GAD-PVI : A General Accelerated Dynamic-Weight Particle-based Variational Inference Framework

Fangyikang Wang[1], **Huminhao Zhu**[1], **Chao Zhang**[*2], **Hanbin Zhao**[2], **Hui Qian**[1,3]

[1]College of Computer Science and Technology, Zhejiang University
[2]Advanced Technology Institute, Zhejiang University
[3]State Key Lab of CAD&CG, Zhejiang University
{wangfangyikang, zhuhuminhao, zczju, zhaohanbin, qianhui}@zju.edu.cn

Abstract

Particle-based Variational Inference (ParVI) methods approximate the target distribution by iteratively evolving finite weighted particle systems. Recent advances of ParVI methods reveal the benefits of accelerated position update strategies and dynamic weight adjustment approaches. In this paper, we propose the first ParVI framework that possesses both accelerated position update and dynamical weight adjustment simultaneously, named the General Accelerated Dynamic-Weight Particle-based Variational Inference (GAD-PVI) framework. Generally, GAD-PVI simulates the semi-Hamiltonian gradient flow on a novel Information-Fisher-Rao space, which yields an additional decrease on the local functional dissipation. GAD-PVI is compatible with different dissimilarity functionals and associated smoothing approaches under three information metrics. Experiments on both synthetic and real-world data demonstrate the faster convergence and reduced approximation error of GAD-PVI methods over the state-of-the-art.

Introduction

Particle-based Variational Inference (ParVI) methods have gained significant attention in the Bayesian inference literature owing to their effectiveness in providing approximations of the target posterior distribution π (Liu and Wang 2016; Zhu, Liu, and Zhu 2020; Shen, Heinonen, and Kaski 2021; Zhang et al. 2022; Li et al. 2023). The essence of ParVI lies in deterministically evolving a system of finite weighted particles by simulating the probability space gradient flow of certain dissimilarity functional $\mathcal{F}(\mu) := \mathcal{D}(\mu|\pi)$ vanishing at $\mu = \pi$ (Liu et al. 2019). Since the seminal work Stein Variational Gradient Descent (SVGD) (Liu and Wang 2016), classical ParVI focus on simulating the *first-order* gradient flow in *Wasserstein* space. By using different dissimilarity and associated smoothing approaches, various effective ParVI methods have been proposed, including the BLOB method(Chen et al. 2018a), the GFSD method(Liu et al. 2019), and the KSDD method(Korba et al. 2021).

Methods \ Features	Accelerated position update	Dynamic weight adjustment	Dissimilarity and associated smoothing approach	Underlying probability space
SVGD[1] (Liu and Wang 2016)	✗	✗	KL-RKHS	Wasserstein
BLOB (Craig and Bertozzi 2016)	✗	✗	KL-BLOB	Wasserstein
KSDD (Korba et al. 2021)	✗	✗	KSD-KSDD	Wasserstein
ACCEL (Taghvaei and Mehta 2019)	✓	✗	KL-GFSD	Wasserstein
WNES,WAG (Liu et al. 2019)	✓	✗	General	Wasserstein
AIG (Wang and Li 2022)	✓	✗	KL-GFSD	Information (General)
DPVI (Zhang et al. 2022)	✗	✓	General	Wasserstein-Fisher-Rao
GAD-PVI (Ours)	✓	✓	General	Information-Fisher-Rao (General)

Table 1: Feature-by-Feature comparison of different ParVIs.

To improve the efficiency of ParVIs, recent works explore different aspects of the underlying geometry structures in the probability space and design two types of refined particle systems with either *accelerated position update* or *dynamic weight adjustment*.

- *Accelerated position update.* By considering the second-order Riemannian information of the Wasserstein probability space, different accelerated position update strategies have been proposed(Liu et al. 2019; Taghvaei and Mehta 2019): Liu et al. (2019) follows the accelerated gradient descend methods in the Wasserstein propability space (Liu et al. 2017; Zhang and Sra 2018) and derives the WNES and WAG methods, which update the particles' positions with an extra momentum; the ACCEL method(Taghvaei and Mehta 2019) directly discretizes the Hamiltonian gradient flow in the Wasserstein space and update the position with the damped velocity field, which effectively decrease the Hamiltonian potential of the particle system. Later, Wang and Li (2022) consider the Hamiltonian gradient flow for general information probability space(Lafferty 1988), and derive novel accelerated position update strategies according to the Kalman-Wasserstein/Stein Hamiltonian flow. They theoretically show that the Hamiltonian flow usually has a faster convergence to the equilibrium compared with the original first-order counterpart under mild condition. Numerous experimental studies demonstrate that these accelerated position update strategies usually drift the particle system to the target distribution more efficiently(Liu et al. 2019; Taghvaei and Mehta 2019; Carrillo, Choi, and Tse 2019; Wang and Li 2022).
- *Dynamic weight adjustment.* Delving into the orthogonality structure of the Wasserstein-Fisher-Rao(WFR)

*Corresponding author.

[1]SVGD methods can be also viewed as using KL dissimilarity without Smoothing Approach under Stein-Metric(Nüsken and Renger 2021).

space, Zhang et al. (2022) develop the first dynamic-weight ParVI (DPVI) methods. Specifically, they derive effective dynamical weight adjustment approaches by mimicing the reaction variational step in a JKO splitting scheme of first-order WFR gradient flow(Gallouët and Monsaingeon 2017; Rotskoff et al. 2019). Compared with the commonly used fixed weight strategy, these dynamical weight adjustment schemes usaully lead to less approximation error, especially when the number of particles is limited(Zhang et al. 2022).

Contribution: In this paper, we propose the first ParVI methods which possess both accelerated position update and dynamical weight adjustment simultaneously. Specifically, we first construct a novel Information-Fisher-Rao (IFR) probability space, which augment the origianl information space with an orthogonal Fisher-Rao structure. Then, we originate a novel Semi-Hamiltonian IFR (SHIFR) flow in this space, which simplifies the influence of the kinetic energy on the velocity feild in the Hamiltonian IFR flow [2]. By discretizing the SHIFR flow, a practical General Accelerated Dynamic-weight Particle-based Variational Inference (GAD-PVI) framework is proposed. The main contribution of our paper are listed as follows:

- We investigate the convergence property of the SHIFR flow and show that the target distribution π is the stationary distribution of the proposed semi-Hamiltonian flow for proper dissimilarity functional $\mathcal{D}(\cdot|\pi)$. Moreover, our theoretical result also shows that the augmented Fisher-Rao structure yields an additional decrease on the local functional dissipation, compared to the Hamiltonian flow in the vanilla information space.
- We derive an effective finite-particle approximation to the SHIFR flow, which directly evolves the position, weight, and velocity of the particles via a set of ordinary differential equations. The finite particle system is compatible with different dissimilarity and associated smoothing approaches. We prove that the mean-field limit of the proposed particle system converges to the exact SHIFR flow under mild condition.
- By adopting explicit Euler discretization to the finite-particle system, we architect the General Accelerated Dynamic-weight Particle-based Variational Inference (GAD-PVI) framework, which update positions in an acceleration manner and dynamically adjust weights. We derive nine GAD-PVI instances by using three different dissimilarity functionals and associated smoothing approaches(KL-BLOB, KL-GFSD and KSD-KSDD) on the Wasserstein/Kalman-Wasserstein/Stein IFR space, respectively.

We evaluate our algorithms on various synthetic and real-world tasks. The empirical results demonstrate the superiority of our GAD-PVI methods.

[2]Though the Hamiltonian IFR flow seems a natural choice, it is generally infeasible to obtain practical algorithm by discretizing this flow. Please check the Appendix A.2 for a detailed discussion of IFR Hamiltonian flow.

Notation. Given a probability measure μ on $\mathbb{R}^d$, we denote $\mu \in \mathcal{P}_2(\mathbb{R}^d)$ if its second moment is finite. For a given functional $\mathcal{F}(\cdot) : \mathcal{P}_2(\mathbb{R}^d) \to \mathbb{R}$, $\frac{\delta \mathcal{F}(\tilde{\mu})}{\delta \mu}(\cdot) : \mathbb{R}^d \to \mathbb{R}$ denote its first variation at $\mu = \tilde{\mu}$. We use $C(\mathbb{R}^n)$ to denote the set of continuous functions map from $\mathbb{R}^n$ to $\mathbb{R}$. We denote $\mathbf{x}^i \in \mathbb{R}^d$ as the i-th particle, for $i \in \{1...M\}$. We denote the Dirac delta distribution with point mass located at $\mathbf{x}^i$ as $\delta_{\mathbf{x}^i}$, and use $f * g : \mathbb{R}^d \to \mathbb{R}$ to denote the convolution operation between $f : \mathbb{R}^d \to \mathbb{R}$ and $g : \mathbb{R}^d \to \mathbb{R}$. Besides, we use ∇ and $\nabla \cdot ()$ to denote the gradient and the divergence operator, respectively. We denote a general information probability space as $(\mathcal{P}(\mathbb{R}^n), G(\mu))$, where $G(\mu)[\cdot]$ denotes the one-to-one information metric tensor mapping elements in the tangent space $T_\mu \mathcal{P}(\mathbb{R}^n) \subset C(\mathbb{R}^n)$ to the cotangent space $T_\mu^* \mathcal{P}(\mathbb{R}^n) \subset C(\mathbb{R}^n)$. The inverse map of $G(\mu)[\cdot]$ is denoted as $G^{-1}(\mu)[\cdot] : T_\mu^* \mathcal{P}(\mathbb{R}^n) \to T_\mu \mathcal{P}(\mathbb{R}^n)$.

Preliminaries

When dealing with Bayesian inference tasks, variational inference methods approximate the target posterior π with an easy-to-sample distribution μ, and recast the inference task as an optimization problem over $\mathcal{P}_2(\mathbb{R}^d)$ (Ranganath, Gerrish, and Blei 2014):

$$\min_{\mu \in \mathcal{P}_2(\mathbb{R}^n)} \mathcal{F}(\mu) := \mathcal{D}(\mu|\pi). \tag{1}$$

To solve this optimization problem, Particle-based Variational Inference (ParVI) methods generally simulate the gradient flow of $\mathcal{F}(\mu)$ in certain probability space with a finite particle system, which transport the initial empirical distribution towards the target distribution π iteratively. Given an information metric tensor $G(\mu)[\cdot]$, the gradient flow in the information probability space $(\mathcal{P}(\mathbb{R}^n), G(\mu))$ takes the following form(Ambrosio, Gigli, and Savaré 2008):

$$\partial_t \mu_t = -G(\mu_t)^{-1} \left[\frac{\delta \mathcal{F}(\mu_t)}{\delta \mu} \right]. \tag{2}$$

Wasserstein Gradient Flow and Classical ParVIs

Since the seminal work Stein Variational Gradient Descent (SVGD)(Liu and Wang 2016), many existing ParVI methods focus on flows in the Wasserstein space, where the inverse of the Wasserstein metric tensor writes

$$G^W(\mu)^{-1}[\Phi] = -\nabla \cdot (\mu \nabla \Phi), \Phi \in T_\mu^* \mathcal{P}(\mathbb{R}^n), \tag{3}$$

and the Wasserstein gradient flow is defined as

$$\partial_t \mu_t = \nabla \cdot (\mu_t \nabla \frac{\delta \mathcal{F}(\mu_t)}{\delta \mu}). \tag{4}$$

Based on the probability flow (4) on the density, existing ParVIs maintain a set of particles $\mathbf{x}_t^i$ and directly modify the particle position according to the following ordinary differential equation

$$\mathrm{d}\mathbf{x}_t^i = \nabla \frac{\delta \mathcal{F}(\tilde{\mu}_t)}{\delta \mu}(\mathbf{x}_t^i)\mathrm{d}t, \tag{5}$$

where $\tilde{\mu}_t = \sum_{i=1}^M w_t^i \delta_{\mathbf{x}_t^i}$ denotes the empirical distribution. Since the first total variation $\frac{\delta \mathcal{F}(\tilde{\mu}_t)}{\delta \mu}$ of $\mathcal{F}$ might be not well-defined for the discrete empirical distribution, various ParVI methods have proposed by choosing different dissimilarity $\mathcal{F}$ and associated smoothing approaches for $\frac{\delta \mathcal{F}(\tilde{\mu}_t)}{\delta \mu}$, e.g., KL-BLOB(Chen et al. 2018a), KL-GFSD(Liu et al. 2019), and KSD-KSDD(Korba et al. 2021).

Hamiltonian Gradient Flows and Accelerated ParVIs

The following Hamiltonian gradient flow in the general information probability space has recently been utilized to derive more efficient ParVI methods

$$\begin{cases} \partial_t \mu_t = \dfrac{\delta}{\delta \Phi} \mathcal{H}(\mu_t, \Phi_t), \\ \partial_t \Phi_t = -\gamma_t \Phi_t - \dfrac{\delta}{\delta \mu} \mathcal{H}(\mu_t, \Phi_t), \end{cases} \tag{6}$$

where Φ_t denote the Hamiltonian velocity and $\mathcal{H}(\mu_t, \Phi_t) = \frac{1}{2} \int \Phi_t G(\mu_t)^{-1} [\Phi_t] \, dx + \mathcal{F}(\mu_t)$ denotes the Hamiltonian potential. Note that the Hamiltonian flow (6) can be regarded as the second-order accelerated version of the information gradient flow (2), and usually converges faster to the equilibrium of the target distribution under mild condition(Carrillo, Choi, and Tse 2019; Taghvaei and Mehta 2019; Wang and Li 2022). Though the form of the Hamitonian flow (6) seems complicated, it induces a simple augmented particle system $(\mathbf{x}_t^i, \mathbf{v}_t^i)$, which evolves the position $\mathbf{x}_t^i$ and velocity $\mathbf{v}_t^i$ of particles simultaneously. As the position update rule of $\mathbf{x}_t^i$ also uses the extra velocity information, the induced system is said to have an acclerated position update. By discretizing the continuous particle system, several accelerated ParVI methods have been proposed, which converge faster to the target distribution in numerous real-world Bayesian inference tasks (Taghvaei and Mehta 2019; Wang and Li 2022).

Wasserstein-Fisher-Rao Flow and Dynamic-weight ParVIs

Recently, the Wasserstein-Fisher-Rao (WFR) Flow has been used to derive effective dynamic weight adjustment approaches to mitigate the fixed-weight restriction of ParVIs(Zhang et al. 2022). The inverse of WFR metric tensor is

$$G^{WFR}(\mu)^{-1} [\Phi] = -\nabla \cdot (\mu \nabla \Phi) + (\Phi - \int \Phi) \mu, \tag{7}$$

where $\Phi \in T_\mu^* \mathcal{P}(\mathbb{R}^n)$, and the WFR gradient flow writes:

$$\partial_t \mu_t = \underbrace{\nabla \cdot (\mu_t \nabla \frac{\delta \mathcal{F}(\mu_t)}{\delta \mu})}_{\text{Wasserstein transport}} - \underbrace{(\frac{\delta \mathcal{F}(\mu_t)}{\delta \mu} - \int \frac{\delta \mathcal{F}(\mu_t)}{\delta \mu} \mathrm{d}\mu_t) \mu_t}_{\text{Fisher-Rao variational distortion}}. \tag{8}$$

Since the WFR space can be regarded as orthogonal sum of the Wasserstein space and the Fisher-Rao space, Zhang et al. (2022) mimic a JKO splitting scheme for the WFR flow, which deal with the position and the weight with the Wasserstein transport and the Fisher-Rao variational distortion, respectively. Given a set of particles with position $\mathbf{x}_t^i$ and weight $\mathbf{w}_t^i$, the Fisher-Rao distortion can be approximated by the following ode

$$\frac{\mathrm{d}}{\mathrm{d}t} w_t^i = -\left(\frac{\delta \mathcal{F}(\tilde{\mu}_t)}{\delta \mu}(\mathbf{x}_t^i) - \sum_{i=1}^M w_t^i \frac{\delta \mathcal{F}(\tilde{\mu}_t)}{\delta \mu}(\mathbf{x}_t^i) \right) w_t^i. \tag{9}$$

Accordidng to the ode (9), Zhang et al. (2022) derive two dynamical weighte-adjustment scheme and propose the Dynamic-Weight Particle-Based Variational Inference (DPVI) framework, which is compatible with several dissimilarity functionals and associated smoothing approaches.

Methodology

In this section, we present our General Accelerated Dynamic-weight Particle-based Variational Inference (GAD-PVI) framework, detailted in Algorithm 1. We first introduce a novel augmented Information-Fisher-Rao space, and originate the Semi-Hamiltonian-Information-Fisher-Rao (SHIFR) flow in the space. The theoretical analysis on SHIFR shows that it usually possesses an additional decrease on the local functional dissipation compared to the Hamiltonian flow in the original information space. Then, effective finite-particle systems, which directly evolve the position, weight, and velocity of the particles via a set of ordinary differential equations, are constructed based on SHIFR flows in several IFR spaces with different underlying information metric tensors. We demonstrate that the mean-field limit of the constructed particle system exactly converges to the SHIFR flow in the corresponding probability space. Next, we develop the GAD-PVI framework by discretizing these continuous-time finite-particles formulations, which enables simultaneous accelerated updates of particles' positions and dynamic adjustment of particles' weights. We present nine effective GAD-PVI algorithms that use different underlying information metric tensors, dissimilarity functionals and the associated finite-particle smoothing approaches.

Information-Fisher-Rao Space and Semi-Hamiltonian-Information-Fisher-Rao Flow

To define the augmented Information-Fisher-Rao probability space, we introduce the Information-Fisher-Rao metric tensor $G^{IFR}(\mu)$, whose inverse is defined as follows.

$$G^{IFR}(\mu)^{-1} [\Phi] = G^I(\mu)^{-1} [\Phi] + (\Phi - \int \Phi \mathrm{d}\mu) \mu, \tag{10}$$

where $\Phi \in T_\mu^* \mathcal{P}(\mathbb{R}^n)$ and $G^I(\mu)$ denotes certain underlying information metric tensor. Note that $G^{IFR}(\mu)$ is formed by the inf-convolution of $G^I(\mu)$ and Fisher-Rao metric tensor.

Based on $G^{IFR}(\mu)$, we introduce the following novel semi-Hamiltonian flow of $\mathcal{F}$ on the Information-Fisher-Rao space $(\mathcal{P}(\mathbb{R}^n), G^{IFR}(\mu))$

$$\begin{cases} \partial_t \mu_t = \dfrac{\delta}{\delta \Phi} \mathcal{H}^{IFR}(\mu_t, \Phi_t), \\ \partial_t \Phi_t = -\gamma_t \Phi_t - \dfrac{1}{2} \dfrac{\delta}{\delta \mu} \left(\displaystyle\int \Phi_t G^I(\mu_t)^{-1} [\Phi_t] \, dx \right) - \dfrac{\delta \mathcal{F}(\mu_t)}{\delta \mu}. \end{cases} \tag{11}$$

where Φ_t denote the Hamiltonian velocity and

$$\mathcal{H}^{IFR}(\mu_t, \Phi_t) = \underbrace{\frac{1}{2}\int \Phi_t G^I(\mu_t)^{-1}\left[\Phi_t\right] dx}_{\text{Information kinetic energy}} + \underbrace{\frac{1}{2}\int \Phi_t(\Phi_t - \int \Phi d\mu_t)d\mu_t}_{\text{Fisher-Rao kinetic energy}} + \underbrace{\frac{\delta \mathcal{F}(\mu_t)}{\delta \mu}}_{\text{potential energy}},$$

denotes the Hamiltonian potential in the IFR space. Compared to the full Hamiltonian flow of $\mathcal{F}$ in the IFR space, the SHIFR flow (11) ignores the influence of the Fisher-Rao kinetic energy on the Hamiltonian field Φ_t. Later, we will show that SHIFR can be directly transformed into a particle system consisting of odes on the positions, velocities and weights of particles for proper underlying information metric tensor, while it is generally infeasible to obtain such a direct particle system according to the corresponding full Hamiltonian flow because it is difficult to handle the Fisher-Rao kinetic energy. As the kenetic engery term vanishes when near the equilibrium of the flow, therefore it is acceptable for the SHIFR flow to neglect this intractable term and still has the target distribution π as its stationary distribution. Morever, this semi-Hamiltonian flow would converge faster compare to the Hamiltonian flow in the original information space on account of extra local descending property. Due to the limit of space, we defer the discussion of the stationary analysis and functional dissipation quantitative analysis of the SHIFR flow to Appendix A.4. Please refer to Proposition 2 and Proposition 3 for details.

