Neural Parametric Fokker-Planck Equations

Shu Liu*[1], Wuchen Li†[2], Hongyuan Zha‡[1], and Haomin Zhou§[1]

[1]Georgia Institute of Technology
[2]University of California, Los Angeles

Abstract

In this paper, we develop and analyze numerical methods for high dimensional Fokker-Planck equations by leveraging generative models from deep learning. Our starting point is a formulation of the Fokker-Planck equation as a system of ordinary differential equations (ODEs) on finite-dimensional parameter space with the parameters inherited from generative models such as normalizing flows. We call such ODEs *neural parametric Fokker-Planck equation*. The fact that the Fokker-Planck equation can be viewed as the L^2-Wasserstein gradient flow of Kullback–Leibler (KL) divergence allows us to derive the ODEs as the constrained L^2-Wasserstein gradient flow of KL divergence on the set of probability densities generated by neural networks. For numerical computation, we design a variational semi-implicit scheme for the time discretization of the proposed ODE. Such an algorithm is sampling-based, which can readily handle Fokker-Planck equations in higher dimensional spaces. Moreover, we also establish bounds for the asymptotic convergence analysis of the neural parametric Fokker-Planck equation as well as its error analysis for both the continuous and discrete (forward-Euler time discretization) versions. Several numerical examples are provided to illustrate the performnace of the proposed algorithms and analysis.

Keywords Optimal transport; Transport information geometry; Deep learning; Neural parametric Fokker-Planck equation; Variational semi-implicit-Euler scheme; Numerical analysis.

1 Introduction

Fokker-Planck equation is a parabolic evolution partial differential equation (PDE) that plays a crucial role in stochastic calculus, statistical physics, biology and modeling [33, 39, 43]. Recently, it has seen many applications in machine learning as well [29, 36, 48]. Fokker-Planck equation describes the evolution of the probability density of a stochastic differential equation (SDE). In this research, we mainly focus on the following linear Fokker-Planck equation

$$\frac{\partial \rho(t,x)}{\partial t} = \nabla \cdot (\rho(t,x)\nabla V(x)) + \beta \Delta \rho(t,x), \tag{1}$$

where $x \in \mathbb{R}^d$, $V\colon \mathbb{R}^d \to \mathbb{R}$ is a given potential function and $\beta > 0$ is a diffusion coefficient. In numerical algorithms, there exist several classical methods [38] such as finite difference [10] or finite element [21] for solving the Fokker Planck equation. These methods are grid based, which may be able to approximate the solution accurately if the grid sizes become small. However, they find limited usage in high dimensional problems, especially for $d > 3$, because the number of unknowns grows exponentially fast as the dimension increases. This is known as the curse of dimensionality. The main goal of this paper is providing an alternative strategy, with provable error estimates, to solve the Fokker-Planck equation in high dimensions.

[*] sliu459@gatech.edu
[†] wcli@math.ucla.edu
[‡] zha@cc.gatech.edu
[§] hmzhou@math.gatech.edu

1.1 Neural parametric Fokker-Planck equation

To overcome the challenges imposed by high dimensionality, we leverage the generative models in machine learning [41] and a new interpretation of the Fokker-Planck equation in the theory of optimal transport [51]. We first introduce the KL divergence defined as:

$$\mathrm{D}_{\mathrm{KL}}(\rho||\rho_*) = \int \rho(x)\log\left(\frac{\rho(x)}{\rho_*(x)}\right)dx \quad \rho_*(x) = \frac{1}{Z_\beta}e^{-\frac{V}{\beta}}, \text{ with } Z_\beta = \int e^{-\frac{V(x)}{\beta}}\,dx.$$

Here $\rho_*(x)$ is the Gibbs distribution. A well-known fact is that the Fokker-Planck equation (1) can be viewed as the gradient flow of the functional $\beta\ \mathrm{D}_{\mathrm{KL}}(\rho||\rho_*)$ (also known as relative entropy) on the probability space $\mathcal{P}$ equipped with Wasserstein metric [16, 34]. Recently, this line of research has been extended to parameter space in the field of information geometry [1, 2, 5], leading to an emergent area called transport information geometry [23, 28, 26, 27].

Inspired by aforementioned work, we study the Fokker-Planck equation defined on parametric space Θ equipped with metric tensor G which is compatible with the Wasserstein metric. In this paper, we focus on the parameter space from generative models using neural networks. Our line of thoughts can be summarized as following, we start with a given reference distribution p, and consider a suitable family of parametric pushforward map $\{T_\theta\}_{\theta\in\Theta}$. The so-called pushforward operator $T_\# : \Theta \to \mathcal{P}(\theta \mapsto T_{\theta\#}p)$ can be treated as an immersion from parametric manifold Θ to probability manifold $\mathcal{P}$. We derive the metric tensor $G(\theta)$ by pulling back the Wasserstein metric via immersion $T_\#$. Once we have established (Θ, G), we compute the G-gradient flow of function $H(\theta) = \beta\ D_{\mathrm{KL}}(T_{\theta\#}p||\rho_*)$ defined on the parameter manifold. This leads to an ODE system that can be viewed as a parametric version of Fokker Planck equation:

$$\dot{\theta}_t = -G(\theta_t)^{-1}\nabla_\theta H(\theta_t), \tag{2}$$

in which we use notation $\rho_\theta = T_{\theta\#}p$. The solution $\{\rho_{\theta_t}\}$ can be used as an approximation to the solution ρ_t in (1).

1.2 Computational method

For the computation of (2), we want to point out that metric tensor $G(\theta)$ doesn't have an explicit form and thus the direct computation of $G(\theta)^{-1}\nabla_\theta H(\theta)$ is not tractable. To deal with this issue, we design a numerical algorithm based on the semi-implicit Euler scheme of (2) with time step size h. To be more precise, at each time step, the algorithm seeks to solve the following saddle point problem:

$$\theta_{k+1} = \underset{\theta\in\Theta}{\mathrm{argmin}}\,\max_{\phi}\left\{\int 2\nabla\phi(x)\cdot((T_\theta - T_{\theta_k})\circ T_{\theta_k}^{-1}(x))\rho_{\theta_k}(x)\ dx - \int |\nabla\phi(x)|^2\rho_{\theta_k}(x)\ dx + 2hH(\theta)\right\}. \tag{3}$$

Here $\phi\colon \mathbb{R}^d \to \mathbb{R}$ is the Kantorovich dual potential variable for constrained probability models in optimal transport theory. Hence (3) is derived following the semi-implicit Euler scheme in the dual variable. The advantage of using this formulation is that it allows us to design an efficient implementation, purely based on sampling techniques which are computational friendly in high dimensional problems, to compute the solution of the parameteric Fokker-Planck equation (2).

In our implementation, we endow the pushforward map T_θ with certain kinds of deep neural network known as Normalizing Flow [42], because it is friendly to our scheme evaluations. The dual variable ϕ in the inner maximization is parametrized by the deep Rectified Linear Unit (ReLU) networks [37] . Once the network structures for T_θ and ϕ are chosen, the optimizations are carried out by stochastic gradient descent method [46], in which all terms involved can by computed using samples from the reference distribution p. We stress that this is critical in scaling up the computations in high dimensions. It is worth mentioning that we use neural network as a computational tool without any actual data. Such "data-poor" computation is in significant contrast to the mainstream of deep learning research.

1.3 Major innovations of the proposed method

There are two main innovative points regarding our proposed method:

- (Dimension reduction) Reducing the high dimensional evolution PDE to a finite dimensional ODE system on parameter space. Equivalently, we use the dynamics in a finite dimensional parametric space to approximate the density evolution of particles that follow the Vlasov-type SDE

  $$\dot{\boldsymbol{X}}_t = -\nabla V(\boldsymbol{X}_t) - \beta \nabla \log \rho_t(\boldsymbol{X}_t),$$

  whose density function ρ_t corresponds to the Fokker-Planck equation (1).

- (Sampling-friendly) We distill the information of ρ_t into parameters $\{\theta_t\}$ by solving the parametric Fokker-Planck equation (2). By doing so, we are able to obtain an efficient sampling technique to generate samples from ρ_t for any time step t. To be more precise, we solve (2) for time-dependent parameters $\{\theta_t\}$, and we can then generate samples from ρ_t by pushing forward the samples drawn from a reference distribution p using the pushforward map T_{θ_t}. It worth mentioning that our method is very different from Langevin Monte Carlo (LMC, MALA) methods [14, 44], which aims at targeting the stationary distribution of the SDE associated to (1); or momentum methods [39] , which focuses on keeping track of certain statistical information of the density ρ_t.

1.4 Sketch of numerical analysis

In addition to the methods proposed for solving (1), we also conducted a mathematical analysis on our algorithms. Specifically, we established asymptotic convergence and error analysis results for the continuous version of the parametric Fokker-Planck equation. They are summarized in the following two theorems:

Theorem 1 (Asymptotic convergence analysis for continuous version)**.** *Consider Fokker-Planck equation (1) with V smooth and strictly convex outside a finite ball. Suppose $\{\theta_t\}$ solves (2). Let $\rho_*(x) = \frac{1}{Z_\beta} e^{-V(x)/\beta}$ be the Gibbs distribution of original equation (1). Then we have the inequality:*

$$D_{KL}(\rho_{\theta_t} \| \rho_*) \leq \frac{\delta_0}{\tilde{\lambda}_\beta \beta^2}(1 - e^{-\beta \tilde{\lambda}_\beta t}) + D_{KL}(\rho_{\theta_0} \| \rho_*) e^{-\beta \tilde{\lambda}_\beta t}.$$

Here $\tilde{\lambda}_\beta > 0$ is a constant related to potential function V and β. δ_0 is a constant depending on the approximation power of pushforward map T_θ.

Theorem 2 (Error analysis for continuous version)**.** *Assume $\{\theta_t\}_{t\geq 0}$ solves (2); and $\{\rho_t\}_{t\geq 0}$ solves (1). Assume that the Hessian of the potential function V in (1) is bounded below by a constant λ, i.e. $\nabla^2 V \succeq \lambda I$. Then:*

$$W_2(\rho_{\theta_t}, \rho_t) \leq \frac{\sqrt{\delta_0}}{\lambda}(1 - e^{-\lambda t}) + e^{-\lambda t} W_2(\rho_{\theta_0}, \rho_0).$$

In general cases, λ is not guaranteed to be positive and the error bound in Theorem 2 increases to $+\infty$ as $t \to \infty$. However, we can improved this result by establishing a uniformly small error bound, this is summarized in the following theorem:

Theorem 3 (Main result on error analysis)**.** *Suppose we keep all the notations in Theorem 1 and 2, then for any time $t > 0$, the 2-Wasserstein error $W_2(\rho_{\theta_t}, \rho_t)$ can be bounded above by $K(E_0 + \sqrt{\delta_0})^\alpha$ with some $0 < \alpha \leq 1$. Here $E_0 = W_2(\rho_{\theta_0}, \rho_0)$, K is a positive constant independent of time t.*

This result generally illustrates that under ideal assumption that both the initial error E_0 and $\sqrt{\delta_0}$ are small enough, we will establish a *uniformly* small upper bound for the error term $W_2(\rho_{\theta_t}, \rho_t)$ at all time $t > 0$. Most of the techniques used in our analysis for establishing such result rely on the theories of optimal transport and Wasserstein manifold, which are still not common in today's relevant literature. Besides error analysis for the continuous version of (2), we are also able to verify the order of W_2-error for the discrete version of (2). To be more precise, we apply forward-Euler algorithm to (2) and obtain $\{\theta_k\}$ at different time nodes $\{t_k\}$, we can show that error at t_k: $W_2(\rho_{\theta_k}, \rho_{t_k})$ is of order $O(\sqrt{\delta_0}) + O(C_N h) + O(E_0)$ for finite time t. This is summarized in the following theorem:

Theorem 4. *We assume that $\lambda I \preceq \nabla^2 V \preceq \Lambda I$. The time step size is h. Assume $\{\rho_t\}_{t\geq 0}$ solves (1), $\{\theta_k\}_{k=0}^N$ is the numerical solution of (2) at time nodes $t_k = kh$ for $k = 0, 1, ..., N$ computed by forward Euler scheme. Suppose we keep the notation δ_0 and E_0 in previous theorems. Then:*

$$W_2(\rho_{\theta_k}, \rho_{t_k}) \leq (\sqrt{\delta_0}h + C_N h^2)\frac{(1-e^{-\lambda t_k})}{1-e^{-\lambda h}} + e^{-\lambda t_k}E_0 \sim O(\sqrt{\delta_0}) + O(C_N h) + O(E_0), \quad 0 \leq k \leq N.$$

Here C_N is some constant depending on N and h. As a result, the W_2-error is dominated by three different error terms: $O(\sqrt{\delta_0})$ is the essential error that originates from the approximation mechanism of parametric Fokker-Planck equation; $O(C_N h)$ term is induced by the finite difference scheme; and $O(E_0)$ term is the initial error.

It worth mentioning that we establish Theorem 4 based on different techniques used for Theorem 3. Since the ODE (2) contains the term $G(\theta)^{-1}\nabla_\theta H(\theta)$, which is difficult to deal with, we decide to switch to particle point of view of the ODE (2) and establish corresponding analysis there and finally combine the results to get the desired Theorem 4. Theorem 4 is compatible with Theorem 2 as time stepsize $h \to 0$. Currently, we are not able to establish discrete version of Theorem 1 and thus a discrete version of Theorem 3. This might be one of our future research directions.

1.5 Literature review

We should point out that there are previous works on applying neural networks to solve PDE of various types [52, 40, 18, 19, 54]. Among them, [52] and [19] focuses on high dimensional parabolic partial differential equations. We point out that our approaches differ from these existing works in many aspects, especially the purposes, ways of applying neural networks and the associated numerical analysis.

For example, in [52], the authors are inspired by the non-linear Feynmann-Kac formula that relates the certain parabolic PDE to the Backward Stochastic Differential Equation (BSDE). They reformulate the BSDE as an optimal control problem (also known as reinforcement learning in machine learning community). By applying deep neural network as the control function and optimizing over network parameters, they are able to evaluate the function value of the solution at certain space-time location. Another example is [19], they mainly focus on computing for the committor function that solves a steady-state (time-independent) Fokker-Planck equation with specific boundary conditions. This committor function can be treated as the solution to a variational problem associated with a certain energy functional. They plug neural network into this variational problem and optimize over network parameter to acquire an approximation to the committor function.

In this paper, we handles the other parabolic PDE, i.e. the time dependent Fokker-Planck equation, which actually differs from the parabolic PDEs considered in [52] and steady-state equation treated in [19]. Here we focus on designing a sampling-friendly method. Our numerical solutions as a stream of probability distributions are presented in sample forms, given by deep learning generative models. Despite all above mentioned works apply deep neural networks as computational tools, our approaches are different in terms of how deep networks are leveraged to approximate the solution to the PDE: We use pushforward of a given reference measure by neural networks to create a generative model. This is to approximate the stream of probability distributions; [52] uses networks to approximate the optimal control of a reinforcement learning problem and [19] directly use the network to approximate the solution. More importantly, we provide several numerical analysis on the asymptotic convergence and error control of machine learning approaches. To name a few: Theorem 1 guarantees the entropy-dissipative property of our proposed neural parametric Fokker-Planck equation, Theorem 3 together with Theorem 4 provide upper bounds for the L^2-Wasserstein error between our numerical solution and real solution for both continuous and discrete schemes.

1.6 Organization of this paper

We organize the paper as follows: In section 2, we briefly introduce some background knowledge of Fokker-Planck equation, including its relation with SDE and its Wasserstein gradient flow structure; Then in section 3, we introduce the Wasserstein statistical manifold (Θ, G) and derive our parametric Fokker-Planck equation as the manifold gradient flow of relative entropy on Θ. We study the geometric property of this equation; An insightful particle point of view of the parametric Fokker-Planck equation will also be provided; In section 4, we design a numerical scheme that is tractable for computing our parametric Fokker-Planck equation under deep learning framework. Some important details of implementation will also be discussed; We present asymptotic convergence analysis and error analysis for the parametric Fokker-Planck equation in section 5; Some numerical examples will be exhibited in section 6.

2 Background on Fokker-Planck equation

In this section, we review some basic knowledge about Fokker-Planck equations that will be used in future discussion. In 2.1, we introduce the relationship between Fokker-Planck equation and Stochastic Differential Equations (SDE); then in 2.2.1, we briefly introduce the Wasserstein manifold $(\mathcal{P}, g^W)$; finally, in 2.2.2 we show that Fokker-Planck equation can be treated as the manifold gradient flow of relative entropy functional on $(\mathcal{P}, g^W)$.

2.1 As the density evolution of stochastic differential equation

The general form of Fokker-Planck equation is:

$$\frac{\partial \rho(x,t)}{\partial t} = -\nabla \cdot (\rho(x,t)\boldsymbol{\mu}(x,t)) + \frac{1}{2}\nabla \cdot (\boldsymbol{D}(x,t)\nabla\rho(x,t)) \quad \rho(x,0) = \rho_0(x).$$

Here $\nabla\cdot$, ∇ is the divergence and gradient operator in $\mathbb{R}^d$, $\boldsymbol{\mu}$ is the drift function and $\boldsymbol{D} = \boldsymbol{\sigma}\boldsymbol{\sigma}^T$ is the diffusion tensor. Here $\boldsymbol{\sigma}(x,t)$ is a $d \times \tilde{d}$ matrix. The derivation of Fokker-Planck Equation originates from considering the following stochastic differential equation(SDE) [43]:

$$d\boldsymbol{X}_t = \boldsymbol{\mu}(\boldsymbol{X}_t, t)\ dt + \boldsymbol{\sigma}(\boldsymbol{X}_t, t)\ d\boldsymbol{B}_t, \quad \boldsymbol{X}_0 \sim \rho_0.$$

Here $\{\boldsymbol{B}_t\}_{t\geq 0}$ is the standard Brownian motion in $\mathbb{R}^{\tilde{d}}$. It is well known that the evolution of the density $\rho(x,t)$ of the stochastic process $\{\boldsymbol{X}_t\}_{t\geq 0}$ is described by the Fokker-Planck equation, i.e. suppose $\boldsymbol{X}_t \sim \rho(t, odt)$, then ρ satisfies (2.1).

In this paper, we consider a more specific type of (2.1) by setting $\boldsymbol{\mu}(x,t) = -\nabla V(x)$, $\boldsymbol{\sigma}(x,t) = \sqrt{2\beta}\ I_{d\times d}$ $(\beta > 0)$ and so $\boldsymbol{D} = 2\beta\ I_{d\times d}$. Here $I_{d\times d}$ is the d by d identity matrix. Then (2.1) is:

$$d\boldsymbol{X}_t = -\nabla V(\boldsymbol{X}_t)\ dt + \sqrt{2\beta}\ d\boldsymbol{B}_t \quad \boldsymbol{X}_0 \sim \rho_0. \tag{4}$$

The above is also called over-damped Langevin dynamics with broad applications in computational physics, computational biology, Bayesian statistics etc. [14, 47, 53]. The corresponding Fokker-Planck equation simplifies to

$$\frac{\partial \rho(x,t)}{\partial t} = \nabla \cdot (\rho(x,t)\nabla V(x)) + \beta\Delta\rho(x,t), \quad \rho(x,0) = \rho_0(x). \tag{5}$$

We should also mention that, despite (4), there is a Vlasov-type SDE corresponding to the Fokker-Planck equation (5):

$$\frac{d\boldsymbol{X}_t}{dt} = -\nabla V(\boldsymbol{X}_t) - \beta\ \nabla \log \rho(\boldsymbol{X}_t, t) \quad \boldsymbol{X}_0 \sim \rho_0 \tag{6}$$

Here we denote $\rho(\cdot, t)$ as the density of distribution of $\boldsymbol{X}_t$. Suppose (6) admits a valid solution, then one can show that the density $\rho(\cdot, t)$ solves Fokker-Planck equation (5). This Vlasov-type SDE (6) will be very useful in our further discussions.

2.2 As the Wasserstein gradient flow of relative entropy

A useful viewpoint of (5) is to treat it as the Wasserstein gradient flow of relative entropy. We briefly present some of the notations and basic results in this regard.

2.2.1 Wasserstein manifold

Denote the probability space supported on $\mathbb{R}^d$ with densities and finite second order momentum as:

$$\mathcal{P} = \left\{ \rho\colon \int \rho(x)dx = 1,\ \rho(x) \geq 0,\ \int |x|^2 \rho(x)\ dx < \infty \right\}.$$

We define the so-called Wasserstein distance (also known as L^2-Wasserstein distance) on $\mathcal{P}$ as [51]:

$$W_2(\rho_1, \rho_2) = \left(\inf_{\pi \in \Pi(\rho_1,\rho_2)} \iint |x-y|^2\ d\pi(x,y) \right)^{1/2}. \tag{7}$$

Here $\Pi(\rho_1, \rho_2)$ is the set of joint distributions defined on $\mathbb{R}^d \times \mathbb{R}^d$ with fixed marginal distributions whose densities are ρ_1, ρ_2. If we treat $\mathcal{P}$ as an infinite dimensional manifold, then the Wasserstein distance W_2 can induce a metric g^W on the tangent bundle $T\mathcal{P}$ and then $\mathcal{P}$ becomes a Riemmanian manifold. We now directly give the definition of g^W: One can identify the tangent space at ρ as:

$$T_\rho \mathcal{P} = \left\{ \dot{\rho}\colon \int \dot{\rho}(x)dx = 0 \right\}.$$

Now for a specific $\rho \in \mathcal{P}$ and $\dot{\rho}_i \in T_\rho\mathcal{P}$, $i = 1, 2$, we define the Wasserstein metric tensor g^W as: [22, 34]

$$g^W(\rho)(\dot{\rho}_1, \dot{\rho}_2) = \int \nabla\psi_1(x) \cdot \nabla\psi_2(x)\rho(x)\ dx, \tag{8}$$

where ψ_1, ψ_2 satisfies

$$\dot{\rho}_i = -\nabla \cdot (\rho_i \nabla \psi_i) \quad i = 1, 2, \tag{9}$$

with boundary conditions

$$\lim_{x\to\infty} \rho(x)\nabla\psi_i(x) = 0 \quad i = 1, 2.$$

Use the above definition, we can also write:

$$g^W(\rho)(\dot{\rho}_1, \dot{\rho}_2) = \int \psi_1(-\nabla\cdot(\rho\nabla\psi_2))\ dx = \int (-\nabla\cdot(\rho\nabla))^{-1}(\dot{\rho}_1)\cdot\dot{\rho}_2\ dx.$$

Thus, we can identify $g^W(\rho)$ as $(-\nabla\cdot(\rho\nabla))^{-1}$. When $\mathrm{supp}(\rho) = \mathbb{R}^d$, $g^W(\rho)$ is a positive definite bilinear form defined on tangent bundle $T\mathcal{P} = \{(\rho, \dot{\rho})\colon \rho \in \mathcal{P},\ \dot{\rho} \in T_\rho\mathcal{P}\}$ and we can treat $\mathcal{P}$ as a Riemannian manifold. From now on, we call the manifold $(\mathcal{P}, g^W)$ Wasserstein manifold [34].

2.2.2 Wasserstein gradient

We denote the Wasserstein gradient grad_W as manifold gradient on $(\mathcal{P}, g^W)$. In Riemannian geometry, the manifold gradient should be compatible with the metric, which implies that for any smooth $\mathcal{F}$ defined on $\mathcal{P}$ and for any $\rho \in \mathcal{P}$, consider arbitrary differentiable curve $\{\rho_t\}_{t\in(-\delta,\delta)}$ with $\rho_0 = \rho$, we always have:

$$\frac{d}{dt}\mathcal{F}(\rho_t)\Big|_{t=0} = g^W(\rho)(\mathrm{grad}_W\mathcal{F}(\rho),\ \dot{\rho}_0).$$

Since we can write

$$\frac{d}{dt}\mathcal{F}(\rho_t)\Big|_{t=0} = \int \frac{\delta\mathcal{F}(\rho)}{\delta\rho(x)}(x)\cdot\dot{\rho}_0(x)\ dx = \left\langle \frac{\delta\mathcal{F}(\rho)}{\delta\rho}, \dot{\rho}_0 \right\rangle_{L^2},$$

here $\frac{\delta\mathcal{F}(\rho)}{\delta\rho(x)}(x)$ is the L^2 variation of $\mathcal{F}$ at point $x \in \mathbb{R}^d$, we then have

$$\left\langle \frac{\delta\mathcal{F}(\rho)}{\delta\rho}, \dot{\rho}_0 \right\rangle_{L^2} = g^W(\rho)(\mathrm{grad}_W\mathcal{F}(\rho),\ \dot{\rho}_0) \quad \forall\ \dot{\rho}_0 \in T_\rho\mathcal{P}.$$

This leads to the following useful formula for computing Wasserstein gradient of functional $\mathcal{F}$:

$$\begin{aligned}\mathrm{grad}_W \mathcal{F}(\rho) =& g^W(\rho)^{-1}\left(\frac{\delta \mathcal{F}}{\delta \rho}\right)(x) \\ =& -\nabla\cdot\left(\rho(x)\nabla\,\frac{\delta\mathcal{F}(\rho)}{\delta\rho(x)}(x)\right),\end{aligned} \tag{10}$$

In particular, consider the KL divergence functional [17]:

$$D_{\mathrm{KL}}\left(\rho\middle\|\rho_*\right) = \int \rho(x)\log\left(\frac{\rho(x)}{\rho_*(x)}\right)dx = \int \frac{1}{\beta}V(x)\rho(x) + \rho(x)\log\rho(x)\ dx + \log Z_\beta \tag{11}$$

Here we denote $\rho_*(x) = \frac{1}{Z_\beta}e^{-\frac{V(x)}{\beta}}$ with $Z_\beta = \int e^{-\frac{V(x)}{\beta}}\ dx$.
In the following discussion, we denote:

$$\mathcal{H}(\rho) = \beta\ D_{\mathrm{KL}}\left(\rho\middle\|\rho_*\right) = \int V(x)\rho(x) + \beta\rho(x)\log\rho(x)\ dx + \beta\ \log Z_\beta$$

for shorthand. $\mathcal{H}$ is also known as the relative entropy functional.
Then we have $\nabla\frac{\delta\mathcal{H}(\rho)}{\delta\rho} = \nabla V + \beta\nabla\log\rho$. Using (10), the Wasserstein gradient flow of $\mathcal{H}$ can be written as:

$$\frac{\partial\rho}{\partial t} = -\mathrm{grad}_W\mathcal{H}(\rho) = \nabla\cdot(\rho\nabla V) + \beta\nabla\cdot(\rho\nabla\log\rho)).$$

Notice $\nabla\log\rho = \frac{\nabla\rho}{\rho}$, then $\nabla\cdot(\rho\nabla\log\rho) = \nabla\cdot(\nabla\rho) = \Delta\rho$. The above equation is exactly the Fokker-Planck equation (5).

3 Parametric Fokker-Planck equation

In this section, we provide detailed derivations and related discussions for our parametric Fokker-Planck equation in this section. In 3.1, we first introduce the parameter space Θ and compute the metric tensor G by pulling back Wasserstein metric g^W from $\mathcal{P}$ to Θ; Then in 3.2 we define our parametric Fokker-Planck equation by computing the manifold gradient flow of relative entropy functional on (Θ, G). Some properties related to submanifold geometry will also be provided; in 3.3 we discover a particle formulation for our parametric Fokker-Planck equation. It relates our parametric equation to a "projected" Vlasov-type Stochastic Differential Equation; An illustrative and analytical example is provided in 3.4.

