right) = 0, \forall \varphi \in C_c^\infty, \quad (10)$$

where $u^{(j)} = u(x^{(j)})$. As analyzed above, for a valid vector field, we have to smooth the function φ.

For the above considerations, we restrict φ in Eq. (10) to be in the Reproducing Kernel Hilbert Space (RKHS) $\mathcal{H}$ of some kernel $K(\cdot, \cdot)$, and convert the equation as the following optimization problem:

$$\min_{\hat{u} \in \mathbb{R}^{D \times N}} \max_{\substack{\varphi \in \mathcal{H}, \\ \|\varphi\|_{\mathcal{H}}=1}} J(\hat{u}, \varphi) := \sum_{j,\alpha} \left(\varphi(x^{(j)})\hat{u}_{\alpha j} - \partial_{x_\alpha^{(j)}}\varphi(x^{(j)}) \right)^2$$

where $\hat{u}_{\alpha j} := u_\alpha(x^{(j)})$. By using the reproducing properties of RKHS, we can write $J(\hat{u}, \varphi)$ as:

$$J(\hat{u}, \varphi) = \sum_\alpha \langle \varphi(\cdot), \zeta_\alpha(\cdot) \rangle_{\mathcal{H}}^2,$$

$$\zeta_\alpha(\cdot) := \sum_j \left(\hat{u}_{\alpha j} K(x^{(j)}, \cdot) - \partial_{x_\alpha^{(j)}} K(x^{(j)}, \cdot) \right).$$

By linear algebra operations, we have:

$$\max_{\varphi \in \mathcal{H}, \|\varphi\|_{\mathcal{H}}=1} J(\hat{u}, \varphi) = \lambda_1(A(\hat{u})),$$

where $\lambda_1(A(\hat{u}))$ is the largest eigenvalue of matrix A, and $A(\hat{u})_{\alpha\beta} = \langle \zeta_\alpha(\cdot), \zeta_\beta(\cdot) \rangle_{\mathcal{H}}$, or:

$$A(\hat{u}) = \hat{u}\hat{K}\hat{u}^\top - (\hat{K}'\hat{u}^\top + \hat{u}\hat{K}'^\top) + \hat{K}'',$$

with $\hat{K}''_{\alpha\beta} := \sum_{i,j} \partial_{x_\alpha^{(i)}} \partial_{x_\beta^{(j)}} K(x^{(i)}, x^{(j)})$. For distinct samples $\hat{K}$ is positive-definite, so we can conduct Cholesky decomposition: $\hat{K} = GG^\top$ with G non-singular. Note that $A(\hat{u}) = (\hat{u}G - \hat{K}'G^{-1\top})(\hat{u}G - \hat{K}'G^{-1\top})^\top + (\hat{K}'' - \hat{K}'\hat{K}^{-1}\hat{K}'^\top)$. So whenever $\hat{u}G \neq \hat{K}'G^{-1\top}$, the first term will be positive semidefinite with positive largest eigenvalue, which makes $\lambda_1(A(u)) > \lambda_1(\hat{K}'' - \hat{K}'\hat{K}^{-1}\hat{K}'^\top)$, a constant with respect to $\hat{u}$. So to minimize $\lambda_1(A(\hat{u}))$, we require $\hat{u}G = \hat{K}'G^{-1\top}$, i.e., $\hat{u} = \hat{K}'(GG^\top)^{-1} = \hat{K}'\hat{K}^{-1}$. This result coincides with the one for vector-valued functions $\phi \in \mathcal{H}^D$.

In practice, for numerical stability, we add a small diagonal matrix to $\hat{K}$ before conducting inversion. This is a common practice. Particularly, it is adopted in Li & Turner (2017) for the same estimate.