With different underlying information metric tensor $G^I(\mu)$ in $\mathcal{H}^{IFR}(\mu_t, \Phi_t)$, we can obtain different SHIFR flows. Suitable $G^I(\mu)$ includes the Wasserstein metric tensor,the Kalman-Wasserstein metric tensor (KW-metric) and the Stein metric tensor (S-metric). For instance, the SHIFR flow with Wasserstein metric (Wasserstein-SHIFR flow) writes:

$$\begin{cases} \partial_t \mu_t = -\nabla \cdot (\mu_t \nabla \Phi_t) - \left(\dfrac{\delta \mathcal{F}(\mu_t)}{\delta \mu} - \displaystyle\int \dfrac{\delta \mathcal{F}(\mu_t)}{\delta \mu} \mathrm{d}\mu_t\right)\mu_t, \\ \partial_t \Phi_t = -\gamma_t \Phi_t - \|\nabla \Phi_t\|^2 - \dfrac{\delta \mathcal{F}(\mu_t)}{\delta \mu}. \end{cases} \tag{12}$$

Note that in the subsequent section, we focus on the Wasserstein-SHIFR flow, and defer the detailed formulations with respect to KW-SHIFR and S-SHIFR to the Appendix B.1 and B.2 due to limited space.

Finite-Particles Formulations to SHIFR flows

Now, we derive the finite-particle approximation to the SHIFR flow, which directly evolves the position $\mathbf{x}_t^i$, weight w_t^i, and velocity $\mathbf{v}_t^i$ of the particles. Specifically, we construct the following ordinary differential euqation system to simulate the Wasserstein-SHIFR flow (12)

$$\begin{cases} \mathrm{d}\mathbf{x}_t^i = \mathbf{v}_t^i \mathrm{d}t, \\ \mathrm{d}\mathbf{v}_t^i = (-\gamma \mathbf{v}_t^i - \nabla \dfrac{\delta \mathcal{F}(\tilde{\mu}_t)}{\delta \mu}(\mathbf{x}_t^i))\mathrm{d}t, \\ \mathrm{d}w_t^i = -\left(\frac{\delta \mathcal{F}(\tilde{\mu}_t)}{\delta \mu}(\mathbf{x}_t^i) - \sum_{i=1}^M w_t^i \frac{\delta \mathcal{F}(\tilde{\mu}_t)}{\delta \mu}(\mathbf{x}_t^i)\right) w_t^i \mathrm{d}t, \\ \tilde{\mu}_t = \sum_{i=1}^M w_t^i \delta_{\mathbf{x}_t^i}. \end{cases} \tag{13}$$

The following proposition demonstrates that the mean-field limit of the particle system (13) corresponds precisely to the Wasserstein-SHIFR flow in (12).

Proposition 1. *Suppose the empirical distribution $\tilde{\mu}_0^M$ of M weighted particles weakly converges to a distribution μ_0 when $M \to \infty$. Then, the path of* (13) *starting from $\tilde{\mu}_0^M$ and Φ_0 with initial velocity* **0** *weakly converges to a solution of the Wasserstein-SHIFR gradient flow* (12) *starting from $\mu_t|_{t=0} = \mu_0 =$ and $\Phi_t|_{t=0} = \mathbf{0}$ as $M \to \infty$:*

The particle systems w.r.t. the KW-SHIFR and S-SHIFR flows and their mean-field analysis are provided in the Appendix B.1 and B.2.

Algorithm 1: General Accelerated Dynamic-weight Particle-based Variational Inference (GAD-PVI) framework

Input: Initial distribution $\tilde{\mu}_0 = \sum_{i=1}^M w_0^i \delta_{\mathbf{x}_0^i}$, position adjusting step-size η_{pos}, weight adjusting step-size η_{wei}, velocity field adjusting step-size η_{vel}, velocity damping parameter γ.

1: Choose a suitable functional $\mathcal{F}$ and its smoothing strategy $U_{\tilde{\mu}} \approx \frac{\delta \mathcal{F}(\tilde{\mu})}{\delta \mu}$ from KL-BLOB/KL-GFSD/KSD-KSDD
2: **for** $k = 0, 1, ..., T-1$ **do**
3: **for** $i = 1, 2, ..., M$ **do**
4: Update positions $\mathbf{x}_{k+1}^i$'s according to (15).
5: **end for**
6: **for** $i = 1, 2, ..., M$ **do**
7: Adjust velocity field $\mathbf{v}_{k+1}^i$'s according to (16).
8: **end for**
9: **for** $i = 1, 2, ..., M$ **do**
10: Adjust weights w_{k+1}^i's according to (17).
11: **end for**
12: **end for**
13: **Output**:$\tilde{\mu}_T = \sum_{i=1}^M w_T^i \delta_{\mathbf{x}_T^i}$.

GAD-PVI Framework

Generally, it is impossible to obtain an analytic solution of the continuous finite-particles formulations (13), thus a numerical integration method is required to derive an approximate solution. Note that any numerical solver, such as the implicit Euler method (Platen and Bruti-Liberati 2010) and higher-order Runge-Kutta method (Butcher 1964) can be used. Here, we follow the tradition of ParVIs to adopt the first-order explicit Euler discretization (Süli and Mayers 2003) since it is efficient and easy-to-implement

(Zhang et al. 2022), and propose our General Accelerated Dynamic-weight Particle-based Variational Inference (GAD-PVI) framework, as listed in Algorithm 1.

Dissimilarity Functionals and Smoothing Approaches To develop practical GAD-PVI methods, we must first select a dissimilarity functional $\mathcal{F}$. The commonly used underlying functionals are KL-divergence (Liu and Wang 2016; Liu et al. 2019; Wang and Li 2022) and Kernel-Stein-Discrepancy (Korba et al. 2020). Once a dissimilarity functional $\mathcal{F}$ has been chosen, we need to select a smoothing approach to approximate the first variation of the empirical approximation, as the value of $\frac{\delta\mathcal{F}(\cdot)}{\mu}$ at an empirical distribution $\tilde{\mu} = \sum_{i=1}^{M} w^i \delta_{\mathbf{x}^i}$ is generally not well-defined. Smoothing strategies allow us to approximate the first variation value at the discrete empirical distribution. Generally, a smoothed approximation to the first total variation is denoted as $U_{\tilde{\mu}}(\cdot) \approx \frac{\delta\mathcal{F}(\tilde{\mu})}{\delta\mu}(\cdot)$. The commonly used smoothing approaches in the ParVI area, namely BLOB (with KL-divergence as $\mathcal{F}$) (Craig and Bertozzi 2016), GFSD (with KL-divergence as $\mathcal{F}$) (Liu et al. 2019), and KSDD (with Kernel Stein Discrepancy as $\mathcal{F}$) (Korba et al. 2021), are all compatible with our GAD-PVI framework.

Here, we describe the dissimilarity functional KL-divergence and the associated BLOB smoothing approach as an example. The first total variation of the KL-divergence is

$$\frac{\delta\mathcal{F}(\mu)}{\delta\mu}(\cdot) := \frac{\delta KL(\mu|\pi)}{\delta\mu}(\cdot) = -\log\pi(\cdot) + \log\mu(\cdot).$$

As $\log\mu(\mathbf{x})$ is ill-defined for the discrete empirical distribution $\tilde{\mu}_k$, BLOB smoothing approach reformulate the intractable term $\log\mu$ as $\frac{\delta}{\delta\mu}\mathbb{E}_\mu[\log\mu]$ and smooth the density with a kernel function K, resulting the following approximation

$$\begin{aligned}\log\tilde{\mu} &\approx \frac{\delta}{\delta\tilde{\mu}}\mathbb{E}_{\tilde{\mu}}\left[\log\left(\tilde{\mu} * K\right)\right] \\ &:= \log\textstyle\sum_{i=1}^{M} w^i K(\cdot,\mathbf{x}^i) + \sum_{i=1}^{M} \frac{w^i K(\cdot,\mathbf{x}^i)}{\sum_{j=1}^{M} w^j K(\mathbf{x}^i,\mathbf{x}^j)}.\end{aligned}$$

for a discrete density $\tilde{\mu} = \sum_{i=1}^{M} w^i \mathbf{x}^i$. This leads to the following approximation results:

$$\begin{aligned}U_{\tilde{\mu}_k}(\mathbf{x}) = &-\log\pi(\mathbf{x}) + \log\textstyle\sum_{i=1}^{M} w_k^i K(\mathbf{x},\mathbf{x}_k^i) \\ &+ \sum_{i=1}^{M} \frac{w_k^i K(\mathbf{x},\mathbf{x}_k^i)}{\sum_{j=1}^{M} w_k^j K(\mathbf{x}_k^i,\mathbf{x}_k^j)}.\end{aligned} \tag{14}$$

Details regarding other dissimilarity functionals and smoothing approaches are included in the Appendix B.3.

Updating rules Once the functional $\mathcal{F}$ and its empirical approximation of the first variation $U_{\tilde{\mu}} \approx \frac{\delta\mathcal{F}(\tilde{\mu})}{\delta\mu}$ is decided, we adopt a Jacobi-type strategy to update the position $\mathbf{x}_k^i$, velocity field $\mathbf{v}_k^i$ and the weight w_k^i, i.e., the calculations in the $k+1$-th iteration are totally based on the variables obtained in the k-th iteration. Therefore, starting from M weighted particles located at $\{\mathbf{x}_0^i\}_{i=1}^M$ with weights $\{w_0^i\}_{i=0}^M$ and $\{\mathbf{v}_0^i = 0\}_{i=0}^M$, GAD-PVI w.r.t. the Wasserstein-SHIFR flow first updates the positions of particles according to the following rule:

$$\mathbf{x}_{k+1}^i = \mathbf{x}_k^i + \eta_{pos}\mathbf{v}_k^i. \tag{15}$$

Then, it adjusts the velocity field as

$$\mathbf{v}_{k+1}^i = (1-\gamma\eta_{vel})\mathbf{v}_k^i - \eta_{vel}\nabla U_{\tilde{\mu}_k}(\mathbf{x}_k^i), \tag{16}$$

and particles' weights as following:

$$w_{k+1}^i = w_k^i - \eta_{wei}(U_{\tilde{\mu}_k}(\mathbf{x}_k^i) - \sum_{j=1}^{M} w_k^j U_{\tilde{\mu}_k}(\mathbf{x}_k^j)). \tag{17}$$

Here $\tilde{\mu}_k = \sum_{i=1}^{M} w_k^i \delta_{\mathbf{x}_k^i}$ denotes the empirical distribution, and $\eta_{pos}/\eta_{vel}/\eta_{wei}$ are the discretization stepsizes. It can be verified that the total mass of $\tilde{\mu}_k$ is conserved and $\tilde{\mu}_k$ remains a valid probability distribution during the whole procedure of GAD-PVI, i.e. $\sum_i w_k^i = 1$ for all k. The detailed updating rules of GAD-PVI w.r.t. the KW-SHIFR and S-SHIFR can be found in Appendix B.3.

Notice that, compared to the classical ParVIs, the position acceleration scheme and dynamic-weight scheme only bring *little* extra computational cost, because the number of time-complexity-bottleneck operation, i.e. calculation of $U_{\tilde{\mu}}$ and $\nabla U_{\tilde{\mu}}$, remains the same.

An alternative Weight Adjusting Approach. Except for Continuous Adjusting (CA) strategy, the Duplicate/Kill (DK) strategy, which is a probabilistic discretization strategy to the Fisher-Rao part of (12), can also be adopt in GAD-PVI. This strategy duplicates/kills particle $\mathbf{x}_{k+1}^i$ according to an exponential clock with instantaneous rate:

$$R_{k+1}^i = -\eta_{wei}(\frac{\delta\mathcal{F}(\tilde{\mu}_k)}{\delta\mu}(\mathbf{x}_k^i) - \sum_{j=1}^{M} w_k^j \frac{\delta\mathcal{F}(\tilde{\mu_k})}{\delta\mu}(\mathbf{x}_k^j)). \tag{18}$$

Specifically, if $R_{k+1}^i > 0$, duplicate the particle $\mathbf{x}_{k+1}^i$ with probability $1-\exp(-R_{k+1}^i)$, and kill another one with uniform probability to conserve the total mass; if $R_{k+1}^i < 0$, kill the particle $\mathbf{x}_{k+1}^i$ with probability $1-\exp(R_{k+1}^i)$, and duplicate another one with uniform probability. By replacing the CA strategy (17) in the GAD-PVI framework, we could obtain the DK variants of GAD-PVI methods.

GAD-PVI instances. With different underlying information metric tensors (W-metric, KW-metic and S-metric), weight adjustment approaches(CA and DK) and dissimilarity functionals/associated smoothing approaches(KL-BLOB, KL-GFSD and KSD-KSDD), we can derive 18 different intances of GAD-PVI, named as WGAD/KWGAD/SGAD-CA/DK-BLOB/GFSD/KSDD.

Experiments

In this section, we conduct empirical studies with our GAD-PVI algorithms. Here, we focus on the instances of

GAD-PVI w.r.t. the W-SHIFR flows, i.e.,WGAD-CA/DK-BLOB/GFSD. The experimental results on methods w.r.t. the KW-SHIFR and S-SHIFR flows are provided in the Appendix C. Note that we do not include GAD-PVI methods with the KSDD smoothing approaches, as they are more computationally expensive and have been widely reported to be less stable(Korba et al. 2020; Zhang et al. 2022). We include four classes of methods as our baseline: classical ParVI algorithms (SVGD, GFSD and BLOB), the Nesterov accelerated ParVI algorithms (WNES-BLOB/GFSD), the Hamiltonian accelerated ParVI algorithms (WAIG-BLOB/GFSD) and the Dynamic-weight ParVI algorithms (DPVI-CA/DK-BLOB/GFSD).

We compare the performance of these algorithms on two simulations, i.e., a 10-D Single-mode Gaussian model (SG) and a Gaussian mixture model (GMM), and two real-world applications, i.e. Gaussian Process (GP) regression and Bayesian neural network (BNN). For all the algorithms, the particles' weights are initialized to be equal. In the first three experiments, we tune the parameters to achieve the best W_2 distance. In the BNN task, we split $1/5$ of the training set as our validation set to tune the parameters. Note that, the position step-size are tuned via grid search for the fixed-weight ParVI algorithms, then used in the corresponding dynamic-weight algorithms. The acceleration parameters and weight adjustment parameters are tuned via grid search for each specific algorithm. We repeat all the experiments 10 times and report the average results. Due to limited space, only parts of the results are reported in this section. We refer readers to the Appendix C for the results on SG and additional results for GMM, GP and BNN.

Gaussian Mixture Model

We consider approximating a 10-D Gaussian mixture model with two components, weighted by $1/3$ and $2/3$ respectively. We run all algorithms with particle number $M \in \{32, 64, 128, 256, 512\}$.

In Figure 1, we report the 2-Wasserstein (W_2) distance between the empirical distribution generated by each algorithm and the target distribution w.r.t. iterations of different ParVI methods. We generate 5,000 samples from the target distribution π as reference to evaluate the W_2 distance by using the POT library [3]. The results demonstrate that our GAD-PVI algorithms consistently outperform their counterpart with only one (or none) of the accelerated position update strategy and dynamic weight adjustment approach. Besides, the CA weight-adjustment approach usually result a lower W_2 compared to the DK scheme, and WGAD-CA-BLOB/GFSD usually have the fastest convergence and the lowest final W_2 distance to the target.

Gaussian Process Regression

The Gaussian Process (GP) model is widely adopted for the uncertainty quantification in regression problems (Rasmussen 2003). We follow the experiment setting in (Chen et al. 2018b), and use the dataset LIDAR which consists of

[3]http://jmlr.org/papers/v22/20-451.html

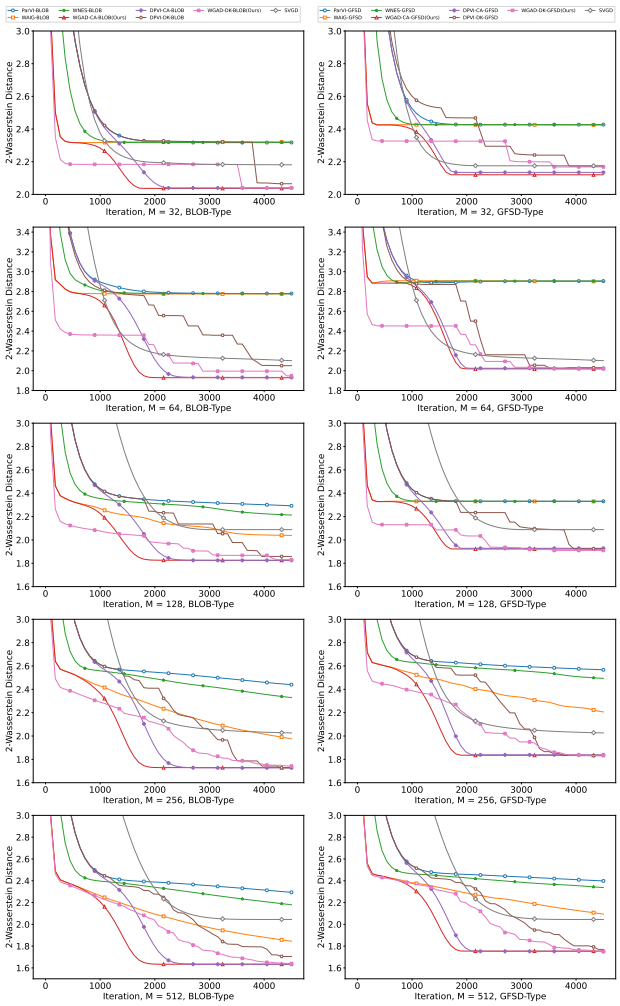

Figure 1: W_2 distance to the target w.r.t. iterations in the Gaussian Mixture Model task.

221 observations. In this task, we set the particle number to $M = 128$ for all the algorithms.

We report the W_2 distance between the empirical distribution after 10000 iterations and the target distribution in Table 2. The target distribution is approximated by 10000 reference particles generated by the HMC method after it achieves its equilibrium(Brooks et al. 2011). It can be observed that both the accelerated position update and the dynamic weight adjustment result in a decreased W_2 and GAD-PVI algorithms consistently achieve loweset W_2 to the target. Besides, the results also show that the CA variants usually outperforms their DK counterpart, as CA is able to adjust the weight continuously on $[0, 1]$ while DK set the weight either to 0 or $1/M$.