3.1 Wasserstein statistical manifold

Consider a parameter space Θ as an open set in $\mathbb{R}^m$, and assume the sample space is $\mathbb{R}^d$. Let T_θ be a map from $\mathbb{R}^d$ to $\mathbb{R}^d$ parametrized by θ. In our discussion, we will always assume that T_θ is invertible and smooth with respect to parameter θ and variable x.

Remark 1. *There are many different choices for T_θ:*

- *We can set $T_\theta(x) = Ux + b$, with $\theta = (U, b), U \in GL_d(\mathbb{R}),\ b \in \mathbb{R}^d$;*
- *We may also choose T_θ as the linear combination of basis functions $T_\theta(x) = \sum_{k=1}^m \theta_k\vec{\Phi}_k(x)$, where $\{\vec{\Phi}_k\}_{k=1}^m$ are the basis functions and the parameter θ will be the coefficients: $\theta = (\theta_1, ..., \theta_m)$;*
- *We can also treat T_θ as neural network. Its general structure can be written as the composition of l affine and non-linear activation functions: $T_\theta(x) = \sigma_l(W_l(\sigma_{l-1}(...\sigma_1(W_1x + b_1)...)) + b_l)$. In this case, the parameter θ will be the weight matrices and bias vectors of the neural network, i.e. $\theta = (W_1, b_1, ..., W_l, b_l)$.*

We introduce the pushforward operation:

Definition 1. *Suppose X, Y are two measurable spaces, λ is a probability measure defined on X; let $T : X \to Y$ be a measurable map. We define $T_{\#}\lambda$ as: $T_{\#}\lambda(E) = \lambda(T^{-1}(E))$ for all measurable $E \subset Y$. We call $T_{\#}p$ the pushforward of measure p by map T.*

Let $p \in \mathcal{P}$ as a reference probability measure with positive density defined on $\mathbb{R}^d$. For example, we can choose p as the standard Gaussian. We denote ρ_θ as the density of $T_{\theta\#}p$. Such kind of mechanism of producing parametric probability distributions is also known as **generative model**, which has broad applications in deep learning research [13, 4, 7]. We further require:

$$\int |z|^2 \rho_\theta(z)\ dz = \int |T_\theta(x)|^2\ dp(x) < \infty \quad \forall\ \theta \in \Theta. \tag{12}$$

This ensures that $\rho_\theta \in \mathcal{P}$ for each $\theta \in \Theta$. In order to introduce Wasserstein metric defined in previous section to the parameter space Θ, we need to add mild condition on $\partial_\theta T_\theta$. Notice that $\partial_\theta T_\theta : \mathbb{R}^d \to \mathbb{R}^{d\times m}$. Assume there exists an $L^1(p)$ function $L(x)$ that can bound the Frobenius norm $\|\partial_\theta T_\theta(x)\|_F$, i.e.,

$$\exists\ L(x),\ s.t.\ \|\partial_\theta T_\theta(x)\|_F \le L(x)\ \forall x \in \mathbb{R}^d, \theta \in \Theta \quad \text{and} \quad \int L(x)\ dp(x) < \infty. \tag{13}$$

Now suppose the parameter space Θ satisfies conditions (12) and (13). We denote the parametric submanifold $\mathcal{P}_\Theta \subset \mathcal{P}$ as:

$$\mathcal{P}_\Theta = \{\rho_\theta \text{ is density function of } \ T_{\theta\#}p \mid \theta \in \Theta\}.$$

The connection between $\mathcal{P}$ and Θ is the pushforward operation $T_{\#} : \Theta \to \mathcal{P}_\Theta \subset \mathcal{P}, \theta \mapsto \rho_\theta$. In order to introduce the Wasserstein metric to parameter space Θ, it is natural to treat the map $T_{\#}$ as an isometric immersion from Θ to $\mathcal{P}$, then the pullback $(T_{\#})^* g^W$ of the Wasserstein metric g^W by $T_{\#}$ should be the metric tensor on Θ. Let us denote $G = (T_{\#})^* g^W$. Then for each θ, $G(\theta)$ is a bilinear form on $T_\theta\Theta \simeq \mathbb{R}^m$, thus $G(\theta)$ can be identified as an $m \times m$ matrix. The formula for $G(\theta)$ is established in the following theorem:

Theorem 5. *Assume Θ satisfies (12),(13). Suppose T_θ is invertible and smooth with respect to θ and x. We equip Θ with the metric $G = (T_{\#})^* g^W$. Then the metric tensor $G(\theta)$ at $\theta \in \Theta$ is $m \times m$ non-negative definite symmetric matrix of the form:*

$$G(\theta) = \int \nabla\boldsymbol{\Psi}(T_\theta(x))\nabla\boldsymbol{\Psi}(T_\theta(x))^T\ dp(x), \tag{14}$$

Or in entry-wised form:

$$G_{ij}(\theta) = \int \nabla\psi_i(T_\theta(x)) \cdot \nabla\psi_j(T_\theta(x))\ dp(x), \quad 1 \le i, j \le m.$$

Here $\boldsymbol{\Psi} = (\psi_1, \cdots, \psi_m)^T$ and $\nabla\boldsymbol{\Psi}$ is $m \times d$ Jacobian matrix of $\boldsymbol{\Psi}$. For each $j = 1, 2, \cdots, m$, ψ_j solves the following equation:

$$\nabla \cdot (\rho_\theta \nabla\psi_j(x)) = \nabla \cdot (\rho_\theta\ \partial_{\theta_j} T_\theta(T_\theta^{-1}(x))). \tag{15}$$

with boundary conditions

$$\lim_{x\to\infty} \rho_\theta(x)\nabla\psi_j(x) = 0.$$

Proof. Suppose $\xi \in T\Theta$ is a vector field on Θ, for a fixed $\theta \in \Theta$, we first compute the pushforward $(T_{\#}|_\theta)_*\xi(\theta)$ of ξ at point θ: We choose any smooth curve $\{\theta_t\}_{t\ge0}$ on Θ with $\theta_0 = \theta$ and $\dot\theta_0 = \xi(\theta)$. If we denote $\rho_{\theta_t} = T_{\theta_t\#}p$, then we have $(T_{\#})_*\xi(\theta) = \frac{\partial\rho_{\theta_t}}{\partial t}\Big|_{t=0}$.

To compute $\frac{\partial\rho_{\theta_t}}{\partial t}\Big|_{t=0}$, we consider an arbitrary $\phi \in C_0^\infty(M)$. On one hand, $\frac{\rho_{\theta_{\Delta t}}(y) - \rho_{\theta_0}(y)}{\Delta t} = \frac{\partial}{\partial t}\rho(\theta_{\tilde{t}_1}, y)$, where $\tilde{t}_1$ is some point between $0, \Delta t$, since $\phi \in C_0^\infty$ and $\rho(\theta_t, y)$ is smooth with respect to t, y, we can show that the function $\varphi(x) = \sup_{s\in[0,\Delta t]} |\phi(x)\frac{\partial}{\partial t}\rho(\theta_s, y)|$ is continuous on a compact set and thus is integrable on $\mathbb{R}^d$. Using dominant convergence theorem, we have:

$$\frac{\partial}{\partial t}\left(\int \phi(y)\rho_{\theta_t}(y)\ dy\right)\Big|_{t=0} = \int \phi(y)\frac{\partial\rho_{\theta_t}(y)}{\partial t}\Big|_{t=0}\ dy. \tag{16}$$

On the other hand, we have:

$$\frac{\phi(T_{\theta_{\Delta t}}(y)) - \phi(T_{\theta_0}(y))}{\Delta t} = \dot{\theta}_{\tilde{t}_2}^T \ \partial_\theta T_{\theta_{\tilde{t}_2}}(x)^T \ \nabla\phi(T_{\theta_{\tilde{t}_2}}(y)),$$

here $\tilde{t}_2$ is between $0, \Delta t$. For any Δt small enough and $\tilde{t}$, we can easily find an upper bound for $\|\dot{\theta}_{\tilde{t}}\| \leq A$ and since $\phi \in C_0^\infty$, we can bound $\|\nabla\phi(\cdot)\|_\infty \leq B$. Then using (13) we can bound:

$$|\dot{\theta}_{\tilde{t}}^T \ \partial_\theta T_{\theta_{\tilde{t}}}(x)^T \ \nabla\phi(T_{\theta_{\tilde{t}}}(y))| \leq AB\|\partial_\theta T_{\theta_{\tilde{t}}}(x)\|_F \leq ABL(x).$$

Since the right hand side is $L^1(p)$, applying dominated convergence theorem, we have:

$$\frac{\partial}{\partial t}\left(\int \phi(T_{\theta_t}(x))dp\right)\Big|_{t=0} = \int \dot{\theta}_t{}^T \partial_\theta T_{\theta_t}(x)^T \nabla\phi(T_{\theta_t}(x))|_{t=0} dp. \tag{17}$$

Now since $\frac{\partial}{\partial t}\int \phi(y)\rho_{\theta_t}(y)\ dy = \frac{\partial}{\partial t}\int \phi(T_{\theta_t}(x))\ dp(x)$, we can equate (16) and (17) to get:

$$\begin{aligned}
\int \phi(y)\frac{\partial \rho_{\theta_t}}{\partial t}(y)\Big|_{t=0}\ dy &= \int \dot{\theta}_t{}^T \partial_\theta T_{\theta_t}(x)^T \nabla\phi(T_{\theta_t}(x))|_{t=0}\ dp(x) \\
&= \int \dot{\theta}_t^T \partial_\theta T_{\theta_t}(T_{\theta_t}^{-1}(x))^T \nabla\phi(x)\ \rho_{\theta_t}(x)|_{t=0}\ dx \\
&= \int \phi(x)\left(-\nabla\cdot(\rho_{\theta_t}\partial_\theta T_{\theta_t}(T_{\theta_t}^{-1}(x))^T\ \dot{\theta}_t)\right)|_{t=0}\ dx.
\end{aligned}$$

This weak formulation reveals that

$$(T_\#|_\theta)_*\xi(\theta) = \frac{\partial\rho_{\theta_t}}{\partial t}\Big|_{t=0} = -\nabla\cdot(\rho_\theta\ \partial_\theta T_\theta(T_\theta^{-1}(x))^T\ \xi(\theta)). \tag{18}$$

Now let us compute the metric tensor G. Since $T_\#$ is isometric immersion from Θ to $\mathcal{P}$, the pullback of g^W by $T_\#$ gives G, i.e. $(T_\#)^*g^W = G$. By definition of pullback map, for any $\xi \in T\Theta$ and for any $\theta \in \Theta$, we have:

$$G(\theta)(\xi(\theta), \xi(\theta)) = g^W(\rho_\theta)((T_\#|_\theta)_*\xi(\theta), (T_\#|_\theta)_*\xi(\theta)) \tag{19}$$

To compute the right hand side of (19), recall (8), we need to solve for φ from:

$$\frac{\partial\rho_{\theta_t}}{\partial t}\Big|_{t=0} = -\nabla\cdot(\rho_\theta\nabla\varphi(x)) \tag{20}$$

By (18), (20) is:

$$\nabla\cdot(\rho_\theta\nabla\varphi(x)) = \nabla\cdot(\rho_\theta\partial_\theta T_\theta(T_\theta^{-1}(\cdot))^T\ \xi(\theta)). \tag{21}$$

We can straightforwardly check that $\varphi(x) = \mathbf{\Psi}^T(x)\xi(\theta)$ is the solution of (21). Then $G(\theta)$ is computed as:

$$\begin{aligned}
G(\theta)(\xi,\xi) &= \int |\nabla\varphi(y)|^2\ \rho_\theta(y)\ dy = \int |\nabla\varphi(T_\theta(x))|^2\ dp(x) \\
&= \int |\nabla\mathbf{\Psi}(T_\theta(x))^T\xi|^2 dp(x) = \xi^T\left(\int \nabla\mathbf{\Psi}(T_\theta(x))\nabla\mathbf{\Psi}(T_\theta(x))^T dp(x)\right)\xi.
\end{aligned}$$

Thus we can verify that:

$$G(\theta) = \int \nabla\mathbf{\Psi}(T_\theta(x))\nabla\mathbf{\Psi}(T_\theta(x))^T\ dp(x),$$

completing the proof. □

Generally speaking, the metric tensor G does not have an explicit form when $d \geq 2$; but for $d = 1$, G has an explicit form and can be computed directly.

Corollary 5.1. *When dimension $d = 1$, the metric tensor $G(\theta)$ has the following explicit form:*

$$G(\theta) = \int \partial_\theta T_\theta(x)^T \partial_\theta T_\theta(x)\ dp(x). \tag{22}$$

Proof. When $d = 1$, (15) is

$$\frac{d}{dx}\left(\rho_\theta(x)\frac{d}{dx}\psi_k(x)\right) = \frac{d}{dx}\left(\rho_\theta(x)\frac{\partial T_\theta}{\partial \theta_k}(T_\theta^{-1}(x))\right), \tag{23}$$

with boundary conditions $\lim_{x\to\pm\infty}\rho_\theta(x)\psi_k'(x) = 0$. And using (13), we know $\partial_\theta T_\theta$ is $L^1(p)$ integrable and so $\rho_\theta(\cdot)\partial_\theta T_\theta(T_\theta^{-1}(\cdot))$ is Lebesgue integrable, we can find a sequence $\{x_m\} \to -\infty$, such that $\rho_\theta(x_m)\partial_{\theta_k}T_\theta(T_\theta^{-1}(x_m)) \to 0$ as $m \to \infty$. Now for any $x \in \mathbb{R}$, integrate (23) from x_m to x and send $m \to \infty$ we get:

$$\rho_\theta(x)\psi_k'(x) = \rho_\theta(x)\partial_{\theta_k}T_\theta(T_\theta^{-1}(x)).$$

Now, on the support on ρ_θ, we have $\psi_k'(x) = \partial_{\theta_k}T_\theta(T_\theta^{-1}(x))$, thus we have:

$$G_{ij}(\theta) = \int \psi_i'(x)\psi_j'(x)\rho_\theta(x)\, dx = \int \partial_{\theta_i}T_\theta(x)\partial_{\theta_j}T_\theta(x)\, dp(x),$$

completing the proof. □

The following theorem mentioned in [25] ensures the positive definiteness of the metric tensor G:

Theorem 6. *We follow the notations and conditions in this section. Then G is Riemmanian metric if and only if For each $\theta \in \Theta$, for any $\xi \in T_\theta\Theta$ ($\xi \neq 0$), we can find $z \in M$ such that $\nabla \cdot (\rho_\theta(z)\partial_\theta T_\theta(T_\theta^{-1}(z))\xi) \neq 0$.*

Proof. We first establish the following identity: according to Theorem 5, for any θ, ξ, x,

$$\nabla \cdot (\rho_\theta(x)\nabla(\xi^T\mathbf{\Psi}(x))) = \nabla \cdot (\rho_\theta(x)\partial_\theta T_\theta(T_\theta^{-1}(x))\xi). \tag{24}$$

($\Leftarrow$): suppose for any $\theta \in \Theta$ and $\xi \in T_\theta\Theta$, at certain $z \in \mathbb{R}^d$, $\nabla \cdot (\rho_\theta(z)\partial_\theta T_\theta(T_\theta^{-1}(z)\xi) \neq 0$, then $\nabla \cdot (\rho_\theta(z)\nabla(\xi^T\mathbf{\Psi}(z))) \neq 0$, thus $\rho_\theta\nabla(\xi^T\mathbf{\Psi})$ is not identically $\mathbf{0}$. Using continuity of $\rho_\theta\nabla(\xi^T\mathbf{\Psi})$, we know that: $|\nabla(\xi^T\mathbf{\Psi}(x))|^2\rho_\theta(x) > 0$ in some small neighbourhood of z. Thus we have:

$$\xi^T G(\theta)\xi = \int |\nabla\mathbf{\Psi}(x)^T\xi|^2\rho_\theta(x)\, dx > 0, \tag{25}$$

holds for any θ and ξ, this leads to the positive definiteness of G.
($\Rightarrow$): Now, (25) holds for all θ, ξ. We have

$$\int -\nabla \cdot (\rho_\theta(x)\nabla(\xi^T\mathbf{\Psi}(x))) \cdot \xi^T\mathbf{\Psi}(x)\, dx > 0.$$

This leads to the existence of a $z \in \mathbb{R}^d$ such that $-\nabla \cdot (\rho_\theta(z)\nabla(\xi^T\mathbf{\Psi}(z))) \neq 0$. Combining (24) completes the proof. □

A more intuitive way to understand the positive definiteness of $G(\theta)$ is illustrated in the following theorem:

Theorem 7. *For $\theta \in \Theta$, let us recall the definition of $\{\psi_k\}_{k=1}^m$ in (15), then $G(\theta)$ is positive definite if and only if $\{\nabla\psi_k\}_{k=1}^m$ as m vectors in the space $L^2(\mathbb{R}^d;\mathbb{R}^d,\rho_{\theta_k})$, are linearly independent.*

For most of the common choices of T_θ like linear combination of basis functions or smooth invertible neural networks, we may assume Theorem 6, 7 holds. To keep our discussion concise, in the following sections, we will always assume $G(\theta)$ is positive definite for every $\theta \in \Theta$.

3.2 Parametric Fokker-Planck equation

Recall the relative entropy functional $\mathcal{H}$ defined in (11), we consider $H = \mathcal{H} \circ T_\# : \Theta \to \mathbb{R}$. Then:

$$H(\theta) = \mathcal{H}(\rho_\theta) = \int V(x)\rho_\theta(x)\, dx + \beta\int \rho_\theta(x)\log\rho_\theta(x)\, dx = \int V(T_\theta(x)) + \beta\log\rho_\theta(T_\theta(x))\, dp(x). \tag{26}$$

As in [1], the gradient flow of H on Wasserstein statistical manifold (Θ, G) satisfies

$$\dot{\theta} = -G(\theta)^{-1}\nabla_\theta H(\theta)^{1}. \tag{27}$$

We call (27) *parametric Fokker-Planck equation.* The ODE (27) as the Wasserstein gradient flow on parameter space (Θ, G) is closely related to Fokker-Planck equation on probability submanifold $\mathcal{P}_\Theta$. We have the following theorem, which is a natural result derived from submanifold geometry:

Theorem 8. *Suppose $\{\theta_t\}_{t\geq 0}$ solves (27). Then $\{\rho_{\theta_t}\}$ is the gradient flow of $\mathcal{H}$ on probability submanifold $\mathcal{P}_\Theta$. Here we always assume that $\mathcal{P}_\Theta$ inherits the metric of $\mathcal{P}$. Furthermore, at any time t,$\dot{\rho}_{\theta_t} = \frac{d}{dt}\rho_{\theta_t} \in T_{\rho_{\theta_t}}\mathcal{P}_\Theta$ is the orthogonal projection of $-grad_W \mathcal{H}(\rho_{\theta_t}) \in T_{\rho_{\theta_t}}\mathcal{P}$ onto the subspace $T_{\rho_{\theta_t}}\mathcal{P}_\Theta$ with respect to the Wasserstein metric g^W.*

Theorem 8 easily follows from the following two general results about manifold gradient:

Theorem 9. *Suppose $(N, g^N), (M, g^M)$ are Riemannian Manifolds. Suppose $\varphi : N \to M$ is isometry. Consider $\mathcal{F} \in \mathcal{C}^\infty(M)$, define $F = \mathcal{F} \circ \varphi \in \mathcal{C}^\infty(N)$. Suppose $\{x_t\}_{t\geq 0}$ is the gradient flow of F on N:*

$$\dot{x} = -grad_N F(x).$$

Then $\{y_t = \varphi(x_t)\}_{t\geq 0}$ is the gradient flow of $\mathcal{F}$ on M. That is, $\{y_t\}$ satisfies $\dot{y} = -grad_M \mathcal{F}(y)$.

Proof. Since we always have $\dot{y}_t = \varphi_*\dot{x}_t = -\varphi_*\text{grad}_N F(x_t)$, we only need to show that $\varphi_*\text{grad}_N F(x_t) = \text{grad}_M \mathcal{F}(\varphi(x_t))$. Fix the time t, consider any curve $\{\xi_\tau\}$ on N passing through x_t at $\tau = 0$, since φ is isometry, we have $g^N = \varphi^* g^M$, thus:

$$\frac{d}{d\tau}F(\xi_\tau)\Big|_{\tau=0} = g^N(\text{grad}_N F(x_t), \dot{\xi}_0) = \varphi^* g^M(\text{grad}_N F(x_t), \dot{\xi}_0) = g^M(\varphi_*\text{grad}_N F(x_t), \varphi_*\dot{\xi}_0).$$

On the other hand, denote $\eta_\tau = \varphi(\xi_\tau)$, we have:

$$\frac{d}{d\tau}F(\xi_\tau)\Big|_{\tau=0} = \frac{d}{d\tau}\mathcal{F}(\eta_\tau)\Big|_{\tau=0} = g^M(\text{grad}_M \mathcal{F}(y_t), \dot{\eta}_0) = g^M(\text{grad}_M \mathcal{F}(y_t), \varphi_*\dot{\xi}_0).$$

As a result, $g^M(\varphi_*\text{grad}_N F(x_t) - \text{grad}_M \mathcal{F}(y_t), \varphi_*\dot{\xi}_0) = 0$ for all $\dot{\xi}_0 \in T_{x_t}N$.
Since φ_* is surjective, thus $\varphi_*\text{grad}_N F(x_t) = \text{grad}_M \mathcal{F}(\varphi(x_t))$.

□

Theorem 10. *Suppose (M, g^M) is Riemannian manifold, $M_{sub} \subset M$ is the submanifold of M. Assume M_{sub} inherits metric g^M, i.e. define $\iota : M_{sub} \to M$ as the inclusion map, then ι is isometry: $g^{M_{sub}} = \iota^* g^M$. For any $\mathcal{F} \in \mathcal{C}^\infty(M)$, we denote the restriction of $\mathcal{F}$ on M_{sub} as $\mathcal{F}^{sub}$. Then the gradient $grad_{M_{sub}}\mathcal{F}^{sub}(x) \in T_x M_{sub}$ is the orthogonal projection of $grad_M \mathcal{F}(x) \in T_x M$ onto subspace $T_x M_{sub}$ with respect to the metric g^M for any $x \in M_{sub}$.*

Proof. For any $x \in M_{\text{sub}}$, consider any curve $\{\gamma_\tau\}$ on $M_{\text{ sub }}$ passing through x at $\tau = 0$. We have

$$\frac{d}{d\tau}\mathcal{F}^{\text{sub}}(\gamma_\tau)\Big|_{\tau=0} = g^{M_{\text{sub}}}(\text{grad}_{M_{\text{sub}}}\mathcal{F}^{\text{sub}}(x), \dot{\gamma}_0) = g^M(\iota_*\text{grad}_{M_{\text{sub}}}\mathcal{F}^{\text{sub}}(x), \iota_*\dot{\gamma}_0) = g^M(\text{grad}_{M_{\text{sub}}}\mathcal{F}^{\text{sub}}(x), \dot{\gamma}_0).$$

The last equality is because ι_* restricted on TM_{sub} is identity. On the other hand, $\mathcal{F}^{\text{sub}}(\gamma_\tau) = \mathcal{F}(\gamma_\tau)$ for all τ. We also have:

$$\frac{d}{d\tau}\mathcal{F}^{\text{sub}}(\gamma_\tau)\Big|_{\tau=0} = g^M(\text{grad}_M \mathcal{F}(x)\dot{\gamma}_0).$$

Combining them we know

$$g^M(\text{grad}_{M_{\text{sub}}}\mathcal{F}^{\text{sub}}(x) - \text{grad}_M \mathcal{F}(x), v) \quad \forall\, v \in T_x M_{\text{sub}} \Rightarrow \text{grad}_{M_{\text{sub}}}\mathcal{F}^{\text{sub}}(x) - \text{grad}_M \mathcal{F}(x) \perp_{g^M} T_x M_{\text{sub}},$$

which proves this result.

□

[1] Here (and for later) dot symbol $\dot{\theta}$ stands for time derivative $\frac{d\theta_t}{dt}$.

Proof. (Theorem 8) To prove the first part of Theorem 8, we apply Theorem 9 with $(N, g^N) = (\Theta, G)$, $M = \mathcal{P}_\Theta$ with its metric inherited from $(\mathcal{P}, g^W)$ and $\varphi = T_\#$. To prove the second part, we apply Theorem 10 with $(M, g^M) = (\mathcal{P}, g^W)$, $M_{\text{sub}} = \mathcal{P}_\Theta$. □

The following theorem is closely related to Theorem 8 and is useful for future discussion:

Theorem 11 (Wasserstein gradient as solution to a least squares problem)**.** *We still use the notations introduced in section 3. For a fixed $\theta \in \Theta$, recall $\boldsymbol{\Psi} \subset \mathbb{R}^m$ as defined in Theorem 5, we have:*

$$G(\theta)^{-1}\nabla_\theta H(\theta) = \underset{\eta \in T_\theta\Theta \cong \mathbb{R}^m}{\arg\min} \left\{ \int |(\nabla \boldsymbol{\Psi}(T_\theta(x)))^T \eta - \nabla\left(V + \beta \log \rho_\theta\right) \circ T_\theta(x)|^2 dp(x) \right\}. \tag{28}$$

Proof. Direct computation shows minimizing the function in (28) is equivalent to minimizing:

$$\eta^T \left(\int \nabla \boldsymbol{\Psi}(T_\theta(x)) \nabla \boldsymbol{\Psi}(T_\theta(x))^T \; dp(x) \right) \eta - 2\; \eta^T \left(\int \nabla \boldsymbol{\Psi}(y) \nabla (V(y) + \beta \log \rho(y)) \rho_\theta(y) \; dy \right),$$

for each entry of the second term, we have:

$$\begin{aligned} &\int \nabla \psi_k(y) \cdot \nabla (V(y) + \beta \log \rho_\theta(y)) \rho_\theta(y) \; dy = \int -\nabla \cdot (\rho_\theta(y) \nabla \psi_k(y)) \cdot (V(y) + \beta \log \rho_\theta(y)) \; dy \\ &= \int -\nabla \cdot (\rho_\theta(y) \partial_{\theta_k} T_\theta(T_\theta^{-1}(y))) \cdot (V(y) + \beta \log \rho_\theta(y)) \; dy \\ &= \partial_{\theta_k} \left(\int (V(T_\theta(x)) + \beta \log \rho_\theta(T_\theta(x))) \; dp(x) \right) = \partial_{\theta_k} H(\theta). \end{aligned}$$

Recall the definition (14) of $G(\theta)$, the target function to be minimized is $\eta^T G(\theta) \eta - 2\eta^T \nabla_\theta H(\theta)$. And the minimizer is clearly $G(\theta)^{-1} \nabla_\theta H(\theta)$. □

Despite this direct proof, Theorem 11 also naturally follows from Theorem 8: denote $\xi = G(\theta)^{-1}\nabla_\theta H(\theta)$, consider $\{\theta_t\}$ starting at $\theta_0 = \theta$ and solves (27). Now by Theorem 8, $\frac{d}{dt}\rho_{\theta_t}\Big|_{t=0} = (T_\#|_\theta)_* \xi \in T_{\rho_\theta}\mathcal{P}_\Theta$ is the orthogonal projection of $\text{grad}_W \mathcal{H}(\rho_\theta)$ onto $T_{\rho_\theta}\mathcal{P}_\Theta$ w.r.t. metric g^W. This is equivalent to that η solves the following least square problem:

$$\min_\eta g^W(\text{grad}_W \mathcal{H}(\rho_\theta) - (T_\#|_\theta)_*\eta, \; \text{grad}_W \mathcal{H}(\rho_\theta) - (T_\#|_\theta)_*\eta). \tag{29}$$

Recall the definition of g^W in section 2.2.1 and by (10), $\text{grad}_W \mathcal{H}(\rho_\theta) = -\nabla \cdot (\rho_\theta \nabla (V + \beta \log \rho_\theta))$; by (18), $(T_\#|_\theta)_*\eta = -\nabla \cdot (\rho_\theta \partial_\theta T_\theta(T_\theta^{-1}(\cdot))\eta)$, solving $-\nabla \cdot (\rho_\theta \nabla \varphi) = \text{grad}_W \mathcal{H}(\rho_\theta) - (T_\#|_\theta)_*\eta$ gives

$$\varphi = (V + \beta \log \rho_\theta) - \boldsymbol{\Psi}^T \eta,$$

and thus least squares problem (29) can be written as

$$\min_\eta \left\{ \int |\nabla \boldsymbol{\Psi}(x)^T \eta - \nabla (V(x) + \beta \log \rho_\theta(x))|^2 \rho_\theta(x) \; dx \right\},$$

which is exactly (28).