C. Details on Accelerated First-Order Methods on the Wasserstein Space $\mathcal{P}_2(\mathcal{X})$

C.1: SIMPLIFICATION OF RIEMANNIAN ACCELERATED GRADIENT (RAG) WITH APPROXIMATIONS

We consider the general version of RAG (Alg. 2 of Liu et al. (2017b)). It updates the target variable q_k as:

$$q_k = \operatorname{Exp}_{r_{k-1}}(\varepsilon v_{k-1}),$$

where $v_{k-1} := -\operatorname{grad} \operatorname{KL}(r_{k-1})$. The update rule for the auxiliary variable r_k is given by the solution of the following non-linear equation (see Alg. 2 and Eq. (5) of Liu et al. (2017b)):

$$\Gamma_{r_k}^{r_{k-1}} \left(\frac{k}{\alpha - 1} \operatorname{Exp}_{r_k}^{-1}(q_k) + \frac{Dv_k}{\|v_k\|_{r_k}} \right)$$
$$= \frac{k-1}{\alpha - 1} \operatorname{Exp}_{r_{k-1}}^{-1}(q_{k-1}) - \frac{k+\alpha - 2}{\alpha - 1}\varepsilon v_{k-1} + \frac{Dv_{k-1}}{\|v_{k-1}\|_{r_{k-1}}}.$$

Here we focus on simplifying this complicated update rule for r_k with moderate approximations. We note that the original work of RAG (Liu et al., 2017b) actually adopted these approximations in experiments, but the simplification of the general algorithm is not given in the work.

Applying $\left(\Gamma_{r_k}^{r_{k-1}} \right)^{-1}$ to both sides of the equation and noticing that $\left(\Gamma_{r_k}^{r_{k-1}} \right)^{-1} = \Gamma_{r_{k-1}}^{r_k}$, the above equation can be

reformulated as:

$$\frac{k}{\alpha-1}\operatorname{Exp}^{-1}_{r_k}(q_k) + \frac{Dv_k}{\|v_k\|_{r_k}}$$
$$=\Gamma^{r_k}_{r_{k-1}}\left(\frac{k-1}{\alpha-1}\operatorname{Exp}^{-1}_{r_{k-1}}(q_{k-1}) - \frac{k+\alpha-2}{\alpha-1}\varepsilon v_{k-1}\right)$$
$$+\frac{D\Gamma^{r_k}_{r_{k-1}}(v_{k-1})}{\|v_{k-1}\|_{r_{k-1}}}.$$

Approximating v_k on the left hand side of the equation by $\Gamma^{r_k}_{r_{k-1}}(v_{k-1})$ and noting that $\left\|\Gamma^{r_k}_{r_{k-1}}(v_{k-1})\right\|_{r_k} = \|v_{k-1}\|_{r_{k-1}}$, we have:

$$\frac{k}{\alpha-1}\operatorname{Exp}^{-1}_{r_k}(q_k)$$
$$=\Gamma^{r_k}_{r_{k-1}}\left(\frac{k-1}{\alpha-1}\operatorname{Exp}^{-1}_{r_{k-1}}(q_{k-1}) - \frac{k+\alpha-2}{\alpha-1}\varepsilon v_{k-1}\right).$$

Using the fact that $\operatorname{Exp}^{-1}_{r_k}(q_k) = -\Gamma^{r_k}_{q_k}(\operatorname{Exp}^{-1}_{q_k}(r_k))$ and applying $\left(\Gamma^{r_k}_{q_k}\right)^{-1} = \Gamma^{q_k}_{r_k}$ to both sides of the equation, we have:

$$\operatorname{Exp}^{-1}_{q_k}(r_k)$$
$$= -\Gamma^{q_k}_{r_k}\Gamma^{r_k}_{r_{k-1}}\left(\frac{k-1}{k}\operatorname{Exp}^{-1}_{r_{k-1}}(q_{k-1}) - \frac{k+\alpha-2}{k}\varepsilon v_{k-1}\right).$$

Approximating $\Gamma^{q_k}_{r_k}\Gamma^{r_k}_{r_{k-1}}$ by $\Gamma^{q_k}_{r_{k-1}}$, we finally have $r_k =$

$$\operatorname{Exp}_{q_k}\left[-\Gamma^{q_k}_{r_{k-1}}\left(\frac{k-1}{k}\operatorname{Exp}^{-1}_{r_{k-1}}(q_{k-1}) - \frac{k+\alpha-2}{k}\varepsilon v_{k-1}\right)\right].$$

C.2: REFORMULATION OF RIEMANNIAN NESTEROV'S METHOD (RNES)