In Figure 2, we plot the contour lines of the log posterior and the particles generated by four representative algorithms, namely BLOB, WAIG-BLOB, DPVI-CA-BLOB, and WGAD-CA-BLOB, at different iterations (0, 100, 500, 2000, 10000). The results indicate that the particles in WAIG-BLOB and WGAD-CA-BLOB exhibit a faster convergence to the high probability area of the target due to

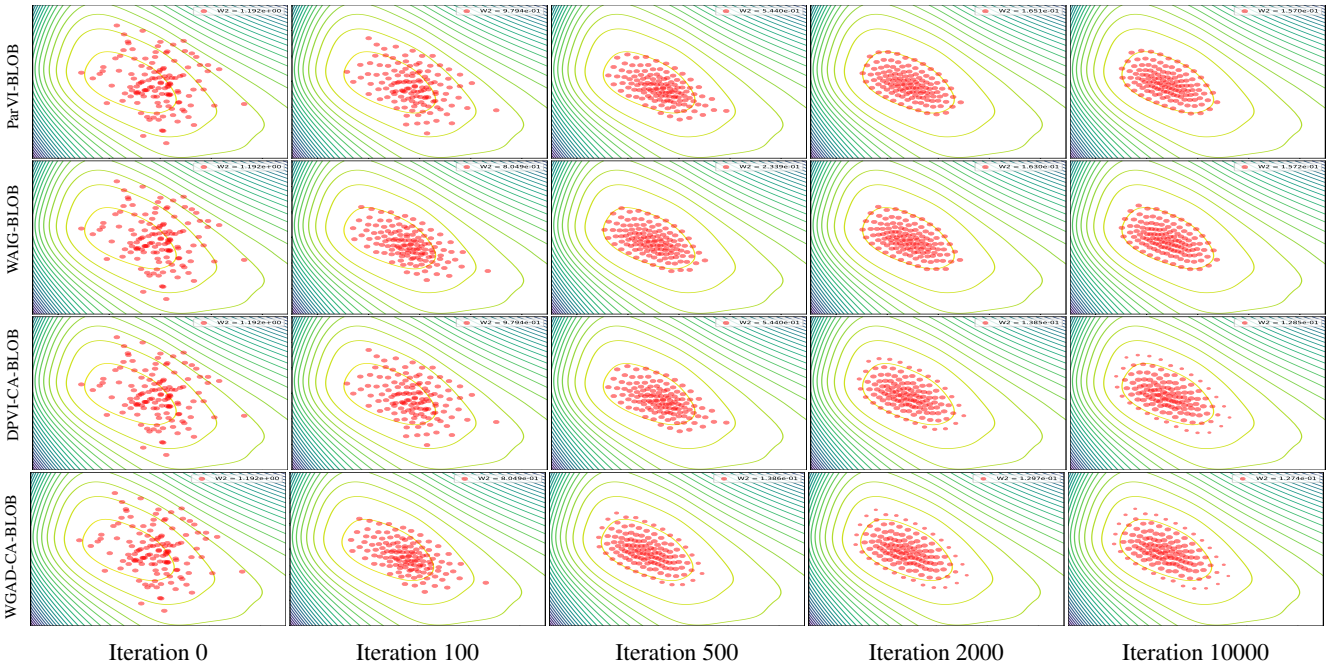

Figure 2: The contour lines of the log posterior in the Gaussian Process task.

Algorithm	Smoothing Strategy	
	BLOB	GFSD
ParVI	1.570e-01 ± 2.210e-04	2.143e-01 ± 7.424e-04
WAIG	1.572e-01 ± 2.070e-04	2.142e-01 ± 7.048e-04
WNES	1.571e-01 ± 3.011e-04	2.138e-01 ± 7.771e-04
DPVI-DK	1.568e-01 ± 1.496e-03	2.142e-01 ± 2.712e-03
DPVI-CA	1.285e-01 ± 2.960e-04	1.638e-01 ± 4.332e-04
WGAD-DK(Ours)	1.561e-01 ± 1.155e-03	2.142e-01 ± 1.501e-03
WGAD-CA(Ours)	**1.274e-01 ± 2.964e-04**	**1.626e-01 ± 4.842e-04**

Table 2: Averaged W_2 for the GP task with dataset LIDAR.

algorithms	Datasets			
	Concrete	kin8nm	RedWine	space
ParVI-SVGD	6.323ee+00	8.020e-02	6.330e-01	9.021e-02
ParVI-BLOB	6.313e+00	7.891e-02	6.318e-01	8.943e-02
WAIG-BLOB	6.063e+00	7.791e-02	6.267e-01	8.775e-02
WNES-BLOB	6.112e+00	7.690e-02	6.264e-01	8.836e-02
DPVI-DK-BLOB	6.285e+00	7.889e-02	6.294e-01	8.853e-02
DPVI-CA-BLOB	6.292e+00	7.789e-02	6.298e-01	8.850e-02
WGAD-DK-BLOB(Ours)	6.058e+00	7.688e-02	6.267e-01	8.716e-02
WGAD-CA-BLOB(Ours)	**6.047e+00**	**7.629e-02**	**6.263e-01**	**8.704e-02**
ParVI-GFSD	6.314e+00	7.891e-02	6.317e-01	8.943e-02
WAIG-GFSD	6.105e+00	7.794e-02	6.265e-01	8.776e-02
WNES-GFSD	6.123e+00	7.756e-02	6.263e-01	8.836e-02
DPVI-DK-GFSD	6.291e+00	7.882e-02	6.277e-01	8.851e-02
DPVI-CA-GFSD	6.290e+00	7.791e-02	6.298e-01	8.852e-02
WGAD-DK-GFSD(Ours)	6.099e+00	7.726e-02	6.265e-01	**8.708e-02**
WGAD-CA-GFSD(Ours)	**6.088e+00**	**7.634e-02**	**6.260e-01**	8.710e-02

Table 3: Averaged Test $RMSE$ in the BNN task.

their accelerated position updating strategy, and the DPVI-CA and WGAD-CA algorithms finally offer a broader final coverage, as the CA dynamic weight adjustment strategy enables the particles to represent the region with arbitrary local density mass instead of a fixed $1/M$ mass.

Bayesian Neural Network

In this experiment, we study a Bayesian regression task with Bayesian neural network on 4 datasets from UCI and LIBSVM. We follow the experiment setting from (Liu and Wang 2016; Zhang et al. 2022), which models the output as a Gaussian distribution and uses a Gamma$(1, 0.1)$ prior for the inverse covariance. We use a one-hidden-layer neural network with 50 hidden units and maintain 128 particles. For all the datasets, we set the batchsize as 128.

We present the Root Mean Squared Error (RMSE) of various ParVI algorithms in Table 3. The results demonstrate that the combination of the accelerated position updating strategy and the dynamically weighted adjustment leads to a lower RMSE. Notably, WGAD-CA type algrithms outperform other methods in the majority of cases.

Conclusion

In this paper, we propose the General Accelerated Dynamic-Weight Particle-based Variational Inference (GAD-PVI) framework, which adopts an accelerated position update scheme and dynamic weight adjustment approach simutaneously. Our GAD-PVI framework is developed by discretizing the Semi-Hamiltonian Information Fisher-Rao (SHIFR) flow on the novel Information-Fisher-Rao space. The theoretical analysis demonstrate that the SHIFR flow yields additional decrease on the local functional dissipation compared to the Hamiltonian flow in the vanilla information space. We propose effective particle system which evolve the position, weight, velocity of particles via a set of odes for the SHIFR flows with different underlying information metrics. By directly discretizing the propsed particle system, we obatain our GAD-PVI framework. Several effective instances of the GAD-PVI framework have been provided by employing three distinct dissimilarity functionals and associated smoothing approaches under the Wasserstein/Kalman-Wasserstein/Stein metric. Empirical studies demonstrate the faster convergence and reduced approximation error of GAD-PVI methods over the SOTAs.

References

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

Brooks, S.; Gelman, A.; Jones, G.; and Meng, X.-L. 2011. *Handbook of markov chain monte carlo*. CRC press.

Butcher, J. C. 1964. Implicit runge-kutta processes. *Mathematics of Computation*, 18(85): 50–64.

Carrillo, J. A.; Choi, Y.-P.; and Tse, O. 2019. Convergence to equilibrium in Wasserstein distance for damped Euler equations with interaction forces. *Communications in Mathematical Physics*, 365: 329–361.

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

Chen, W. Y.; Mackey, L.; Gorham, J.; Briol, F.-X.; and Oates, C. 2018b. Stein points. In *ICML*, 844–853. PMLR.

Craig, K.; and Bertozzi, A. 2016. A blob method for the aggregation equation. *Mathematics of computation*, 85(300): 1681–1717.

Gallouët, T. O.; and Monsaingeon, L. 2017. A JKO Splitting Scheme for Kantorovich–Fisher–Rao Gradient Flows. *SIAM Journal on Mathematical Analysis*, 49(2): 1100–1130.

Garbuno-Inigo, A.; Hoffmann, F.; Li, W.; and Stuart, A. M. 2020. Interacting Langevin diffusions: Gradient structure and ensemble Kalman sampler. *SIAM Journal on Applied Dynamical Systems*, 19(1): 412–441.

Korba, A.; Aubin-Frankowski, P.-C.; Majewski, S.; and Ablin, P. 2021. Kernel Stein Discrepancy Descent. In Meila, M.; and Zhang, T., eds., *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, 5719–5730. PMLR.

Korba, A.; Salim, A.; Arbel, M.; Luise, G.; and Gretton, A. 2020. A non-asymptotic analysis for Stein variational gradient descent. *NeurIPS*, 33.

Lafferty, J. D. 1988. The Density Manifold and Configuration Space Quantization. *Transactions of the American Mathematical Society*, 305(2): 699–741.

Li, L.; qiang liu; Korba, A.; Yurochkin, M.; and Solomon, J. 2023. Sampling with Mollified Interaction Energy Descent. In *The Eleventh International Conference on Learning Representations*.

Liu, C.; Zhuo, J.; Cheng, P.; Zhang, R.; and Zhu, J. 2019. Understanding and accelerating particle-based variational inference. In *ICML*, 4082–4092.

Liu, Q.; Lee, J.; and Jordan, M. 2016. A kernelized Stein discrepancy for goodness-of-fit tests. In *ICML*, 276–284. PMLR.

Liu, Q.; and Wang, D. 2016. Stein variational gradient descent: A general purpose bayesian inference algorithm. *arXiv preprint arXiv:1608.04471*.

Liu, Y.; Shang, F.; Cheng, J.; Cheng, H.; and Jiao, L. 2017. Accelerated first-order methods for geodesically convex optimization on Riemannian manifolds. *Advances in Neural Information Processing Systems*, 30.

Lu, Y.; Lu, J.; and Nolen, J. 2019. Accelerating langevin sampling with birth-death. *arXiv preprint arXiv:1905.09863*.

Mroueh, Y.; and Rigotti, M. 2020. Unbalanced Sobolev Descent. *NeurIPS*, 33.

Nüsken, N.; and Renger, D. 2021. Stein Variational Gradient Descent: many-particle and long-time asymptotics. *arXiv preprint arXiv:2102.12956*.

Platen, E.; and Bruti-Liberati, N. 2010. *Numerical solution of stochastic differential equations with jumps in finance*, volume 64. Springer Science & Business Media.

Ranganath, R.; Gerrish, S.; and Blei, D. 2014. Black box variational inference. In *Artificial intelligence and statistics*, 814–822. PMLR.

Rasmussen, C. E. 2003. Gaussian processes in machine learning. In *Summer school on machine learning*, 63–71. Springer.

Rotskoff, G.; Jelassi, S.; Bruna, J.; and Vanden-Eijnden, E. 2019. Global convergence of neuron birth-death dynamics. *arXiv preprint arXiv:1902.01843*.

Santambrogio, F. 2017. {Euclidean, metric, and Wasserstein} gradient flows: an overview. *Bulletin of Mathematical Sciences*, 7(1): 87–154.

Shen, Z.; Heinonen, M.; and Kaski, S. 2021. Derandomizing MCMC dynamics with the diffusion Stein operator. In Ranzato, M.; Beygelzimer, A.; Dauphin, Y.; Liang, P.; and Vaughan, J. W., eds., *Advances in Neural Information Processing Systems*, volume 34, 17507–17517. Curran Associates, Inc.

Süli, E.; and Mayers, D. F. 2003. *An introduction to numerical analysis*. Cambridge university press.

Taghvaei, A.; and Mehta, P. 2019. Accelerated flow for probability distributions. In *International Conference on Machine Learning*, 6076–6085. PMLR.

Von Mises, R.; Geiringer, H.; and Ludford, G. S. S. 2004. *Mathematical theory of compressible fluid flow*. Courier Corporation.

Wang, Y.; and Li, W. 2022. Accelerated Information Gradient Flow. *Journal of Scientific Computing*, 90: 11.

Zhang, C.; Li, Z.; Du, X.; and Qian, H. 2022. DPVI: A Dynamic-Weight Particle-Based Variational Inference Framework. In Raedt, L. D., ed., *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, 4900–4906. International Joint Conferences on Artificial Intelligence Organization. Main Track.

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

Zhang, J.; Zhang, R.; Carin, L.; and Chen, C. 2020. Stochastic particle-optimization sampling and the non-asymptotic convergence theory. In *Artificial Intelligence and Statistics*, 1877–1887. PMLR.

Zhu, M.; Liu, C.; and Zhu, J. 2020. Variance Reduction and Quasi-Newton for Particle-Based Variational Inference. In *ICML*, 11576–11587. PMLR.

Appendix A

A.1 Definition of Information Metric in Probability Space

To make general information gradient flow on probability space defined, we briefly review the definition of information metrics in probability space (Ambrosio, Gigli, and Savaré 2008):

Definition 1. *(Information metric in probability space). Denote the tangent space at $\mu \in \mathcal{P}(\mathbb{R}^n)$ by $T_\mu\mathcal{P}(\mathbb{R}^n) = \{\sigma \in C(\mathbb{R}^n) : \int \sigma dx = 0\}$. The cotangent space at μ is denoted as $T_\mu^*\mathcal{P}(\mathbb{R}^n)$, can be treated as the quotient space $C(\mathbb{R}^n)/\mathbb{R}$. An information metric tensor $G(\mu)[\cdot] : T_\mu\mathcal{P}(\mathbb{R}^n) \to T_\mu^*\mathcal{P}(\mathbb{R}^n)$ is an invertible mapping from $\mathcal{P}(\mathbb{R}^n)$ to $T_\mu^*\mathcal{P}(\mathbb{R}^n)$. This information metric tensor defines the information metric (as well as inner product) on tangent space $\mathcal{P}(\mathbb{R}^n)$, for $\sigma_1, \sigma_2 \in T_\mu\mathcal{P}(\mathbb{R}^n)$ and $\Phi_i = G(\mu)[\sigma_i], i = 1, 2$ as*

$$g_\mu(\sigma_1, \sigma_2) = \int \sigma_1 G(\mu)\sigma_2 dx = \int \Phi_1 G(\mu)^{-1}\Phi_2 dx, \tag{19}$$

then we denote the general information probability space as $(\mathcal{P}(\mathbb{R}^n), G(\mu))$. As long as a metric is specified, the probability space $\mathcal{P}(\mathbb{R}^n)$ together with the metric can be taken as an infinite dimensional Riemannian manifold which is so-called density manifold (Lafferty 1988), which enables the definition of gradient flow.

A.2 Full Hamiltonian Flow on the IFR Space and the Fisher-Rao Kinetic Energy

To develop ParVI methods which possess both accelerated position update and dynamical weight adjustment simultaneously, a natural choice is to directly simulate the Hamiltonian flow on the augmented IFR space By substituting the IFR metric (10) into the general Hamiltonian flow (6), we derive the Full Hamiltonian flow on the IFR space as the direct accelerated probabilistic flow on the IFR space with the form

$$\begin{cases} \partial_t \mu_t = G(\mu_t)^{-1}[\Phi_t] + \left(\Psi_t - \int \Psi_t d\mu_t\right)\mu_t \\ \partial_t \Phi_t + \gamma_t \Phi_t + \underbrace{\frac{1}{2}\frac{\delta}{\delta\mu}\left(\int \Phi_t G(\mu_t)^{-1}[\Phi_t]\, dx\right)}_{\text{Information kinetic energy}} + \underbrace{\frac{1}{2}\Phi_t^2 - \Phi_t \int \Phi_t d\mu_t}_{\text{Fisher-Rao kinetic energy}} + \frac{\delta \mathcal{F}(\mu_t)}{\delta \mu} = 0. \end{cases} \tag{20}$$

where the Φ_t is the velocity field; the $\frac{\delta \mathcal{F}(\mu_t)}{\delta \mu}$ represent the potential energy dissipation; the $\frac{1}{2}\frac{\delta}{\delta\mu}(\int \Phi_t G(\mu_t)^{-1}[\Phi_t]\, dx)$ represent the kinetic energy dissipation of information transport; the $\frac{1}{2}\Phi_t^2 - \Phi_t \int \Phi_t d\mu_t$ represent the kinetic energy dissipation of Fisher-Rao distortion. As far as we know, the particle formulation of the Full Hamiltonian flow on the IFR space (20) is intractable due to the Fisher-Rao kinetic energy term. When deriving particle systems, the $\nabla\Phi_t$ can be straightforwardly approximated by the velocities $\mathbf{v}_t^i$, but the Hamiltonian field Φ_t is hard to be approximated by finite points and iteratively updated. Actually, even the particle formulation of the accelerated Fisher-Rao flow has not been derived due to great difficulty (Wang and Li 2022). Therefore, we ignore the influence of the Fisher-Rao kinetic energy on the Hamiltonian field and derive the SHIFR (11). We point out that the Fisher-Rao kinetic energy would vanish as the flow converge to the equilibrium of $(\mu_\infty = \pi, \Phi_\infty = \mathbf{0})$ which fit the behavior of kinetic energy in physical dynamic system. Therefore, the ignorance of the Fisher-Rao kinetic energy is tenable and the SHIFR still have the target distribution as its stationary distribution which will be shown in Proposition 3.

A.3 Proof of Proposition 1

First we Introduce a technical lemma for the proof of Proposition 1.