3.3 A particle point of view of the parametric Fokker Planck Equation

The motion of parameter θ_t solving (27) will naturally induce a stochastic dynamics on $\mathbb{R}^d$ whose density evolution is exactly $\{\rho_{\theta_t}\}$. To see this, notice that $\{\theta_t\}$ directly leads to a time dependent map $\{T_{\theta_t}\}$. Now we have a random variable $\boldsymbol{Z} \sim p$, i.e. $\boldsymbol{Z}$ is distributed according to the reference distribution p. Then set $\boldsymbol{Y}_0 = T_{\theta_0}(\boldsymbol{Z}) \sim \rho_{\theta_0}$. Now at any time t, the map T_{θ_t} will send $\boldsymbol{Y}_0$ to $\boldsymbol{Y}_t = T_{\theta_t}(T_{\theta_0}^{-1}(\boldsymbol{Y}_0)) \sim \rho_{\theta_t}$. Thus, we constructed a sequence of random variables $\{\boldsymbol{Y}_t\}$ whose density evolution is exactly $\{\rho_{\theta_t}\}$. We can characterize the dynamical system satisfied by $\{\boldsymbol{Y}_t\}$ by taking time derivative: $\dot{\boldsymbol{Y}}_t = \partial_\theta T_{\theta_t}(\boldsymbol{Z})\dot{\theta}_t = \partial_\theta T_{\theta_t}(T_{\theta_t}^{-1}(\boldsymbol{Y}_t))\dot{\theta}_t$. It is actually more insightful to consider the following dynamic:

$$\dot{\boldsymbol{X}}_t = \nabla\boldsymbol{\Psi}_t(\boldsymbol{X}_t)^T\ \dot{\theta}_t, \quad \boldsymbol{X}_0 = T_{\theta_0}(\boldsymbol{Z}) \sim \rho_{\theta_0}. \tag{30}$$

Here $\boldsymbol{\Psi}_t$ is obtained from (15) with parameter θ_t. Based on (15), it is not hard to show that for any time t, $\boldsymbol{X}_t$ and $\boldsymbol{Y}_t$ has the same distribution. Thus $\boldsymbol{X}_t \sim \rho_{\theta_t}$ for all $t \geq 0$. Now recall $\dot{\theta}_t = -G(\theta_t)^{-1}\nabla_\theta H(\theta_t)$, we are able to rewrite (30) as:

$$\dot{\boldsymbol{X}}_t = \nabla\boldsymbol{\Psi}_t(\boldsymbol{X}_t)^T \underbrace{\left(\int \nabla\boldsymbol{\Psi}_t(x)\nabla\boldsymbol{\Psi}_t(x)^T\ \rho_{\theta_t}(x)\ dx\right)}_{G(\theta_t)}{}^{-1} \underbrace{\left(\int \nabla\boldsymbol{\Psi}_t(\eta)(-\nabla V(\eta) - \beta\nabla\log\rho_{\theta_t}(\eta))\ \rho_{\theta_t}(\eta)\ d\eta\right)}_{-\nabla_\theta H(\theta_t)}. \tag{31}$$

If we define the kernel function $K_\theta : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}^{d\times d}$ as

$$K_\theta(x,\eta) = \nabla\boldsymbol{\Psi}^T(x)\left(\int \nabla\boldsymbol{\Psi}(x)\nabla\boldsymbol{\Psi}(x)^T\ \rho_\theta(x)\ dx\right)^{-1}\nabla\boldsymbol{\Psi}(\eta).$$

This K_θ will induce a linear operator $\mathcal{K}_\theta : L^2(\mathbb{R}^d;\mathbb{R}^d,\rho_\theta) \to L^2(\mathbb{R}^d;\mathbb{R}^d,\rho_\theta)$ by:

$$\mathcal{K}_\theta[\vec{v}] = (\mathcal{K}_\theta * \vec{v})(\cdot) = \int K_\theta(\cdot,\eta)\ \vec{v}(\eta)\ \rho_\theta(\eta)\ d\eta.$$

It can be verified that $\mathcal{K}_\theta$ is an orthogonal projection defined on the Hilbert space $L^2(\mathbb{R}^d;\mathbb{R}^d,\rho_\theta)$. The range of such projection is the subspace span$\{\nabla\psi_1,...,\nabla\psi_m\} \subset L^2(\mathbb{R}^d;\mathbb{R}^d,\rho_\theta)$. Here $\psi_1,...,\psi_m$ are the m components of $\boldsymbol{\Psi}$ solved from (15). Now (31) can also be written as:

$$\dot{\boldsymbol{X}}_t = -\mathcal{K}_{\theta_t}[\nabla V + \beta\nabla\log\rho_{\theta_t}](\boldsymbol{X}_t), \quad \text{where } \rho_{\theta_t} \text{ is the probability density of } \boldsymbol{X}_t \quad \boldsymbol{X}_0 \sim \rho_{\theta_0}. \tag{32}$$

We can compare (32) with the following dynamic without projection:

$$\dot{\tilde{\boldsymbol{X}}}_t = -(\nabla V + \beta\nabla\log\rho_t)(\tilde{\boldsymbol{X}}_t), \quad \text{where } \rho_t \text{ is the probability density of } \tilde{\boldsymbol{X}}_t \quad \boldsymbol{X}_0 \sim \rho_0. \tag{33}$$

Recall section 2.1, (33) is the Vlasov-type SDE that involves the density of random particle, if we assume (33) admits a regular solution, then $\rho(x,t) = \rho_t(x)$ solves the original Fokker Planck equation (5). Now it is clear that the approximate solution ρ_{θ_t} of (5) is actually originated from the projection of vector field that drives the SDE (33).

The expectation of ℓ^2 discrepancy between $\nabla V + \beta\nabla\log\rho$ and its $\mathcal{K}_\theta$ projection is:

$$\mathbb{E}_{\boldsymbol{X}\sim\rho_\theta}|\mathcal{K}_\theta[\nabla V + \beta\nabla\log\rho_\theta](\boldsymbol{X}) - (\nabla V + \beta\nabla\log\rho_\theta)(\boldsymbol{X})|^2 = \int |\nabla\boldsymbol{\Psi}(x)^T\xi - (-\nabla V - \beta\nabla\log\rho_\theta)(x)|^2\rho_\theta(x)\ dx. \tag{34}$$

here $\xi = -G(\theta)^{-1}\nabla_\theta H(\theta)$. This is an essential error term appeared in later error analysis part.

Remark 2. *Figure 1 illustrates the relation between (5), (27), (33) and (32). It worth mentioning that the probability manifold point of view discussed in Theorem 8 will be useful for numerical analysis of continuous scheme (27), while particle point of view helps us on establishing numerical analysis for discrete scheme (i.e. forward-Euler) of (27).*

3.4 An example of parametric Fokker-Planck equation with quadratic potential

The solution of Fokker-Planck equation on statistical manifold (27) can serve as an approximation to the solution of the original equation (5). However, in some special cases, ρ_{θ_t} exactly solves (5). In this section, we demonstrate such examples.

Let us consider Fokker-Planck equations with quadratic potentials whose initial conditions are Gaussian:

$$V(x) = \frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu) \quad \text{and} \quad \rho_0 \sim \mathcal{N}(\mu_0,\Sigma_0). \tag{35}$$

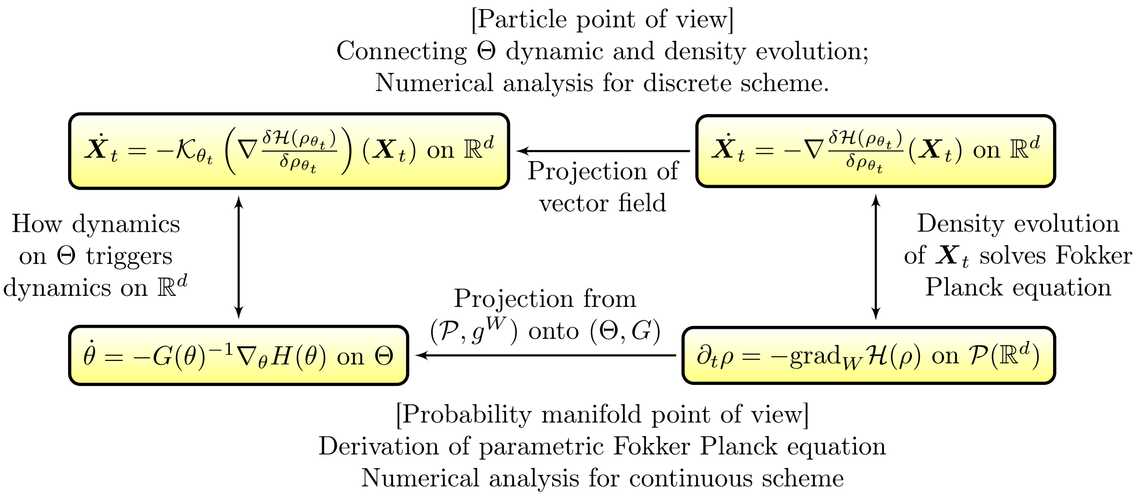

[Particle point of view]
Connecting Θ dynamic and density evolution;
Numerical analysis for discrete scheme.
$\dot{\boldsymbol{X}}_t = -\mathcal{K}_{\theta_t}\left(\nabla \frac{\delta \mathcal{H}(\rho_{\theta_t})}{\delta \rho_{\theta_t}}\right)(\boldsymbol{X}_t)$ on $\mathbb{R}^d$
Projection of vector field
$\dot{\boldsymbol{X}}_t = -\nabla \frac{\delta \mathcal{H}(\rho_{\theta_t})}{\delta \rho_{\theta_t}}(\boldsymbol{X}_t)$ on $\mathbb{R}^d$
How dynamics on Θ triggers dynamics on $\mathbb{R}^d$
Density evolution of $\boldsymbol{X}_t$ solves Fokker Planck equation
$\dot{\theta} = -G(\theta)^{-1}\nabla_\theta H(\theta)$ on Θ
Projection from $(\mathcal{P}, g^W)$ onto (Θ, G)
$\partial_t \rho = -\mathrm{grad}_W \mathcal{H}(\rho)$ on $\mathcal{P}(\mathbb{R}^d)$
[Probability manifold point of view]
Derivation of parametric Fokker Planck equation
Numerical analysis for continuous scheme

Figure 1: Illustrative diagram

Here $\mathcal{N}(\mu, \Sigma)$ denotes Gaussian distribution with mean μ and covariance Σ. We consider parameter space $\Theta = (\Gamma, b) \subset \mathbb{R}^m$ $(m = d(d+1))$, where Γ is a $d \times d$ invertible matrix with $\det(\Gamma) > 0$ and $b \in \mathbb{R}^d$. We define the parametric map as $T_\theta(x) = \Gamma x + b$. We choose the reference measure $p = \mathcal{N}(0, I)$. Here is the lemma we have to use:

Lemma 12. *Let $\mathcal{H}$ be the relative entropy defined in (11) and H defined in (26). For $\theta \in \Theta$, If the vector function $\nabla\left(\frac{\delta \mathcal{H}}{\delta \rho}\right) \circ T_\theta$ can be written as the linear combination of $\{\frac{\partial T_\theta}{\partial \theta_1}, ..., \frac{\partial T_\theta}{\partial \theta_m}\}$, i.e. there exists $\zeta \in \mathbb{R}^m$, such that $\nabla\left(\frac{\delta \mathcal{H}}{\delta \rho}\right) \circ T_\theta(x) = \partial_\theta T_\theta(x)\zeta$. Then:*
1) $\zeta = G(\theta)^{-1}\nabla_\theta H(\theta)$, which is the Wasserstein gradient of F at θ.
2) Recall that the Wasserstein gradient of $\mathcal{H}$ is $\mathrm{grad}_W \mathcal{H}(\rho_\theta)$ and we denote the gradient of $\mathcal{H}$ on the sub-manifold $\mathcal{P}_\Theta$ as $\mathrm{grad}_W \mathcal{H}(\rho_\theta)|_{\mathcal{P}_\Theta}$, then $\mathrm{grad}_W \mathcal{H}(\rho_\theta)|_{\mathcal{P}_\Theta} = \mathrm{grad}_W \mathcal{H}(\rho_\theta)$.

Proof. Suppose $\zeta \in \mathbb{R}^m$ satisfies $\nabla\left(\frac{\delta \mathcal{H}}{\delta \rho}\right) \circ T_\theta(x) = \partial_\theta T_\theta(x)\zeta$, then we have

$$\int |\partial_\theta T_\theta(x)\zeta - \nabla(\frac{\delta \mathcal{H}}{\delta \rho}) \circ T_\theta(x)|^2 \, dp(x) = 0.$$

We need to apply Lemma 15 mentioned in 4.2.2 here. Use the notation in (15) and notice that

$$(\nabla \boldsymbol{\Psi})^T \zeta - \nabla\left(\frac{\delta \mathcal{H}}{\delta \rho}\right) = \mathrm{Proj}_{\rho_\theta}[\partial_\theta T_\theta \circ T_\theta^{-1}\zeta - \nabla\left(\frac{\delta \mathcal{H}}{\delta \rho}\right)],$$

we know:

$$\int |(\nabla \boldsymbol{\Psi}(T_\theta(x)))^T \zeta - \nabla\left(\frac{\delta \mathcal{H}}{\delta \rho}\right) \circ T_\theta(x)|^2 \, dp(x) \leq 0.$$

As a result,

$$\inf_\eta \int |(\nabla \boldsymbol{\Psi}(T_\theta(x)))^T \eta - \nabla\left(\frac{\delta \mathcal{H}}{\delta \rho}\right) \circ T_\theta(x)|^2 \, dp(x) = \int |(\nabla \boldsymbol{\Psi}(T_\theta(x)))^T \zeta - \nabla\left(\frac{\delta \mathcal{H}}{\delta \rho}\right) \circ T_\theta(x)|^2 \, dp(x) = 0.$$

Now by Theorem 11, we get $\zeta = G(\theta)^{-1}\nabla_\theta H(\theta)$ and: $\|(T_\#|_\theta)_*\zeta - \mathrm{grad}_W \mathcal{H}(\rho_\theta)\|_{g^W(\rho_\theta)} = 0$. According to Theorem 8, $\mathrm{grad}_W \mathcal{H}(\rho_\theta)|_{\mathcal{P}_\Theta} = (T_\#|_\theta)_*\zeta$. Thus we have $\mathrm{grad}_W \mathcal{H}(\rho_\theta)|_{\mathcal{P}_\Theta} = \mathrm{grad}_W \mathcal{H}(\rho_\theta)$. □

Return to our example, we can compute

$$\rho_\theta(x) = T_{\theta\#}p(x) = \frac{f(T_\theta^{-1}(x))}{|\det(\Gamma)|} = \frac{f(\Gamma^{-1}(x-b))}{|\det(\Gamma)|}, \quad f(x) = \frac{\exp(-\frac{1}{2}|x|^2)}{(2\pi)^{\frac{d}{2}}}.$$

Then we have:

$$\nabla\left(\frac{\delta\mathcal{H}(\rho_\theta)}{\delta\rho}\right)\circ T_\theta(x)=\nabla(V+\beta\log\rho_\theta)\circ T_\theta(x)=\Sigma^{-1}(\Gamma x+b-\mu)-\beta\Gamma^{-T}x$$

is affine w.r.t. x.

Notice that $\partial_{\Gamma_{ij}}T_\theta(x)=(\dots,0,\dots,\underset{i-\text{th}}{x_j},\dots,0,\dots)^T$ and $\partial_{b_i}T_\theta=(\dots,0,\dots,\underset{i-\text{th}}{1},\dots,0,\dots)^T$. We can verify that $\zeta=(\Sigma^{-1}\Gamma-\beta\Gamma^{-T},\Sigma^{-1}(b-\mu))$ solves $\nabla\left(\frac{\delta\mathcal{F}(\rho_\theta)}{\delta\rho}\right)\circ T_\theta(x)=\partial_\theta T_\theta(x)\zeta$. By 1) of Lemma 12, $\zeta=G(\theta)^{-1}\nabla_\theta F(\theta)$. Thus ODE (27) for our example is:

$$\dot{\Gamma}=-\Sigma^{-1}\Gamma+\beta\Gamma^{-T}\quad\Gamma_0=\sqrt{\Sigma_0},\tag{36}$$

$$\dot{b}=\Sigma^{-1}(\mu-b)\quad b_0=\mu_0.\tag{37}$$

By 2) of Lemma 12, we know $\mathrm{grad}_{\mathrm{W}}\mathcal{H}(\rho_\theta)|_{\mathcal{P}_\Theta}=\mathrm{grad}_{\mathrm{W}}\mathcal{H}(\rho_\theta)$ for all $\theta\in\Theta$. This indicates that there is no local error for our scheme, one can verify that the solution to the parametric Fokker-Planck equation also solves the original equation.

In addition to the previous results, we have the following corollary:

Corollary 12.1. *The solution of Fokker-Planck equation (5) with condition(35) is Gaussian distribution for all $t>0$.*

Proof. If we denote $\{\Gamma_t,b_t\}$ as the solutions to (36),(37), set $\theta_t=(\Gamma_t,b_t)$, then $\rho_t=T_{\theta_t\#}p$ solves the Fokker Planck Equation (5) with conditions (35). Since the pushforward of Gaussian distribution p by an affine transform T_θ is still a Gaussian, we conclude that for any $t>0$, the solution $\rho_t=T_{\theta_t\#}p$ is always Gaussian distribution. □

Remark 3. *This is already a well known property for Ornstein–Uhlenbeck process [11]. We give an alternative proof using our framework.*

4 Numerical methods

In this section, we introduce the sampling efficient numerical method for computing the proposed parametric Fokker-Planck equations.

When dimension $d=1$, according to Corollary 5.1, $G(\theta)$ has explicit solution. Thus, push-forward approximation of 1D Fokker-Planck equation can be directly computed by solving the ODE system (27) with forward-Euler scheme [25]. In this section, we will mainly focus on numerical methods for (27) with dimension $d\geq 2$.

When dimension $d\geq 2$, we are unable to compute (27) via a forward-Euler scheme directly. There are mainly two reasons:

- When $d\geq 2$, as shown in (14), $G(\theta)$ doesn't have an explicit formula, directly compute it could be very expensive;
- When dimension d gets higher, to ensure our efficient, we choose to implement it using deep neural networks. However, $G(\theta)$ is generally a dense matrix. multiplying its inverse to $\nabla_\theta H(\theta)$ cannot be computed efficiently using deep neural networks.

Although there are some efficient approximation methods for Fisher natural gradient [31], whether there are efficient ways to compute Wasserstein natural gradeint $G(\theta)^{-1}\nabla_\theta H(\theta)$ remains an open problem. As a result, in order to solve (27), we need to seek for alternative schemes other than forward-Euler. It is worth mentioning that the JKO scheme [17] for numerically computing Wasserstein gradient flows [9]:

$$\partial_t\rho_t=-\mathrm{grad}_W\mathcal{F}(\rho_t)\quad\Longleftrightarrow\quad\rho_{k+1}=\underset{\rho\in\mathcal{P}}{\mathrm{argmin}}\left\{\frac{W_2^2(\rho,\rho_k)}{2h}+\mathcal{F}(\rho)\right\}.\tag{38}$$

Here h is the time step size, $\mathcal{F}$ could be a suitable functional defined on $\mathcal{P}$.

Some related work has already been done in [24]. Based on (38), the authors mainly invented two schemes, one can be treated as a scheme for solving: $\dot{\theta} = -\hat{G}^{-1}(\theta)\nabla_\theta F(\theta)$ with a simplified Wasserstein metric tensor $\hat{G}(\theta) = \int_M \frac{\partial T_\theta(x)}{\partial \theta}^T \frac{\partial T_\theta(x)}{\partial \theta}\, dp(x)$; Another scheme approximates the Wasserstein distance $W_2^2(\rho_\theta, \rho_{\theta'})$ by solving a variational problem restricted to a finite dimensional vector space with chosen basis functions. Both schemes are not computing for the exact Wasserstein gradient flow since they either simplify the metric tensor $G(\theta)$ or restrict the computation on low dimensional space in order to acquire a tractable algorithm. In our research, we try to directly tackle with the computation of the exact Wasserstein gradient flow. We will design schemes with accuracy guarantee and develope algorithm that is able to run efficiently under deep learnign framework.

In 4.1, we introduce a typical parametrized pushforward map called Normalizing Flow, which has been proved to be an efficient tool for distribution approximation. We will use it as our computational tool in this project; In 4.2, we exhibit the derivation of our numerical scheme and provide local error analysis between our scheme and the semi-implicit scheme for the parametric Fokker-Planck equation; complete algorithm and details of implementation are also provided.

4.1 Normalizing Flow as push forward maps

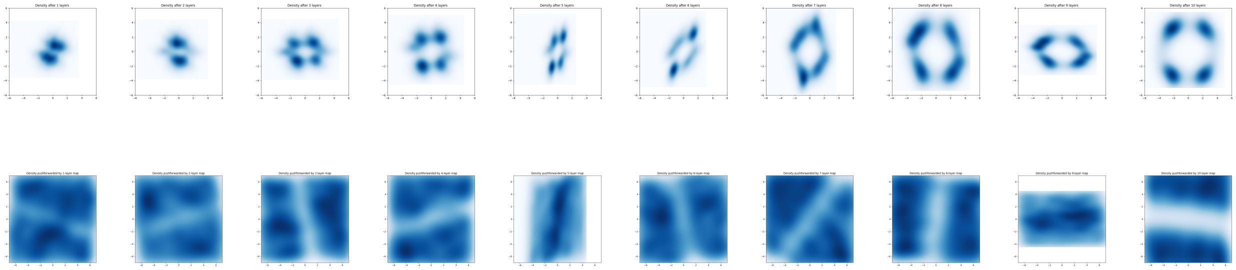
To this end, we choose T_θ as the so-called normalizing flow [42]. Here is a brief sketch of T_θ's structure: T_θ is written as the composition of K invertible nonlinear transforms:

$$T_\theta = f_K \circ f_{K-1} \circ ... \circ f_2 \circ f_1.$$

Where each f_k $(1 \leq k \leq K)$ takes the form

$$f_k(x) = x + h(w_k^T x + b_k)u_k.$$

where $w_k, u_k \in M$, $b_k \in \mathbb{R}$. And h is a nonlinear function, one can choose it as tanh, for example. In [42], it has been shown that f_k is invertible iff $w_k^T u_k \geq -1$. The following shows several examples of how a normalizing flow T_θ with length equal to 10 pushes forward standard Gaussian distribution to a certain distribution:



Among these series of images, the first row displays (from left to right) the probability density of distributions $f_{1\#}p, (f_2 \circ f_1)_{\#}p, ..., (f_{10} \circ f_9 \circ ... \circ f_1)_{\#}p$, the last image displays our target distribution; the second row exhibits the push-forward effect of each single-layer transformation f_k $(1 \leq k \leq 10)$. i.e. the images (from left to right) display the density of distributions $f_{1\#}m, f_{2\#}m, ..., f_{10\#}m$, here m represents the uniform distribution defined on the square.

Using normalizing flow, the parameters are: $\theta = (w_1, u_1, b_1, ..., w_K, u_K, b_K)$. The determinant of the Jacobi matrix of T_θ can be explicitly computed as:

$$\mathbf{det}\left(\frac{\partial T_\theta(x)}{\partial x}\right) = \prod_{k=1}^{K}(1 + h'(w_k^T x_k + b_k)w_k^T u_k).$$

Here $x_k = f_k \circ f_{k-1} \circ ... \circ f_1(x)$. Thus the logarithm of the density ρ_θ of $T_{\theta\#}p$ can be written as

$$\log \rho_\theta(x) = \log p \circ t_\theta^{-1}(x) + \sum_{k=1}^{K} \log(1 + h'(w_k^T \tilde{x}_k)w_k^T u_k) \quad \text{Here } \tilde{x}_k = f_k \circ ... \circ f_1(T_\theta^{-1}(x)) = f_{k+1}^{-1} \circ ... \circ f_K^{-1}(x). \tag{39}$$

Thus we can explicitly write the relative entropy functional $H(\theta)$ defined in (26) as:

$$H(\theta) = \mathbb{E}_{\mathbf{X}\sim p}[V(T_\theta(\mathbf{X})) + \mathcal{L}_\theta(\mathbf{X})]. \tag{40}$$

Here $\mathcal{L}_\theta$ is defined as:

$$\mathcal{L}_\theta(\cdot) = \log p(\cdot) + \sum_{k=1}^{K} \log(1 + h'(w_k^T F_k(\cdot)) w_k^T u_k) \quad F_k(\cdot) = f_k \circ f_{k-1} \circ ... \circ f_1(\cdot).$$

Once $H(\theta)$ can be explicitly computed, the gradient $\nabla_\theta H(\theta)$ can also be explicitly computed. Here we summarize the main advantages of normalizing flows:

- As shown in [42], normalizing flow has sufficient expression power to approximate complicated distributions on $\mathbb{R}^d$.
- Due to the special structure of normalizing flow, relative entropy $H(\theta)$ will have a very concise form (40). Then the gradient of $H(\theta)$ can be conveniently computed.

Remark 4. *We should emphasize here that the normalizing flow is not the only choice for T_θ, any other choices satisfying the two advantages mentioned above may serve as a candidate for T_θ. Our proposed Algorithm 1 works generally for other class of T_θ as well.*

4.2 Numerical scheme

4.2.1 Derivation

We consider the semi-implicit scheme of (27):

$$\frac{\theta_{k+1} - \theta_k}{h} = -G^{-1}(\theta_k)\nabla_\theta H(\theta_{k+1}).$$

There is a natural proximal-type algorithm that computes for θ_{k+1}:

$$\theta_{k+1} = \operatorname*{argmin}_\theta \left\{ \langle \theta - \theta_k, G(\theta_k)(\theta - \theta_k) \rangle + 2hH(\theta) \right\}. \tag{41}$$

The main difficulty of (41) is the computation of the first term. To derive an efficient method to compute $\langle \theta - \theta_k, G(\theta_k)(\theta - \theta_k) \rangle$, let us recall the definition (14) of $G(\theta_k)$, if we set $\psi(x) = (\theta - \theta_k)^T \mathbf{\Psi}(x)$, then $\int |\nabla\psi(x)|^2 \rho_{\theta_k}(x)\, dx = \langle \theta - \theta_k, G(\theta_k)(\theta - \theta_k) \rangle$. We know ψ satisfies

$$-\nabla \cdot (\rho_{\theta_k}(x)\nabla\psi(x)) = -\nabla \cdot (\rho_{\theta_k}(x)\partial_\theta T_{\theta_k}(T_{\theta_k}^{-1}(x))(\theta - \theta_k)). \tag{42}$$

We replace $\partial_\theta T_{\theta_k}(T_{\theta_k}^{-1}(x))(\theta - \theta_k)$ by finite difference approximation $(T_\theta - T_{\theta_k}) \circ T_{\theta_k}^{-1}(x)$ and denote $\hat{\psi}$ as the solution of (42) after this replacement. Furthermore, let

$$\mathcal{E}(\phi) = \int (2\nabla\phi(x) \cdot ((T_\theta - T_{\theta_k}) \circ T_{\theta_k}^{-1}(x)) - |\nabla\phi(x)|^2)\rho_{\theta_k}(x)\, dx. \tag{43}$$

Then we can verify that $\hat{\psi}$ solves the variational problem: $\hat{\psi} = \operatorname*{argmax}_\phi \mathcal{E}(\phi)$ with maximum value

$$\max_\phi \mathcal{E}(\phi) = \int |\nabla\hat{\psi}(x)|^2 \rho_{\theta_k}(x)\, dx. \tag{44}$$

If $\hat{\psi}$ is a valid approximation of ψ, then $\max_\phi \mathcal{E}(\phi)$ will be an approximation of $\langle \theta - \theta_k, G(\theta_k)(\theta - \theta_k) \rangle$.