We consider the constant step version of RNes (Alg. 2 of (Zhang & Sra, 2018)). It introduces an additional auxiliary variable $s_k \in \mathcal{P}_2(\mathcal{X})$, and update the variables in iteration k as:

$$r_{k-1} = \operatorname{Exp}_{q_{k-1}}\left(c_1 \operatorname{Exp}^{-1}_{q_{k-1}}(s_{k-1})\right), \tag{11a}$$

$$q_k = \operatorname{Exp}_{r_{k-1}}(\varepsilon v_{k-1}), \tag{11b}$$

$$s_k = \operatorname{Exp}_{r_{k-1}}\left(\frac{1-\alpha}{1+\beta}\operatorname{Exp}^{-1}_{r_{k-1}}(s_{k-1}) + \frac{\alpha}{(1+\beta)\gamma}v_{k-1}\right), \tag{11c}$$

where $v_{k-1} := -\operatorname{grad}\operatorname{KL}(r_{k-1})$, and the coefficients α, γ, c_1 are set by a step size $\varepsilon > 0$, a shrinkage parameter $\beta > 0$, and a parameter $\mu > 0$ upper bounding the Lipschitz coefficient of the gradient of the objective, in the

following way:

$$\alpha = \frac{\sqrt{\beta^2 + 4(1+\beta)\mu\varepsilon} - \beta}{2},$$
$$\gamma = \frac{\sqrt{\beta^2 + 4(1+\beta)\mu\varepsilon} - \beta}{\sqrt{\beta^2 + 4(1+\beta)\mu\varepsilon} + \beta}\mu,$$
$$c_1 = \frac{\alpha\gamma}{\gamma + \alpha\mu}.$$

Now we simplify the update rule by collapsing the variable s. Referring to Eq. (11a), the variable s_{k-1} can be expressed by:

$$s_{k-1} = \operatorname{Exp}_{q_{k-1}}\left(\frac{1}{c_1}\operatorname{Exp}^{-1}_{q_{k-1}}(r_{k-1})\right).$$

This result indicates that s_{k-1} lies on the $1/c_1$ portion of the geodesic from q_{k-1} to r_{k-1}, which is the $(1 - 1/c_1)$ portion of the geodesic from r_{k-1} to q_{k-1}. According to this knowledge, we have:

$$\operatorname{Exp}^{-1}_{r_{k-1}}(s_{k-1}) = \left(1 - \frac{1}{c_1}\right)\operatorname{Exp}^{-1}_{r_{k-1}}(q_{k-1}).$$

Substitute this result into Eq. (11c), we have:

$$s_k = \operatorname{Exp}_{r_{k-1}}\left(\frac{1-\alpha}{1+\beta}\left(1 - \frac{1}{c_1}\right)\operatorname{Exp}^{-1}_{r_{k-1}}(q_{k-1})\right.$$
$$\left. + \frac{\alpha}{(1+\beta)\gamma\varepsilon}\operatorname{Exp}^{-1}_{r_{k-1}}(q_k)\right),$$

where we have also substituted v_{k-1} with $\frac{1}{\varepsilon}\operatorname{Exp}^{-1}_{r_{k-1}}(q_k)$ according to Eq. (11b). Leveraging Eq. (**??**) to simplify the coefficients in the above equation, we get:

$$s_k = \operatorname{Exp}_{r_{k-1}}\left((1-c_2)\operatorname{Exp}^{-1}_{r_{k-1}}(q_{k-1}) + c_2\operatorname{Exp}^{-1}_{r_{k-1}}(q_k)\right),$$

where the coefficient $c_2 := 1/\alpha$. Replacing $k \to k+1$ in Eq. (11a) and substitute with the above result, we have the update rule for r_k:

$$r_k = \operatorname{Exp}_{q_k}\left\{c_1 \operatorname{Exp}^{-1}_{q_k}\left[\operatorname{Exp}_{r_{k-1}}\left((1-c_2)\operatorname{Exp}^{-1}_{r_{k-1}}(q_{k-1})\right.\right.\right.$$
$$\left.\left.\left. + c_2\operatorname{Exp}^{-1}_{r_{k-1}}(q_k)\right)\right]\right\},$$

which builds the update rule of RNes together with Eq. (11a).