Lemma 1. *The following probability flow dynamic formulation and particle system formulation is equivalent:*

$$\begin{cases} \partial_t \mu_t + \nabla \cdot (\mu_t \nabla \Phi_t) = 0 \\ \partial_t \Phi_t + \gamma_t \Phi_t + \frac{1}{2}\left\|\nabla \Phi_t\right\|^2 + \frac{\delta \mathcal{F}(\mu_t)}{\delta \mu} = 0 \end{cases} \tag{21}$$

$$\begin{cases} \frac{d}{dt} X_t = V_t \\ \frac{d}{dt} V_t = -\gamma_t V_t - \nabla\left(\frac{\delta \mathcal{F}(\mu_t)}{\delta \mu}\right)(X_t) \end{cases} \tag{22}$$

Proof. We start with the calculation of gradient of kinetic term. For a twice differentiable $\Phi(x)$, we have:

$$\frac{1}{2}\nabla\left\|\nabla\Phi\right\|^2 = \nabla^2\Phi\nabla\Phi = (\nabla\Phi \cdot \nabla)\nabla\Phi \tag{23}$$

From (21), we have:

$$\partial_t \mu_t + \nabla \cdot (\mu_t \nabla \Phi_t) = 0$$

which is the continuity equation of μ_t under vector field $\nabla \Phi_t$ (Santambrogio 2017). Hence, we have following equation on partical level (denoting $V_t = \nabla \Phi_t(X_t)$):

$$\begin{aligned} \frac{d}{dt} X_t &= \nabla \Phi_t(X_t) \\ &= V_t \end{aligned}$$

Then, the vector field shall follow:

$$\begin{aligned} \frac{d}{dt} V_t &= \frac{d}{dt} \nabla \Phi_t(X_t) \\ &\overset{(1)}{=} (\partial_t + \nabla \Phi_t(X_t) \cdot \nabla) \nabla \Phi_t(X_t) \\ &\overset{(2)}{=} -\gamma_t \nabla \Phi_t(X_t) - \frac{1}{2} \nabla \left\| \nabla \Phi_t(X_t) \right\|^2 - \nabla (\frac{\delta \mathcal{F}(\mu_t)}{\delta \mu})(X_t) + (\nabla \Phi_t(X_t) \cdot \nabla) \nabla \Phi_t(X_t) \\ &\overset{(3)}{=} -\gamma_t \nabla \Phi_t(X_t) - \nabla (\frac{\delta \mathcal{F}(\mu_t)}{\delta \mu})(X_t) \\ &\overset{(4)}{=} -\gamma_t V_t - \nabla (\frac{\delta \mathcal{F}(\mu_t)}{\delta \mu})(X_t) \end{aligned}$$

where equation (1) becomes valid from material derivative in fluid dynamic (Von Mises, Geiringer, and Ludford 2004), equation (2) comes from the PDEs of Φ_t in (21), equation (3) comes from cancelling terms on each side of (23), equation (4) comes from the definition of V_t. □

Now we are ready to give the proof of Proposition 1.

Proof. (Proof of Proposition 1) Let $\Psi : \mathcal{P}_2(\mathbb{R}^d) \to \mathbb{R}$ be a functional on the probability space, $\tilde{\mu}_t^M$ be the distribution produced by the continuous-time composite flow (13) at time t. With μ_t denoting the mean-field limit of $\tilde{\mu}_t^M$ as $M \to \infty$, we have

$$\partial_t \Psi[\mu_t] = (\mathcal{L}\Psi)[\mu_t],$$

where,

$$\mathcal{L}\Psi[\mu] = \int \langle \nabla \Phi(\mathbf{x}), \nabla_{\mathbf{x}} \frac{\delta \Psi(\mu)}{\delta \mu}(\mathbf{x}) \rangle \mu(\mathbf{x}) \mathrm{d}\mathbf{x} - \int \frac{\delta \Psi(\mu)}{\delta \mu}(\mathbf{x}) \left(\frac{\delta \mathcal{F}(\mu)}{\delta \mu}(\mathbf{x}) - \mathbb{E}_\mu [\frac{\delta \mathcal{F}(\mu)}{\delta \mu}(\mathbf{x})] \right) \mu(\mathbf{x}) \mathrm{d}\mathbf{x}, \tag{24}$$

in which Φ_t abides

$$\begin{cases} \partial_t \Phi_t + \gamma_t \Phi_t + \frac{1}{2} \left\| \nabla \Phi_t \right\|^2 + \frac{\delta \mathcal{F}(\mu_t)}{\delta \mu} = 0 \\ \Phi_0 = \mathbf{0} \end{cases}$$

and $\frac{\delta \Psi(\mu)}{\delta \mu}(\cdot)$ denotes the first variation of functional Ψ at μ satisfying

$$\int \frac{\delta \Psi(\mu)}{\delta \mu}(\mathbf{x}) \xi(\mathbf{x}) \mathrm{d}\mathbf{x} = \lim_{\epsilon \to 0} \frac{\Psi(\mu + \epsilon \xi) - \Psi(\mu)}{\epsilon}$$

for all signed measure $\int \xi(\mathbf{x}) \mathrm{d}\mathbf{x} = 0$. Let $(\mathcal{L}^{Pos}\Psi)[\mu]$ be the first term of (24), and $(\mathcal{L}^{Wei}\Psi)[\mu]$ be the second term of (24), We have:

$$\mathcal{L}\Psi[\mu] = (\mathcal{L}^{Pos}\Psi)[\mu] + (\mathcal{L}^{Wei}\Psi)[\mu]$$

For the measure valued composite flow $\tilde{\mu}_t^M$ (13), the infinitesimal generator of Ψ w.r.t. $\tilde{\mu}_t^M$ is defined as follows:

$$(\mathcal{L}_M \Psi)[\tilde{\mu}^M] := \lim_{t \to 0^+} \frac{\mathbb{E}_{\tilde{\mu}_0^M = \tilde{\mu}^M}(\Psi[\tilde{\mu}_t^M]) - \Psi(\tilde{\mu}^M)}{t},$$

where $\mathbb{E}_{\tilde{\mu}_0^M=\tilde{\mu}^M}(\Psi[\tilde{\mu}_t^M])$ denotes the expectation of the functional Ψ evaluated along the trajectory $\tilde{\mu}_t^M$ taken conditional on the initialization $\tilde{\mu}_0^M = \tilde{\mu}^M$. As

$$\begin{aligned}\mathbb{E}_{\tilde{\mu}_0^M=\tilde{\mu}^M}(\Psi[\tilde{\mu}_t^M]) - \Psi(\tilde{\mu}^M) =& \underbrace{\mathbb{E}_{\tilde{\mu}_0^M=\tilde{\mu}^M}(\Psi[\sum_{i=1}^M w_t^i \delta_{\mathbf{x}_t^i}]) - \mathbb{E}_{\tilde{\mu}_0^M=\tilde{\mu}^M}(\Psi[\sum_{i=1}^M w_0^i \delta_{\mathbf{x}_t^i}])}_{\text{weight adjusting infinitesimal}} \\ &\underbrace{+ \mathbb{E}_{\tilde{\mu}_0^M=\tilde{\mu}^M}(\Psi[\sum_{i=1}^M w_0^i \delta_{\mathbf{x}_t^i}]) - \Psi(\sum_{i=1}^M w_0^i \delta_{\mathbf{x}_0^i}),}_{\text{position adjusting infinitesimal}}\end{aligned}$$

we follow the same idea from (Mroueh and Rigotti 2020; Rotskoff et al. 2019; Zhang et al. 2022) to divide $(\mathcal{L}_M\Psi)[\tilde{\mu}^M]$ into two parts $(\mathcal{L}_M^{Pos}\Psi)[\tilde{\mu}^M]$ and $(\mathcal{L}_M^{Wei}\Psi)[\tilde{\mu}^M]$ corresponding to the position update and the weight adjustment, respectively. According to the definition of first variation, it can be calculated that

$$\begin{aligned}(\mathcal{L}_M^{Wei}\Psi)[\tilde{\mu}^M] &= \lim_{t\to 0^+} \frac{\mathbb{E}_{\tilde{\mu}_0^M=\tilde{\mu}^M}(\Psi[\sum_{i=1}^M w_t^i \delta_{\mathbf{x}_t^i}]) - \mathbb{E}_{\tilde{\mu}_0^M=\tilde{\mu}^M}(\Psi[\sum_{i=1}^M w_0^i \delta_{\mathbf{x}_t^i}])}{t} \\ &= \int \frac{\delta\Psi(\mu^M)}{\delta\mu}(\mathbf{x}) \sum_{i=1}^M \partial_t \rho(w_t^i \mathbf{x}_0^i) \mathrm{d}\mathbf{x} \\ &= -\int \frac{\delta\Psi(\mu^M)}{\delta\mu}(\mathbf{x})(\frac{\delta\mathcal{F}(\mu^M)}{\delta\mu}(\mathbf{x}) - \mathbb{E}_{\mu^M}[\frac{\delta\mathcal{F}(\mu^M)}{\delta\mu}(\mathbf{x})])\mu^M(\mathbf{x})\mathrm{d}\mathbf{x}.\end{aligned}$$

and

$$\begin{aligned}(\mathcal{L}_n^{Pos}\Psi)[\tilde{\mu}^M] &= \lim_{t\to 0^+} \frac{\mathbb{E}_{\tilde{\mu}_0^M=\tilde{\mu}^M}(\Psi[\sum_{i=1}^M w_0^i \delta_{\mathbf{x}_t^i}]) - \Psi(\sum_{i=1}^M w_0^i \delta_{\mathbf{x}_0^i})}{t} \\ &= \int \frac{\delta\Psi(\mu^M)}{\delta\mu}(\mathbf{x}) \sum_{i=1}^M w_0^i \partial_t \rho(\mathbf{x}_t^i) \mathrm{d}\mathbf{x} \\ &= \int \langle V_{\mu^M}(\mathbf{x}), \nabla_{\mathbf{x}} \frac{\delta\Psi(\mu^M)}{\delta\mu}(\mathbf{x})\rangle \mu^M(\mathbf{x})\mathrm{d}\mathbf{x}\end{aligned}$$

where V_{μ^M} abides

$$\begin{cases}\dfrac{dV_{\mu^M}}{dt} = -\gamma_t V_{\mu^M} - \nabla\dfrac{\delta\mathcal{F}(\mu^M)}{\delta\mu} \\ V_{\mu_0^M} = \mathbf{0}\end{cases}$$

Combining the above equalities, we have

$$\mathcal{L}_M\Psi[\mu_M] = \mathcal{L}_M^{Pos}\Psi[\mu_M] + \mathcal{L}_M^{Wei}\Psi[\mu_M]$$

If we take the limit of $\mathcal{L}_M\Psi[\mu_M]$ as $M \to \infty$ on a sequence such that $\mu^M \rightharpoonup \mu$ (i.e. μ^M weakly converges to μ) a.s., and $\frac{\delta\mathcal{F}(\mu^M)}{\delta\mu} \rightharpoonup \frac{\delta\mathcal{F}(\mu)}{\delta\mu}$ a.s., we can deduce that $V_{\mu^M} \rightharpoonup \nabla\Phi$ in light of Lemma 1. Those allow to conclude that $\mathcal{L}_M^{Wei}\Psi[\mu_M] \to \mathcal{L}^{Wei}\Psi[\mu]$ and $\mathcal{L}_M^{Pos}\Psi[\mu_M] \to \mathcal{L}^{Pos}\Psi[\mu]$, thus $\mathcal{L}_M\Psi[\mu_M] \to \mathcal{L}\Psi[\mu]$.

Since $\partial_t\Psi(\mu_t^M) = \mathcal{L}_M\Psi[\mu^M]$ and $\partial_t\Psi(\mu_t) = \mathcal{L}_M\Psi[\mu_t]$, we have

$$\lim_{n\to\infty} \Psi(\mu_t^M) = \Psi(\mu_t),$$

which indicates that $\mu_t^M \rightharpoonup \mu_t$ if $\mu_0^M \rightharpoonup \mu_0$. Since μ_t satisfying $\partial_t\Psi(\mu_t) = \mathcal{L}\Psi[\mu_t]$ solves the partial differential equation (12), we conclude that the path of (13) starting from $\tilde{\mu}_0^M$ weakly converges to a solution of the partial differential equation (12) starting from μ_0 as $M \to \infty$.

□

A.4 The Extra Decrease of Functional Dissipation of the SHIFR Flow (11)

Here, we investigate the extra decrease property in terms of functional dissipation of the SHIFR gradient flow (11) comparing to the Hamiltonian flow (6) in vanilla information space in the following proposition. Here we only illustrate the Wasserstein case for readers can easily check for general information space case in the same routine.

Proposition 2. *For arbitrary $\overline{\mu} \in \mathcal{P}(\mathbb{R}^n)$ and $\overline{\Phi} \in C(\mathbb{R}^n)$, The local dissipation of functional $\frac{\mathrm{d}\mathcal{F}(\mu_t)}{\mathrm{d}t}$ following the SHIFR gradient flow (11) starting from $(\overline{\mu}, \overline{\Phi})$ has an additional functional dissipation term comparing to the ones following the Hamiltonian flow in non-augmented space (6).*

Take the Wasserstein case as an example. Denote the probability path starting from $(\overline{\mu}, \overline{\Phi})$ following W-SHIFR flow as $(\mu_t^{SHIFR}, \Phi_t^{SHIFR})$, following Hamiltonian flow in vanilla space as (μ_t^H, Φ_t^H). We have:

$$\left.\frac{\mathrm{d}\mathcal{F}(\mu_t^{SHIFR})}{\mathrm{d}t}\right|_{t=0} \leq \left.\frac{\mathrm{d}\mathcal{F}(\mu_t^H)}{\mathrm{d}t}\right|_{t=0} \tag{25}$$

Proof. For W-SHIFR case, according to the result in (24), the following eqaulity holds for any functional $\Psi : \mathcal{P}_2(\mathbb{R}^d) \to \mathbb{R}$ on the probability space $\mathcal{P}_2(\mathbb{R}^d)$, where,

$$\begin{aligned}\partial_t \Psi[\mu_t^{SHIFR}] = &\int \langle \nabla \Phi(\mathbf{x}), \nabla_{\mathbf{x}} \frac{\delta \Psi(\mu_t^{SHIFR})}{\delta \mu}(\mathbf{x}) \rangle \mu_t^{SHIFR}(\mathbf{x}) \mathrm{d}\mathbf{x} \\ &- \int \frac{\delta \Psi(\mu_t^{SHIFR})}{\delta \mu}(\mathbf{x}) \left(\frac{\delta \mathcal{F}(\mu_t^{SHIFR})}{\delta \mu}(\mathbf{x}) - \mathbb{E}_{\mu}[\frac{\delta \mathcal{F}(\mu_t^{SHIFR})}{\delta \mu}(\mathbf{x})] \right) \mu_t^{SHIFR}(\mathbf{x}) \mathrm{d}\mathbf{x},\end{aligned}$$

in which $\mu_t|_{t=0} = \overline{\mu}$ and Φ_t abides

$$\begin{cases} \partial_t \Phi_t + \gamma_t \Phi_t + \frac{1}{2} \|\nabla \Phi_t\|^2 + \frac{\delta \mathcal{F}(\mu_t)}{\delta \mu} = 0 \\ \Phi_t|_{t=0} = \overline{\Phi} \end{cases}$$

By substituting $\Psi(\mu_t^{SHIFR}) = \mathcal{F}(\mu_t^{SHIFR})$, $U_{\mu_t^{SHIFR}} = \frac{\delta \mathcal{F}(\mu_t^{SHIFR})}{\delta \mu}$ and $t = 0$ in the above equality, we have:

$$\begin{aligned}\left.\frac{\mathrm{d}\mathcal{F}(\mu_t^{SHIFR})}{\mathrm{d}t}\right|_{t=0} &= -\int \langle \nabla_{\mathbf{x}} \frac{\delta \mathcal{F}(\overline{\mu})}{\delta \mu}(\mathbf{x}), \nabla \overline{\Phi}(\mathbf{x}) \rangle \overline{\mu}(\mathbf{x}) \mathrm{d}\mathbf{x} - \int \frac{\delta \mathcal{F}(\overline{\mu})}{\delta \mu}(\mathbf{x}) (U_{\overline{\mu}}(\mathbf{x}) - \mathbb{E}_{\overline{\mu}}[U_{\overline{\mu}}(\mathbf{x})]) \overline{\mu}(\mathbf{x}) \mathrm{d}\mathbf{x} \\ &= -\int \langle \nabla_{\mathbf{x}} \frac{\delta \mathcal{F}(\overline{\mu})}{\delta \mu}(\mathbf{x}), \nabla \overline{\Phi}(\mathbf{x}) \rangle \overline{\mu}(\mathbf{x}) \mathrm{d}\mathbf{x} - \left(\int \left(\frac{\delta \mathcal{F}(\overline{\mu})}{\delta \mu}(\mathbf{x}) \right)^2 \overline{\mu}(\mathbf{x}) \mathrm{d}\mathbf{x} - \left(\int \frac{\delta \mathcal{F}(\overline{\mu})}{\delta \mu}(\mathbf{x}) \overline{\mu}(\mathbf{x}) \mathrm{d}\mathbf{x} \right)^2 \right).\end{aligned} \tag{26}$$

Similarly, we can get following result for the Hamiltonian flow in the non-augmented Wasserstein space case:

$$\left.\frac{\mathrm{d}\mathcal{F}(\mu_t^H)}{\mathrm{d}t}\right|_{t=0} = -\int \langle \nabla_{\mathbf{x}} \frac{\delta \mathcal{F}(\overline{\mu})}{\delta \mu}(\mathbf{x}), \nabla \overline{\Phi}(\mathbf{x}) \rangle \overline{\mu}(\mathbf{x}) \mathrm{d}\mathbf{x}. \tag{27}$$

Since the second term of (26) is always less or equal to zero and the fisrt term of (26) is the same as the first term of (27), we can reach to the conclusion that the local dissipation of SHIFR flow has an additional functional dissipation term compared to the Hamiltonian flow in the non-augmented space:

$$\left.\frac{\mathrm{d}\mathcal{F}(\mu_t^{SHIFR})}{\mathrm{d}t}\right|_{t=0} \leq \left.\frac{\mathrm{d}\mathcal{F}(\mu_t^H)}{\mathrm{d}t}\right|_{t=0}$$

For general information space, readers can follow the same routine to get the extra functional dissipation property. □

A.5 The stationary analysis of the SHIFR Flow (11)

Following proposition establish the stationary property of the SHIFR Flow (11) with dissimilarity functional $\mathcal{D}(\cdot|\pi)$ which vanishes at $\mu = \pi$.

Proposition 3. *The target distribution and zero-velocity $(\mu_\infty = \pi, \Phi_\infty = \mathbf{0})$ ($\mathbf{0}$ means a funtion defined on $\mathbb{R}^n$ that always map to zero) is the stationary distribution of the SHIFR flow (11) with dissimilarity functional $\mathcal{D}(\cdot|\pi)$ which satisfy $\mathcal{D}(\pi|\pi) = 0$ with any Information metric tensor $G^I(\mu)[\cdot]$.*

Proof. The SHIFR flow under an Information metric writes:

$$\begin{cases} \partial_t \mu_t = G^I(\mu_t)^{-1}\left[\Phi_t\right] - \left(\dfrac{\delta \mathcal{F}(\mu_t)}{\delta \mu} - \displaystyle\int \dfrac{\delta \mathcal{F}(\mu_t)}{\delta \mu} \mathrm{d}\mu_t\right)\mu_t, \\ \partial_t \Phi_t + \gamma_t \Phi_t + \dfrac{1}{2}\dfrac{\delta}{\delta \mu}(\displaystyle\int \Phi_t G^I(\mu_t)^{-1}\left[\Phi_t\right] dx) + \dfrac{\delta \mathcal{F}(\mu_t)}{\delta \mu} = 0. \end{cases} \tag{28}$$

Because the functional $\mathcal{F}(\cdot)$ is specified as some dissimilarity functional $\mathcal{D}(\cdot|\pi)$, we have:

$$\frac{\delta \mathcal{F}(\mu_\infty)}{\delta \mu} = \frac{\delta \mathcal{F}(\pi)}{\delta \mu} = \frac{\delta \mathcal{D}(\pi, \pi)}{\delta \mu} = \mathbf{0}.$$

From the element of gradient flow, we also have:

$$G^I(\mu)^{-1}\left[\Phi_\infty\right] = G^I(\mu)^{-1}\left[\mathbf{0}\right] = \mathbf{0}.$$

Substituting into (28) that $(\mu_\infty = \pi, \Phi_\infty = \mathbf{0})$, we can get :

$$\partial_t \Phi_t|_{t=\infty} = -\gamma_\infty \Phi_\infty - \frac{1}{2}\frac{\delta}{\delta \mu}(\int \Phi_\infty G^I(\mu_t)^{-1}\left[\Phi_\infty\right] dx) - \frac{\delta \mathcal{F}(\mu_\infty)}{\delta \mu} = \mathbf{0}, \tag{29}$$

$$\partial_t \mu_t|_{t=\infty} = G^I(\mu_\infty)^{-1}\left[\Phi_\infty\right] - \left(\frac{\delta \mathcal{F}(\mu_\infty)}{\delta \mu} - \int \frac{\delta \mathcal{F}(\mu_\infty)}{\delta \mu} \mathrm{d}\mu_\infty\right)\mu_\infty = \mathbf{0}. \tag{30}$$

These suffice for proof.