Now replace $\langle \theta - \theta_k, G(\theta_k)(\theta - \theta_k) \rangle$ in (41) by $\max_\phi \mathcal{E}(\phi)$ we derived our numerical scheme for solving (27):

$$\theta_{k+1} = \operatorname*{argmin}_\theta \max_\phi \left\{ \int 2\nabla\phi(x) \cdot ((T_\theta - T_{\theta_k}) \circ T_{\theta_k}^{-1}(x))\rho_{\theta_k}(x)\, dx - \int |\nabla\phi(x)|^2 \rho_{\theta_k}(x)\, dx + 2hH(\theta) \right\}. \tag{45}$$

Remark 5. *Our proposed scheme can actually be treated as an approximation to the JKO scheme* (38)

$$\theta_{k+1} = \arg\min_{\theta} \left\{ \frac{W_2^2(\rho_\theta, \rho_{\theta_k})}{2h} + H(\theta) \right\} = \arg\min_{\theta} \left\{ W_2^2(\rho_\theta, \rho_{\theta_k}) + 2hH(\theta) \right\} \tag{46}$$

To see why let us denote $\vec{v}_h(x) = \frac{T_\theta \circ T_{\theta_k}^{-1}(x) - x}{h}$, *under mild conditions, one can verify that*

$$W_2^2(\rho_\theta, \rho_{\theta_k}) = W_2^2((Id + h\vec{v}_h)_{\#}\rho_{\theta_k}, \rho_{\theta_k}) = \int |\nabla\hat{\psi}|^2 \rho_{\theta_k} \; dx + o(h^2) = \max_{\phi} \mathcal{E}(\phi) + o(h^2). \tag{47}$$

If we replace $W_2^2(\rho_\theta, \rho_{\theta_k})$ *in* (46) *by its approximation* $\max_\phi \mathcal{E}(\phi)$, *we will obtain our proposed* (38).

Remark 6. *It is worth mentioning that the variational problem* $\max_\phi \mathcal{E}(\phi)$ *is equivalent to:*

$$\min_{\phi} \left\{ \int_M |\nabla\phi(x) - ((T_\theta - T_{\theta_k}) \circ T_{\theta_k}^{-1}(x))|^2 \rho_{\theta_k}(x) \; dx \right\}. \tag{48}$$

Then the gradient field $\nabla\hat{\psi}$ *from the optimal* $\hat{\psi}$ *can be treated as the* $L^2(\rho_{\theta_k})$ *orthogonal projection of the vector field* $(T_\theta - T_{\theta_k}) \circ T_{\theta_k}^{-1}(\cdot)$ *onto the subspace of gradient fields.*

4.2.2 Local error of the proposed scheme

We are now in a position to analyze the local error of scheme (45) compared with the semi-implicit scheme (41), or equivalently:

$$\theta_{k+1} \quad \text{solves} \quad \theta_{k+1} = \theta_k - hG^{-1}(\theta_k)\nabla_\theta H(\theta_{k+1}).$$

Let us denote $\max_\phi \mathcal{E}(\phi)$ as $\widehat{W}_2^2(\theta, \theta_k)$ (Here $\widehat{W}_2$ is treated as an approximation of 2-Wasserstein distance (remark 5)). It is straightforward to verify $\widehat{W}_2(\theta, \theta') \geq 0$ and $\widehat{W}_2(\theta, \theta) = 0$. Consider the following **assumption**:

$$\widehat{W}_2^2(\theta, \theta') \geq l(|\theta - \theta'|) \qquad \text{for any} \quad \theta, \theta' \in \Theta. \tag{49}$$

Here $l : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ satisfies $l(0) = 0$. $l(r)$ is continuous, strictly increasing when $r \leq r_0$ and is bounded below by $\lambda_0 > 0$ when $r > r_0$. Notice that this assumption generally guarantees positive definiteness of $\widehat{W}_2$. Clearly, (49) only depends on the structure of T_θ, we should expect that (49) holds for most kinds of neural networks used as pushforward maps.

We have the following result:

Theorem 13. *Suppose* ***assumption*** (49) *holds true for the class of push-frward maps* $\{T_\theta\}$. *Then the local error of scheme* (45) *is of order* h^2, *i.e., assume that* θ_{k+1} *is the optimal solution to* (45), *then*

$$|\theta_{k+1} - \theta_k + hG(\theta_k)^{-1}\nabla_\theta H(\theta_{k+1})| \sim O(h^2). \tag{50}$$

or equivalently: $\limsup_{h\to 0^+} \frac{|\theta_{k+1} - \theta_k + hG(\theta_k)^{-1}\nabla_\theta H(\theta_{k+1})|}{h^2} < +\infty$.

To prove this theorem, we need the following lemmas:

Lemma 14. *[Danskin's Theorem [6]] Suppose* $F : \mathbb{R}^m \times \mathcal{B} \to \mathbb{R}$, *here* $\mathcal{B}$ *is a Banach space. Suppose for any* $\xi \in \mathcal{B}$, $F(\cdot, \xi)$ *is smooth; also assume that for any* $x \in \mathbb{R}^m$, *there is unique* $\xi_x \in \mathcal{B}$ *such that* $F(x, \xi_x) = \sup_{\xi\in\mathcal{B}} F(x, \xi)$. *Now denote:* $\Gamma(x) = \sup_{\xi\in\mathcal{B}} F(x, \xi)$. *Then* Γ *is differentialbe on* $\mathbb{R}^m$ *and its derivative can be computed as:*

$$\nabla\Gamma(x) = \partial_x F(x, \xi_x).$$

We now introduce a shorthand notation: for $\vec{v} \in L^2(\mathbb{R}^d; \mathbb{R}^d, \rho)$, $\text{Proj}_\rho[\vec{v}] = \nabla\psi$ as $L^2(\rho)$-orthogonal projection of $\vec{v}$ onto the subspace of gradient fields, i.e. $\psi = \underset{\psi}{\text{argmin}} \left\{ \int |\vec{v}(x) - \nabla\psi(x)|^2 \rho(x) \; dx \right\}$, or equivalently, ψ solves $-\nabla \cdot (\rho(x)\nabla\psi(x)) = -\nabla \cdot (\rho(x)\vec{v}(x))$.

Lemma 15. *Suppose $\vec{u}, \vec{v}$ are two vector fields on $M = \mathbb{R}^d$, denote $Proj_\rho[\vec{u}] = \nabla\varphi$ and $Proj_\rho[\vec{v}] = \nabla\psi$. Then:*

$$\int \vec{u}(x) \cdot \nabla\psi(x)\rho(x)\ dx = \int \nabla\varphi(x) \cdot \nabla\psi(x)\rho(x)\ dx; \tag{51}$$

$$\int |\nabla\psi(x)|^2\rho(x)\ dx \leq \int |\vec{v}(x)|^2\rho(x)\ dx. \tag{52}$$

Proof. For (51):

$$\int \vec{u}(x)\cdot\nabla\psi(x)\rho(x)\ dx = \int -\nabla\cdot(\rho(x)\vec{u}(x))\psi(x)\ dx = \int -\nabla\cdot(\rho(x)\nabla\varphi(x))\psi(x)\ dx = \int \nabla\varphi(x)\cdot\nabla\psi(x)\rho(x)\ dx.$$

For (52):

$$\begin{aligned}\int |\vec{v}(x)|^2\rho(x)\ dx &= \int (|\nabla\psi(x)|^2 + 2(\vec{v}(x) - \nabla\psi(x)) \cdot \nabla\psi(x) + |\vec{v}(x) - \nabla\psi(x)|^2)\rho(x)\ dx \\ &= \int |\nabla\psi(x)|^2 + |\vec{v}(x) - \nabla\psi(x)|^2)\rho(x)\ dx \geq \int |\nabla\psi(x)|^2\rho(x)\ dx.\end{aligned}$$

The second equality is due to (51). □

The following lemma gives a prior estimation of $|\theta_{k+1} - \theta_k|$:

Lemma 16. *Under assumption(49), recall θ_{k+1} is the optimal solution of (45), which depends on time step size h then*

$$|\theta_{k+1} - \theta_k| \sim o(1) \quad i.e. \quad \lim_{h\to 0^+} |\theta_{k+1} - \theta_k| = 0. \tag{53}$$

Proof. Denote the function to be minimized in (45) as $J(\theta) = \widehat{W}(\theta, \theta_k) + 2hH(\theta)$. First, we choose $\theta = \theta_k$ in (45), then $J(\theta_k) = 2hH(\theta_k)$. Thus $J(\theta_{k+1}) \leq J(\theta_k) = 2hH(\theta_k)$. Since $H(\theta_k) \geq 0$, this leads to $\widehat{W}_2^2(\theta_{k+1}, \theta_k) \leq 2hH(\theta_k)$. When h is small enough, $|\theta_{k+1} - \theta_k| \leq k(2hH(\theta_k))$, here k is the inverse function of l defined on $[0, l(r_0)]$. We know $k(0) = 0$ and k is also continuous and increasing function. This leads to $\lim_{h\to 0^+} |\theta_{k+1} - \theta_k| \leq \lim_{h\to 0^+} k(2hH(\theta_k)) = 0$. □

Before proving Theorem 13, we introduce some additonal notations: we define ϵ ball in parameter space as $B_\epsilon(\theta_k) = \{\theta \mid |\theta - \theta_k| \leq \epsilon\}$; Let $T_\theta^{(i)}$ be the i-th component $(1 \leq i \leq d)$ of map T_θ. We denote:

$$L(\theta_k, \epsilon) = \sum_{i=1}^d \mathbb{E}_{x\sim p} \sup_{\theta\in B_\epsilon(\theta_k)} \left\{|\partial_\theta T_\theta^{(i)}(x)|^2\right\}, \quad H(\theta_k, \epsilon) = \sum_{i=1}^d \mathbb{E}_{x\sim p} \sup_{\theta\in B_\epsilon(\theta_k)} \left\{\|\partial^2_{\theta\theta} T_\theta^{(i)}(x)\|_2^2\right\}. \tag{54}$$

Proof of Theorem 13. We denote

$$F(\theta, \phi) = \int (2\nabla\phi(x) \cdot (T_\theta - T_{\theta_k}) \circ T_{\theta_k}^{-1}(x) - |\nabla\phi(x)|^2)\ \rho_{\theta_k}(x)\ dx + 2hH(\theta).$$

As discussed before, $\hat{\psi}_\theta = \underset{\phi}{\text{argmax}}\{F(\theta, \phi)\}$ solves

$$-\nabla \cdot (\rho_{\theta_k}(x)\nabla\hat{\psi}_\theta(x)) = -\nabla \cdot (\rho_{\theta_k}(x)(T_\theta - T_{\theta_k}) \circ T_{\theta_k}^{-1}(x)).$$

We write

$$\nabla\hat{\psi}_\theta = \text{Proj}_{\rho_{\theta_k}}[(T_\theta - T_{\theta_k}) \circ T_{\theta_k}^{-1}].$$

Now denote $\Gamma(\theta) = \sup_\phi F(\theta, \phi)$, apply Lemma 14, we can compute:

$$\nabla_\theta\Gamma(\theta) = 2\left(\int\ \partial_\theta T_\theta(T_{\theta_k}^{-1}(x))\ \nabla\hat{\psi}_\theta(x)\ \rho_{\theta_k}(x)\ dx + h\ \nabla_\theta H(\theta)\right).$$

Due to the differentiability of $\Gamma(\theta)$, at the optimizer θ_{k+1}, the gradient must vanish, i.e.

$$\int \partial_\theta T_{\theta_{k+1}}(T_{\theta_k}^{-1}(x))\ \nabla\hat{\psi}_{\theta_{k+1}}(x)\ \rho_{\theta_k}(x)\ dx + h\nabla_\theta H(\theta_{k+1}) = 0. \tag{55}$$

We can Taylor expand at θ_{k+1}: $T_{\theta_{k+1}} - T_{\theta_k} = \partial_\theta T_{\theta_k}(\theta_{k+1}-\theta_k) + R(\theta_{k+1},\theta_k)$, here $R(\theta,\theta')(\cdot) \in L^2(\mathbb{R}^d;\mathbb{R}^m,\rho_{\theta_k})$, the i-th entry of $R(\theta,\theta')$ is $R_i(\theta,\theta')(x) = \frac{1}{2}(\theta-\theta')^T\partial^2_{\theta\theta}T^{(i)}_{\tilde{\theta}_i(x)}(x)(\theta-\theta'),\ 1\le i\le m$, where each $\tilde{\theta}_i(x) = \lambda_i(x)\theta + (1-\lambda_i(x))\theta'$ for some $\lambda_i(x)\in[0,1]$. Then we can write:

$$\nabla\hat{\psi}_{\theta_{k+1}} = \mathrm{Proj}_{\rho_{\theta_k}}[(T_{\theta_{k+1}} - T_{\theta_k})\circ T_{\theta_k}^{-1}] = \mathrm{Proj}_{\rho_{\theta_k}}[\partial_\theta T_{\theta_k}\circ T_{\theta_k}^{-1}(\theta_{k+1}-\theta_k)] + \mathrm{Proj}_{\rho_{\theta_k}}[R(\theta_{k+1},\theta_k)\circ T_{\theta_k}^{-1}]. \tag{56}$$

On the other hand,

$$\partial_\theta T_{\theta_{k+1}} = \partial_\theta T_{\theta_k} + r(\theta_{k+1},\theta_k). \tag{57}$$

Here $r(\theta,\theta')\in L^2(\mathbb{R}^d;\mathcal{L}(\mathbb{R}^m;\mathbb{R}^d),\rho_{\theta_k})$, the i-j entry of $r(\theta,\theta')(x)$ is $(\theta_{k+1}-\theta_k)^T\partial_\theta(\partial_{\theta_j}T^{(i)}_{\tilde{\theta}_{ij}(x)}(x)),\ 1\le i\le d,\ 1\le j\le m$, where each $\tilde{\theta}_{ij}(x) = \mu_{ij}(x)\theta_{k+1} + (1-\mu_{ij}(x))\theta_k$, for some $\mu_i j(x)\in(0,1)$. Now apply (57), (56) to (55), we obtain

$$\begin{aligned}
&\int \partial_\theta T_{\theta_k}(T_{\theta_k}^{-1}(x))\mathrm{Proj}_{\rho_{\theta_k}}[\partial_\theta T_{\theta_k}\circ T_{\theta_k}^{-1}(x)(\theta_{k+1}-\theta_k)]\ \rho_{\theta_k}(x)\ dx\\
&+\int \partial_\theta T_{\theta_k}(T_{\theta_k}^{-1}(x))\mathrm{Proj}_{\rho_{\theta_k}}[R(\theta_{k+1},\theta_k)\circ T_{\theta_k}^{-1}](x)\ \rho_{\theta_k}(x)\ dx\\
&+\int r(\theta_{k+1},\theta_k)(T_{\theta_k}^{-1}(x))\mathrm{Proj}_{\rho_{\theta_k}}[(T_{\theta_{k+1}}-T_{\theta_k})\circ T_{\theta_k}^{-1}](x)\ \rho_{\theta_k}(x)\ dx \quad = -h\nabla_\theta H(\theta_{k+1}).
\end{aligned}\tag{58}$$

Recall definition of $\mathbf{\Psi}$ in Theorem 5, use (51) of lemma 15, we know the first term on left hand side of (58) equals

$$\int \nabla\mathbf{\Psi}(x)\nabla\mathbf{\Psi}(x)^T(\theta_{k+1}-\theta_k)\ \rho_{\theta_k}(x)\ dx = G(\theta_k)(\theta_{k+1}-\theta_k).$$

Apply Cauchy inequality and (52) in lemma 15, every i-th entry of the second term of (58) can be bounded by:

$$\begin{aligned}
&\left(\int |\partial_\theta T^{(i)}_{\theta_k}(x)|^2\ dp(x)\cdot\int\sum_{i=1}^d |(\theta_{k+1}-\theta_k)\partial^2_{\theta\theta}T^{(i)}_{\tilde{\theta}_i(x)}(x)(\theta_{k+1}-\theta_k)|^2\ dp(x)\right)^{\frac{1}{2}}\\
&\le\left(\mathbb{E}_p|\partial_\theta T^{(i)}_{\theta_k}(x)|^2\cdot\mathbb{E}_p\left[\sum_{i=1}^d\|\partial^2_{\theta\theta}T^{(i)}_{\tilde{\theta}_i(x)}(x)\|_2\right]\right)^{\frac{1}{2}}|\theta_{k+1}-\theta_k|^2 \overset{\text{denote as}}{=} A^{(i)}|\theta_{k+1}-\theta_k|^2.
\end{aligned}$$

To bound the third term in (58), we first consider $T_{\theta_{k+1}}(x) - T_{\theta_k}(x)$, the i-th entry can be written as

$$T^{(i)}_{\theta_{k+1}}(x) - T^{(i)}_{\theta_k}(x) = (\theta_{k+1}-\theta_k)^T\partial_\theta T_{\bar{\theta}_i(x)}(x),$$

here $\bar{\theta}_i(x) = \zeta_i(x)\theta_{k+1} + (1-\zeta_i(x))\theta_k$ for some $\zeta_i(x)\in(0,1)$. Now the i-th entry of the third term of (58) can be bounded by:

$$\begin{aligned}
&\left(\int\sum_{i=1}^d|(\theta_{k+1}-\theta_k)^T\partial_{\theta\theta}T^{(i)}_{\tilde{\theta}_{ij}(x)}(x)|^2\ dp(x)\cdot\int|T^{(i)}_{\theta_{k+1}}(x) - T^{(i)}_{\theta_k}(x)|^2\ dp(x)\right)^{\frac{1}{2}}\\
&\le\left(\mathbb{E}_p\left[\sum_{i=1}^d\|\partial^2_{\theta\theta}T_{\tilde{\theta}_{ij}(x)}(x)\|_2^2\right]\cdot\mathbb{E}_p|\partial_\theta T^{(i)}_{\bar{\theta}_i(x)}(x)|^2\right)^{\frac{1}{2}}|\theta_{k+1}-\theta_k|^2 \overset{\text{denote as}}{=} B^{(i)}|\theta_{k+1}-\theta_k|^2.
\end{aligned}$$

We set $A\in\mathbb{R}^m$ with entries $A^{(i)}$, $1\le i\le m$ and similarly $B\in\mathbb{R}^m$ with entries $B^{(i)}$, $1\le i\le m$. (58) now leads to the following inequality,

$$|\theta_{k+1}-\theta_k + hG(\theta_k)^{-1}\nabla_\theta H(\theta_{k+1})| \le \|G(\theta_k)^{-1}\|_2(|A|+|B|)\ |\theta_{k+1}-\theta_k|^2.$$

As we have shown in Lemma 16 that $|\theta_{k+1} - \theta_k| \sim o(1)$, for any $\epsilon > 0$, when step size h is small enough, we always have $\theta_{k+1} \in B_\epsilon(\theta_k)$. Recall the notations in (54), we have $|A|, |B| \leq \sqrt{L(\theta_k, \epsilon)H(\theta_k, \epsilon)}$. Thus we have

$$|\theta_{k+1} - \theta_k + hG(\theta_k)^{-1}\nabla_\theta H(\theta_{k+1})| \leq 2\sqrt{L(\theta_k, \epsilon)H(\theta_k, \epsilon)}\|G(\theta_k)^{-1}\|_2|\theta_{k+1} - \theta_k|^2.$$

Denote $\theta_{k+1} - \theta_k = \eta$, $G(\theta_k)^{-1}\nabla_\theta H(\theta_{k+1}) = \xi$ and $C = 2\sqrt{L(\theta_k, \epsilon)H(\theta_k, \epsilon)}\|G(\theta_k)^{-1}\|_2$, the previous inequality is

$$|\eta \ - \ h\ \xi| \leq C|\eta|^2. \tag{59}$$

Since $|\eta - h\xi| \geq |\eta| - h|\xi|$, we have

$$C|\eta|^2 \geq |\eta| - h|\xi|. \tag{60}$$

Solving (60) gives

$$|\eta| \leq \frac{2|\xi|h}{1 + \sqrt{1 - 4C|\xi|h}} \quad \text{or} \quad |\eta| > \frac{1 + \sqrt{1 - 4Ch|\xi|}}{2C}.$$

The second inequality leads to $|\theta_{k+1} - \theta_k| > \frac{1}{2C}$ for any $h > 0$, which avoids $|\theta_{k+1} - \theta_k| \sim o(1)$. Thus, when h is sufficiently small, we have

$$|\eta| \leq \frac{2|\xi|h}{1 + \sqrt{1 - 4C|\xi|h}}. \tag{61}$$

Combining (61) and (59), we have:

$$|\theta_{k+1} - \theta_k + hG(\theta_k)^{-1}\nabla_\theta H(\theta_{k+1})| \leq \frac{C|\xi|^2}{(1 + \sqrt{1 - 4C|\xi|h})}\ h^2 \leq C|\xi|^2h^2. \tag{62}$$

This proves the result. □

Remark 7. *One should be aware of the relation between the positive definite condition (49) and the positive definiteness of the metric tensor $G(\theta_k)$: Positive definite $G(\theta)$ guarantees the inequality: $\widehat{W}_2^2(\theta, \theta') \geq C|\theta - \theta'|^2$ for $\theta' \in B_{r_0}(\theta)$ (r_0 depends on θ is small enough). But we are not able to bound $\widehat{W}_2^2(\theta, \theta')$ from below when $|\theta - \theta'| > r_0$. On the other hand, (49) is a locally weaker condition than positive definiteness of $G(\theta)$. Thus, positive definiteness of $G(\theta)$ and assumption (49) are related but not equivalent.*

4.2.3 Details of implementation

From the previous sections, we know that one can solve ODE (27) at every time step t_k by solving the saddle point problem (45). We now provide some detailed discussion on how we deal with (27):

- As in Remark 6, we may solve (48) instead of $\max_\phi \mathcal{E}(\phi)$ in every inner loop of the saddle point problem (45). Although they are mathematically equivalent, (48) has a more concise form. And according to our experience, using (48) makes our code run more efficiently than directly solving $\max_\phi \mathcal{E}(\phi)$. Thus we can formulate the following scheme that is equivalent to (45):

$$\theta_{k+1} = \underset{\theta}{\text{argmin}} \left\{ \int 2\nabla\hat{\psi}(x) \cdot ((T_\theta - T_{\theta_k}) \circ T_{\theta_k}^{-1}(x))\rho_{\theta_k}(x)\ dx - \int |\nabla\hat{\psi}(x)|^2\rho_{\theta_k}(x)\ dx + 2hH(\theta) \right\} \tag{63}$$

$$\overset{\text{denote as}}{=} \underset{\theta}{\text{argmin}}\ J(\theta),$$

$$\text{where } \hat{\psi} \text{ solves } \min_\phi \left\{ \int |\nabla\phi(x) - ((T_\theta - T_{\theta_k}) \circ T_{\theta_k}^{-1}(x))|^2\rho_{\theta_k}(x)\ dx \right\}.$$

- In numerical computation, we are not able to optimize over the entire function space of ψ. Instead, we treat $\psi_\lambda : M \to \mathbb{R}$ as a ReLU neural network parametrized by λ [12] . We know that in this case, ψ_λ is a piece-wise affine function and its gradient $\nabla\psi_\lambda(\cdot)$ forms a piece-wise constant vector field. Check Figure 2, 3 for an example.

- The entire procedure of solving (63) can be formulated as nested loops:

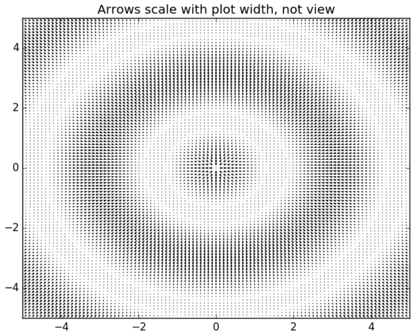


Figure 2: gradient field of $\psi(x) = \sin(|x|)$

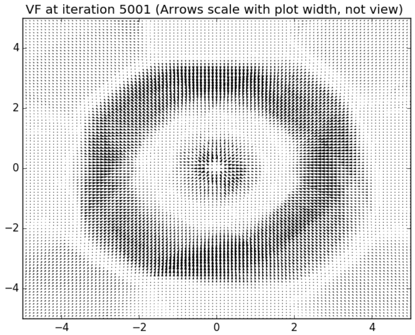


Figure 3: approximation by the gradient of a ReLU function ψ_λ

- (inner loop) Every inner loop aims at solving (48) on ReLU functions ψ_λ, i.e. solving:

$$\min_\lambda \left\{ \mathbb{E}_{\boldsymbol{X}\sim p} |\nabla \psi_\lambda(T_{\theta_k}(\boldsymbol{X})) - (T_\theta(\boldsymbol{X}) - T_{\theta_k}(\boldsymbol{X}))|^2 \right\}. \tag{64}$$

  One can use Stochastic Gradient Descent (SGD) methods like RMSProp [46] or Adam [20] with learning rate α_{in} to deal with this inner loop optimization. In our implementation, we will stop after M_{in} iterations. Let us denote the optimal λ in each inner loop as $\hat{\lambda}$;

- (outer loop) We apply similar SGD method to $J(\theta)$: using Lemma 14, we are able to compute $\nabla_\theta J(\theta)$ as:

$$\nabla_\theta J(\theta) = \partial_\theta \left(\int 2\nabla\hat{\psi}(x) \cdot (T_\theta \circ T_{\theta_k}^{-1}(x)) \rho_{\theta_k}(x)\, dx + 2hH(\theta) \right).$$

  If we treat optimal $\hat{\psi}$ as $\psi_{\hat{\lambda}}$, what we need to do in each outer loop is to consider:

$$\tilde{J}(\theta) = \mathbb{E}_{\boldsymbol{X}\sim p}\ 2[\nabla \psi_{\hat{\lambda}}(T_{\theta_k}(\boldsymbol{X})) \cdot T_\theta(\boldsymbol{X})] + 2h[V(T_\theta(\boldsymbol{X})) + \mathcal{L}_\theta(\boldsymbol{X})] \tag{65}$$

  and update θ for one step by our chosen SGD method with learning rate α_{out} applied to optimize $\tilde{J}(\theta)$. In our actual computation, we will stop the outer loop after M_{out} iterations.

- We now present the entire algorithm for computing (27) based on the scheme (45). This algorithm contains the following parameters: $T, N; M_{\text{out}}, K_{\text{out}}, \alpha_{\text{out}}; M_{\text{in}}, K_{\text{in}}, \alpha_{\text{in}}$. Recall we set reference distribution p as standard Gaussian on $M = \mathbb{R}^d$.