In our implementation, the parameters are tackled with ε, β, μ instead of setting c_1, c_2 directly. The shrinkage parameter β is set in the scale of $\sqrt{\mu\varepsilon}$. In our Alg. 1, the coefficient $c_1(c_2 - 1)$ can be expressed as:

$$1 + \beta - \frac{2(1+\beta)(2+\beta)\mu\varepsilon}{\sqrt{\beta^2 + 4(1+\beta)\mu\varepsilon} - \beta + 2(1+\beta)\mu\varepsilon}.$$

C.3: DEDUCTION OF WASSERSTEIN ACCELERATED GRADIENT (WAG) AND WASSERSTEIN NESTEROV'S METHOD (WNES) (ALG. 1)

First consider developing WAG based on RAG. We denote the vector field $\zeta_{k-1} := \frac{k-1}{k}\text{Exp}_{r_{k-1}}^{-1}(q_{k-1}) - \frac{k+\alpha-2}{k}\varepsilon v_{k-1}$ for simplicity, so $r_k = \text{Exp}_{q_k}\left[-\Gamma_{r_{k-1}}^{q_k}(\zeta_{k-1})\right]$, due to the update rule of RAG. We assume that $\{x_{k-1}^{(i)}\}_{i=1}^N$ of q_{k-1} and $\{y_{k-1}^{(i)}\}_{i=1}^N$ of r_{k-1} are pairwise close, so from Section 4 we know that $\text{Exp}_{r_{k-1}}^{-1}(q_{k-1})(y_{k-1}^{(i)}) = x_{k-1}^{(i)} - y_{k-1}^{(i)}$, thus $\zeta_{k-1}(y_{k-1}^{(i)}) = \frac{k-1}{k}(x_{k-1}^{(i)} - y_{k-1}^{(i)}) - \frac{k+\alpha-2}{k}\varepsilon v_{k-1}^{(i)}$. Due to the update rule for q_k that we already discovered: $x_k^{(i)} = y_{k-1}^{(i)} + \varepsilon v_{k-1}^{(i)}$, we know that $\{x_k^{(i)}\}_{i=1}^N$ of q_k and $\{y_{k-1}^{(i)}\}_{i=1}^N$ of r_{k-1} are pairwise close, for small enough step size ε. Using the parallel transport estimate developed above with Schild's ladder method, $\left(\Gamma_{r_{k-1}}^{q_k}(\zeta_{k-1})\right)(x_k^{(i)}) \approx \zeta_{k-1}(y_{k-1}^{(i)})$. So finally, we assign $y_k^{(i)} = x_k^{(i)} - \left(\Gamma_{r_{k-1}}^{q_k}(\zeta_{k-1})\right)(x_k^{(i)}) \approx x_k^{(i)} - \zeta_{k-1}(y_{k-1}^{(i)}) = x_k^{(i)} - \frac{k-1}{k}(x_{k-1}^{(i)} - y_{k-1}^{(i)}) + \frac{k+\alpha-2}{k}\varepsilon v_{k-1}^{(i)}$ as a sample of r_k.

We note that initially $x_0^{(i)} = y_0^{(i)}$. Assume $\{x_{k-1}^{(i)}\}_{i=1}^N$ and $\{y_{k-1}^{(i)}\}_{i=1}^N$ are pairwise close, so for sufficiently small ε, $\zeta_{k-1}(y_{k-1}^{(i)})$ is an infinitesimal vector for all i. This, in turn, indicates that $\{x_k^{(i)}\}_{i=1}^N$ of q_k and $\{y_k^{(i)}\}_{i=1}^N$ of r_k are pairwise close, which provides the assumption for the next iteration. The derivation of WNes based on RNes can be developed similarly, and we omit verbosing the procedure.

D. Details on the HE Method for Bandwidth Selection

We first note that the bandwidth selection problem cannot be solved using theories of heat kernels, which aims to find the evolving density under the Brownian motion with known initial distribution, while in our case the density is unknown and we want to find an update on samples to approximate the effect of Brownian motion.