□

Appendix B

B.1 Kalman-Wasserstein-SHIFR Flow and KWGAD-PVI Algorithms

Combining the Kalman filter to estimate the probability distributions of a dynamic system over time and the Wasserstein metric to measure the difference between these estimated distributions, the Kalman-Wasserstein metric is proposed in ensemble Kalman sampling literature (Garbuno-Inigo et al. 2020). The inverse of Kalman-Wasserstein metric tensor write:

$$G^{KW}(\mu)^{-1}\left[\Phi\right] = -\nabla\cdot(\mu C^{\lambda}(\mu)\nabla\Phi), \Phi \in T^*_{\mu}\mathcal{P}(\mathbb{R}^n), \tag{31}$$

where $\Phi \in T^*_{\mu}\mathcal{P}(\mathbb{R}^n)$, substituting into (2), the gradient flow of Kalman-Wasserstein metric writes:

$$\partial_t\mu_t = G^{KW}(\mu_t)^{-1}\left[\frac{\delta\mathcal{F}(\mu_t)}{\delta\mu}\right] = -\nabla\cdot(\mu_t C^{\lambda}(\mu_t)\nabla\frac{\delta\mathcal{F}(\mu_t)}{\delta\mu}). \tag{32}$$

where $\lambda \geq 0$ is the regularization constant and $C^{\lambda}(\mu)$ is the linear transformation follows:

$$C^{\lambda}(\mu) = \int (x-m(\mu))(x-m(\mu))^T\mu dx + \lambda I, m(\mu) = \int x\mu dx. \tag{33}$$

Substituting the Kalman-Wasserstein metric into the SHIFR flow (11) gives the Kalman-Wasserstein-SHIFR flow:

$$\begin{cases} \partial_t\mu_t = -\nabla\cdot\left(\mu_t C^{\lambda}(\mu_t)\nabla\Phi_t\right) - \left(\dfrac{\delta\mathcal{F}(\mu_t)}{\delta\mu} - \displaystyle\int\frac{\delta\mathcal{F}(\mu_t)}{\delta\mu}\mathrm{d}\mu_t\right)\mu_t, \\ \partial_t\Phi_t + \gamma_t\Phi_t + \dfrac{1}{2}((x-m(\mu_t))^T\displaystyle\int\nabla\Phi_t\nabla\Phi_t^T d\mu_t(x-m(\mu_t)) + \nabla\Phi_t^T C^{\lambda}(\mu_t)\nabla\Phi_t) + \dfrac{\delta\mathcal{F}(\mu_t)}{\delta\mu} = 0. \end{cases} \tag{34}$$

We claim that the finite-particles formulation of the Kalman-Wasserstein-SHIFR flow (34) evolves the positions x^i's, the weights w^i's of M particles and velocity field $\mathbf{v}$ following:

$$\begin{cases} \mathrm{d}\mathbf{x}_t^i = C^{\lambda}(\tilde{\mu}_t)\mathbf{v}_t^i\mathrm{d}t, \\ \mathrm{d}\mathbf{v}_t^i = (-\gamma\mathbf{v}_t^i - \mathbb{E}[\mathbf{v}_t\mathbf{v}_t^T](\mathbf{x}^i - \mathbb{E}[\mathbf{x}]) - \nabla\dfrac{\delta\mathcal{F}(\tilde{\mu}_t)}{\delta\mu}(\mathbf{x}_t^i))\mathrm{d}t, \\ \mathrm{d}w_t^i = -\left(\frac{\delta\mathcal{F}(\tilde{\mu}_t)}{\delta\mu}(\mathbf{x}_t^i) - \sum_{i=1}^M w_t^i\frac{\delta\mathcal{F}(\tilde{\mu}_t)}{\delta\mu}(\mathbf{x}_t^i)\right)w_t^i\mathrm{d}t, \\ \tilde{\mu}_t = \sum_{i=1}^M w_t^i\delta_{\mathbf{x}_t^i}. \end{cases} \tag{35}$$

Here the expectation is taken over the empirical distribution of particles.

Then, the proposition below show the mean-field limit of the finite-particles formulation (35) is exactly the Kalman-Wasserstein-SHIFR flow (34).

Proposition 4. *Suppose the empirical distribution $\tilde{\mu}_0^M$ of M weighted particles weakly converges to a distribution μ_0 when $M\to\infty$. Then, the path of* (35) *starting from $\tilde{\mu}_0^M$ and Φ_0 with initial velocity* **0** *weakly converges to a solution of the Kalman-Wasserstein-SHIFR gradient flow* (34) *starting from $\mu_t|_{t=0} = \mu_0 =$ and $\Phi_t|_{t=0} = \mathbf{0}$ as $M\to\infty$:*

Similar to the proof scheme of proposition 1, we start from proofing a technical lemma first:

Lemma 2. *The following fluid dynamic formulation and particle dynamic formulation is equivalent:*
(Suppose that $X_t\sim\mu_t$ and $V_t = \nabla\Phi_t(X_t)$, expectation is taken over particles)

$$\begin{cases} \partial_t\mu_t + \nabla\cdot\left(\mu_t C^{\lambda}(\mu_t)\nabla\Phi_t\right) = 0, \\ \partial_t\Phi_t + \gamma_t\Phi_t + \dfrac{1}{2}((x-m(\mu_t))^T\displaystyle\int\nabla\Phi_t\nabla\Phi_t^T d\mu_t(x-m(\mu_t)) + \nabla\Phi_t^T C^{\lambda}(\mu_t)\nabla\Phi_t) + \dfrac{\delta\mathcal{F}(\mu_t)}{\delta\mu} = 0. \end{cases} \tag{36}$$

$$\begin{cases} \dfrac{d}{dt}X_t = C^{\lambda}(\mu_t)V_t, \\ \dfrac{d}{dt}V_t = -\gamma_t V_t - \mathbb{E}[V_tV_t^T](X_t - \mathbb{E}[X_t]) - \nabla(\dfrac{\delta\mathcal{F}(\mu_t)}{\delta\mu})(X_t). \end{cases} \tag{37}$$

Proof. First we establish two equations, For $i = 1\dots n$, we have:

$$\begin{aligned} (C^{\lambda}(\mu_t)\nabla\Phi_t\cdot\nabla)\nabla_i\Phi_t(X_t) &= \sum_{j=1}^n (C^{\lambda}(\mu_t)\nabla\Phi_t)_j\nabla_j\nabla_i\Phi_t(X_t) \\ &= \sum_{j=1}^n \nabla_{ij}\Phi_t(X_t)(C^{\lambda}(\mu_t)\nabla\Phi_t)_j \\ &= (\nabla^2\Phi_t C^{\lambda}(\mu_t)\nabla\Phi_t)_i. \end{aligned} \tag{38}$$

Them, according to chain rule, we have:

$$\nabla(\nabla\Phi_t(x)^T C^\lambda(\mu_t)\nabla\Phi_t(x)) = 2\nabla^2\Phi_t(x)C^\lambda(\mu_t)\nabla\Phi_t(x). \tag{39}$$

Since the first equation of (36) is actually the continuity equation with velocity field $C^\lambda(\mu_t)\nabla\Phi_t$, it is obvious to have $\frac{d}{dt}X_t = C^\lambda(\mu_t)V_t$. Then we deduce the second equation of (37):

$$\begin{aligned}
\frac{d}{dt}V_t &= \frac{d}{dt}\nabla\Phi_t(X_t) \\
&\overset{(1)}{=} (\partial_t + C^\lambda(\mu_t)\nabla\Phi_t\cdot\nabla)\nabla\Phi_t(X_t) \\
&\overset{(2)}{=} \partial_t\nabla\Phi_t + \nabla^2\Phi_t C^\lambda(\mu_t)\nabla\Phi_t \\
&\overset{(3)}{=} -\gamma_t\nabla\Phi_t - \int\nabla\Phi_t\nabla\Phi_t^T d\mu_t(x-m(\mu_t)) - \frac{1}{2}\nabla(\nabla\Phi_t(x)^T C^\lambda(\mu_t)\nabla\Phi_t(x)) - \nabla(\frac{\delta\mathcal{F}(\mu_t)}{\delta\mu})(X_t) + \nabla^2\Phi_t C^\lambda(\mu_t)\nabla\Phi_t \\
&\overset{(4)}{=} -\gamma_t\nabla\Phi_t - \int\nabla\Phi_t\nabla\Phi_t^T d\mu_t(x-m(\mu_t)) - \nabla(\frac{\delta\mathcal{F}(\mu_t)}{\delta\mu})(X_t) \\
&\overset{(5)}{=} -\gamma_t V_t - \mathbb{E}[V_tV_t^T](X_t-\mathbb{E}[X_t]) - \nabla(\frac{\delta\mathcal{F}(\mu_t)}{\delta\mu})(X_t).
\end{aligned}$$

where equation (1) becomes valid from material derivative in fluid dynamic (Von Mises, Geiringer, and Ludford 2004), equation (2) comes from the equation (38), equation (3) comes from the PDE (36), equation (4) comes from cancelling terms on each side of (39), equation (5) comes from the definition of V_t and X_t. □

Proof. (Proof of Proposition 4) Substituting Lemma 1 by Lemma 2, the proof scheme of proposition 4 is the same with the proof scheme of proposition 1. □

By discretizing (35), we derive the KWGAD-PVI algorithms which update the positions of particles according to the following rule:

$$\mathbf{x}_{k+1}^i = \mathbf{x}_k^i + \eta_{pos}[C_k^\lambda\mathbf{v}_k]^i, \tag{40}$$

and adjusts the velocity field as following:

$$\mathbf{v}_{k+1}^i = (1-\gamma\eta_{vel})\mathbf{v}_k^i - \frac{\eta_{vel}}{M}[\sum_{j=1}^N w_k^j(V_k^j)(V_k^j)^T](\mathbf{x}_k^i - m_k) - \eta_{vel}\nabla U_{\tilde{\mu}_k}(\mathbf{x}_k). \tag{41}$$

Here, C_k^λ and m_k are calculated at each round by:

$$m_k = \frac{1}{N}\sum_{i=1}^M w_k^i x_k^i, C_k^\lambda = \frac{1}{N-1}\sum_{i=1}^M w_k^i(\mathbf{x}_k^i - m_k)(\mathbf{x}_k^i - m_k)^T + \lambda I. \tag{42}$$

B.2 Stein-SHIFR Flow and SGAD-PVI Algorithms

Involving reproducing kernel Hilbert space norm into probability space, the Stein metric is proposed for geometrical analysis (Nüsken and Renger 2021). The gradient flow of Stein metric writes:

$$\partial_t\mu_t = G^S(\mu_t)^{-1}\frac{\delta\mathcal{F}(\mu_t)}{\delta\mu} = -\nabla\cdot(\mu_t\int k(\cdot,y)\mu_t(y)\nabla_y\frac{\delta\mathcal{F}(\mu_t)}{\delta\mu}(y)dy). \tag{43}$$

Substituting the Stein metric into the SHIFR flow (11) gives the Stein-SHIFR flow:

$$\begin{cases}
\partial_t\mu_t = -\nabla\cdot(\mu_t\int k(\cdot,y)\mu_t(y)\nabla_y\Phi_t(y)dy) - \left(\frac{\delta\mathcal{F}(\mu_t)}{\delta\mu} - \int\frac{\delta\mathcal{F}(\mu_t)}{\delta\mu}\mathrm{d}\mu_t\right)\mu_t, \\
\partial_t\Phi_t + \gamma_t\Phi_t + \int\nabla\Phi_t(\cdot)^T\nabla\Phi_t(y)k(\cdot,y)\mu_t(y)dy + \frac{\delta\mathcal{F}(\mu_t)}{\delta\mu} = 0.
\end{cases} \tag{44}$$

We claim that the finite-particles formulation of the Stein-SHIFR flow (44) evolves the positions x^i's, the weights w^i's of M particles and velocity field $\mathbf{v}$ following:

$$\begin{cases} \mathrm{d}\mathbf{x}_t^i = [\int k(\mathbf{x}_t, y)\nabla\Phi_t(y)\tilde{\mu}_t(y)\mathrm{d}y]^i\mathrm{d}t, \\ \mathrm{d}\mathbf{v}_t^i = (-\gamma\mathbf{v}_t^i - [\int \mathbf{v}_t^T\nabla\Phi_t(y)\nabla_x k(\mathbf{x}_t, y)\tilde{\mu}_t(y)\mathrm{d}y]^i - \nabla\frac{\delta\mathcal{F}(\tilde{\mu}_t)}{\delta\mu}(\mathbf{x}_t^i))\mathrm{d}t, \\ \mathrm{d}w_t^i = -\left(\frac{\delta\mathcal{F}(\tilde{\mu}_t)}{\delta\mu}(\mathbf{x}_t^i) - \sum_{i=1}^M w_t^i\frac{\delta\mathcal{F}(\tilde{\mu}_t)}{\delta\mu}(\mathbf{x}_t^i)\right)w_t^i\mathrm{d}t, \\ \tilde{\mu}_t = \sum_{i=1}^M w_t^i\delta_{\mathbf{x}_t^i}. \end{cases} \tag{45}$$

Then, the proposition below show the mean-field limit of the finite-particles formulation (45) is exactly the Stein-SHIFR flow (44).

Proposition 5. *Suppose the empirical distribution $\tilde{\mu}_0^M$ of M weighted particles weakly converges to a distribution μ_0 when $M \to \infty$. Then, the path of* (45) *starting from $\tilde{\mu}_0^M$ and Φ_0 with initial velocity $\mathbf{0}$ weakly converges to a solution of the Stein-SHIFR gradient flow* (44) *starting from $\mu_t|_{t=0} = \mu_0 =$ and $\Phi_t|_{t=0} = \mathbf{0}$ as $M \to \infty$:*

Similarly, fist proof a technical lemma:

Lemma 3. *The following fluid dynamic formulation and particle dynamic formulation is equivalent:*
(Suppose that $X_t \sim \mu_t$ and $V_t = \nabla\Phi_t(X_t)$)

$$\begin{cases} \partial_t\mu_t + \nabla\cdot(\mu_t\int k(\cdot, y)\mu_t(y)\nabla_y\Phi_t(y)dy) = 0, \\ \partial_t\Phi_t + \gamma_t\Phi_t + \int\nabla\Phi_t(\cdot)^T\nabla\Phi_t(y)k(\cdot, y)\mu_t(y)dy + \frac{\delta\mathcal{F}(\mu_t)}{\delta\mu} = 0. \end{cases} \tag{46}$$

$$\begin{cases} \frac{d}{dt}X_t = \int k(X_t, y)\nabla\Phi_t(y)\mu_t(y)dy, \\ \frac{d}{dt}V_t = -\gamma_t V_t - \int V_t^T\nabla\Phi_t(y)\nabla_x k(X_t, y)\mu_t(y)dy - \nabla(\frac{\delta\mathcal{F}(\mu_t)}{\delta\mu})(X_t). \end{cases} \tag{47}$$

Proof. First we notice the following equation:

$$\nabla(\int\nabla\Phi(x)^T\nabla\Phi(y)k(x, y)\mu_t(y)dy) = \nabla^2\Phi(x)\int\nabla\Phi(y)k(x, y)\mu(y)dy + \int\nabla\Phi(x)^T\nabla\Phi(y)\nabla_x k(x, y)\mu(y)dy. \tag{48}$$

Since the first equation of (46) is actually the continuity equation with velocity field $\int k(\cdot, y)\mu_t(y)\nabla_y\Phi_t(y)dy$, it is obvious to have $\frac{d}{dt}X_t = \int k(X_t, y)\nabla\Phi_t(y)\mu_t(y)dy$. Then we deduce the second equation of (47):

$$\begin{aligned} \frac{d}{dt}V_t &= \frac{d}{dt}\nabla\Phi_t(X_t) \\ &\overset{(1)}{=} \partial_t\nabla\Phi_t(X_t) + \nabla^2\Phi_t(X_t)(\int k(X_t, y)\nabla\Phi_t(y)\mu_t(y)dy) \\ &\overset{(2)}{=} -\gamma_t V_t - \nabla(\int\nabla\Phi(x)^T\nabla\Phi(y)k(x, y)\mu_t(y)dy) - \nabla(\frac{\delta\mathcal{F}(\mu_t)}{\delta\mu})(X_t) + \nabla^2\Phi_t(X_t)(\int k(X_t, y)\nabla\Phi_t(y)\mu_t(y)dy) \\ &\overset{(3)}{=} -\gamma_t V_t - \int\nabla\Phi(x)^T\nabla\Phi(y)\nabla_x k(x, y)\mu(y)dy - \nabla(\frac{\delta\mathcal{F}(\mu_t)}{\delta\mu})(X_t) \\ &\overset{(4)}{=} -\gamma_t V_t - \int V_t^T\nabla\Phi_t(y)\nabla_x k(X_t, y)\mu_t(y)dy - \nabla(\frac{\delta\mathcal{F}(\mu_t)}{\delta\mu})(X_t). \end{aligned}$$

where equation (1) becomes valid from material derivative in fluid dynamic (Von Mises, Geiringer, and Ludford 2004), equation (2) comes from the PDE (46), equation (3) comes from leveraging (48), equation (4) comes from the definition of V_t. □

Proof. (Proof of Proposition 5) Substituting Lemma 1 by Lemma 3, the proof scheme of proposition 5 is the same with the proof scheme of proposition 1. □

By discretizing (45) Stein-GAD-PVI algorithm updates the positions of particles according to the following rule:

$$\mathbf{x}_{k+1}^i = \mathbf{x}_k^i + \frac{\eta_{pos}}{M}\sum_{j=1}^{M} K(\mathbf{x}_k^i, \mathbf{x}_k^j)\mathbf{v}_k^i, \tag{49}$$

and adjusts the velocity field as following:

$$\mathbf{v}_{k+1}^i = (1-\gamma\eta_{vel})\mathbf{v}_k^i - \frac{\eta_{vel}}{M}\sum_{j=1}^{M}(\mathbf{v}_k^i)^T\mathbf{v}_k^j\nabla_1 K(\mathbf{x}_k^i, \mathbf{x}_k^j) - \eta_{vel}\nabla U_{\tilde{\mu}_k}(\mathbf{x}_k). \tag{50}$$