Remark 8. *In our implementation, $T_\theta(\boldsymbol{X}) - T_{\theta_k}(\boldsymbol{X})$ is usually of order $O(\alpha_{out})$, which is very small quantity. We can rescale it so that we solve each inner loop problem in a more stable way with larger stepsize (learning rate). That is to say, we choose some small $\epsilon \sim O(\alpha_{out})$ and consider*

$$\min_\lambda \left\{ \mathbb{E}_{\boldsymbol{X}\sim p} \left| \nabla \psi_\lambda(T_{\theta_k}(\boldsymbol{X})) - \left(\frac{T_\theta(\boldsymbol{X}) - T_{\theta_k}(\boldsymbol{X})}{\epsilon} \right) \right|^2 \right\}, \tag{66}$$

instead of (64) in each inner loop and set:

$$\tilde{J}(\theta) = \mathbb{E}_{\boldsymbol{X}\sim p}\ 2[\nabla \psi_{\hat{\lambda}}(T_{\theta_k}(\boldsymbol{X})) \cdot T_\theta(\boldsymbol{X})] + \frac{2h}{\epsilon}[V(T_\theta(\boldsymbol{X})) + \mathcal{L}_\theta(\boldsymbol{X})] \tag{67}$$

in each outer loop. In actual experiments, we usually set $\epsilon = \alpha_{out}$.

Remark 9. *It worth mentioning that the sample size K_{in}, K_{out} in each SGD step (especially K_{in}) should be chosen reasonably large so that the inner optimization problem can be solved with enough accuracy. In our practice, we usually choose $K_{in} = K_{out} = \max\{1000, 300d\}$. Here d is the dimension of sample space. This is very different from the small batch technique applied to training neural network in deep learning researches [32].*

Algorithm 1 Computing (27) by scheme (45) on the time interval $[0, T]$

```
Initialize $\theta$
for $i = 1, ...N$ do
    Save current parameter value to $\theta_0$: $\theta_0 = \theta$
    for $j = 1, ...M_{\text{outer}}$ do
        for $p = 1, ..., M_{\text{in}}$ do
            Sample $\{\mathbf{X}_1, ..., \mathbf{X}_{K_{\text{in}}}\}$ from $p$
            Apply one SGD (RMSProp, Adam etc.) step with learning rate $\alpha_{\text{in}}$ to loss function (of variable $\lambda$)
                $$\frac{1}{K_{\text{in}}} \left( \sum_{k=1}^{K_{\text{in}}} |\nabla \psi_\lambda (T_{\theta_0}(\mathbf{X}_k)) - (T_\theta(\mathbf{X}_k) - T_{\theta_0}(\mathbf{Y}_k))|^2 \right)$$
        end for
        Sample $\{\mathbf{X}_1, ..., \mathbf{X}_{K_{\text{out}}}\}$ from $p$
        Apply one SGD (RMSProp, Adam etc.) step with learning rate $\alpha_{\text{out}}$ to loss function
            $$\frac{1}{K_{\text{out}}} \sum_{k=1}^{K_{\text{out}}} 2[\nabla \psi_\lambda (T_{\theta_0}(\boldsymbol{X}_k)) \cdot T_\theta(\boldsymbol{X}_k)] + 2h[V(T_\theta(\boldsymbol{X}_k)) + \mathcal{L}_\theta(\boldsymbol{X}_k)]$$
    end for
    Set $\theta_i = \theta$
end for
The sequence of probability distributions $\{T_{\theta_0 \#} p, T_{\theta_1 \#} p, ..., T_{\theta_N \#} p\}$ will be the numerical solution of $\{\rho_{t_0}, \rho_{t_1}, ..., \rho_{t_N}\}$, where $t_i = i\frac{T}{N}$ $(i = 0, 1, ..., N-1, N)$. Here $\rho_t$ solves original Fokker-Planck equation (5).
```

5 Numerical analysis

In this section, we establish numerical analysis for parametric Fokker-Planck equation (27). In 5.1, we introduce an important quantity δ_0, which will play an essential role in our numerical analysis; In 5.2, we establish the asymptotic convergence analysis for equation (27); In 5.3, we work out the error analysis for both continuous version and discrete version (forward-Euler) of equation (27).

5.1 An important quantity

Before our analysis, we first introduce an important quantity that will play an essential role in our numerical analysis. Let us recall the optimal value of the least square problem (28) in Theorem 11 of section 3.2, or equivalently (29) of section 3.2, (34) of section 3.3. If we denote the upper bound of all possible values to be δ_0, i.e.

$$\delta_0 = \sup_{\theta \in \Theta} \min_{\xi \in T_\theta \Theta} \left\{ \int |(\nabla \mathbf{\Psi}(T_\theta(x))^T \xi - \nabla (V + \beta \log \rho_\theta) \circ T_\theta(x)|^2 \, dp(x) \right\}, \tag{68}$$

this quantity provides crucial error bound between our parametric equation and original equation in the forthcoming analysis. Ideally, we hope δ_0 to be sufficiently small. And this can be guaranteed if the neural network we select has universal approximation power. A closer examination may relax such a requirement. In fact, we only need require the neural network to be able to approximate a family of vector fields, more specifically, we want $\partial_\theta T_\theta$ to be able to approximate $\{\nabla(V + \beta \log \rho_\theta)\}_{\theta \in \Theta}$. In our numerical experiments, we found that using normalizing flow as T_θ works fine in various test examples. We believe that such an approximation property is shared by a large number of commonly used deep neural networks. This assertion can be further illustrated from another perspective. Let us consider T_θ with linear structure: i.e., set $T_\theta(x) = \sum_{i=1}^m \theta_i \vec{\Phi}_i(x)$, here $\{\vec{\Phi}_i\}_{i=1}^m$ are basis functions like gradient of radial basis functions(RBF). Then by (52) of Lemma 15, it is not hard to show:

$$\delta_0 \le \sup_{\theta\in\Theta} \min_{\xi\in\mathbb{R}^m} \left\{ \int |\sum_{k=1}^{m} \xi_k \vec{\Phi}_i(x) - \nabla(V + \beta \log\rho_\theta)\circ T_\theta(x)|^2 \; dp(x) \right\}.$$

This inequality indicates that δ_0 is no worse than the approximation error of using linear combination of classical RBF functions [8], which can be viewed as a one-layer network with large width. It is widely believed that nonlinear deep neural networks have better flexibility and approximation power than linear approximations, which may explain why normalizing flow can achieve accurate computations (small δ_0) in high dimensional space in our examples.

It also worth mentioning that δ_0 is used for *a priori* estimate in this section, because we don't know the exact trajectory of $\{\theta_t\}$ when solving ODE (27), and we take supremum over Θ to obtain δ_0. Once solved for $\{\theta_t\}$, denote $\mathcal{C}$ as set covering its trajectory, i.e.

$$\mathcal{C} = \{\theta \mid \exists\, t \ge 0, \text{ s.t. } \theta = \theta_t\} \tag{69}$$

We define another quantity δ_1:

$$\delta_1 = \sup_{\theta\in\mathcal{C}} \min_{\xi\in T_\theta\Theta} \left\{ \int |(\nabla\boldsymbol{\Psi}(T_\theta(x))^T\xi - \nabla\left(V + \beta\log\rho_\theta\right)\circ T_\theta(x)|^2 \; dp(x) \right\}. \tag{70}$$

Clearly, we have $\delta_1 \le \delta_0$. We can obtain corresponding *posterior* estimates for the asymptotic convergence and error analysis by replacing δ_0 with δ_1.

5.2 Asymptotic Convergence Analysis

In this section, we consider the solution $\{\theta_t\}_{t\ge0}$ of our parametric Fokker-Planck equation (27). We define:

$$\mathcal{V} = \left\{ V \;\middle|\; \begin{matrix} V \in \mathcal{C}^2(\mathbb{R}^d),\ V \text{ can be decomposed as: } V = U + \phi, \text{ with } U, \phi \in \mathcal{C}^2(\mathbb{R}^d); \\ \nabla^2 U \succeq KI[2] \text{ with } K > 0 \text{ and } \phi \in L^\infty(\mathbb{R}^d) \end{matrix} \right\}$$

As we know, for Fokker-Planck equation (5), when the potential $V \in \mathcal{V}$, $\{\rho_t\}$ will converge to the Gibbs distribution $\rho_* = \frac{1}{Z_\beta} e^{-V(x)/\beta}$ as $t \to \infty$ under the measure of KL divergence [15]. For (27), we wish to study its asymptotic convergence property. We come up with the following apriori result:

Theorem 17 (*a priori* estimation on asymptotic convergence)**.** *Consider Fokker-Planck equation* (5) *with the potential $V \in \mathcal{V}$. Suppose $\{\theta_t\}$ solves the parametric Fokker-Planck equation* (27)*, denote δ_0 as in* (68)*. Let $\rho_*(x) = \frac{1}{Z_\beta} e^{-V(x)/\beta}$ be the Gibbs distribution of original equation* (5)*. Then we have the inequality:*

$$D_{KL}(\rho_{\theta_t}\|\rho_*) \le \frac{\delta_0}{\tilde{\lambda}_\beta \beta^2}(1 - e^{-\beta\tilde{\lambda}_\beta t}) + D_{KL}(\rho_{\theta_0}\|\rho_*) e^{-\beta\tilde{\lambda}_\beta t}. \tag{71}$$

Here $\tilde{\lambda}_\beta > 0$ is the constant asscoiated to the Logarithm-Sobolev inequality discussed in Lemma 18 with potential function $\frac{1}{\beta}V$.

To prove Theorem 17, we need the following two lemmas:

Lemma 18. *[Holley-Stroock Perturbation] Suppose the potential $V \in \mathcal{V}$ is decomposed as $V = U + \phi$ where $\nabla^2 U \succeq KI$ and $\phi \in L^\infty$. Let $\tilde{\lambda} = Ke^{-osc(\phi)}$, here $osc(\phi) = \sup\phi - \inf\phi$. Then the following logarithm-Sobolev inequality holds for any probability density ρ:*

$$D_{KL}(\rho\|\rho_*) \le \frac{1}{\tilde{\lambda}} \mathcal{I}(\rho|\rho_*). \tag{72}$$

Here $\rho_* = \frac{1}{Z} e^{-V}$ and $\mathcal{I}(\rho|\rho_*)$ is the Fisher information functional defined as:

$$\mathcal{I}(\rho|\rho_*) = \int \left|\nabla \log\left(\frac{\rho(x)}{\rho_*(x)}\right)\right|^2 \rho(x)\; dx.$$

Lemma 18 is first proved in [15].

Lemma 19. *Recall δ_0 defined in (68), for any $\theta \in \Theta$, we have:*

$$\beta^2 \ \mathcal{I}(\rho_\theta|\rho_*) \leq \delta_0 + \nabla_\theta H(\theta) \cdot G(\theta)^{-1} \nabla_\theta H(\theta). \tag{73}$$

Proof of Lemma 19. Suppose $\{\theta_t\}$ solves (27) with $\theta_0 = \theta$. We denote $\xi = G(\theta)^{-1}\nabla_\theta H(\theta)$ for shorthand. By Theorem 8, $\frac{d}{dt}\rho_{\theta_t}\Big|_{t=0} = -(T_\#|_\theta)_* \xi$ is orthogonal projection of $-\mathrm{grad}_W \mathcal{H}(\rho_\theta)$ onto $T_{\rho_\theta}\mathcal{P}$ w.r.t metric g^W. Thus the orthogonal relation gives:

$$\begin{aligned} g^W(-\mathrm{grad}_W \mathcal{H}(\rho_\theta), -\mathrm{grad}_W \mathcal{H}(\rho_\theta)) = g^W(&\mathrm{grad}_W \mathcal{H}(\rho_\theta) - (T_\#|_\theta)_* \xi, \mathrm{grad}_W \mathcal{H}(\rho_\theta) - (T_\#|_\theta)_* \xi) \\ &+ g^W((T_\#|_\theta)_*\xi, (T_\#|_\theta)_*\xi). \end{aligned} \tag{74}$$

One can verify that left hand side of (74) is:

$$g^W(-\mathrm{grad}_W \mathcal{H}(\rho_\theta), -\mathrm{grad}_W \mathcal{H}(\rho_\theta)) = \int |\nabla(V(x) + \beta \log \rho_\theta(x))|^2 \rho(x) \ dx = \beta^2 \ \mathcal{I}(\rho_\theta|\rho_*). \tag{75}$$

Recall the equivalence between (28) and (29) and the definition (68) of δ_0, we know the first term on the right hand side of (74) is upper bounded by:

$$g^W(\mathrm{grad}_W \mathcal{H}(\rho_\theta) - (T_\#|_\theta)_* \xi, \mathrm{grad}_W \mathcal{H}(\rho_\theta) - (T_\#|_\theta)_* \xi) \leq \delta_0. \tag{76}$$

The second term on the right hand side of (74) is:

$$\begin{aligned} g^W((T_\#|_\theta)_*\xi, (T_\#|_\theta)_*\xi) = (T_\#|_\theta)^* g^W(\xi, \xi) &= G(\theta)(G(\theta)^{-1}\nabla_\theta H(\theta), \ G(\theta)^{-1}\nabla_\theta H(\theta)) \\ &= \nabla_\theta H(\theta) \cdot G(\theta)^{-1} \nabla_\theta H(\theta) \end{aligned} \tag{77}$$

Combining (74), (75),(76) and (77) yield to (73). □

Proof of Theorem 17. First, we recall the relationship between KL divergence and relative entropy:

$$D_{\mathrm{KL}}(\rho\|\rho_*) = \frac{1}{\beta}\mathcal{H}(\rho) + \log(Z_\beta).$$

We are actually treating $KL(\rho_\theta\|\rho_*)$ as the Lyapunov function for our ODE (27): take time derivative of $\mathrm{KL}(\rho_{\theta_t}\|\rho_*)$:

$$\frac{d}{dt} D_{\mathrm{KL}}(\rho_{\theta_t}\|\rho_*) = \frac{1}{\beta}\frac{d}{dt}\mathcal{H}(\rho_{\theta_t}) = \frac{1}{\beta}\dot{\theta}_t \cdot \nabla H(\theta_t) = -\frac{1}{\beta}\nabla H(\theta_t) \cdot G^{-1}(\theta_t)\nabla H(\theta_t).$$

Use the inequality in Lemma 19, we are able to show:

$$\frac{d}{dt} D_{\mathrm{KL}}(\rho_{\theta_t}\|\rho_*) \leq \frac{\delta_0}{\beta} - \beta \ \mathcal{I}(\rho_{\theta_t}|\rho_*).$$

Now by Lemma 18, we have:

$$\frac{d}{dt} D_{\mathrm{KL}}(\rho_{\theta_t}\|\rho_*) \leq \frac{\delta_0}{\beta} - \beta \ \tilde{\lambda}_\beta \ D_{\mathrm{KL}}(\rho_{\theta_t}\|\rho_*).$$

Then by Grownwall's inequality, we are able to show:

$$D_{\mathrm{KL}}(\rho_{\theta_t}\|\rho_*) \leq \frac{\delta_0}{\tilde{\lambda}_\beta \beta^2}(1 - e^{-\beta\tilde{\lambda}_\beta t}) + D_{\mathrm{KL}}(\rho_{\theta_0}\|\rho_*) e^{-\beta\tilde{\lambda}_\beta t}.$$

□

Remark 10. *Follow the previous proof, we can show the similar convergence estimation for the solution $\{\rho_t\}_{t\geq 0}$ of (5). Recall $\rho_*(x) = \frac{1}{Z_\beta}e^{-\frac{1}{\beta}V(x)}$, we have the inequality:*

$$D_{KL}(\rho_t\|\rho_*) \leq D_{KL}(\rho_0\|\rho_*) \ e^{-\beta\tilde{\lambda}_\beta t} \quad \forall \ t > 0. \tag{78}$$

It is natural to establish the posterior version of our asymptotic convergence analysis Theorem 17:

Theorem 20 (Posterior estimation on asymptotic convergence)**.** *We keep all the notations in Theorem 17, recall δ_1 defined in (70) then:*

$$D_{KL}(\rho_{\theta_t}\|\rho_*) \leq \frac{\delta_1}{\tilde{\lambda}_\beta \beta^2}(1 - e^{-\beta\tilde{\lambda}_\beta t}) + D_{KL}(\rho_{\theta_0}\|\rho_*) e^{-\beta\tilde{\lambda}_\beta t}.$$

5.3 Error Analysis

In this section we establish our error analysis for both continuous and discrete version of parametric Fokker-Planck equation (27) as an approximation of original equation (5).

5.3.1 Error analysis for continuous version

Suppose we exactly solved for $\{\theta_t\}_{t\geq 0}$ from (27). The following theorem provides an upper bound for the approximation error:

Theorem 21. *Assume that $\{\theta_t\}_{t\geq 0}$ solves (27); and $\{\rho_t\}_{t\geq 0}$ solves (5). Assume that the Hessian of the potential function V in (5) is bounded below by a constant λ, i.e. $\nabla^2 V \succeq \lambda\, I$. Then we have:*

$$W_2(\rho_{\theta_t}, \rho_t) \leq \frac{\sqrt{\delta_0}}{\lambda}(1 - e^{-\lambda t}) + e^{-\lambda t} W_2(\rho_{\theta_0}, \rho_0). \tag{79}$$

To prove this inequality, we need the following lemmas:

Lemma 22 (Constant speed of geodesic)**.** *Recall the geodesic equation [51],[30] connecting $\rho_0, \rho_1 \in \mathcal{P}(M)$ is described by the following equation system:*

$$\begin{cases} \frac{\partial \rho_t}{\partial t} + \nabla\cdot(\rho \nabla \psi_t) = 0 \\ \frac{\partial \psi_t}{\partial t} + \frac{1}{2}|\nabla\psi_t|^2 = 0 \end{cases} \qquad \rho_t|_{t=0} = \rho_0, \;\; \rho_t|_{t=1} = \rho_1. \tag{80}$$

Denote $\dot\rho_t = \partial_t \rho_t = -\nabla\cdot(\rho_t\nabla\psi_t) \in T_{\rho_t}\mathcal{P}(M)$. Then $g^W(\dot\rho_t, \dot\rho_t)$ is constant for $0 \leq t \leq 1$ and $g^W(\dot\rho_t, \dot\rho_t) = W_2^2(\rho_0, \rho_1)$ for $0 \leq t \leq 1$.

Proof. Recall definition (8) of Wasserstein metric g^W: $g^W(\dot\rho_t, \dot\rho_t) = \int |\nabla\psi_t|^2 \rho_t\, dx$. Since $\{\rho_t\}$ is the geodesic on $(\mathcal{P}(M), g^W)$, the speed $g^W(\sigma_t, \sigma_t)$ should remain constant. To directly verify this, we compute the time derivative:

$$\frac{d}{dt} g^W(\dot\rho_t, \dot\rho_t) = \frac{d}{dt}\left(\int |\nabla\psi_t|^2 \rho_t\, dx\right) = \int \frac{\partial}{\partial t}|\nabla\psi_t|^2 \rho_t\, dx + \int |\nabla\psi_t|^2 \partial_t\rho_t\, dx,$$

use the first equation in (80),

$$\int |\nabla\psi_t|^2 \partial_t \rho_t\, dx = \int |\nabla\psi_t|^2 \cdot (-\nabla\cdot(\rho_t\nabla\psi_t))\, dx = \int \nabla(|\nabla\psi_t|^2)\cdot\nabla\psi_t\rho_t\, dx,$$

take space gradient of the second equation in (80)

$$\partial_t(\nabla\psi_t) = -\nabla(\frac{1}{2}|\nabla\psi_t|^2).$$

Then

$$\int \frac{\partial}{\partial t}|\nabla\psi_t|^2\rho_t\, dx = \int 2\partial_t(\nabla\psi_t)\cdot\nabla\psi_t\rho_t\, dx = \int -\nabla(|\nabla\psi_t|^2)\cdot\nabla\psi_t\rho_t\, dx.$$

Adding them together, we have verified $\frac{d}{dt}g^W(\dot\rho_t, \dot\rho_t) = 0$, since $\int_0^1 g^W(\dot\rho_t, \dot\rho_t)\, dt = W_2^2(\rho_0, \rho_1)$, we know $g^W(\dot\rho_t, \dot\rho_t) = W_2^2(\rho_0, \rho_1)$ for any $0 \leq t \leq 1$. □

Lemma 23 (Displacement convexity of relative entropy)**.** *Suppose $\{\rho_t\}$ solves (80). Recall $\mathcal{H}$ as the relative entropy functional with potential V (11). Suppose $\nabla^2 V \succeq \lambda I$, then:*

$$\frac{d}{dt} g^W(grad_W \mathcal{H}(\rho_t), \dot\rho_t) \geq \lambda W_2^2(\rho_0, \rho_1).$$

Or equivalently, we have: $\frac{d^2}{dt^2}\mathcal{H}(\rho_t) \geq \lambda W_2^2(\rho_0, \rho_1)$.

Proof. We first write:

$$g^W(\text{grad}_W\mathcal{H}(\rho_t), \dot\rho_t) = \int \nabla(V + \beta \log \rho_t) \cdot \nabla\psi_t \ \rho_t \ dx.$$

Then:

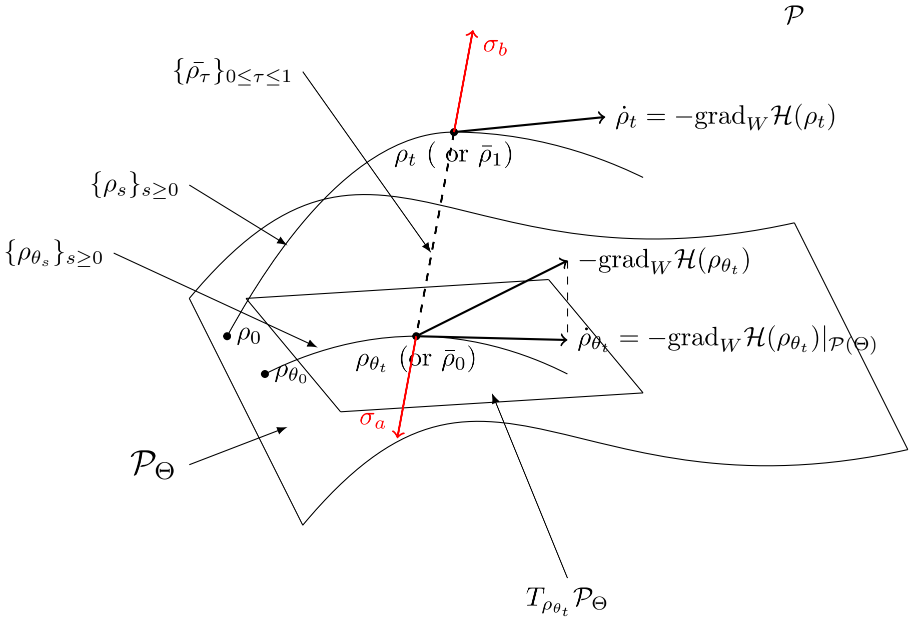
$$\frac{d}{dt}g^W(\text{grad}_W\mathcal{H}(\rho_t), \dot\rho_t) = \frac{d}{dt}\left(\int \nabla(V + \beta \log \rho_t) \cdot \nabla\psi_t \ \rho_t \ dx\right) = \int (\nabla\psi_t^T\nabla^2 V\nabla\psi_t + \text{Tr}(\nabla^2\psi_t\nabla^2\psi_t)) \ \rho_t \ dx.$$

The second equality can be carried out by direct calculations. Due to its length, we omit the details here. One can check [50] or [51] for complete derivation. Making use of $\nabla^2 V \succeq \lambda I$, we get:

$$\frac{d}{dt}g^W(\text{grad}_W\mathcal{H}(\rho_t), \dot\rho_t) \geq \int \lambda|\nabla\psi_t|^2\rho_t \ dx = \lambda \ g^W(\dot\rho_t, \dot\rho_t) = \lambda W_2^2(\rho_0, \rho_1).$$

The last equality is due to Lemma 22. By definition of Wasserstein gradient (10), $\frac{d}{dt}\mathcal{H}(\rho_t) = g^W(\text{grad}_W\mathcal{H}(\rho_t), \dot\rho_t)$, thus we also proved $\frac{d^2}{dt^2}\mathcal{H}(\rho_t) \geq \lambda W_2^2(\rho_0, \rho_1)$.