According to the derivation in the main context, we write the dimensionless final objective explicitly:

$$\frac{1}{h^{D+2}}\sum_k \lambda(x^{(k)})^2$$
$$= \frac{1}{h^{D+2}}\sum_k \Big[\Delta\tilde{q}(x^{(k)};\{x^{(i)}\}_i)$$
$$+ \sum_j \nabla_{x^{(j)}}\tilde{q}(x^{(k)};\{x^{(i)}\}_i)\cdot\nabla\log\tilde{q}(x^{(j)};\{x^{(i)}\}_i)\Big]^2.$$

For $\tilde{q}(x;\{x^{(j)}\}_j) = (1/Z)\sum_j c(\|x - x^{(j)}\|^2/(2h))$, the above objective becomes:

$$\sum_k\Big(\sum_j\Big[c_j''(x)\|x - x^{(j)}\|^2 + Dhc_j'(x)$$
$$+ \frac{\left(\sum_i c_{ij}'x^{(i)}\right) - \left(\sum_i c_{ij}\right)x^{(j)}}{\left(\sum_i c_{ij}\right)}\cdot(x - x^{(j)})c_j'(x)\Big]\Big)^2,$$

where $c_j'(x) = c'(\|x - x^{(j)}\|^2/(2h))$, $c_{ij}' = c_j'(x^{(i)})$, $c_{ij} = c(\|x^{(i)} - x^{(j)}\|^2/(2h))$. For Gaussian kernel $c(r) = (2\pi h)^{-\frac{D}{2}}e^{-r}$, denoting $g_k^2(h)$ as the summand for k of the l.h.s. of the above equation, we have:

$$(2\pi)^{\frac{D}{2}}g_k(h)$$
$$= \left(\sum_j e_{kj}\|d_{kj}\|^2\right) - hD\left(\sum_j e_{kj}\right)$$
$$- \sum_j\left(\sum_i e_{ij}\right)^{-1}e_{jk}d_{jk}\cdot\left(\sum_i e_{ij}d_{ij}\right),$$

$$(2\pi)^{\frac{D}{2}}g_k'(h)$$
$$= \frac{1}{2h^2}\left(\sum_j e_{jk}\|d_{jk}\|^4\right) - \frac{D}{h}\left(\sum_j e_{jk}\|d_{jk}\|^2\right)$$
$$+ \left(\frac{D^2}{2} - D\right)\left(\sum_j e_{jk}\right)$$
$$- \frac{1}{2h^2}\sum_j\left(\sum_i e_{ij}\right)^{-1}e_{jk}d_{jk}\cdot\left(\sum_i e_{ij}\|d_{ij}\|^2 d_{ij}\right)$$
$$- \frac{1}{2h^2}\sum_j\left(\sum_i e_{ij}\right)^{-1}e_{jk}\|d_{jk}\|^2 d_{jk}\cdot\left(\sum_i e_{ij}d_{ij}\right)$$
$$+ \frac{1}{2h^2}\sum_j\left(\sum_i e_{ij}\right)^{-2}\left(\sum_i e_{ij}\|d_{ij}\|^2\right)e_{jk}d_{jk}\cdot\left(\sum_i e_{ij}d_{ij}\right)$$
$$+ \frac{D}{2h}\sum_j\left(\sum_i e_{ij}\right)^{-1}e_{jk}d_{jk}\cdot\left(\sum_i e_{ij}d_{ij}\right),$$

where $d_{ij} = x^{(i)} - x^{(j)}$, $e_{ij} = e^{-\|d_{ij}\|^2/(2h)-(D/2)\log h}$.

Although the evaluation of $g_k(h)$ may induce some computation cost, the optimization is with respect to a scalar. In each particle update iteration, before estimating the vector field v, we first update the previous bandwidth by one-step exploration with quadratic interpolation, which only requires one derivative evaluation and two value evaluations.

E. Detailed Settings and Parameters of Experiments

E.1: DETAILED SETTINGS AND PARAMETERS OF THE SYNTHETIC EXPERIMENT

The bimodal toy target distribution is inspired by the one of (Rezende & Mohamed, 2015). The logarithm of the target

density $p(z)$ for $z = (z_1, z_2) \in \mathbb{R}^2$ is given by:

$$\log p(z) = -2(\|z\|_2^2 - 3)^2 + \log(e^{-2(z_1-3)^2} + e^{-2(z_1+3)^2})$$
$$+ \text{const.}$$

The region shown in each figure is $[-4, 4] \times [-4, 4]$. The number of particles is 200, and all particles are initialized with standard Gaussian $\mathcal{N}(0, 1)$.