B.3 GAD-PVI Algorithms in details

various Dissimilarity Functionals and Smoothing Approaches To develop practical GAD-PVI methods, we must first select a dissimilarity functional $\mathcal{F}$. Once a dissimilarity functional $\mathcal{F}$ has been chosen, we need to select a smoothing approach to approximate the first variation of the empirical approximation, as the value of $\frac{\delta\mathcal{F}(\cdot)}{\mu}$ at an empirical distribution $\tilde{\mu} = \sum_{i=1}^M w^i\delta_{\mathbf{x}^i}$ is generally not well-defined. Smoothing strategies allow us to approximate the first variation value at the discrete empirical distribution. Generally, a smoothed approximation to the first total variation is denoted as $U_{\tilde{\mu}}(\cdot) \approx \frac{\delta\mathcal{F}(\tilde{\mu})}{\delta\mu}(\cdot)$. The commonly BLOB (with KL-divergence as $\mathcal{F}$) (Craig and Bertozzi 2016) has been introduced in (14), now we give detailed formulations of GFSD (with KL-divergence as $\mathcal{F}$) (Liu et al. 2019) and KSDD (with Kernel Stein Discrepancy as $\mathcal{F}$) (Korba et al. 2021), which are all compatible with our GAD-PVI framework. *KL-GFSD* In order to deal with the intractable $\log\mu(\mathbf{x})$ of the first variation of the KL divergence, GFSD directly approximate μ by smoothing the empirical distribution $\tilde{\mu}$ with a kernel function K: $\hat{\mu} = \tilde{\mu} * K = \sum_{i=1}^M w^i K(\cdot, \mathbf{x}^i)$, which leads to the following approximations:

$$U_{\tilde{\mu}_k}(\mathbf{x}) = -\log\pi(\mathbf{x}) + \log\textstyle\sum_{i=1}^M w_k^i K(\mathbf{x}, \mathbf{x}_k^i), \tag{51}$$

$$\nabla U_{\tilde{\mu}_k}(\mathbf{x}) = -\nabla\log\pi(\mathbf{x}) + \frac{\sum_{i=1}^M w_k^i\nabla_{\mathbf{x}}K(\mathbf{x}, \mathbf{x}_k^i)}{\sum_{i=1}^M w_k^i K(\mathbf{x}, \mathbf{x}_k^i)}. \tag{52}$$

Remark 1. *In the above approximations, we call the terms defined through the interaction with other particles as the repulsive terms. It can be observed that the BLOB-type approximations* (14) *have an extra repulsive term (the term in the second line) compared to the GFSD-type approximations* (51) *and* (52)*. Practically, this extra repulsive term would drive the particles away from each other further, and result in a better exploration of particles in the probability space. Actually, the BLOB-type methods usually outperforms the GFSD-type methods empirically.*

Remark 2. *Since GFSD and BLOB (partly) smooth the original empirical distribution $\tilde{\mu}$ with a kernel function K, the underlying evolutionary distribution is actually a smoothed version of $\tilde{\mu}$. To update the positions and the weights in the smoothed empirical distribution, one should solve a system of linear equations to obtain the new positions $\mathbf{x}_{k+1}^i$'s and weights w_{k+1}^i's in the k-th iteration. Nevertheless, with a proper kernel function K, such as the RBF kernel, the density $\mu(\mathbf{x}^i)$ at a given position $\mathbf{x}^i$ mainly comes from its corresponding weight w^i. Actually, as the RBF kernel $e^{-h\|x-x_i\|^2}$ approaches 0 when x becomes far from x_i and equals 1 when $x = x_i$, it can be observed that the density at x_i mainly from w^i. Hence, we can still update the positions and weights in a splitting scheme respectively. This approximation performs well in practice.*

KSD-KSDD Except for the KL-divergence, KSD is recently adopted as the dissimilarity functional in the non-accelerated fixed-weight ParVI method KSDD (Korba et al. 2021), whose first variation and the corresponding vector field are defined as

$$\begin{aligned}\tfrac{\delta\mathcal{F}(\mu)}{\delta\mu}(\mathbf{x}) &= \mathbb{E}_{\mathbf{x}'\sim\mu}\left[k_\pi(\mathbf{x}', \mathbf{x})\right], \\ \nabla\tfrac{\delta\mathcal{F}(\mu)}{\delta\mu}(\mathbf{x}) &= \mathbb{E}_{\mathbf{x}'\sim\mu}\left[\nabla_{\mathbf{x}}k_\pi(\mathbf{x}', \mathbf{x})\right].\end{aligned} \tag{53}$$

Here, k_π denotes the Stein kernel (Liu, Lee, and Jordan 2016), and it is defined by the score of π: $s(\mathbf{x}) = \nabla\log\pi(\mathbf{x})$ and a positive semi-definite kernel function K:

$$k_\pi(\mathbf{x}, \mathbf{y}) = s(\mathbf{x})^T s(\mathbf{y})K(\mathbf{x}, \mathbf{y}) + s(\mathbf{x})^T\nabla_{\mathbf{y}}K(\mathbf{x}, \mathbf{y}) + \nabla_{\mathbf{x}}K(\mathbf{x}, \mathbf{y})^T s(\mathbf{y}) + \nabla_{\mathbf{x}}\cdot\nabla_{\mathbf{y}}K(\mathbf{x}, \mathbf{y}).$$

The the first variation and its gradient in (53) can be directly approximated via the empirical distribution $\tilde{\mu}$. We construct the following finite-particle approximations:

$$\begin{aligned}U_{\tilde{\mu}_k}(\mathbf{x}) &= \textstyle\sum_{i=1}^M w_k^i k_\pi(\mathbf{x}_k^i, \mathbf{x}), \\ \nabla U_{\tilde{\mu}_k}(\mathbf{x}) &= \textstyle\sum_{i=1}^M w_k^i\nabla_{\mathbf{x}}k_\pi(\mathbf{x}_k^i, \mathbf{x}).\end{aligned}$$

The Detailed GAD-PVI algorithms Adopting different underlying information metric tensors (W-metric, KW-metic and S-metric), weight adjustment approaches(CA and DK) and dissimilarity functionals/associated smoothing approaches(KL-BLOB, KL-GFSD and KSD-KSDD), we can derive 18 different intances of GAD-PVI, named as WGAD/KWGAD/SGAD-CA/DK-BLOB/GFSD/KSDD. Here we present our General Accelerated Dynamic-weight Particle-based Variational Inference (GAD-PVI) framework, in a more detailed version of Algorithm 1 as Algorithm 2.

Algorithm 2: General Accelerated Dynamic-weight Particle-based Variational Inference (GAD-PVI) framework in details

Input: Initial distribution $\tilde{\mu}_0 = \sum_{i=1}^M w_0^i \delta_{\mathbf{x}_0^i}$, position adjusting step-size η_{pos}, weight adjusting step-size η_{wei}, velocity field adjusting step-size η_{vel}, velocity damping parameter γ.

Choose a suitable functional $\mathcal{F}$ and its smoothing strategy $U_{\tilde{\mu}} \approx \frac{\delta \mathcal{F}(\tilde{\mu})}{\delta \mu}$ from KL-BLOB/KL-GFSD/KSD-KSDD

$$U_{\tilde{\mu}}(\mathbf{x}) \approx \frac{\delta \mathcal{F}(\tilde{\mu})}{\delta \mu}(\mathbf{x}) = \begin{cases} -\log \pi(\mathbf{x}) + \dfrac{\sum_{i=1}^M K(\mathbf{x}, \mathbf{x}_k^i)}{\sum_{i=1}^M K(\mathbf{x}, \mathbf{x}_k^i)} + \sum\limits_{i=1}^M \dfrac{K(\mathbf{x}, \mathbf{x}_k^i)}{\sum_{l=1}^M K(\mathbf{x}_k^i, \mathbf{x}_k^l)} \text{ (KL-BLOB)}, \\ -\log \pi(\mathbf{x}) + \dfrac{\sum_{i=1}^M K(\mathbf{x}, \mathbf{x}_k^i)}{\sum_{i=1}^M K(\mathbf{x}, \mathbf{x}_k^i)} \text{ (KL-GFSD)}, \\ -\frac{1}{M} \sum_{i=1}^M k_\pi(\mathbf{x}_k^i, \mathbf{x}) \text{ (KSD-KSDD)}. \end{cases}$$

for $k = 0, 1, ..., T-1$ **do**
  for $i = 1, 2, ..., M$ **do**

$$\text{Update positions } \mathbf{x}_k^i \text{ according to } \begin{cases} (15)(\text{WGAD}), \\ (40)(\text{KWGAD}), \\ (49)(\text{SGAD}). \end{cases}$$

$$\text{Update positions } \mathbf{v}_k^i \text{ according to } \begin{cases} (16)(\text{WGAD}), \\ (41)(\text{KWGAD}), \\ (50)(\text{SGAD}). \end{cases}$$

  end for
  if Adopt CA strategy for weight adjustment **then**
    Update weights w_k^i according to (17)
  end if
  if Adopt DK strategy for weight adjustment **then**
    for $i = 1, 2, ..., M$ **do**
      Calculate the duplicate/kill rate: $R_{k+1}^i = -\lambda\eta \left(U_{\tilde{\mu}_k}(\mathbf{x}_{k+1}^i) - \frac{1}{M} \sum_{i=1}^M U_{\tilde{\mu}_k}(\mathbf{x}_{k+1}^i) \right)$
    end for
    for $i = 1, 2, ..., M$ **do**
      if $R_{k+1}^i > 0$ **then**
        Duplicate the particle $\mathbf{x}_{k+1}^i$ with probability $1 - \exp(-R_{k+1}^i)$ and kill one which is uniformly chosen from the rest.
      else
        Kill the particle $\mathbf{x}_{k+1}^i$ with probability $1 - \exp(R_{k+1}^i)$ and duplicate one which is uniformly chosen from the rest.
      end if
    end for
  end if
end for
Output:$\tilde{\mu}_T = \frac{1}{M} \sum_{i=1}^M \delta_{\mathbf{x}_T^i}$.

Appendix C

In this section we list the details of experiments setting, parameter tuning and additional results of our empirical studies.

C.1 Experiments Settings

Density of the Gaussian mixture model. The density of the Gaussian mixture model is defined as follows:

$$\pi(\mathbf{x}) \propto \frac{2}{3}\exp\left(-\frac{1}{2}\left\|\mathbf{x}-\mathbf{a}\right\|^2\right)+\frac{1}{3}\exp\left(-\frac{1}{2}\left\|\mathbf{x}+\mathbf{a}\right\|^2\right),$$

where $\mathbf{a} = 1.2 * \mathbf{1}$.

Density of the Gaussian Process task. We follow the experiment setting in (Chen et al. 2018b; Zhang et al. 2022), and use the dataset LIDAR (denoted as $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N$) which consists of 221 observations of scalar variable x_i and y_i. Denote $\mathbf{x} = [x_1, x_2, ..., x_N]^T$ and $\mathbf{y} = [y_1, y_2, ..., y_N]^T$, the target log-posterior w.r.t. the model parameter $\phi = (\phi_1, \phi_2)$ is defined as follows:

$$\log p(\phi|\mathcal{D}) = -\frac{\mathbf{y}^T\mathbf{K}_y^{-1}\mathbf{y}}{2} - \frac{\log\det\left(\mathbf{K}_y\right)}{2} - \log\left(1+\mathbf{x}^T\mathbf{x}\right).$$

Here, $\mathbf{K}_y$ is a covariance function $\mathbf{K}_y = \mathbf{K} + 0.04\mathbf{I}$ with $\mathbf{K}_{i,j} = \exp(\phi_1)\exp(-\exp(\phi_2)(x_i - x_j)^2)$ and $\mathbf{I}$ represents the identity matrix.

Training/Validation/Test dataset in Bayesian neural network. For each dataset in the Bayesian neural network task, we split it into 90% training data and 10% test data randomly, which follows the settings from (Liu and Wang 2016; Zhang et al. 2020, 2022). Besides, we also randomly choose $1/5$ of the training set as the validation set for parameter tuning.

Initialization of particles' positions. In the Gaussian mixture model, we initialize particles according to the standard Gaussian distribution. In the Gaussian process regression task, we initialize particles with mean vector $[0, -10]^T$ and covariance $0.09 * \mathbf{I}_{2\times 2}$ for all the algorithms. As for the Bayesian neural network task, we follow the initialization convention in (Liu and Wang 2016; Zhang et al. 2022).

Bandwidth of Kernel Function in Different Algorithms For all the experiments, we adopt RBF kernel as the kernel function K: $K(\mathbf{x}, \mathbf{y}) = \exp(-\|\mathbf{x}-\mathbf{y}\|_2^2/h)$, where the parameter h is known as the bandwidth (Liu et al. 2019). We follow the convention in (Zhang et al. 2022) and set the parameter $h = \frac{1}{M}\sum_{i=1}^M \left(\min_{j\neq i}\|\mathbf{x}^i - \mathbf{x}^j\|_2^2\right)$ for GFSD-type algorithms and BLOB-type algorithms.

WNES and WAG Liu et al. (2019) follows the accelerated gradient descend methods in the Wasserstein propability space (Liu et al. 2017; Zhang and Sra 2018) and derives the WNES and WAG methods, which update the particles' positions with an extra momentum. Though their methods have WNES and WAG type, we only conduct empirical studies of WNES as baseline because the authors report WNES algorithms are usually more robust and efficient than WAG type algorithms(Liu et al. 2019).

C.2 Parameters Tuning

Detailed Settings for η_{pos}, η_{wei}, η_{vel} and γ Here we present the parameter settings for position adjusting step-size η_{pos}, weight adjusting step-size η_{wei}, velocity field adjusting step-size η_{vel}, velocity damping parameter γ of different algorithms are provided in Table 4, 5, and 6. All the parameters are chosen by grid search. For the position adjusting step-size η_{pos}, we first find a suitable range by a coarse-grain grid search and then fine tune it. Note that, the position step-size are tuned via grid search for the fixed-weight ParVI algorithms, then used in the corresponding dynamic-weight algorithms. The acceleration parameters and weight adjustment parameters are tuned via grid search for each specific algorithm. As a result, it can be observed that the position adjusting step-size for any specific fixed-weight ParVI algorithm, its corresponding dynamic algorithm and the DK variant are the same in these tables. For ease of understanding, we use the rate of weight adjusting step-size η_{wei} divided by the position adjusting step-size η_{pos} to illustrate the tuning. Moreover, inspired by the effective warmup strategy in tuning hyper-parameters, we follow the settings of (Zhang et al. 2022) and construct the weight adjusting step-size parameter schedule using the hyperbolic tangent: $\lambda\tanh\left(2*(t/T)^5\right)$, with t being the current time step and T the total number of steps.

C.3 Additional Experiments Results

Results for SG In this section, we give empirical results on approximating a single-mode Gaussian distribution, whose density is defined as:

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

where $\Sigma_{ii} = 1.0$ and correlation $\Sigma_{ij,i\neq j} = 0.8$. To investigate the influence of number M in this task, we run all the algorithms with $M \in \{32, 64, 128, 256, 512\}$. All the particles are initialized from a Gaussian distribution with zero mean and covariance matrix $0.5 * \mathbf{I}_{10\times 10}$.

Algorithm	tasks	
	Single Gaussian	Gaussian Mixture Model
ParVI-BLOB	1.0e-2, -, -, -	1.0e-2, -, -, -
WAIG-BLOB	1.0e-2, -, 1.0, 0.3	1.0e-2, -, 1.0, 0.3
WNES-BLOB	1.0e-2, -, 1.0, 0.2	1.0e-2, -, 1.0, 0.2
DPVI-CA-BLOB	1.0e-2, 1.0, -, -	1.0e-2, 1.0, -, -
DPVI-DK-BLOB	1.0e-2, 1.0, -, -	1.0e-2, 1.0, -, -
WGAD-CA-BLOB	1.0e-2, 1.0, 1.0, 0.3	1.0e-2, 1.0, 1.0, 0.3
WGAD-DK-BLOB	1.0e-2, 5e-2, 1.0, 0.3	1.0e-2, 5e-2, 1.0, 0.3
KWAIG-BLOB	1.0e-2, -, -, -	1.0e-2, -, -, -
KWGAD-CA-BLOB	1.0e-2, 5e-3, 1.0, 0.9	1.0e-2, 5e-3, 1.0, 0.9
KWGAD-DK-BLOB	1.0e-2, 5e-2, 1.0, 0.9	1.0e-2, 5e-2, 1.0, 0.9
SAIG-BLOB	5.0e-2, -, -, -	2.5e-2, -, -, -
SGAD-CA-BLOB	5.0e-2, 5e-3, 1.0, 0.9	2.5e-2, 5e-3, 1.0, 0.9
SGAD-DK-BLOB	5.0e-2, 5e-2, 1.0, 0.9	2.5e-2, 5e-2, 1.0, 0.9
ParVI-GFSD	1.0e-2, -, -, -	1.0e-2, -, -, -
WAIG-GFSD	1.0e-2, -, 1.0, 0.3	1.0e-2, -, 1.0, 0.3
WNES-GFSD	1.0e-2, -, 1.0, 0.2	1.0e-2, -, 1.0, 0.2
DPVI-CA-GFSD	1.0e-2, 1.0, -, -	1.0e-2, 0.8, -, -
DPVI-DK-GFSD	1.0e-2, 1.0, -, -	1.0e-2, 1.0, -, -
WGAD-CA-GFSD	1.0e-2, 1.0, 1.0, 0.3	1.0e-2, 0.8, 1.0, 0.3
WGAD-DK-GFSD	1.0e-2, 5e-2, 1.0, 0.3	1.0e-2, 5e-2, 1.0, 0.3
KWAIG-GFSD	1.0e-2, -, -, -	1.0e-2, -, -, -
KWGAD-CA-GFSD	1.0e-2, 5e-3, 1.0, 0.9	1.0e-2, 5e-3, 1.0, 0.9
KWGAD-DK-GFSD	1.0e-2, 5e-2, 1.0, 0.9	1.0e-2, 5e-2, 1.0, 0.9
SAIG-GFSD	5.0e-2, -, -, -	2.5e-2, -, -, -
SGAD-CA-GFSD	5.0e-2, 5e-3, 1.0, 0.9	2.5e-2, 5e-3, 1.0, 0.9
SGAD-DK-GFSD	5.0e-2, 5e-2, 1.0, 0.9	2.5e-2, 5e-2, 1.0, 0.9

Table 4: Parameters of different algorithms in SG and GMM(η_{pos}, $\frac{\eta_{wei}}{\eta_{pos}}$, η_{vel} and γ).

In Figure 3 and 4, we plot the W_2 distance to the target of the samples generated by each algorithm w.r.t. iteration and time. We generate 5000 samples from the target distribution π as reference to evaluate W_2. As this task is a simple Single Gaussian model, the approximation error difference between the GAD-PVI algorithms and fixed-weight ones is not so obvious. When particles number gets more, the effect of the dynamic weight adjustment scheme is smaller, for such a large number of fix-weight particles is also sufficed for approximating this simple distribution. Meanwhile, the faster convergence effect of the accelerated position update strategies is quite obvious in the figures. Moreover, we can see that the GFSD-type algorithms cannot outperform the BLOB-type algorithms and SVGD, this may due to the lack of the repulsive mechanism in GFSD which lead to particle system collapse in single-mode task. This also coincide with the discussion in Appendix B.3.