□


$\mathcal{P}$
σ_b
$\{\bar\rho_\tau\}_{0\leq\tau\leq1}$
$\dot\rho_t = -\text{grad}_W\mathcal{H}(\rho_t)$
ρ_t (or $\bar\rho_1$)
$\{\rho_s\}_{s\geq0}$
$\{\rho_{\theta_s}\}_{s\geq0}$
$-\text{grad}_W\mathcal{H}(\rho_{\theta_t})$
ρ_0
$\dot\rho_{\theta_t} = -\text{grad}_W\mathcal{H}(\rho_{\theta_t})|_{\mathcal{P}(\Theta)}$
ρ_{θ_0}
ρ_{θ_t} (or $\bar\rho_0$)
σ_a
$\mathcal{P}_\Theta$
$T_{\rho_{\theta_t}}\mathcal{P}_\Theta$

Proof of Theorem 21. For a given time t, we construct a geodesic $\{\bar\rho_\tau\}_{0\leq\tau\leq1}$ on Wasserstein manifold $\mathcal{P}(M)$ that starts at ρ_{θ_t} and ends at ρ_t. Such geodesic solves:

$$\begin{cases} \frac{\partial\bar\rho_\tau}{\partial\tau} + \nabla\cdot(\bar\rho_\tau\nabla\psi_\tau) = 0, \\ \frac{\partial\psi_\tau}{\partial\tau} + \frac{1}{2}|\nabla\psi_\tau|^2 = 0. \end{cases} \quad \text{with boundary conditions:} \bar\rho_0 = \rho_{\theta_t}, \ \bar\rho_1 = \rho_t.$$

We differentiate $W_2^2(\rho_{\theta_t}, \rho_t)$ with respect to time t, according to Theorem 23.9 of [51], we are able to deduce that:

$$\frac{d}{dt}W_2^2(\rho_{\theta_t}, \rho_t) = 2g^W(\dot\rho_{\theta_t}, -\dot{\bar\rho}_0) + 2g^W(\dot\rho_t, \dot{\bar\rho}_1), \tag{81}$$

here $\dot{\bar\rho}_0 = \partial_\tau\bar\rho_\tau|_{\tau=0} = -\nabla\cdot(\bar\rho_0\nabla\psi_0)$, $\dot{\bar\rho}_1 = \partial_\tau\bar\rho_\tau|_{\tau=1} = -\nabla\cdot(\bar\rho_1\nabla\psi_1)$. Notice that

$$\dot\rho_{\theta_t} = (T_\#|_{\theta_t})_*\dot\theta_t \quad \dot\rho_t = -\text{grad}_W\mathcal{H}(\rho_t) = \nabla\cdot(\rho_t\nabla(V + \beta\log\rho_t)).$$

Use the definition (8) of Wasserstein metric, we can compute (recall that $\rho_{\theta_t} = \bar\rho_0$, $\rho_t = \bar\rho_1$):

$$g^W(\dot\rho_{\theta_t}, \dot{\bar\rho}_0) = \int \nabla(V + \beta\log\bar\rho_0)\cdot\psi_0 \ \bar\rho_0 \ dx \quad g^W(\dot\rho_t, \dot{\bar\rho}_1) = \int \nabla(V + \beta\log\bar\rho_1)\cdot\psi_1 \ \bar\rho_1 \ dx.$$

We can now write (81) as:

$$\begin{aligned}\frac{1}{2}\frac{d}{dt}W_2^2(\rho_{\theta_t},\rho_t) =& g^W((T_\#|_{\theta_t})_*\dot{\theta}_t - (-\mathrm{grad}_W\mathcal{H}(\rho_{\theta_t})), -\dot{\bar{\rho}}_0) + g^W(-\mathrm{grad}_W\mathcal{H}(\rho_{\theta_t}), -\dot{\bar{\rho}}_0) + g^W(-\mathrm{grad}_W\mathcal{H}(\rho_t), \dot{\bar{\rho}}_1) \\ \overset{\text{set: } \xi=-\dot{\theta}_t}{=}& g^W(\mathrm{grad}_W\mathcal{H}(\rho_{\theta_t}) - (T_\#|_{\theta_t})_*\xi, -\dot{\bar{\rho}}_0) - (g^W(\mathrm{grad}_W\mathcal{H}(\bar{\rho}_1), \dot{\bar{\rho}}_1) - g^W(\mathrm{grad}_W\mathcal{H}(\bar{\rho}_0), \dot{\bar{\rho}}_0)).\end{aligned} \tag{82}$$

For the first term in (82), we use Cauchy inequality. By Lemma 22, we know $g(\sigma_a,\sigma_a) = W_2^2(\rho_{\theta_t},\rho_t)$. Now under (68), we will have:

$$\begin{aligned}g^W(\mathrm{grad}_W\mathcal{H}(\rho_{\theta_t}) - (T_\#|_{\theta_t})_*\xi, -\dot{\bar{\rho}}_0) \leq & \sqrt{g^W(\mathrm{grad}_W\mathcal{H}(\rho_{\theta_t}) - (T_\#|_{\theta_t})_*\xi, \mathrm{grad}_W\mathcal{H}(\rho_{\theta_t}) - (T_\#|_{\theta_t})_*\xi)}\sqrt{g^W(\dot{\bar{\rho}}_0,\dot{\bar{\rho}}_0)} \\ \leq & \sqrt{\delta_0}W(\rho_{\theta_t},\rho_t).\end{aligned} \tag{83}$$

For the second term in (82) , we write it as:

$$g^W(\mathrm{grad}_W\mathcal{H}(\bar{\rho}_1), \dot{\bar{\rho}}_1) - g^W(\mathrm{grad}_W\mathcal{H}(\bar{\rho}_0), \dot{\bar{\rho}}_0) = \int_0^1 \frac{d}{d\tau} g^W(\mathrm{grad}_W\mathcal{H}(\bar{\rho}_\tau), \dot{\bar{\rho}}_\tau)\, d\tau. \tag{84}$$

By Lemma 23, we have:

$$g^W(\mathrm{grad}_W\mathcal{H}(\bar{\rho}_1), \dot{\bar{\rho}}_1) - g^W(\mathrm{grad}_W\mathcal{H}(\bar{\rho}_0), \dot{\bar{\rho}}_0) \geq \lambda\ W_2^2(\rho_{\theta_t},\rho_t). \tag{85}$$

Combining inequalities (83), (85) and (82):

$$\frac{1}{2}\frac{d}{dt}W_2^2(\rho_{\theta_t},\rho_t) \leq -\lambda W_2^2(\rho_{\theta_t},\rho_t) + \sqrt{\delta_0}\ W_2(\rho_{\theta_t},\rho_t).$$

This is:

$$\frac{d}{dt}W_2(\rho_{\theta_t},\rho_t) \leq -\lambda W_2(\rho_{\theta_t},\rho_t) + \sqrt{\delta_0}.$$

Then Grownwall's inequality gives

$$W_2(\rho_{\theta_t},\rho_t) \leq \frac{\sqrt{\delta_0}}{\lambda}(1-e^{-\lambda t}) + e^{-\lambda t}W_2(\rho_{\theta_0},\rho_0).$$

□

When potential V is strictly convex, i.e. $\lambda > 0$. (79) in Theorem 21 provides a nice estimation: the error term $W_2(\rho_{\theta_t},\rho_t)$ at any time t is always upper bounded by $\max\{\frac{\sqrt{\delta_0}}{\lambda}, W_2(\rho_{\theta_0},\rho_0)\}$.

But in many cases, potential V may not be strictly convex, i.e. λ could be negative. In such cases, the right hand side in (79) may increase to infinity when time $t \to \infty$. However, (71) and (78) reveals that both ρ_{θ_t} and ρ_t will finally stay in a small neighbourhood of the Gibbs ρ_* when t is large. Taking this into account, the error term $W_2(\rho_{\theta_t},\rho_t)$ will never go crazy. We thus hope that the error can be controlled by a uniformly bounded value depending on t. This is summarized in the following theorem:

Theorem 24. *Suppose $\{\rho_t\}_{t\geq 0}$ solves (5) and $\{\rho_{\theta_t}\}_{t\geq 0}$ solves (27). We assume the potential $V \in \mathcal{V}$ and its Hessian can be bounded from below by λ, i.e. $\nabla^2 V \succeq \lambda I$. Keep all the notations in Theorem 17 and Theorem 21. Then we may improve the error estimation in Theorem 21 :*

$$W_2(\rho_{\theta_t},\rho_t) \leq \min\left\{\frac{\sqrt{\delta_0}}{\lambda} + \left(E_0 - \frac{\sqrt{\delta_0}}{\lambda}\right)\ e^{-\lambda t},\ \sqrt{\frac{2\delta_0}{\tilde{\lambda}_\beta^2\beta^2}} + \left(\sqrt{2K_1 - \frac{2\delta_0}{\tilde{\lambda}_\beta^2\beta^2}} + \sqrt{\frac{2K_2}{\tilde{\lambda}_\beta}}\right) e^{-\frac{\tilde{\lambda}_\beta}{2}\beta t}\right\}. \tag{86}$$

Here we denote $E_0 = W_2(\rho_{\theta_0},\rho_0)$, $K_1 = D_{KL}(\rho_{\theta_0}\|\rho_*)$, $K_2 = D_{KL}(\rho_0\|\rho_*)$.

Lemma 25 (Talagrand inequality [51],[35])**.** *Suppose $\rho_* = \frac{1}{Z}e^{-V}$. If ρ_* satisfies log-Sobolev inequality (72) with constant $\tilde{\lambda} > 0$. Then ρ_* also satisfies Talagrand inequality:*

$$\sqrt{2\frac{D_{KL}(\rho\|\rho_*)}{\tilde{\lambda}}} \geq W_2(\rho,\rho_*). \quad \textit{for any } \rho \in \mathcal{P}. \tag{87}$$

Proof of Theorem 24. The first term is already provided in Theorem 21, the second term is just a quick result of Theorem 17 and Talagrand inequality: for t fixed, (71) together with Talagrand inequality (87) gives:

$$W_2(\rho_{\theta_t}, \rho_*) \le \sqrt{2\frac{D_{\mathrm{KL}}(\rho_{\theta_t}\|\rho_*)}{\tilde{\lambda}_\beta}} \le \sqrt{\frac{2\delta_0}{\tilde{\lambda}_\beta^2\beta^2}(1-e^{-\tilde{\lambda}_\beta\beta t}) + 2K_1 e^{-\tilde{\lambda}_\beta\beta t}} \le \sqrt{\frac{2\delta_0}{\tilde{\lambda}_\beta^2\beta^2}} + \sqrt{2K_1 - \frac{2\delta_0}{\tilde{\lambda}_\beta^2\beta^2}}e^{-\frac{\tilde{\lambda}_\beta}{2}\beta t}.$$

Similarly, (78) and (87) gives

$$W_2(\rho_t, \rho_*) \le \sqrt{2\frac{D_{\mathrm{KL}}(\rho_t\|\rho_*)}{\tilde{\lambda}_\beta}} \le \sqrt{\frac{2K_2}{\tilde{\lambda}_\beta}}e^{-\frac{\tilde{\lambda}_\beta}{2}\beta t}.$$

Apply triangle inequality of Wasserstein distance $W_2(\rho_{\theta_t}, \rho_t) \le W_2(\rho_{\theta_t}, \rho_*) + W_2(\rho_t, \rho_*)$ we will get (86). □

We can take a further analysis on the upper bound of Theorem 24 to provide the following apriori uniform error bound:

Theorem 26 (Main Theorem on apriori error analysis of parametric Fokker-Planck equation)**.** *We follow previous notations and assumptions. The approximation error $W_2(\rho_{\theta_t}, \rho_t)$ at any time $t > 0$ can be uniformly bounded by constant number depending on $E_0 = W_2(\rho_{\theta_0}, \rho_0)$ and δ_0 defined in (68). To be more precise,*

1. *When $\lambda \ge 0$, the error $W_2(\rho_{\theta_t}, \rho_t)$ can be at least uniformly bounded by $O(E_0 + \sqrt{\delta_0})$ term;*
2. *When $\lambda < 0$, the error $W_2(\rho_{\theta_t}, \rho_t)$ can be at least uniformly bounded by $O((E_0 + \sqrt{\delta_0})^{\frac{\tilde{\lambda}_\beta\beta}{2|\lambda|+\tilde{\lambda}_\beta\beta}})$ term.*

Proof of Theorem 26 . Let us denote the right hand side of (86) as:

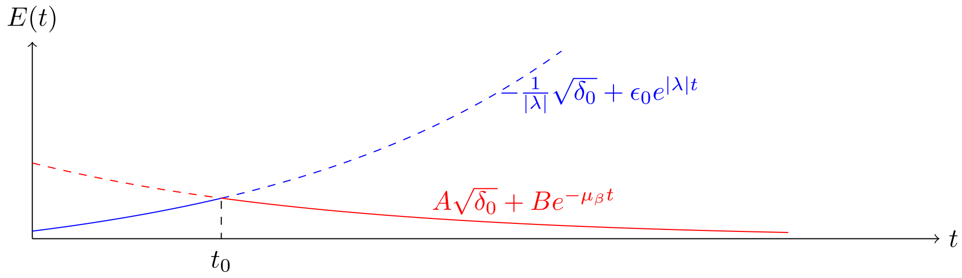
$$E(t) = \min\left\{-\frac{1}{|\lambda|}\sqrt{\delta_0} + \epsilon_0\ e^{|\lambda|t}, A\sqrt{\delta_0} + Be^{-\mu_\beta t}\right\}. \tag{88}$$

for shorthand, where

$$\epsilon_0 = E_0 + \frac{\sqrt{\delta_0}}{|\lambda|}, \quad A = \frac{\sqrt{2}}{\tilde{\lambda}_\beta\beta}, \quad B = \sqrt{2K_1 - \frac{2\delta_0}{\tilde{\lambda}_\beta^2\beta^2}} + \sqrt{\frac{2K_2}{\tilde{\lambda}_\beta}}, \quad \mu_\beta = \frac{\tilde{\lambda}_\beta\beta}{2}.$$

are all positive numbers.
(A) When $\lambda > 0$, $E(t) \le -\frac{1}{|\lambda|}\sqrt{\delta_0} + \epsilon_0\ e^{|\lambda|t} \lesssim O(\epsilon) = O(E_0 + \sqrt{\delta_0})$;
(B) The first term in (88) is increasing as a function of time t while the second term is decreasing. Let us denote $t_0 = \mathrm{argmax}_{t\ge 0}E(t)$, then t_0 should solve:

$$-\frac{1}{|\lambda|}\sqrt{\delta_0} + \epsilon_0\ e^{|\lambda|t_0} = A\sqrt{\delta_0} + Be^{-\mu_\beta t_0}. \tag{89}$$



Since $A > 0$, (89) leads to $\epsilon_0 e^{|\lambda|t_0} > Be^{-\mu_\beta t_0}$, thus

$$t_0 > \frac{\log\left(\frac{B}{\epsilon_0}\right)}{|\lambda| + \mu_\beta}. \tag{90}$$

Using (90), we are able to show:

$$\max_{t\geq 0} E(t) = E(t_0) = A\sqrt{\delta_0} + B\ e^{-\mu_\beta t_0} < A\sqrt{\delta_0} + B^{\frac{|\lambda|}{|\lambda|+\mu_\beta}}\ \epsilon_0^{\frac{\mu_\beta}{|\lambda|+\mu_\beta}}. \tag{91}$$

As a result, $W_2(\rho_{\theta_t}, \rho_t)$ can be uniformly bounded by the right hand side of (91). Since A, B are $O(1)$ coefficients and $\epsilon_0 > \sqrt{\delta_0}$, this uniform bound is dominated by $O(\epsilon_0^{\frac{\mu_\beta}{|\lambda|+\mu_\beta}}) = O((E_0 + \sqrt{\delta_0})^{\frac{\tilde{\lambda}_\beta \beta}{2|\lambda|+\tilde{\lambda}_\beta \beta}})$. □

Remark 11. *We can make further discussions on the error order $\alpha = \frac{\tilde{\lambda}_\beta \beta}{2|\lambda|+\tilde{\lambda}_\beta \beta}$ when $V \in \mathcal{V}$ is not convex. Suppose V is decomposed as $V = U + \phi$. with $\nabla^2 U \succeq KI$ $(K > 0)$ and $\nabla^2\phi \succeq K_\phi I$. We also assume $\nabla^2 V \succeq \lambda I$ with $\lambda < 0$. Then it is not hard to verify that $K_\phi < 0$ and $|K_\phi| - K \geq |\lambda|$. On the other hand, one can compute $\tilde{\lambda}_\beta = \frac{K}{\beta} e^{-\frac{osc(\phi)}{\beta}}$. Combining these together, we can provide a lower bound $\gamma(\beta, U, \phi)$ for order α:*

$$\alpha \geq \gamma(\beta, U, \phi) = \frac{1}{1 + 2\left(\frac{|K_\phi|}{K} - 1\right) e^{\frac{osc(\phi)}{\beta}}}$$

One can verify that increasing the diffusion coefficient β or convexity K, or decreasing the oscillation $osc(\phi)$ and convexity K_ϕ will both improve the lower bound $\gamma(\beta, U, \phi)$ for order α.

At the end of this section, we remark that it is natural to establish the corresponding posterior estimation on error term $W_2(\rho_{\theta_t}, \rho_t)$:

Theorem 27 (Posterior error analysis of parametric Fokker-Planck equation)**.** *We follow previous notations and assumptions. Then $W_2(\rho_{\theta_t}, \rho_t)$ at any time $t > 0$ can be uniformly bounded by constant number depending on $E_0 = W_2(\rho_{\theta_0}, \rho_0)$ and δ_1 defined in (70):*

1. *When $\lambda \geq 0$, $W_2(\rho_{\theta_t}, \rho_t)$ can be at least uniformly bounded by $O(E_0 + \sqrt{\delta_1})$;*
2. *When $\lambda < 0$, $W_2(\rho_{\theta_t}, \rho_t)$ can be at least uniformly bounded by $O((E_0 + \sqrt{\delta_1})^{\frac{\tilde{\lambda}_\beta \beta}{2|\lambda|+\tilde{\lambda}_\beta \beta}})$.*

5.3.2 Error analysis for discrete version

To solve (27) numerically, we need to apply discrete scheme. In this section, we will mainly focus on the forward Euler scheme: Suppose we apply forward-Euler scheme to solve (27) and compute for θ_k at each time node. We denote $\rho_{\theta_k} = T_{\theta_k \#} p$, our main purpose is to estimate the W_2-error between our numerical solution ρ_{θ_k} and the real solution ρ_{t_k}. Our main conclusion is exhibited in the following theorem:

Theorem 28 (Apriori error analysis of forward-Euler scheme)**.** *Suppose the potential function $V \in \mathcal{C}^2(\mathbb{R}^d)$ and its Hessian can be bounded from above and below, i.e. $\lambda I \preceq \nabla^2 V \preceq \Lambda I$. Suppose we apply forward-Euler scheme to solve (27) on the time interval $[0, T]$ with time stepsize $h = \frac{T}{N}$. Denote the corresponding solution at every time node $t_k = kh$ as θ_k $(k = 0, 1, ..., N)$. Assume $\{\rho_t\}_{t\geq 0}$ solves the Fokker-Planck Equation (5). Then we have:*

$$W_2(\rho_{\theta_k}, \rho_{t_k}) \leq (\sqrt{\delta_0} h + C_N h^2) \frac{1 - e^{-\lambda t_k}}{1 - e^{-\lambda h}} + e^{-\lambda t_k} W_2(\rho_{\theta_0}, \rho_0) \quad \textit{for any} \quad t_k = kh, \quad 0 \leq k \leq N. \tag{92}$$

The explicit definition of the constant C_N is in (107).

In order to estimate $W_2(\rho_{\theta_k}, \rho_{t_k})$, we use the triangle inequality of W_2 distance [51] to separate it into three parts:

$$W_2(\rho_{\theta_k}, \rho_{t_k}) \leq W_2(\rho_{\theta_k}, \tilde{\rho}^\star_{t_k}) + W_2(\tilde{\rho}^\star_{t_k}, \tilde{\rho}_{t_k}) + W_2(\tilde{\rho}_{t_k}, \rho_{t_k}). \tag{93}$$

Here $\{\tilde{\rho}_t\}_{t_{k-1}\leq t\leq t_k}$ satisfies:

$$\frac{\partial \tilde{\rho}_t}{\partial t} = \nabla \cdot (\tilde{\rho}_t \nabla V) + \beta \Delta \tilde{\rho}_t \ , \quad \tilde{\rho}_{t_{k-1}} = \rho_{\theta_{k-1}}. \tag{94}$$

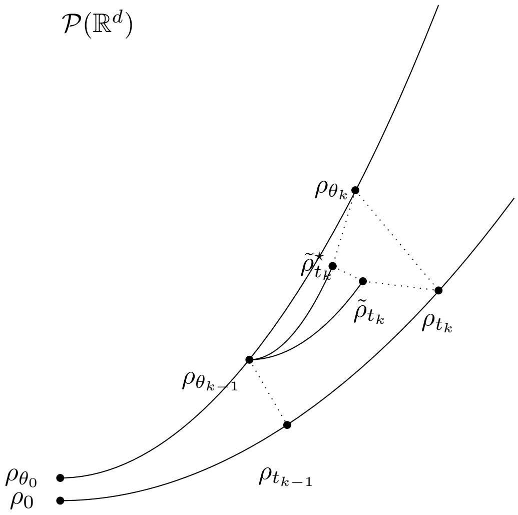


Figure 4: Trajectory of $\{\rho_{\theta_k}\}_{k=0,\dots,N}$ is our numerical solution; trajectory of $\{\rho_t\}_{t\geq 0}$ is the real solution of Fokker-Planck Equation; $\{\tilde{\rho}_t\}_{t\geq t_{k-1}}$ solves (94); $\{\tilde{\rho}^\star_t\}_{t\geq t_{k-1}}$ solves (95).

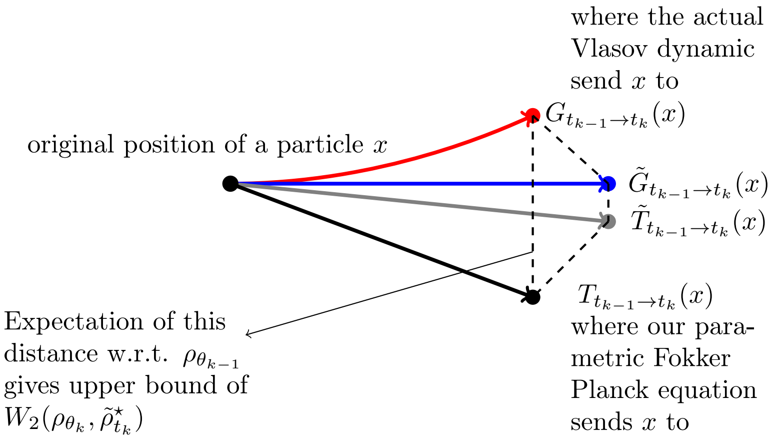

where the actual
Vlasov dynamic
send x to
$G_{t_{k-1}\to t_k}(x)$
original position of a particle x
$\tilde{G}_{t_{k-1}\to t_k}(x)$
$\tilde{T}_{t_{k-1}\to t_k}(x)$
$T_{t_{k-1}\to t_k}(x)$
where our parametric Fokker
Planck equation
sends x to
Expectation of this
distance w.r.t. $\rho_{\theta_{k-1}}$
gives upper bound of
$W_2(\rho_{\theta_k}, \tilde{\rho}^\star_{t_k})$

Figure 5: Illustration of proof strategy

Thus $\{\tilde{\rho}_t\}_{t_{k-1}\leq t\leq t_k}$ solves the real Fokker-Planck equation with initial condition $\rho_{\theta_{k-1}}$.
And we assume $\{\tilde{\rho}^\star_t\}_{t\geq t_{k-1}}$ satisfies:

$$\frac{\partial \tilde{\rho}^\star_t}{\partial t} = \nabla\cdot(\tilde{\rho}^\star_t \nabla(V + \beta \log \rho_{\theta_{k-1}}))\ , \quad \tilde{\rho}^\star_{t_{k-1}} = \rho_{\theta_{k-1}}. \tag{95}$$

Suppose we fix the vector field $-\nabla V - \beta\nabla\log\rho_{\theta_{k-1}}$ at time t_{k-1} and let the particles obeying distribution $\rho_{\theta_{k-1}}$ flow along this fixed vector field, the distribution of these particles at time t will be $\tilde{\rho}^\star_t$.
Figure 4 shows the relations of different items used in our proof.
Now we provide estimations for the three terms appeared in (93). We separate our results into three lemmas.

Lemma 29. *The first term $W_2(\rho_{\theta_k}, \tilde{\rho}^\star_{t_k})$ in (93) can be upper bounded by $\sqrt{\delta_0}h + O(h^2)$.*

An explicit formula for the coefficient of h^2 is included in the following proof.

Proof. We will establish the desired estimation by introducing several different pushforward maps and then applying triangle inequality.
(1) We know $\rho_{\theta_{k-1}} = T_{\theta_{k-1}\#}p$ and $\rho_{\theta_k} = T_{\theta_k\#}p$, let us we denote $T_{t_{k-1}\to t_k} = T_{\theta_k}\circ T^{-1}_{\theta_{k-1}}$. Then $\rho_{\theta_k} =$

${T_{t_{k-1}\to t_k}}_\# \rho_{\theta_{k-1}}$.
(2) We let $\xi_{k-1} = \dot{\theta}_{k-1} = -G(\theta_{k-1})^{-1}\nabla_\theta H(\theta_{k-1})$ and by convention, we denote $\mathbf{\Psi}$ as solution of (15). We consider the map $\tilde{T}_{t_{k-1}\to t_k}(\cdot) = \mathrm{Id} + h\nabla\mathbf{\Psi}(\cdot)^T\xi_{k-1}$.
(3) We denote $\zeta_{k-1}(\cdot) = V(\cdot) + \beta\log\rho_{\theta_{k-1}}(\cdot)$. The particle version (recall (6)) of (95) is:

$$\dot{z}_t = -\nabla\zeta_{k-1}(z_t) \quad 0 \le t \le h \quad \text{with initial condition } z_0 = x. \tag{96}$$

we denote the solution map of (96) by $G_{t_{k-1}\to t_k}(x) = z_{t_k}$. Then $\tilde{\rho}^\star_{t_k} = {G_{t_{k-1}\to t_k}}_\#\rho_{\theta_{k-1}}$.
(4) The map $G_{t_{k-1}\to t_k}$ is obtained by solving an ODE, in order to compare the difference with $T_{t_{k-1}\to t_k}$, we consider the ODE with fixed initial vector field:

$$\dot{\tilde{z}}_t = -\nabla\zeta_{k-1}(x) \quad 0 \le t \le h \quad \tilde{z}_0 = x. \tag{97}$$

This ODE will induce the solution map $\tilde{G}_{t_{k-1}\to t_k}(\cdot) = \mathrm{Id} - h\nabla\zeta_{k-1}(\cdot)$.
With the maps defined in (1),(2),(3),(4), use the triangle inequality of W_2 distance, we can estimate:

$$\begin{aligned}W_2(\rho_{\theta_k}, \tilde{\rho}^\star_{t_k}) &= W_2(T_{t_{k-1}\to t_k\#}\rho_{\theta_{k-1}}, G_{t_{k-1}\to t_k\#}\rho_{\theta_{k-1}}) \\ &\le \underbrace{W_2(T_{t_{k-1}\to t_k\#}\rho_{\theta_{k-1}}, \tilde{T}_{t_{k-1}\to t_k\#}\rho_{\theta_{k-1}})}_{\text{(A)}} + \underbrace{W_2(\tilde{T}_{t_{k-1}\to t_k\#}\rho_{\theta_{k-1}}, \tilde{G}_{t_{k-1}\to t_k\#}\rho_{\theta_{k-1}})}_{\text{(B)}} \\ &+ \underbrace{W_2(\tilde{G}_{t_{k-1}\to t_k\#}\rho_{\theta_{k-1}}, G_{t_{k-1}\to t_k\#}\rho_{\theta_{k-1}})}_{\text{(C)}}.\end{aligned}$$

We now give upper bounds for distances (A),(B) and (C):

(A) Set $\theta(\tau) = \theta_{k-1} + \frac{\tau}{h}(\theta_k - \theta_{k-1}) = \theta_{k-1} + \tau\xi_{k-1}$. For any x, consider $x_\tau = T_{\theta(\tau)}(T^{-1}_{\theta_{k-1}}(x))$ with $0 \le \tau \le h$. such $\{x_\tau\}_{0\le\tau\le h}$ satisfies

$$\dot{x}_\tau = \partial_\theta T_{\theta(\tau)}(T^{-1}_{\theta(\tau)}(x_\tau))\xi_{k-1} \quad 0 \le \tau \le h. \tag{98}$$

If we assume $x_0 \sim \rho_{\theta_{k-1}}$ in (98), it is clear that $x_h \sim {T_{t_{k-1}\to t_k}}_\#\rho_{\theta_{k-1}}$. Furthermore, we denote the distribution of x_τ as ρ_τ. Now assume that $\{\psi_\tau\}$ solves

$$-\nabla\cdot(\rho_\tau(x)\partial_\theta T_{\theta(\tau)}(T^{-1}_{\theta(\tau)}(x))) = -\nabla\cdot(\rho_\tau(x)\nabla\psi_\tau(x)) \quad 0 \le \tau \le h, \tag{99}$$

and consider

$$\dot{y}_\tau = \nabla\psi_\tau(y_\tau) \quad 0 \le \tau \le h \quad \text{with } y_0 \sim \rho_{\theta_{k-1}}.$$

Denote ϱ_τ as the distribution of y_τ, by continuity equation and (99), one knows $\rho_\tau = \varrho_\tau$ for $0 \le \tau \le h$, thus $y_h \sim {T_{t_{k-1}\to t_k}}_\#\rho_{\theta_{k-1}}$. On the other hand, when $\tau = 0$, (99) shows $\nabla\psi_0(x) = \nabla\mathbf{\Psi}(x)^T\xi_{k-1}$. Combine these together, we can estimate term (A) as:

$$\begin{aligned}W_2^2(T_{t_{k-1}\to t_k\#}\rho_{\theta_{k-1}}, \tilde{T}_{t_{k-1}\to t_k\#}\rho_{\theta_{k-1}}) &\le \mathbb{E}_{y_0\sim\rho_{\theta_{k-1}}}|y_h - (y_0 + h\nabla\psi_0(y_0))|^2 \\ &\le \mathbb{E}_{y_0\sim\rho_{\theta_{k-1}}}\left|\int_0^h \nabla\psi_\tau(y_\tau) - \nabla\psi_0(y_0)\, d\tau\right|^2\end{aligned}$$

If we define the constant (only depends on θ_{k-1} and h):

$$M(\theta_{k-1}, h) = \left(\underset{y_0\sim\rho_{\theta_{k-1}}}{\mathbb{E}}\left[\sup_{0\le\tau\le h}\left|\frac{\nabla\psi_\tau(y_\tau) - \nabla\psi_0(y_0)}{\tau}\right|^2\right]\right)^{1/2} \tag{100}$$

Then we are able to show:

$$W_2(T_{t_{k-1}\to t_k\#}\rho_{\theta_{k-1}}, G_{t_{k-1}\to t_k\#}\rho_{\theta_{k-1}}) \le \frac{1}{2}M(\theta_{k-1}, h)h^2.$$

(B) We have

$$W_2^2(\tilde{T}_{t_{k-1}\to t_k\#}\rho_{\theta_{k-1}}, \tilde{G}_{t_{k-1}\to t_k\#}\rho_{\theta_{k-1}}) \leq \int |\tilde{T}_{t_{k-1}\to t_k}(x) - \tilde{G}_{t_{k-1}\to t_k}(x)|^2 \rho_{\theta_{k-1}}(x)\ dx$$
$$= h^2 \left(\int |\nabla \mathbf{\Psi}(x)^T \xi_{k-1} - (-\nabla \zeta_{k-1}(x))|^2 \rho_{\theta_{k-1}}(x)\ dx \right)$$
$$= h^2 \left(\int |\nabla \mathbf{\Psi}(T_{\theta_{k-1}}(x))^T \xi_{k-1} - (-\nabla(V + \beta \log \rho_{\theta_{k-1}}) \circ T_{\theta_{k-1}}(x))|^2\ dp(x) \right) \leq \delta_0\ h^2.$$

The last inequality is due to Theorem 11 and definition (68).