All methods are run for 400 iterations, and all follows the plain WGD method. SVGD uses fixed step size 0.3, while other methods (Blob, GFSD, GFSF) share the fixed step size 0.01. This is because that the updating direction of SVGD is a kernel smoothed one, so it may have a different scale from other methods. Note that the AdaGrad with momentum method in the original SVGD paper (Liu & Wang, 2016) is not used. For GFSF, a small diagonal matrix $0.01I$ is added to $\hat{K}$ before conducting inversion, as discussed at the end of Appendix B.2.

E.2: DETAILED SETTINGS AND PARAMETERS OF THE BLR EXPERIMENT

We adopt the same settings as (Liu & Wang, 2016), which is also adopted by (Chen et al., 2018a). The Covertype dataset contains 581,012 items with each 54 features. Each run uses a random 80%-20% split of the dataset.

For the model, parameters of the Gamma prior on the precision of the Gaussian prior of the weight are $a_0 = 1.0$, $b_0 = 100$ (b_0 is the scale parameter, not the rate parameter). All methods use 100 particles, randomly initialized by the prior. Batch size for all methods is 50.

Detailed parameters of various methods are provided in Table 2. The WGD column provides the step size. The format of the PO column is "PO parameters(decaying exponent, remember rate, injected noise variance), step size". Both methods use a fixed step size, while the WAG and WNes methods use a decaying step size. The format of WAG column is "WAG parameter α, (step size decaying exponent, step size)" (see Alg. 1), and the format of WNes column is "Wnes parameters (μ, β), (step size decaying exponent, step size)" (see Appendix C.2). One exception is that SVGD-WGD uses the AdaGrad with momentum method to reproduce the results of (Liu & Wang, 2016), which uses remember rate 0.9 and step size 0.03. For GFSF, the small diagonal matrix is $(1e-5)I$.

E.3: DETAILED SETTINGS AND PARAMETERS OF THE BNN EXPERIMENT

We follow the same settings as Liu & Wang (2016). For each run, a random 90%-10% train-test split is conducted. The BNN model contains one hidden layer with 50 hidden nodes, and sigmoid activation is used. The parameters of the Gamma prior on the precision parameter of the Gaussian

Table 2. Parameters of various methods in the BLR experiment

	WGD	PO
SVGD	3e-2	(1.0, 0.7, 1e-7), 3e-6
Blob	1e-6	(1.0, 0.7, 1e-7), 3e-7
GFSD	1e-6	(1.0, 0.7, 1e-7), 3e-7
GFSF	1e-6	(1.0, 0.7, 1e-7), 3e-7
	WAG	**WNes**
SVGD	3.9, (0.9, 1e-6)	(300, 0.2), (0.8, 3e-4)
Blob	3.9, (0.9, 1e-6)	(1000, 0.2), (0.9, 1e-5)
GFSD	3.9, (0.9, 1e-6)	(1000, 0.2), (0.9, 1e-5)
GFSF	3.9, (0.9, 1e-6)	(1000, 0.2), (0.9, 1e-5)

Table 3. Parameters of various methods in the BNN experiment

	WGD	PO
SVGD	1e-3	(1.0, 0.6, 1e-7), 1e-4
Blob	(0.5, 3e-5)	(1.0, 0.8, 1e-7), (0.5, 3e-5)
GFSD	(0.5, 3e-5)	(1.0, 0.8, 1e-7), (0.5, 3e-5)
GFSF	(0.5, 3e-5)	(1.0, 0.8, 1e-7), (0.5, 3e-5)
	WAG	**WNes**
SVGD	3.6, 1e-6	(1000, 0.2), 1e-4
Blob	3.5, (0.5, 1e-5)	(3000, 0.2), (0.6, 1e-4)
GFSD	3.5, (0.5, 1e-5)	(3000, 0.2), (0.6, 1e-4)
GFSF	3.5, (0.5, 1e-5)	(3000, 0.2), (0.6, 1e-4)

prior of the weights are $a_0 = 1.0$, $b_0 = 0.1$. Batch size is set to 100. Number of particles is fixed as 20 for all methods. Results are collected after 8,000 iterations for every method.