In Table 7, we further report the final W_2 distance between the empirical distribution generated by each algorithm and the target distribution. It can be observed that, in the Wasserstein metric case, CA strategy constantly outperform their fixed-weight counterparts with the same number of particles, the DK variants are weakened due to single-modality of this task. However, in the KW or S metric case, it can be observed that the GAD-DK algorithms outperform than others in majority of cases, this is due to the poor transportation ability of KW and S metric, a direct duplicate/kill mechanism greatly enhance the transport speed from low probability region to high probability region. We also find that the KW and Stype algorithms achieve poor approximation result comparing to WGAD algorithms in terms of the Wasserstein distance to reference points, this may because the WGAD algorithms aim to implicitly minimize the Wasserstein distance, and this is not the case for other two type.

Additional Results for GMM We provide additional results for Gaussian Mixture Model experiment. In Figure 1 and Figure 5, we plot the W_2 w.r.t iteration of each algorithm for all $M = \{32, 63, 128, 256, 512\}$. From these figures, we can observe that, compared with the baseline ParVI algorithms, our GAD-PVI of either CA strategy or DK variants result in a better performance, i.e., less approximation error and faster convergence. Actually, as we have discussed in the methodology part, while the weight-adjustment step in GAD-PVI greatly enhances the expressiveness of particles' empirical distribution, the accelerated position update strategy also bring in faster convergence. From Figure 1, we find that DK variants decrease quite fast at first in the WGAD case, that is because the duplicate/kill scheme will greatly enhance the particle transport ability thus move more particles to high probability region at first comparing to moving particle step by step. In the Figure 5, we observe that DK variants also show better result comparing to other algorithms, this may because of the slow transportation property of KW and S type algorithms.

In Table 8, we further report the final W_2 distance between the empirical distribution generated by each algorithm and the

Algorithm	Smoothing Approaches	
	BLOB	GFSD
ParVI	1.0e-2, –, –, –	1.0e-2, –, –, –
WAIG	1.0e-2, –, 1.0, 0.4	1.0e-2, –, 1.0, 0.3
WNES	1.0e-2, –, 1.0, 0.4	1.0e-2, –, 1.0, 0.4
DPVI-CA	1.0e-2, 0.1, –, –	1.0e-2, 0.3, –, –
DPVI-DK	1.0e-2, 0.01, –, –	1.0e-2, 0.01, –, –
WGAD-CA	1.0e-2, 0.1, 1.0, 0.4,	1.0e-2, 0.3, 1.0, 0.3
WGAD-DK	1.0e-2, 0.01, 1.0, 0.4,	1.0e-2, 0.01, 1.0, 0.3
KWAIG	5.0e-3, –, 1.0, 0.8	1.0e-3, –, 1.0, 0.7
KWGAD-CA	5.0e-3, 0.1, 1.0, 0.8	1.0e-3, 0.3, 1.0, 0.7
KWGAD-DK	5.0e-3, 0.01, 1.0, 0.8	1.0e-3, 0.01, 1.0, 0.7
SAIG	2.0e-2, –, 1.0, 0.7	1.0e-2, –, 1.0, 0.6
SGAD-CA	2.0e-2, 0.1, 1.0, 0.7	1.0e-2, 0.3, 1.0, 0.6,
SGAD-DK	2.0e-2, 0.01, 1.0, 0.7	1.0e-2, 0.01, 1.0, 0.6

Table 5: Parameters of different algorithms in GP(η_{pos}, $\frac{\eta_{wei}}{\eta_{pos}}$, η_{vel} and γ).

target distribution. It can be observed that GAD-PVI algorithms constantly achieve better approximation result than existing algorithms. Notably, for this complex multi-mode task, DK variants show their advantage as the duplicate/kill operation allows transferring particles from low-probability region to distant high-probability area (e.g. among different local modes) especially in KW/S case. However, the DK variants in Wasserstein case are not so competitive with CA algorithms, this difference could lie in that the KW/S metric space are much more influenced by the potential barrier and need DK scheme to transport particles among different local modes but Wasserstein metric are more robust from multi-modality and CA strategy suffice. Besides, the GAD-PVI algorithms with the CA strategy are usually more stable than their counterpart with DK, which may be ascribed to the fluctuations induced by the discrete weight adjustment (0 or $1/M$) in DK.

Additional Results for GP Here, we provide additional results of KW/S-type method for Gaussian Process Regression task in Table 9. The result is quite similar to the Wasserstein case, i.e., both the accelerated position update and the dynamic weight adjustment result in a decreased W_2 and GAD-PVI algorithms consistently achieve loweset W_2 to the target. Note that difference between DK type GAD-PVI and their fixed weight counterpart is not that obvious, due to the fact that the one-mode nature of GP greatly weaken the advantage of DK, i.e., transferring particles from low-probability region to distant high-probability area (e.g. among different local modes).

Additional Results for BNN We provide additional test Negative Log-likelihood results for Bayesian Neural Network experiment for all algorithms in Table 10 and the test RMSE result under KW/S-Type algorithms in Table 11. The results demonstrate that the combination of the accelerated position updating strategy and the dynamically weighted adjustment leads to a lower NLL and RMSE under difference specific IFR space, and WGAD-PVI algorithms with CA usually achieve the best performance in Wasserstein case while KW/SGAD-PVI algorithms with CA or DK are comparable to each others. Note that the position step-size of GAD-PVI are set to the value tuned for their fixed weight counterpart. Actually, if we retune the position step-size for all GAD-PVI algorithms, they are expected to achieve a better performance than existing result.

Results for GAD-KSDD Newly derived KSDD methods proposed by (Korba et al. 2020) evolve particle system according to the direct minimizing the Kernel Stein Discrepancy(KSD) of particles w.r.t. the target distribution. The KSDD method is the first ParVI that introduce the dissimilarity functional whose first variation is well-defined at discrete empirical distribution, thus result in no approximation error when particles number is infinite. Though theoretically impressive, the experimental performance of KSDD is not satisfying, for they are more computationally expensive and have been widely reported to be less stable. Furthermore, KSDD are also reported to make particles easily trapped at saddle points and demanding for convexity of task and sensitive to parameters.

As shown in Figure 6, we make simple Wasserstein experiments of KSDD type algorithms on SG task to illustrate that our GAD-PVI framwork is compatible with the KSD-KSDD approach. The bandwidth of KSDD are reported that should be carefully determined(Korba et al. 2020; Zhang et al. 2022), we follow the conventions in (Lu, Lu, and Nolen 2019) and (Korba et al. 2021), and set the parameter h via grid search. From Figure 6, it can be observed that our GAD-PVI algorithms achieve the best result in different particles number settings. These illustrate our framework can corporate with this new smoothing approach. However, due to the limit of the KSDD itself, it is not realistic to fine tune parameters and conduct empirical studies of KSDD ParVI algorithms on complex tasks. Additionally, in this simple SG task, the final result of KSDD type algorithms is not competitive to other methods at all. So we exclude KSDD type experiments in GMM, GP and BNN.

Algorithm	Datasets			
	Concrete	kin8nm	RedWine	space
ParVI-BLOB	4.0e-6, –, –, –	1.0e-6, –, –, –	3.4e-6, –, –, –	3.0e-6, –, –, –
WAIG-BLOB	4.0e-6, –, 1.0, 0.2	1.0e-6, –, 1.0, 0.3	3.4e-6, –, 1.0, 0.5	3.0e-6, –, 1.0, 0.5
WNES-BLOB	4.0e-6, –, 1.0, 0.3	1.0e-6, –, 1.0, 0.2	3.4e-6, –, 1.0, 0.2	3.0e-6, –, 1.0, 0.2
DPVI-CA-BLOB	4.0e-6, 1.0, –, –	1.0e-6, 0.8, –, –	3.4e-6, 0.5, –, –	3.0e-6, 1.0, –, –
DPVI-DK-BLOB	4.0e-6, 1.0, –, –	1.0e-6, 0.8, –, –	3.4e-6, 0.5, –, –	3.0e-6, 1.0, –, –
WGAD-CA-BLOB	4.0e-6, 1.0, 1.0, 0.2	1.0e-6, 0.8, 1.0, 0.3	3.4e-6, 0.5, 1.0, 0.5	3.0e-6, 1.0, 1.0, 0.5
WGAD-DK-BLOB	4.0e-6, 1.0, 1.0, 0.2	1.0e-6, 0.8, 1.0, 0.3	3.4e-6, 0.1, 1.0, 0.5	3.0e-6, 1.0, 1.0, 0.5
KWAIG-BLOB	4.0e-6, –, 1.0, 0.7	1.0e-6, –, 1.0, 0.3	3.4e-6, –, 1.0, 0.5	2.0e-6, –, 1.0, 0.8
KWGAD-CA-BLOB	4.0e-6, 1.0, 1.0, 0.7	1.0e-6, 0.8, 1.0, 0.3	3.4e-6, 0.5, 1.0, 0.5	2.0e-6, 1.0, 1.0, 0.8
KWGAD-DK-BLOB	4.0e-6, 1.0, 1.0, 0.7	1.0e-6, 0.8, 1.0, 0.3	3.4e-6, 0.1, 1.0, 0.5	2.0e-6, 1.0, 1.0, 0.8
SAIG-BLOB	1.0e-5, –, 1.0, 0.6	2.0e-6, –, 1.0, 0.3	7.0e-6, –, 1.0, 0.5	8.0e-6, –, 1.0, 0.8
SGAD-CA-BLOB	1.0e-5, 1.0, 1.0, 0.6	2.0e-6, 0.8, 1.0, 0.3	7.0e-6, 0.5, 1.0, 0.5	8.0e-6, 1.0, 1.0, 0.8
SGAD-DK-BLOB	1.0e-5, 1.0, 1.0, 0.6	2.0e-6, 0.8, 1.0, 0.3	7.0e-6, 0.1, 1.0, 0.5	8.0e-6, 1.0, 1.0, 0.8
ParVI-GFSD	4.0e-6, –, –, –	1.0e-6, –, –, –	3.4e-6, –, –, –	3.0e-6, –, –, –
WAIG-GFSD	4.0e-6, –, 1.0, 0.1	1.0e-6, –, 1.0, 0.3	3.4e-6, –, 1.0, 0.5	3.0e-6, –, 1.0, 0.5
WNES-GFSD	4.0e-6, –, 1.0, 0.3	1.0e-6, –, 1.0, 0.3	3.4e-6, –, 1.0, 0.2	3.0e-6, –, 1.0, 0.2
DPVI-CA-GFSD	4.0e-6, 1.0, –, –	1.0e-6, 0.8, –, –	3.4e-6, 0.5, –, –	3.0e-6, 1.0, –, –
DPVI-DK-GFSD	4.0e-6, 1.0, –, –	1.0e-6, 0.8, –, –	3.4e-6, 0.5, –, –	3.0e-6, 1.0, –, –
WGAD-CA-GFSD	4.0e-6, 1.0, 1.0, 0.1	1.0e-6, 0.8, 1.0, 0.3	3.4e-6, 0.5, 1.0, 0.5	3.0e-6, 1.0, 1.0, 0.5
WGAD-DK-GFSD	4.0e-6, 1.0, 1.0, 0.1	1.0e-6, 0.8, 1.0, 0.3	3.4e-6, 0.1, 1.0, 0.5	3.0e-6, 1.0, 1.0, 0.5
KWAIG-GFSD	4.0e-6, –, 1.0, 0.6	1.0e-6, –, 1.0, 0.3	3.4e-6, –, 1.0, 0.5	3.0e-6, –, 1.0, 0.3
KWGAD-CA-GFSD	4.0e-6, 1.0, 1.0, 0.6	1.0e-6, 0.8, 1.0, 0.3	3.4e-6, 0.5, 1.0, 0.5	3.0e-6, 1.0, 1.0, 0.3
KWGAD-DK-GFSD	4.0e-6, 1.0, 1.0, 0.6	1.0e-6, 0.8, 1.0, 0.3	3.4e-6, 0.1, 1.0, 0.5	3.0e-6, 1.0, 1.0, 0.3
SAIG-GFSD	4.0e-6, –, 1.0, 0.2	2.0e-6, –, 1.0, 0.3	7.0e-6, –, 1.0, 0.5	6.0e-6, –, 1.0, 0.3
SGAD-CA-GFSD	4.0e-6, 1.0, 1.0, 0.2	2.0e-6, 0.8, 1.0, 0.3	7.0e-6, 0.5, 1.0, 0.5	6.0e-6, 1.0, 1.0, 0.3
SGAD-DK-GFSD	4.0e-6, 1.0, 1.0, 0.2	2.0e-6, 0.8, 1.0, 0.3	7.0e-6, 0.1, 1.0, 0.5	6.0e-6, 1.0, 1.0, 0.3

Table 6: Parameters of different algorithms in BNN(η_{pos}, $\frac{\eta_{wei}}{\eta_{pos}}$, η_{vel} and γ).

Algorithm	Number of particles				
	32	64	128	256	512
ParVI-SVGD	1.320e+00	1.228e+00	1.162e+00	1.102e+00	1.027e+00
ParVI-BLOB	1.315e+00	1.229e+00	1.164e+00	1.102e+00	1.038e+00
WAIG-BLOB	1.314e+00	1.230e+00	1.164e+00	1.100e+00	1.038e+00
WNes-BLOB	1.315e+00	1.230e+00	1.164e+00	1.101e+00	1.037e+00
DPVI-DK-BLOB	1.313e+00	1.229e+00	1.163e+00	1.100e+00	1.035e+00
DPVI-CA-BLOB	1.309e+00	1.227e+00	1.162e+00	1.102e+00	1.037e+00
WGAD-DK-BLOB(Ours)	1.313e+00	1.229e+00	1.163e+00	**1.098e+00**	**1.034e+00**
WGAD-CA-BLOB(Ours)	**1.300e+00**	**1.226e+00**	**1.161e+00**	1.099e+00	1.036e+00
KWAIG-BLOB	1.943e+00	1.922e+00	1.889e+00	1.827e+00	1.777e+00
KWGAD-DK-BLOB(Ours)	**1.920e+00**	**1.884e-02**	**1.848e+00**	**1.798e+00**	**1.757e+00**
KWGAD-CA-BLOB(Ours)	1.942e+00	1.901e+00	1.865e+00	1.816e+00	1.764e+00
SAIG-BLOB	1.451e+00	1.412e+00	1.429e+00	1.436e+00	1.479e+00
SGAD-DK-BLOB(Ours)	**1.435e+00**	**1.355e+00**	**1.341e+00**	**1.219e+00**	**1.143e+00**
SGAD-CA-BLOB(Ours)	1.444e+00	1.396e+00	1.412e+00	1.405e+00	1.407e+00
ParVI-GFSD	1.453e+00	1.353e+00	1.267e+00	1.198e+00	1.136e+00
WAIG-GFSD	1.449e+00	1.353e+00	1.264e+00	1.196e+00	1.134e+00
WNes-GFSD	1.450e+00	1.353e+00	1.265e+00	1.197e+00	1.135e+00
DPVI-DK-GFSD	1.448e+00	1.347e+00	1.267e+00	1.197e+00	1.135e+00
DPVI-CA-GFSD	1.446e+00	1.349e+00	1.259e+00	1.195e+00	1.133e+00
WGAD-DK-GFSD(Ours)	1.448e+00	1.345e+00	1.264e+00	1.195e+00	1.134e+00
WGAD-CA-GFSD(Ours)	**1.398e+00**	**1.332e+00**	**1.252e+00**	**1.191e+00**	**1.131e+00**
KWAIG-GFSD	2.246e+00	2.171e+00	2.134e+00	2.082e-02	2.046e+00
KWGAD-DK-GFSD(Ours)	2.215e+00	**2.154e+00**	**2.092e+00**	**2.073e-02**	**2.022e+00**
KWGAD-CA-GFSD(Ours)	**2.204e+00**	2.160e+00	2.110e+00	2.081e-02	2.035e+00
SAIG-GFSD	1.823e+00	1.789e+00	1.760e+00	1.626e-02	1.609e+00
SGAD-DK-GFSD(Ours)	1.881e+00	**1.782e+00**	**1.609e+00**	**1.437e-02**	**1.401e+00**
SGAD-CA-GFSD(Ours)	**1.820e+00**	1.783e+00	1.720e+00	1.602e-02	1.592e+00

Table 7: Averaged Test W_2 distances for different ParVI methods in SG task.

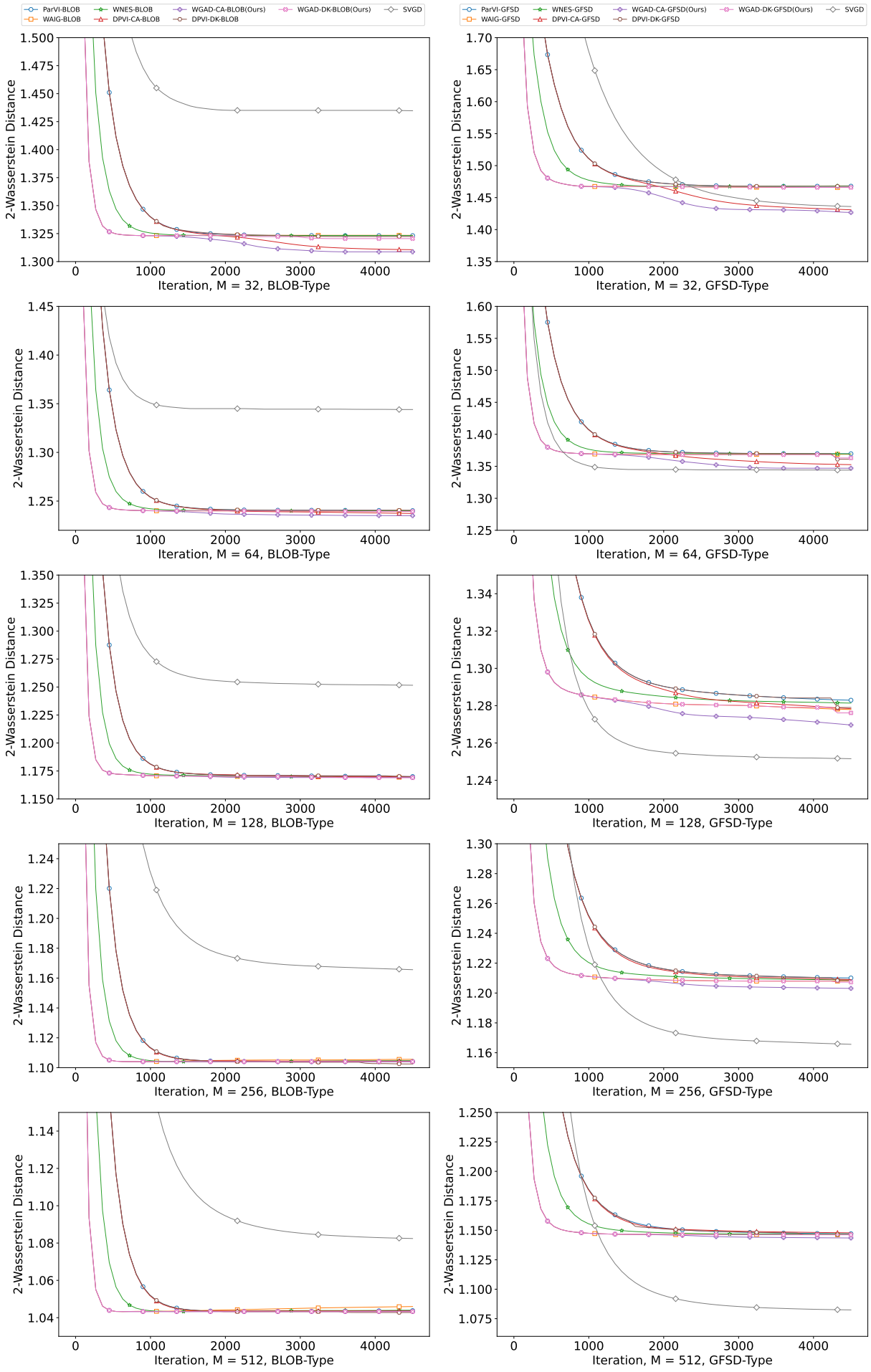

Figure 3: Averaged Test W_2 distance to the target w.r.t. iterations in the SG task for algorithms(W-Type).