(C) Recall $\{z_t\}, \{\tilde{z}_t\}$ solve (96) and (97) with initial condition $z_0 = \tilde{z}_0 = x$, then we can estimate term (C) as:

$$W_2^2(\tilde{G}_{t_{k-1}\to t_k\#}\rho_{\theta_{k-1}}, G_{t_{k-1}\to t_k\#}\rho_{\theta_{k-1}}) \leq \mathbb{E}_{x\sim\rho_{\theta_{k-1}}} |z_h - \tilde{z}_h|^2 = \mathbb{E}_{x\sim\rho_{\theta_{k-1}}} \left| \int_0^h \nabla\zeta_{k-1}(x) - \nabla\zeta_{k-1}(z_\tau)\ d\tau \right|$$

If we denote the constant (only depends θ_{k-1} and h)

$$N(\theta_{k-1}, h) = \left(\underset{x\sim\rho_{\theta_{k-1}}}{\mathbb{E}} \left[\sup_{0\leq\tau\leq h} \left| \frac{\nabla\zeta_{k-1}(x) - \nabla\zeta_{k-1}(z_\tau)}{\tau} \right|^2 \right] \right)^{1/2} \tag{101}$$

Similar to (A), we have:

$$W_2(\tilde{G}_{t_{k-1}\to t_k\#}\rho_{\theta_{k-1}}, G_{t_{k-1}\to t_k\#}\rho_{\theta_{k-1}}) \leq \frac{1}{2} N(\theta_{k-1}, h) h^2$$

Now, combining previous estimates of term (A),(B) and (C), we obtain:

$$W_2(\rho_{\theta_k}, \tilde{\rho}^\star_{t_k}) \leq \sqrt{\delta_0}\ h + \frac{M(\theta_{k-1}, h) + N(\theta_{k-1}, h)}{2}\ h^2.$$

□

Lemma 30. *The second term in (93) can be upper bounded by $O(h^2)$.*

An explicit formula for the coefficient of h^2 is included in the following proof.

Proof. Recall $\tilde{\rho}_t$ is defined by (94) and $\tilde{\rho}^*_t$ is defined by (95). We can rewrite (95) as:

$$\frac{\partial \tilde{\rho}^\star_t}{\partial t} = \nabla \cdot (\tilde{\rho}^\star_t(\nabla V + \beta\nabla \log \rho_{\theta_{k-1}} - \nabla \log \tilde{\rho}^\star_t)) + \beta\Delta\tilde{\rho}^\star_t \quad t_{k-1} \leq t \leq t_k$$

Now we fix Brownian Motion $\{\boldsymbol{B}_\tau\}_{0\leq\tau\leq h}$, we consider the following Stochastic Differential Equations (SDEs) sharing the same $\{\boldsymbol{B}_\tau\}$ and initial condition:

$$dx_\tau = -\nabla V(x_\tau) d\tau + \sqrt{2\beta}\ d\boldsymbol{B}_\tau \tag{102}$$
$$dx^\star_\tau = -\nabla V(x^\star_\tau) d\tau + (\beta\nabla \log \tilde{\rho}^\star_{t_{k-1}+\tau}(x^\star_\tau) - \beta\nabla \log \rho_{\theta_{k-1}}(x^\star_\tau)) d\tau + \sqrt{2\beta}\ d\boldsymbol{B}_\tau \tag{103}$$

with initial condition: $x_0 = x^\star_0 \sim \rho_{\theta_{k-1}}$ and $0 \leq \tau \leq h$.

We denote $\vec{r}(x,\tau) = \beta\nabla \log \tilde{\rho}^\star_{t_{k-1}+\tau}(x) - \beta\nabla \log \rho_{\theta_{k-1}}(x)$. Then subtracting (102) from (103) will lead to:

$$x^\star_\tau - x_\tau = \int_0^\tau \nabla V(x_s) - \nabla V(x^\star_s) + \vec{r}(x^\star_s, s)\ ds$$

Then we have:

$$\begin{aligned}\mathbb{E}|x^\star_\tau - x_\tau|^2 &= \mathbb{E}\left|\int_0^\tau \nabla V(x_s) - \nabla V(x^\star_s) + \vec{r}(x^\star_s, s)\ ds\right|^2 \leq 2\ \mathbb{E}\left|\int_0^\tau \nabla V(x_s) - \nabla V(x^\star_s)\ ds\right|^2 + 2\ \mathbb{E}\left|\int_0^\tau \vec{r}(x^\star_s, s)\ ds\right|^2 \\ &\leq 2\ \mathbb{E}\left[\tau \int_0^\tau |\nabla V(x_s) - \nabla V(x^\star_s)|^2\ ds\right] + 2\ \mathbb{E}\left[\tau \int_0^\tau |\vec{r}(x^\star_s, s)|^2\ ds\right] \\ &= 2\tau\left(\int_0^\tau \mathbb{E}|\nabla V(x_s) - \nabla V(x^\star_s)|^2 + \mathbb{E}|\vec{r}(x^\star_s, s)|^2\ ds\right)\end{aligned}$$

Since Hessian of V is bounded above by Λ, $|\nabla V(x) - \nabla V(y)| \leq \Lambda|x-y|$ for any $x, y \in \mathbb{R}^d$. Thus we have the inequality:

$$\mathbb{E}|x^\star_\tau - x_\tau|^2 \leq 2\tau\Lambda^2 \int_0^\tau \mathbb{E}|x^\star_s - x_s|^2\ ds + 2\tau \int_0^\tau \mathbb{E}|\vec{r}(x^\star_s, s)|^2\ ds \tag{104}$$

We denote $U_\tau = \int_0^\tau \mathbb{E}|x^\star_s - x_s|^2\ ds$ and $R_\tau = \int_0^\tau \mathbb{E}|\vec{r}(x^\star_s, s)|^2\ ds$, then (104) becomes:

$$U'_\tau \leq 2\Lambda^2\tau U_\tau + 2\tau R_\tau$$

By integrating this inequality, $U_\tau \leq \int_0^\tau 2e^{\Lambda(\tau^2 - s^2)} sR_s\ ds$ so $U'_\tau \leq 4\Lambda^2\tau \int_0^\tau e^{\Lambda(\tau^2-s^2)} sR_s ds + 2\tau R_\tau$, thus:

$$W_2(\tilde{\rho}^\star_{t_k}, \tilde{\rho}_{t_k}) \leq \sqrt{\mathbb{E}|x^*_h - x_h|^2} = U'_h \leq \sqrt{4\Lambda^2 h \int_0^h e^{\Lambda(h^2-s^2)} sR_s\ ds + 2hR_h}$$

Let us define the constant

$$L_{k-1}(\theta_{k-1}, h) = \sup_{0\leq\tau\leq h}\left\{\mathbb{E}\left|\frac{\nabla \log \rho_{t_{k-1}+\tau}(x^\star_\tau) - \nabla \log \rho_{t_{k-1}}(x^\star_\tau)}{\tau}\right|\right\}$$

Then for any $0 \leq \tau \leq h$, we can estimate: $R_\tau \leq \int_0^h |\beta L_{k-1}(\theta_{k-1}, h)s|^2\ ds \leq \frac{1}{3}\beta^2 L_{k-1}(\theta_{k-1}, h)^2 h^3$. Thus (5.3.2) leads to:

$$W_2(\tilde{\rho}^\star_{t_k}, \tilde{\rho}_{t_k}) \leq \sqrt{4\Lambda^2 h \int_0^h e^{\Lambda(h^2-s^2)} sR_s\ ds + 2hR_h} \leq \sqrt{\frac{4}{3}\Lambda^2 e^{\Lambda h^2}\beta^2 L^2 h^6 + \frac{2}{3}\beta^2 L^2 h^4} \tag{105}$$

Here we denote L as $L_{k-1}(\theta_{k-1}, h)$ for shorthand. When h is small, the h^4 term in (105) is dominating the upper bound term. Thus we may assert that when h is small enough,

$$W_2(\tilde{\rho}^\star_{t_k}, \tilde{\rho}_{t_k}) \leq \beta L_{k-1}(\theta_{k-1}, h)h^2$$

□

Remark 12. *Analyzing the discrepancy of stochastic particles under different movements will provide a natural upper bound for W_2 distance. Both Lemma 29 and Lemma 30 are derived by making use of the particle version of their corresponding density evolutions. Such proving strategy was motivated from section 3.3.*

Lemma 31. *For third term in (93), we have:*

$$W_2(\rho_{t_k}, \tilde{\rho}_{t_k}) \leq e^{-\lambda h} W_2(\rho_{t_{k-1}}, \rho_{\theta_{k-1}})$$

This lemma is a direct corollary of the following theorem:

Theorem 32. *Suppose the potential $V \in C^2(\mathbb{R}^d)$ and its convexity is bounded below: $\nabla^2 V \succeq \lambda I$ (i.e. the matrix $\nabla^2 V(x) - \lambda I$ is semi-positive definite for any $x \in \mathbb{R}^d$; here λ is a finite real number and need not to be positive). Consider $\rho_1, \rho_2 \in \mathcal{P}$ and two Fokker-Planck equations with different initial distributions:*

$$\begin{aligned}\frac{\partial \rho^{(1)}_t}{\partial t} &= \nabla\cdot(\rho^{(1)}_t \nabla V) + \beta\Delta\rho^{(1)}_t \quad \rho^{(1)}_0 = \rho_1; \\ \frac{\partial \rho^{(2)}_t}{\partial t} &= \nabla\cdot(\rho^{(2)}_t \nabla V) + \beta\Delta\rho^{(2)}_t \quad \rho^{(2)}_0 = \rho_2.\end{aligned}$$

Then we have:

$$W_2(\rho^{(1)}_t, \rho^{(2)}_t) \leq e^{-\lambda t} W_2(\rho_1, \rho_2) \tag{106}$$

This is a known stability result on Wasserstein gradient flows. One can find its proof in [3] or [51]. Once we have proven Lemma 29,30,31, we are able to prove theorem 28:

Proof. (Proof of Theorem 28) Let's denote:

$$\mathrm{Err}_k = W_2(\rho_{\theta_k}, \rho_{t_k}) \quad k = 0, 1, ..., N.$$

Combining Lemma 29, Lemma 30 and Lemma 31, the triangle inequality (93) becomes:

$$\mathrm{Err}_k \leq \sqrt{\delta_0}\ h + \left(\frac{M(\theta_{k-1}, h) + N(\theta_{k-1}, h)}{2} + \beta L_{k-1}(\theta_{k-1}, h)\right) h^2 + e^{-\lambda h}\ \mathrm{Err}_{k-1}.$$

Let us denote:

$$C_N = \max_{0\leq k\leq N-1} \left\{\frac{M(\theta_{k-1}, h) + N(\theta_{k-1}, h)}{2} + \beta L_{k-1}(\theta_{k-1}, h)\right\}. \tag{107}$$

Then we have:

$$\mathrm{Err}_k \leq \sqrt{\delta_0} h + C_N h^2 + e^{-\lambda h}\mathrm{Err}_{k-1} \tag{108}$$

Multiply $e^{\lambda kh}$ to both sides of (108), we get:

$$e^{\lambda kh}\mathrm{Err}_k \leq (\sqrt{\delta_0}\ h + C_N\ h^2)e^{\lambda kh} + e^{\lambda(k-1)h}\mathrm{Err}_{k-1}. \tag{109}$$

For any n, $1 \leq n \leq N$, summing (109) from 1 to n:

$$e^{\lambda nh}\mathrm{Err}_n \leq (\sqrt{\delta_0} h + C_N h^2)\left(\sum_{k=1}^{n} e^{\lambda kh}\right) + \mathrm{Err}_0 = (\sqrt{\delta_0} h + C_N h^2)\frac{e^{\lambda(n+1)h} - e^{\lambda h}}{e^{\lambda h} - 1} + \mathrm{Err}_0.$$

Recall each $t_n = nh$ for $1 \leq n \leq N$, it leads to:

$$\mathrm{Err}_n \leq (\sqrt{\delta_0} h + C_N h^2)\frac{1 - e^{-\lambda t_n}}{1 - e^{-\lambda h}} + e^{-\lambda t_n}\mathrm{Err}_0 \quad n = 1, ..., N.$$

□

Theorem 28 indicates that the error $W_2(\rho_{\theta_k}, \rho_{t_k})$ is upper bounded by $O(\sqrt{\delta_0}) + O(C_N h) + O(W_2(\rho_{\theta_0}, \rho_0))$. Here $O(\sqrt{\delta_0})$ is the essential error term that originates from the approximation mechanism of our parametric Fokker-Planck equation; the $O(C_N h)$ error term is induced by the finite difference scheme; the $O(W_2(\rho_{\theta_0}, \rho_0))$ term is the initial error.

It worth mentioning that the error bound for forward-Euler scheme in (92) matches the error bound for the continuous scheme (79) as we remove the effects introduced by finite difference. To be more precise, under the assumption $\lim_{h\to 0} C_N h = 0$, we have:

$$\begin{aligned}&\lim_{h\to 0}(\sqrt{\delta_0} h + C_N h^2)\frac{1 - e^{-\lambda t}}{1 - e^{-\lambda h}} + e^{-\lambda t}W_2(\rho_{\theta_0}, \rho_0)\\ =&\lim_{h\to 0}(\sqrt{\delta_0} + C_N h)(1 - e^{-\lambda t})\frac{h}{1 - e^{-\lambda h}} + e^{-\lambda t}W_2(\rho_{\theta_0}, \rho_0) = \frac{\sqrt{\delta_0}}{\lambda}(1 - e^{-\lambda t}) + e^{-\lambda t}W_2(\rho_{\theta_0}, \rho_0)\end{aligned}$$

this indicates that error bounds (92) and (79) are compatible as $h \to 0$.

Similar to the discussion in previous sections, we can naturally extend Theorem 28 to posterior version:

Theorem 33 (Posterior error analysis of forward-Euler scheme)**.** *Suppose we keep all the notations in Theorem 28. Recall δ_1 defined in (70). Then we have:*

$$W_2(\rho_{\theta_k}, \rho_{t_k}) \leq (\sqrt{\delta_1} h + C_N h^2)\frac{1 - e^{-\lambda t_k}}{1 - e^{-\lambda h}} + e^{-\lambda t_k}W_2(\rho_{\theta_0}, \rho_0) \quad \textit{for any} \quad t_k = kh, \quad 0 \leq k \leq N.$$

The explicit definition of the constant C_N is in (107).

It worth mentioning that in section 5.3.2, we mainly analyze the error term for the forward-Euler (explicit) scheme. However, in our actual implementation, we use the scheme (45), which can be treated as the semi-implicit scheme (with $O(h^2)$ local error). The following theorem compares the difference between the numerical solution of forward-Euler scheme and semi-implicit scheme.

Theorem 34 (Relation between forward-Euler scheme and semi-implicit scheme)**.** *Recall the parametric Fokker-Planck equation (27) as an ODE: $\dot{\theta} = G(\theta)^{-1}\nabla_\theta H(\theta)$. We consider two numerical schemes:*

$$\theta_{n+1} = \theta_n - hG(\theta_n)^{-1}\nabla_\theta H(\theta_n) \quad \theta_0 = \theta,\ n = 1, 2, ..., N \quad \textit{Forward-Euler scheme}; \tag{110}$$

$$\hat{\theta}_{n+1} = \hat{\theta}_n - hG(\hat{\theta}_n)^{-1}\nabla_\theta H(\hat{\theta}_{n+1}) \quad \hat{\theta}_0 = \theta,\ n = 1, 2, ..., N \quad \textit{Semi-Implicit-Euler scheme} \tag{111}$$

We denote $F(\theta') = G(\theta')^{-1}\nabla_\theta F(\theta'')$, we set:

$$L_1 = \max_{1\le n\le N}\left\{\|F(\theta_n) - F(\hat{\theta}_n)\|/\|\theta_n - \hat{\theta}_n\|\right\} \quad L_2 = \max_{1\le k\le N-1}\{\|\nabla_\theta H(\hat{\theta}_n) - \nabla_\theta H(\hat{\theta}_{n+1})\|/\|\hat{\theta}_n - \hat{\theta}_{n+1}\|\}$$

$$M_1 = \max_{1\le n\le N}\{\|G(\hat{\theta}_n)^{-1}\|\}, \quad M_2 = \max_{1\le n\le N}\{\|\nabla_\theta H(\hat{\theta}_n)\|\}$$

Here $\|\|$ is a certain vector norm (or its corresponding matrix norm). Then we have:

$$\|\theta_n - \hat{\theta}_n\| \le ((1 + L_1 h)^n - 1)\frac{M_1^2 M_2 L_2}{L_1}h \quad n = 1, 2, ..., N$$

If we assume that we are solving the ODE on $[0, T]$ with time stepsize h, i.e. $Nh = T$, all the differences $\|\theta_n - \hat{\theta}_n\|$ can be upper bounded by $(e^{L_1 T} - 1)\frac{M_1^2 M_2 L_2}{L_1}h$.

When the upper bounds $L_1, L_2, M_1, M_2 \sim O(1)$ as $h \to 0$ (or equivalently $N \to \infty$), then the differences between the semi-implicit scheme and forward-Euler scheme can be bounded by $O(h)$. Hence, we are still able to establish $O(h)$ error bound for our proposed scheme (45).

proof of Theorem 34. We subtract (111) from (110):

$$(\theta_{n+1} - \hat{\theta}_{n+1}) = (\theta_n - \hat{\theta}_n) - h(G(\theta_n)^{-1}\nabla_\theta H(\theta_n) - G(\hat{\theta}_n)^{-1}\nabla_\theta H(\hat{\theta}_{n+1}))$$

denote $e_n = \theta_n - \hat{\theta}_n$, we may rewrite this equation as:

$$e_{n+1} = e_n - h(F(\theta_n) - F(\hat{\theta}_n) + G(\hat{\theta}_n)^{-1}(\nabla_\theta H(\hat{\theta}_n) - \nabla_\theta H(\hat{\theta}_{n+1})))$$

Recall the definitions of L_1, L_2, M_1, we have

$$\|e_{n+1}\| \le \|e_n\| + hL_1\|e_n\| + hM_1L_2\|\hat{\theta}_{n+1} - \hat{\theta}_n\|$$

By semi-simplicit scheme, we have

$$\hat{\theta}_{n+1} - \hat{\theta}_n = -hG(\hat{\theta}_n)^{-1}\nabla_\theta H(\hat{\theta}_{n+1})$$

Then $|\hat{\theta}_{n+1} - \hat{\theta}_n\| \le hM_1M_2$. Now we have recurrent inequality:

$$\|e_{n+1}\| \le \|e_n\| + hL_1\|e_n\| + M_1^2 M_2 L_2 h^2$$

This inequality gives

$$\left(\|e_{n+1}\| + \frac{M_1^2 M_2 L_2}{L_1}h\right) \le (1 + hL_1)\left(\|e_n\| + \frac{M_1^2 M_2 L_2}{L_1}h\right) \quad n = 0, 1, ..., N-1$$

This will lead to:

$$\|e_n\| \le ((1 + hL_1)^n - 1)\frac{M_1^2 M_2 L_2}{L_1}h$$

When we are solving the ODE on $[0, T]$ with $h = T/N$, we have $(1 + hL_1)^n \le (1 + hL_1)^N = \left(1 + \frac{L_1 T}{N}\right)^N \le e^{L_1 T}$. This means all terms $\{\|e_n\|\}_{1\le n\le N}$ can be upper bounded by $(e^{L_1 T} - 1)\frac{M_1^2 M_2 L_2}{L_1}h$. □

We end this section with the following two remarks:

Remark 13. *In order to make our argument clear and concise, we omitted the errors introduced by the approximation of ReLU function ψ_λ. Careful analysis on how well $\nabla\psi_\lambda$ can approximate general gradient fields may serve as one of our future research directions.*

Remark 14. *The convergence property of the Stochastic Gradient Descent methods (mainly Adam method [20]) used in our Algorithm 1 are not discussed in details. One can check the detailed convergence analysis in the paper [20].*

6 Numerical examples

In this section, we consider solving Fokker-Planck equation (5)

$$\frac{\partial \rho}{\partial t} = \nabla \cdot (\rho \nabla V) + \beta \Delta \rho.$$

on $\mathbb{R}^d$ with $\beta = 1$ and initial condition $\rho_0(x) = \mathcal{N}(0, I_d)$[3] by using Algorithm 1. We demonstrate several numerical examples that solves (5) with different potential functions V.

In the following experiments, we choose the length of normalizing flow T_θ as 60. And we set $\psi_\lambda : \mathbb{R}^d \to \mathbb{R}$ as ReLU network with length 6 and hidden dimension 20. We use Adam (Adaptive Moment Estimation) Stochastic Gradient Descent method [20] with default $\beta_1 = 0.9, \beta_2 = 0.999; \epsilon = 10^{-8}$.

For the parameters of Algorithm 1, we choose $\alpha_{\text{out}} = 0.005$, $\alpha_{\text{in}} = 0.0005$. We follow Remark 9 to choose $K_{\text{in}}, K_{\text{out}} = \max\{1000, 300d\}$. Based on our experience, we set $M_{\text{out}} = O(\frac{h}{\alpha_{\text{out}}})$; the suitable value of M_{in} can be chosen after several quick tests of different choices of M_{in}–We need to make sure that every inner optimization problem (64) can be solved thoroughly.

6.1 Quadratic Potential

We first apply our method to Fokker-Planck equation (5) with quadratic potential V. We can compute for the explicit solution of (5) when V is quadratic, so these examples can serve as verifications of our proposed method.

6.1.1 2D cases

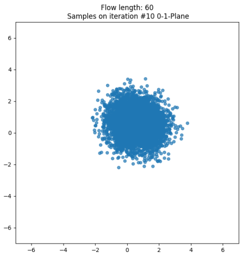
Suppose $d = 2$. We set $V(x) = \frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)$, we let $\mu = \begin{bmatrix} 3 \\ 3 \end{bmatrix}$ and $\Sigma = \begin{bmatrix} \frac{1}{4} & \\ & \frac{1}{4} \end{bmatrix}$. We can explicitly solve (5) in this case:

$$\rho_t = \mathcal{N}(\mu_t, \Sigma_t) \quad \mu_t = (1 - e^{-4t})\mu, \;\; \Sigma_t = \left(\frac{1}{4} + \frac{3}{4}e^{-8t}\right)\Sigma \quad t \geq 0.$$

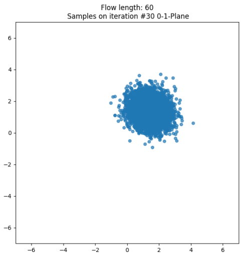
In our algorithm, we consider solving the equation on $[0, 0.7]$ with time stepsize 0.005. We set $M_{\text{out}} = 20$ and $M_{\text{in}} = 100$. Here are the results. At a given time t_k, we draw 6000 samples from reference distribution p and pushforward them by using the map T_{θ_k}, here θ_k is the value of θ at k-th time step solved from ODE (27) by using our proposed algorithm. We demonstrate the the pushforwarded points below (from $t = 0.05$ to $t = 0.70$):

One can check that the distribution of our numerical computed samples is gradually converging to the Gibbs distribution $\mathcal{N}(\mu, \Sigma)$.

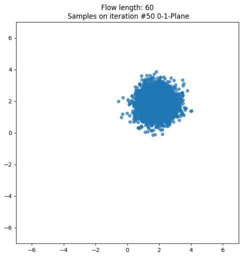
[3] We can set initial value θ_0 so that $T_{\theta_0} = Id$ and thus $\rho_0 = {T_{\theta_0}}_{\#}p$ is standard Gaussian distribution.



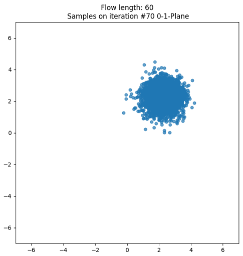
t=0.05



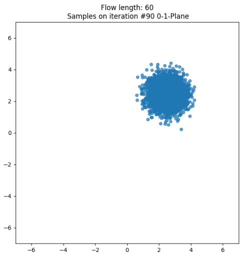
t=0.15



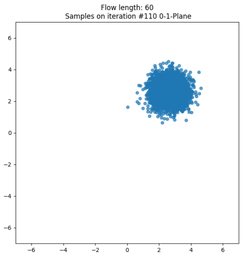
t=0.25



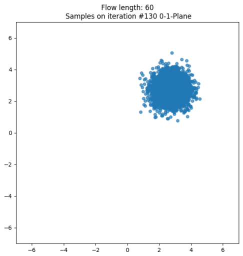
t=0.35



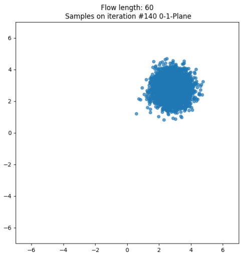
t=0.45



t=0.55



t=0.65



t=0.70

At each time node t_k, we sample $\{\boldsymbol{X}_1, ..., \boldsymbol{X}_M\} \sim T_{\theta_k \#} p$ and use $\hat{\mu}^k = \frac{1}{M}\sum_{j=1}^M \boldsymbol{X}_j$, $\hat{\Sigma}^k = \frac{1}{M-1}\sum_{j=1}^M (\boldsymbol{X}_j - \hat{\mu}_k)(\boldsymbol{X}_j - \hat{\mu}_k)^T$ to compute for the empirical mean and covariance of $\hat{\rho}_k$ at t_k. We then can plot the curve $\{\hat{\mu}^{(k)}\}$, $\{(\hat{\Sigma}_{11}^{(k)}, \hat{\Sigma}_{22}^{(k)})\}$, $\{(\hat{\mu}_1^{(k)}, \hat{\Sigma}_{11}^{(k)})\}$ and then compare them with the explicit solution $\{\mu_t\}$, $\{(\Sigma_{t11}, \Sigma_{t22})\}$, $\{(\mu_{t1}, \Sigma_{t11})\}$. Recall that $\mu_{t1} = \mu_{t2} = 3(1 - e^{-4t})$, $\Sigma_{t11} = \Sigma_{t22} = \frac{1}{4}(1 - 3e^{-8t})$.