Detailed parameters of various methods are provided in Table 3. The format of each column is the same as illustrated in Appendix E.2, except that all SVGD methods uses the AdaGrad with momentum method with remember rate 0.9, so we only provide the step size. WGD and PO methods also adopt the decaying step size, so we provide the decaying exponent. For GFSF, the small diagonal matrix is $(1e-2)I$.

E.4: DETAILED SETTINGS AND PARAMETERS OF THE LDA EXPERIMENT

We follow the same settings as Ding et al. (2014). The dataset is the ICML dataset[5] that contains 765 documents with vocabulary size 1,918. We also adopt the Expanded-Natural parameterization (Patterson & Teh, 2013), and collapsed Gibbs sampling for stochastic gradient estimation. In each document, 90% of words are used for training the topic proportion of the document, and the left 10% words are used for evaluation. For each run, a random 80%-20%

[5] https://cse.buffalo.edu/~changyou/code/ SGNHT.zip

Table 4. Parameters of various methods in the LDA experiment

	WGD	PO
SVGD	3.0	(0.7, 0.7, 1e-4), 10.0
Blob	0.3	(0.7, 0.7, 1e-4), 0.30
GFSD	0.3	(0.7, 0.7, 1e-4), 0.30
GFSF	0.3	(0.7, 0.7, 1e-4), 0.30
	WAG	**WNes**
SVGD	2.5, 3.0	(3.0, 0.2), 10.0
Blob	2.1, 3e-2	(0.3, 0.2), 0.30
GFSD	2.1, 3e-2	(0.3, 0.2), 0.30
GFSF	2.1, 3e-2	(0.3, 0.2), 0.30

(a) SVGD   (b) Blob

(c) GFSD   (d) GFSF

Figure 8. Acceleration effect of WAG and WNes on BLR on the Covertype dataset, measured in log-likelihood. See Appendix E.2 for detailed experiment settings and parameters.

train-test split of the dataset is done.

For the LDA model, we fix the parameter of the Dirichlet prior on topics as 0.1, and mean and standard deviation of the Gaussian prior on the topic proportion as 0.1 and 1.0, respectively. The number of topics is fixed as 30, and batch size is fixed as 100 for all methods. Collapsed Gibbs sampling is run for 50 iterations for each stochastic gradient estimation. Particle size is fixed as 20 for all methods.

Detailed parameters of all methods are provided in Table 4. The format of each column is the same as illustrated in Appendix E.2, except that all methods uses a decaying step size with decaying exponent 0.55 and initial steps 1,000, so we only provide the step size for all methods. SVGD methods do not use the AdaGrad with momentum method. For GFSF, the small diagonal matrix is $(1e\text{-}5)I$.

For SGNHT, both its sequential and parallel simulations use the fixed step size of 0.03, and its mass and diffusion parameters are set to 1.0 and 22.4.

F. More Experimental Results

F.1: MORE RESULTS ON THE BLR EXPERIMENT

The results measured in log-likelihood on test dataset corresponding to the results measured in test accuracy in Fig. 4 is provided in Fig. 8. Acceleration effect of our WAG and WNes methods is again clearly demonstrated, making a consistent support. Particularly, the WNes method is 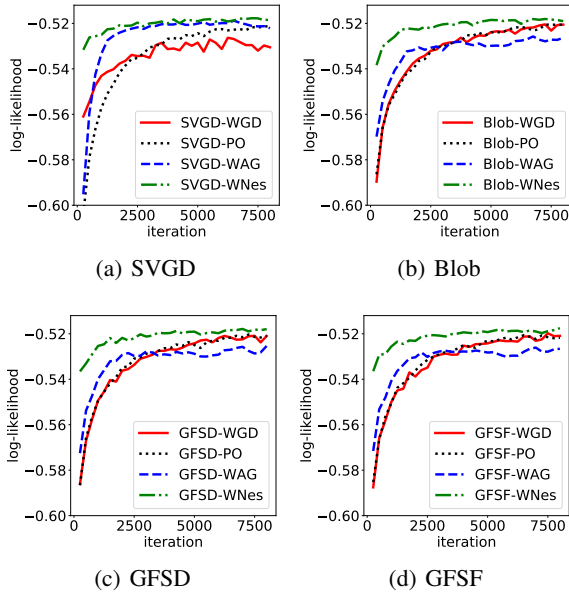more stable than the WAG method.