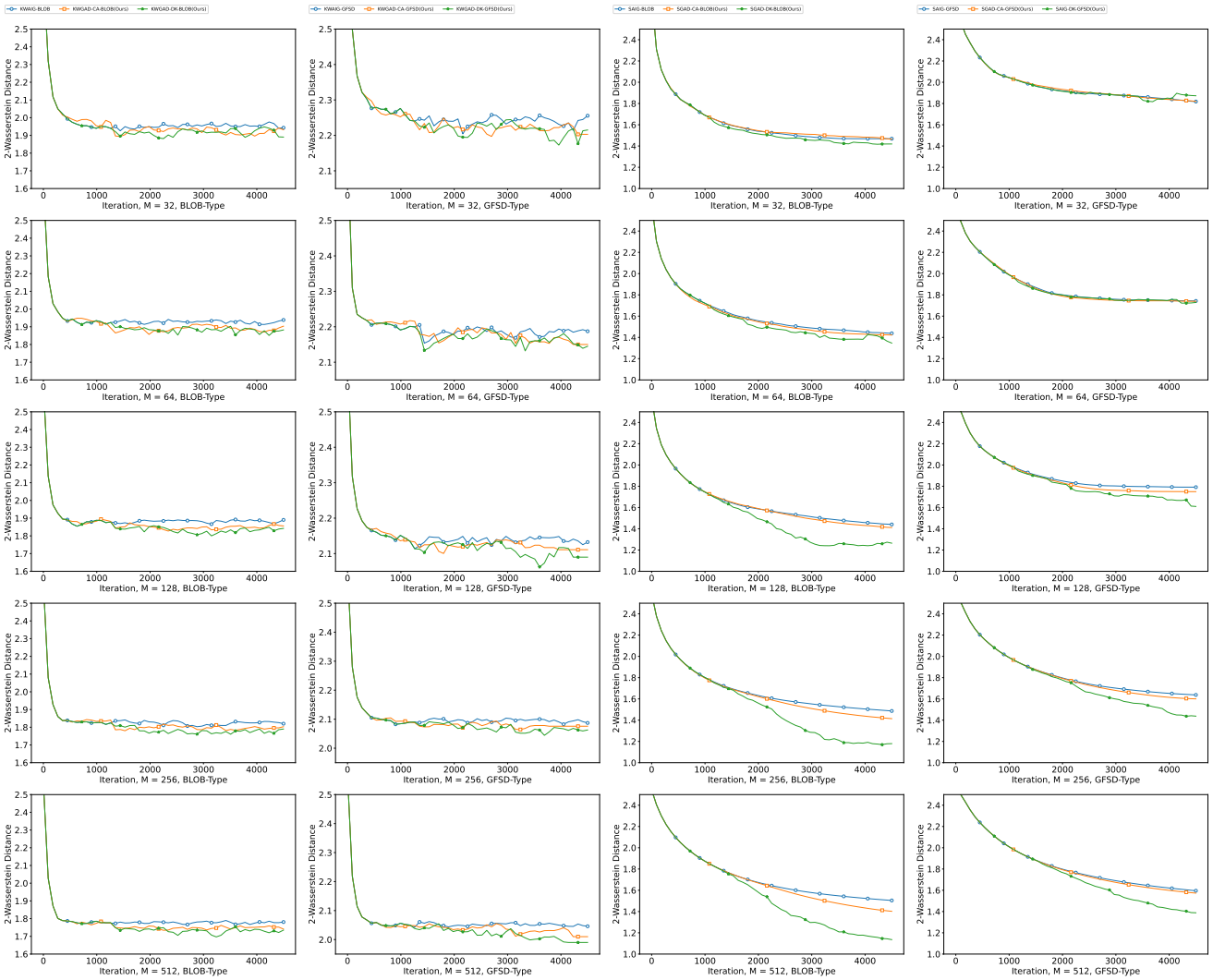

Figure 4: Averaged Test W_2 distance to the target w.r.t. iterations in the SG task for algorithms(KW/S-Type).

Algorithm	Number of particles				
	32	64	128	256	512
ParVI-SVGD	2.175e+00	2.101e+00	2.088e+00	2.026e+00	2.044e+00
ParVI-BLOB	2.317e+00	2.779e+00	2.292e+00	2.440e+00	2.294e+00
WAIG-BLOB	2.317e+00	2.775e+00	2.039e+00	1.976e+00	1.845e+00
WNes-BLOB	2.317e+00	2.777e+00	2.213e+00	2.329e+00	2.180e+00
DPVI-DK-BLOB	2.065e+00	2.066e+00	1.859e+00	1.735e+00	1.704e+00
DPVI-CA-BLOB	2.039e+00	1.934e+00	1.825e+00	1.727e+00	1.633e+00
WGAD-DK-BLOB(Ours)	2.064e+00	1.933e+00	1.831e+00	1.727e+00	1.650e+00
WGAD-CA-BLOB(Ours)	**2.037e+00**	**1.929e+00**	**1.824e+00**	**1.725e+00**	**1.632e+00**
KWAIG-BLOB	4.937e+00	4.674e+00	4.600e+00	4.242e+00	4.221e+00
KWGAD-DK-BLOB(Ours)	**2.854e+00**	**2.622e+00**	**2.400e+00**	**2.542e+00**	**2.248e+00**
KWGAD-CA-BLOB(Ours)	4.565e+00	4.206e+00	4.094e+00	3.836e+00	3.767e+00
SAIG-BLOB	5.070e+00	4.632e+00	4.554e+00	4.140e+00	4.032e+00
SGAD-DK-BLOB(Ours)	**2.863e+00**	**2.914e+00**	2.760e+00	**1.1890e+00**	**2.247e+00**
SGAD-CA-BLOB(Ours)	3.406e+00	3.051e+00	**2.659e+00**	2.595e+00	2.492e+00
ParVI-GFSD	2.427e+00	2.888e+00	2.331e+00	2.567e+00	2.398e+00
WAIG-GFSD	2.425e+00	2.885e+00	2.328e+00	2.206e+00	2.094e+00
WNes-GFSD	2.426e+00	2.888e+00	2.330e+00	2.494e+00	2.337e+00
DPVI-DK-GFSD	2.151e+00	2.025e+00	1.924e+00	1.837e+00	1.769e+00
DPVI-CA-GFSD	2.134e+00	2.025e+00	1.928e+00	1.838e+00	1.755e+00
WGAD-DK-GFSD(Ours)	2.150e+00	**2.017e+00**	1.924e+00	**1.834e+00**	1.755e+00
WGAD-CA-GFSD(Ours)	**2.120e+00**	2.019e+00	**1.923e+00**	1.835e+00	**1.754e+00**
KWAIG-GFSD	5.072e+00	4.780e+00	4.706e+00	4.381e+00	4.359e+00
KWGAD-DK-GFSD(Ours)	**3.086e+00**	**2.682e+00**	**2.817e+00**	**2.385e+00**	**2.393e+00**
KWGAD-CA-GFSD(Ours)	4.597e+00	4.389e+00	4.262e+00	3.960e+00	3.929e+00
SAIG-GFSD	4.828e+00	4.577e+00	4.555e+00	4.151e+00	4.075e+00
SGAD-DK-GFSD(Ours)	**2.994e+00**	**3.411e+00**	**2.881e+00**	**2.347e+00**	**2.324e+00**
SGAD-CA-GFSD(Ours)	3.937e+00	4.142e+00	3.676e+00	3.617e+00	4.007e+00

Table 8: Averaged Test W_2 distances for different ParVI methods in GMM task.

Algorithm	Smoothing Strategy	
	BLOB	GFSD
KWAIG	1.571e-01 $\pm$ 2.190e-04	2.146e-01 $\pm$ 8.608e-04
GAD-KW-DK	1.566e-01 $\pm$ 1.791e-03	2.072e-01 $\pm$ 1.772e-03
GAD-KW-CA	**1.341e-01 $\pm$ 1.494e-04**	**1.991e-01 $\pm$ 4.415e-04**
SAIG	1.570e-01 $\pm$ 3.791e-04	2.084e-01 $\pm$ 7.575e-03
GAD-S-DK	1.552e-01 $\pm$ 1.090e-02	2.012e-01 $\pm$ 4.960e-03
GAD-S-CA	**1.236e-01 $\pm$ 2.872e-04**	**1.691e-01 $\pm$ 3.499e-03**

Table 9: Averaged W_2 distances after 10000 iterations with different KW/S-type algorithms in the GP task with dataset LIDAR.

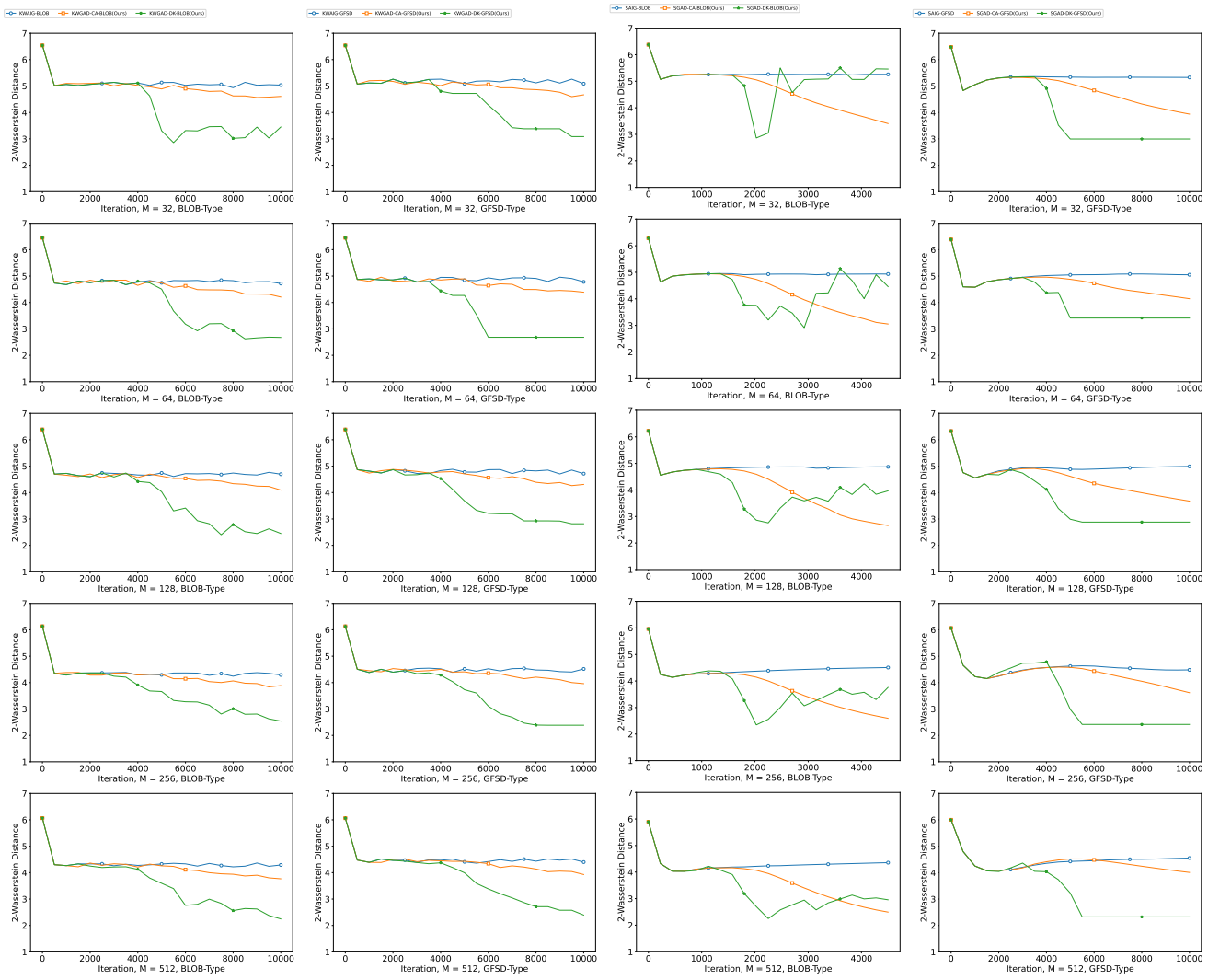

Figure 5: Averaged Test W_2 distance to the target w.r.t. iterations in the GMM task for algorithms(KW/S-Type).

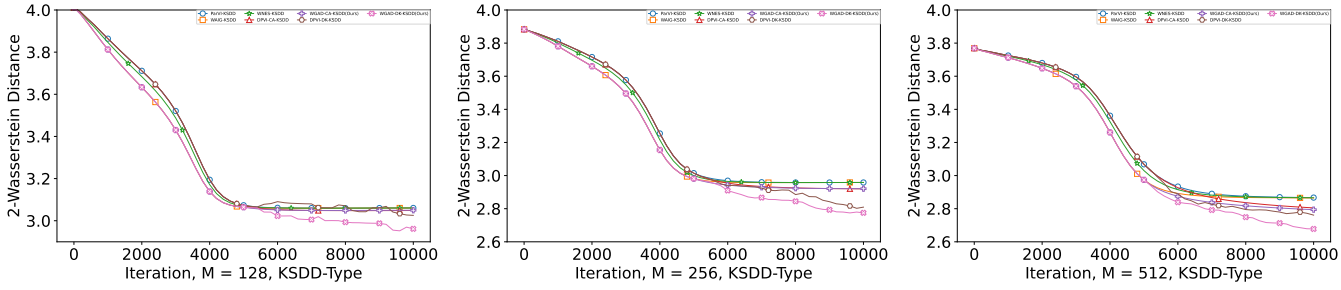

Figure 6: Averaged Test W_2 distance to the target w.r.t. iterations in the SG task for algorithms (WGAD-KSDD-Type).

Algorithm	Datasets			
	Concrete	kin8nm	RedWine	space
ParVI-SVGD	1.738e+00	1.160e+00	6.943e+00	2.739e+00
ParVI-BLOB	1.849e+00	1.122e+00	6.900e+00	2.742e+00
WAIG-BLOB	1.710e+00	1.093e+00	6.790e+00	2.638e+00
WNES-BLOB	1.734e+00	1.065e+00	6.799e+00	2.679e+00
DPVI-DK-BLOB	1.833e+00	1.122e+00	6.848e+00	2.687e+00
DPVI-CA-BLOB	1.837e+00	1.093e+00	6.856e+00	2.685e+00
WGAD-DK-BLOB(Ours)	1.703e+00	1.065e+00	6.785e+00	2.602e+00
WGAD-CA-BLOB(Ours)	**1.697e+00**	**1.048e+00**	**6.782e+00**	**2.595e+00**
KWAIG-BLOB	1.794e+00	1.199e+00	9.322e-01	2.751e+00
KWGAD-DK-BLOB(Ours)	**1.788e+00**	**1.169e+00**	**9.300e-01**	2.743e+00
KWGAD-CA-BLOB(Ours)	1.789e+00	1.173e+00	9.304e-01	**2.736e+00**
SAIG-BLOB	1.832e+00	1.136e+00	9.349e-01	2.711e+00
SGAD-DK-BLOB(Ours)	**1.821e+00**	**1.068e+00**	9.292e-01	2.708e+00
SGAD-CA-BLOB(Ours)	1.824e+00	1.119e+00	**9.270e-01**	**2.642e+00**
ParVI-GFSD	1.850e+00	1.122e+00	6.899e+00	2.742e+00
WAIG-GFSD	1.724e+00	1.094e+00	6.785e+00	2.638e+00
WNES-GFSD	1.740e+00	1.108e+00	6.781e+00	2.679e+00
DPVI-DK-GFSD	1.836e+00	1.120e+00	6.812e+00	2.687e+00
DPVI-CA-GFSD	1.836e+00	1.093e+00	6.857e+00	2.687e+00
WGAD-DK-GFSD(Ours)	1.722e+00	1.075e+00	6.780e+00	**2.593e+00**
WGAD-CA-GFSD(Ours)	**1.720e+00**	**1.050e+00**	**6.774e+00**	2.597e+00
KWAIG-GFSD	1.820e+00	1.199e+00	9.337e-01	2.759e+00
KWGAD-DK-GFSD(Ours)	1.813e+00	1.176e+00	**9.305e-01**	**2.740e+00**
KWGAD-CA-GFSD(Ours)	**1.812e+00**	**1.169e+00**	9.319e-01	2.746e+00
SAIG-GFSD	1.814e+00	1.116e+00	9.444e-01	2.782e+00
SGAD-DK-GFSD(Ours)	1.809e+00	**1.062e+00**	9.391e-01	**2.555e+00**
SGAD-CA-GFSD(Ours)	**1.800e+00**	1.098e+00	**9.359e-01**	2.745e+00

Table 10: Averaged Test NLL distances for different ParVI methods in BNN task.

Algorithm	Datasets			
	Concrete	kin8nm	RedWine	space
KWAIG-BLOB	6.217e+00	8.159e-02	6.916e+00	8.829e-02
KWGAD-DK-BLOB(Ours)	**6.207e+00**	8.069e-02	6.910e+00	8.760e-02
KWGAD-CA-BLOB(Ours)	6.208e+00	**8.058e-02**	**6.896e+00**	**8.753e-02**
SAIG-BLOB	6.323e+00	7.936e-02	6.856e+00	8.899e-02
SGAD-DK-BLOB(Ours)	6.293e+00	**7.695e-02**	6.828e+00	8.867e-02
SGAD-CA-BLOB(Ours)	**6.269e+00**	7.872e-02	**6.811e+00**	**8.783e-02**
KWAIG-GFSD	6.330e+00	8.158e-02	6.924e+00	8.948e-02
KWGAD-DK-GFSD(Ours)	6.274e+00	8.079e-02	**6.854e+00**	**8.918e-02**
KWGAD-CA-GFSD(Ours)	**6.251e+00**	**8.057e-02**	6.917e+00	8.927e-02
SAIG-GFSD	6.266e+00	7.864e-02	6.861e+00	8.998e-02
SGAD-DK-GFSD(Ours)	6.257e+00	**7.679e-02**	6.804e+00	**8.638e-02**
SGAD-CA-GFSD(Ours)	**6.228e+00**	7.801e-02	**6.793e+00**	8.931e-02

Table 11: Averaged Test $RMSE$ for different ParVI methods in BNN task.