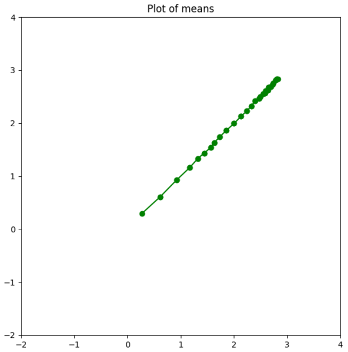


Figure 6: $\{\hat{\mu}^{(k)}\}$

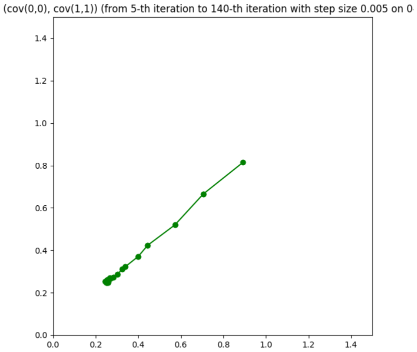


Figure 7: $\{(\hat{\Sigma}_{11}^{(k)}, \hat{\Sigma}_{22}^{(k)})\}$

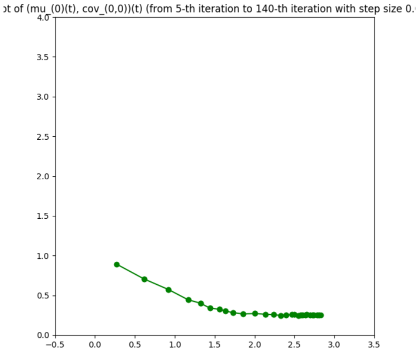


Figure 8: $\{(\hat{\mu}_{(k)}, \hat{\Sigma}_{11}^{(k)})\}$

We can directly evaluate the error between $\hat{\mu}^{(k)}$ and μ_{t_k}; $\hat{\Sigma}^{(k)}$ and Σ_{t_k}. We plot the error curve of $\|\hat{\mu}^{(k)} - \mu_{t_k}\|_2$ (Figure 9) and $\|\hat{\Sigma}^{(k)} - \Sigma_{t_k}\|_F$ (Figure 10). Here $\|\cdot\|_F$ is the Frobenius norm of the a matrix.

Figure 11 captures the exponential decay of H along its Wasserstein gradient flow, this verifies the entropy dissipation property of Fokker-Planck equation with convex potential function V.

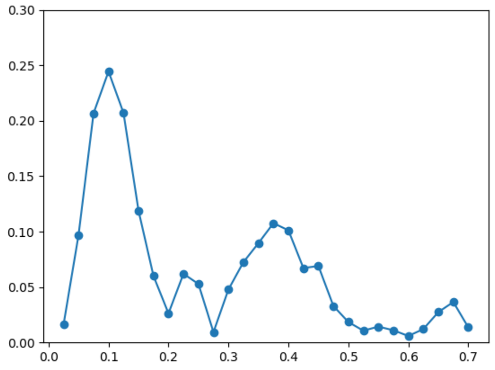


Figure 9: Plot of mean value error (in l_2 norm)

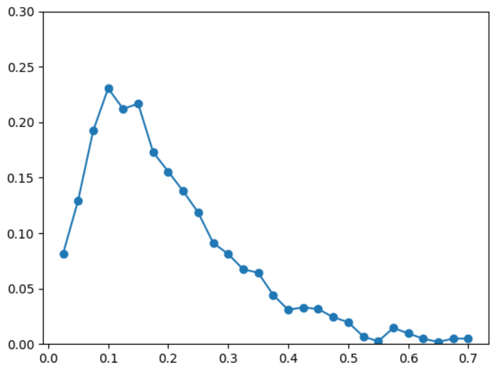


Figure 10: Plot of covariance error (in Frobenius norm)

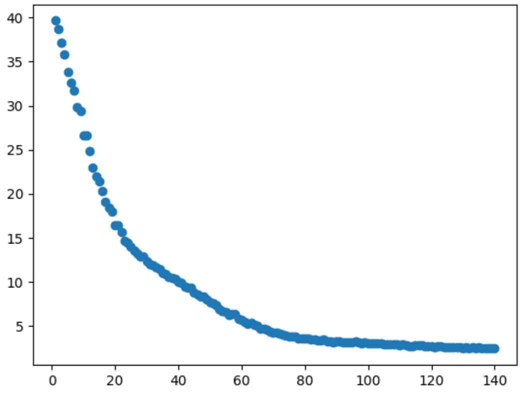
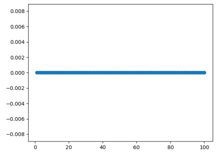
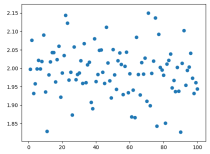


Figure 11: Plot of $\{H(\theta)\}$

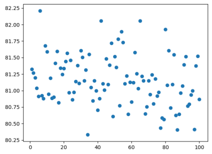
We can also take a closer look at the inner loops loss (Figure 12). The following figures are the first 10 (out of 20) loss plots when applying SGD method to solve (66) when $k = 30$ ($t = 30 \cdot h = 0.15$).



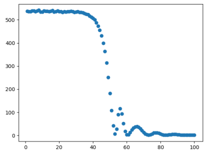
1st inner iteration



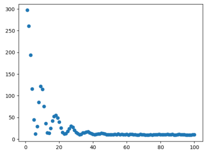
2nd inner iteration



3rd inner iteration

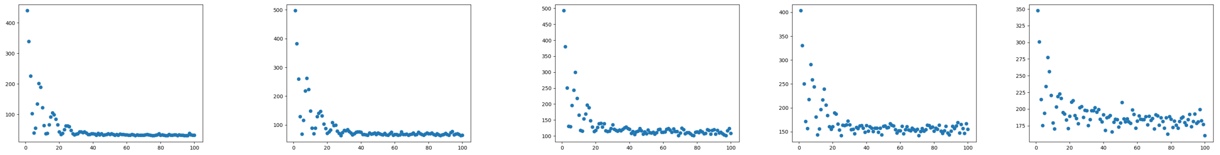


4th inner iteration



5th inner iteration

The remaining loss plots from the 11th outer iteration to 20th iteration are similar to the second row plots. The situations are similar for other time step k. We can thus tell that $M_{\text{in}} = 100$ works well in this problem, the SGD method we used can thoroughly solve the variational problem (66) for each outer loop. Whether M_{out} is suitable for our algorithm remains a hard problem since the function $\tilde{J}(\theta)$ we used in computation is not the functional $J(\theta)$ that we really minimizes. In our computations, we set $M_{\text{out}} = 2\frac{h}{\alpha_{\text{out}}}$ based on our experiences. Our choice of M_{out} provide valid results to most of the numerical experiments done by us.



6th inner iteration 7th inner iteration 8th inner iteration 9th inner iteration 10th inner iteration

Figure 12: Plots of inner loop losses

At last, let us verify the graph of $\psi_{\hat{\lambda}}$ trained at the end of each outer iteration. Generally speaking, the gradient field $\nabla\psi_{\hat{\lambda}}$ reflects the movements of the particles under the Vlasov-typed dynamic (6) at every time step. Here are the graph of $\psi_{\hat{\lambda}}$ at $k = 10, k = 140$ (Figure 13, Figure 14).

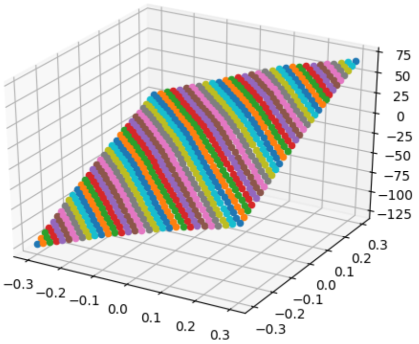


Figure 13: Graph of $\psi_{\hat{\lambda}}$ after $M_{\text{out}} = 20$ outer iterations at $k = 10$th time step

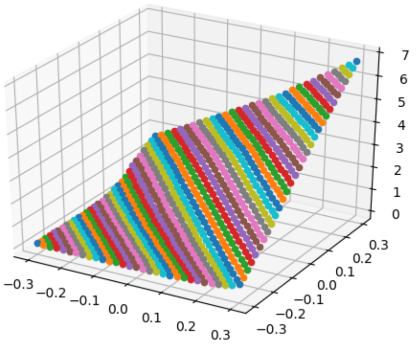


Figure 14: Graph of $\psi_{\hat{\lambda}}$ after $M_{\text{out}} = 20$ outer iterations at $k = 140$th time step

As we can see from these two graphs of $\psi_{\hat{\lambda}}$, the gradient field is in the same direction, but judging from the variation of two $\psi_{\hat{\lambda}}$s, when $k = 10$, $|\nabla\psi_{\hat{\lambda}}|$ is much greater than itself when $k = 140$. This is because when $t = 140$, the distribution is already close to the Gibbs distribution, the particles no longer need to move for a long distance to reach their final destination.

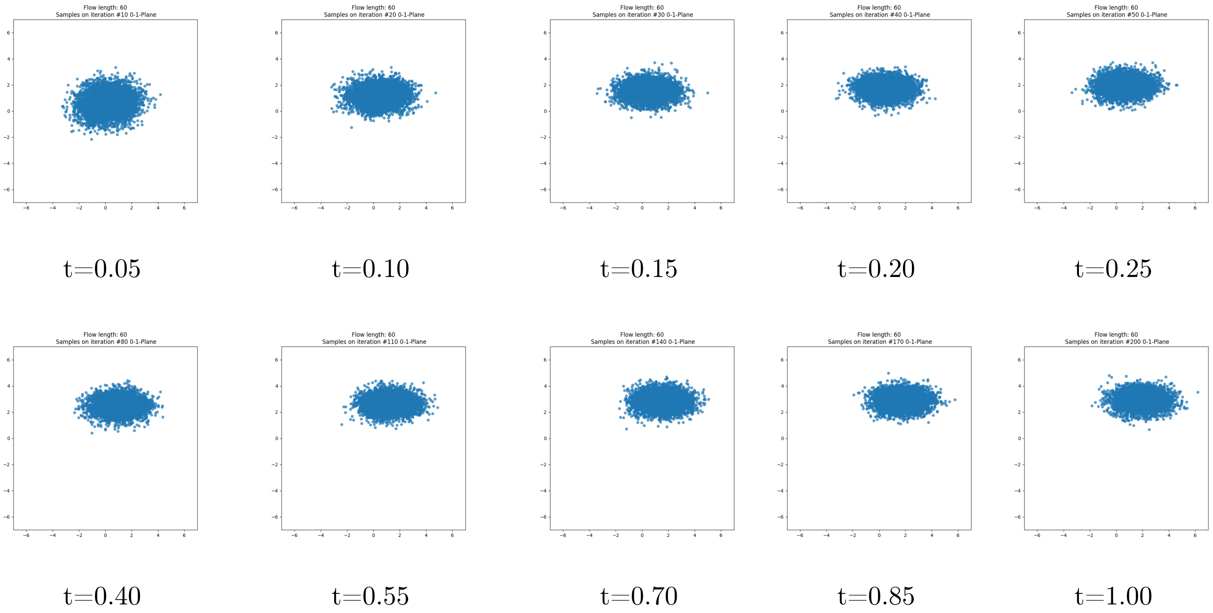
We apply our algorithm to the Fokker-Planck equation with non-isotropic potential:

$$V(x) = \frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu) \quad \mu = \begin{bmatrix} 3 \\ 3 \end{bmatrix} \text{ and } \Sigma = \begin{bmatrix} 1 & \\ & \frac{1}{4} \end{bmatrix}.$$

One can verify that the solution to (5) with such V is

$$\rho_t = \mathcal{N}(\mu_t, \Sigma_t) \quad \mu_t = \begin{bmatrix} 3(1-e^{-t}) \\ 3(1-e^{-4t}) \end{bmatrix}, \ \Sigma_t = \begin{bmatrix} 1 & \\ & \frac{1}{4}(1+3e^{-8t}) \end{bmatrix}.$$

We use the same parameters for our algorithm as before. We solve (5) on $[0, 1.4]$ with time step size 0.005. Here are the sample results at different time steps.



t=0.05 t=0.10 t=0.15 t=0.20 t=0.25

t=0.40 t=0.55 t=0.70 t=0.85 t=1.00

Similarly, we can also plot the empirical mean trajectory, one can compare it with the true solution $(3(1-e^{-t}), 3(1-e^{-4t}))$. Both the curvature and the exponential convergence to μ are captured by our numerical result. Here we can also compare Figure 15 with Figure 16, which is the mean trajectory obtained by computing the flat gradient flow $\dot{\theta} = -\nabla_\theta H(\theta)$. This reveals very different behavior of the flat gradient (∇_θ) flow and Wasserstein gradient $(G(\theta)^{-1}\nabla_\theta)$ flow.

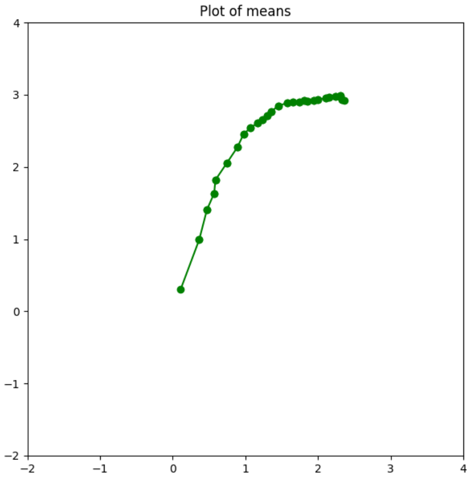


Figure 15: mean trajectory of $\{\rho_{\theta_t}\}$ w.r.t. $\dot{\theta} = -G(\theta)^{-1}\nabla_\theta H(\theta)$

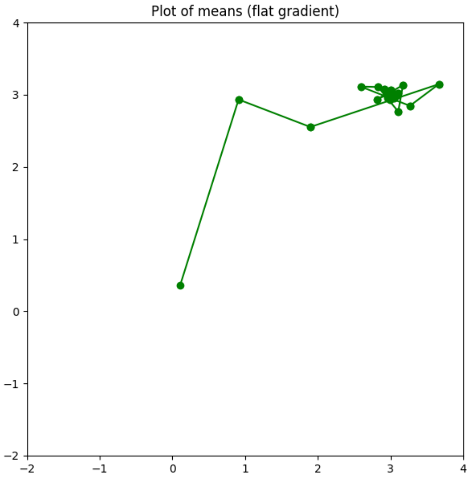


Figure 16: mean trajectory of $\{\rho_{\theta_t}\}$ w.r.t. $\dot{\theta} = -\nabla_\theta H(\theta)$

We plot the error curve of $\|\hat{\mu}^{(k)} - \mu_{t_k}\|_2$ (Figure 17) and $\|\hat{\Sigma}^{(k)} - \Sigma_{t_k}\|_F$ (Figure 18):

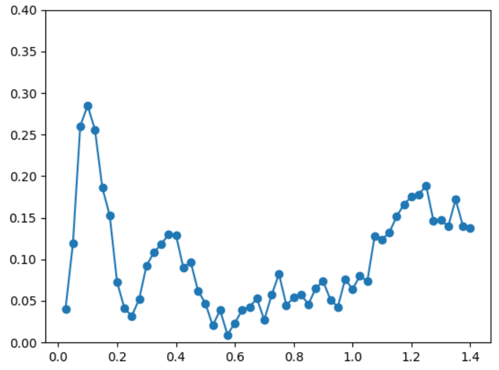


Figure 17: Plot of mean value error (in l_2 norm)

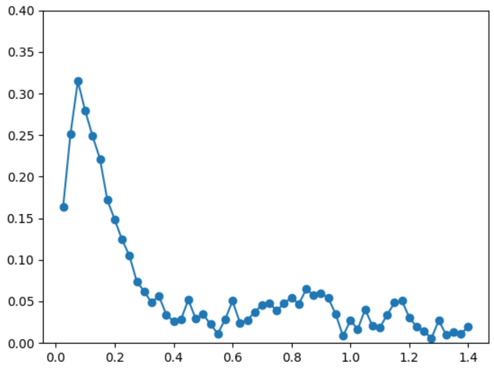


Figure 18: Plot of covariance error (in Frobenius norm)

The exponential decay of $\{H(\theta_k)\}$ is very similar to the isotropic case. $\{H(\theta_k)\}$ also shows exponential decay. And $M_{\text{out}} = 20, M_{\text{in}} = 100$ also works well in this problem.
It is also interesting to compare the graph of trained $\psi_{\hat{\lambda}}$ at different time steps $k = 10, 140$ (Figure 19, 20) with that of the previous example: The directions of $\nabla\psi_{\hat{\lambda}}$ at $k = 10$ and $k = 140$ is different from the previous example. This is caused by the non-isotropic quadratic (Gaussian) potential V used in this problem.

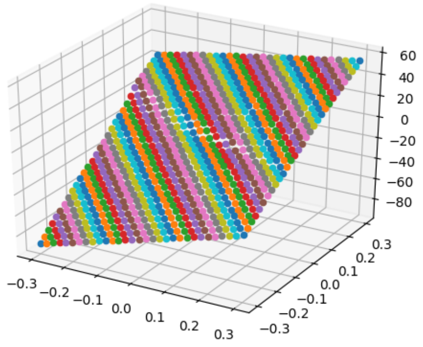


Figure 19: Graph of $\psi_{\hat{\lambda}}$ after $M_{\text{out}} = 20$ outer iterations at $k = 10$th time step

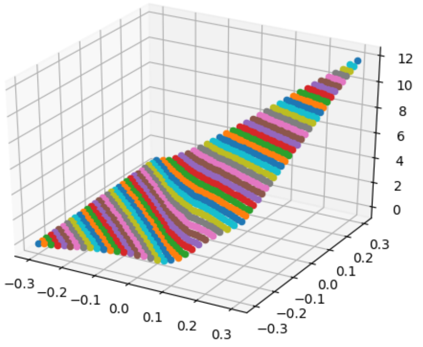


Figure 20: Graph of $\psi_{\hat{\lambda}}$ after $M_{\text{out}} = 20$ outer iterations at $k = 140$th time step

6.1.2 Higher dimension

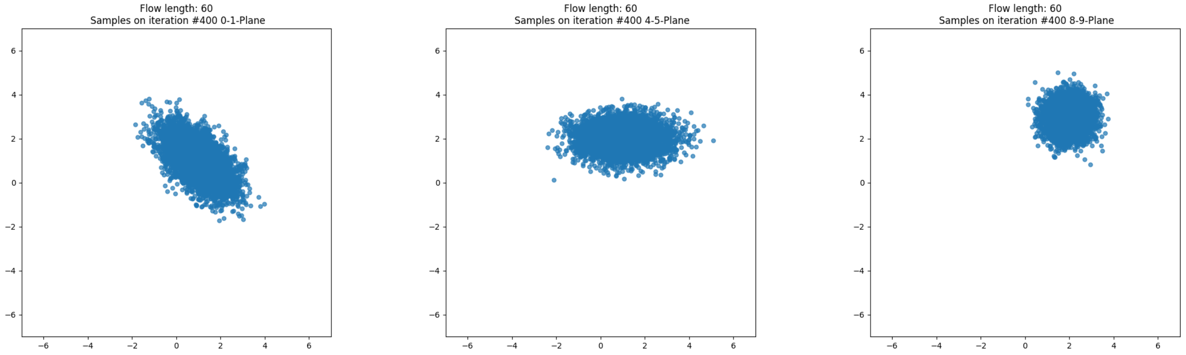
We can implement our algorithm in higher dimensional space, we try $d = 10$: consider the quadratic potential

$$V(x) = \frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu) \quad \Sigma = \text{diag}(\Sigma_A, I_2, \Sigma_B, I_2, \Sigma_C) \quad \mu = (1,1,0,0,1,2,0,0,2,3)^T.$$

Here we set the diagonal blocks as:

$$\Sigma_A = \begin{bmatrix} \frac{5}{8} & -\frac{3}{8} \\ -\frac{3}{8} & \frac{5}{8} \end{bmatrix} \quad \Sigma_B = \begin{bmatrix} 1 & \\ & \frac{1}{4} \end{bmatrix} \quad \Sigma_C = \begin{bmatrix} \frac{1}{4} & \\ & \frac{1}{4} \end{bmatrix}.$$

We solve (5) on $[0,2]$ with time step size $h = 0.005$. We set $K_{\text{in}} = K_{\text{out}} = 3000$ and choose $M_{\text{out}} = 30$, $M_{\text{in}} = 100$.
Here are the samples at the last time step $k = 400$, we exhibit the projection of the samples on $0-1$, $4-5$ and $8-9$ plane in Figure 21.



projection of samples on 0-1 plane   projection of samples on 4-5 plane   projection of samples on 8-9 plane

Figure 21: Plot of samples on different planes

6.2 Experiments with more general potentials

In this section, we exhibit two examples with more general potentials in higher dimensional space.

6.2.1 Styblinski-Tang potential

In this example, we set dimension $d = 30$. We consider the Styblinski–Tang function [49]:

$$V(x) = \frac{3}{50}\left(\sum_{i=1}^{d} x_i^4 - 16x_i^2 + 5x_i\right).$$

Here are the plot and heat map of V when dimension $d = 2$:

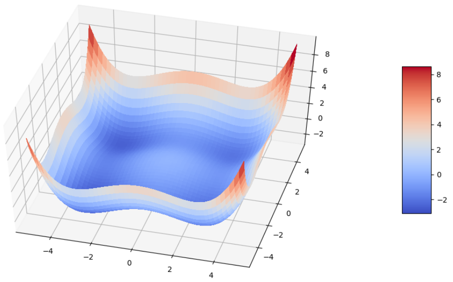


Figure 22: Styblinski–Tang function

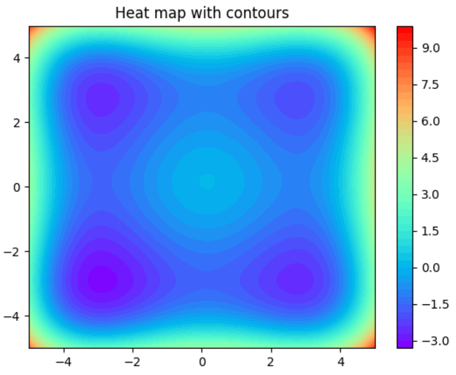
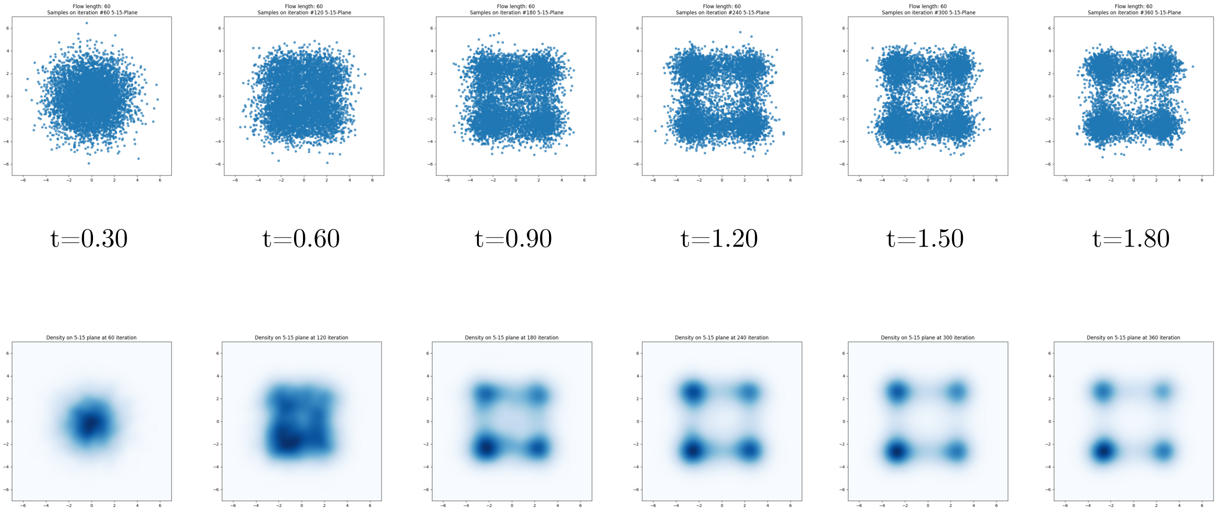


Figure 23: Heat map

We solve (5) with potential V on $[0,3]$ with time step size $h = 0.005$; we set $K_{\text{in}} = K_{\text{out}} = 9000$ and $M_{\text{in}} = 100$, $M_{\text{out}} = 30$.

To exhibit sample results, due to the symmetricity of the potential function, we just project the sample points in $\mathbb{R}^{30}$ to some random plane. Here we project the samples to $5-15$ plane. The sample plots and their estimated densities are presented in Figure 24.

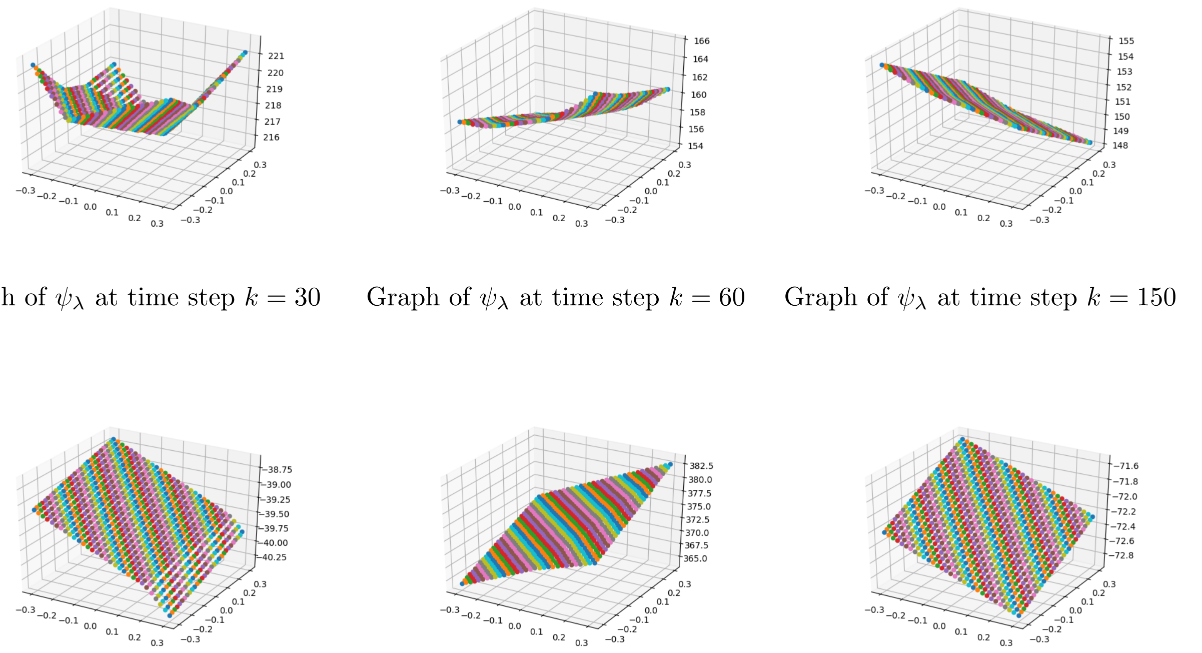


t=0.30 t=0.60 t=0.90 t=1.20 t=1.50 t=1.80

t=0.30 t=0.60 t=0.90 t=1.20 t=1.50 t=1.80

Figure 24: Plot of samples and estimated densities on $5-15$ plane

We also exhibit the graphs of ψ_λ on $5-15$ plane trained at different time steps in Figure 25.



Graph of ψ_λ at time step $k = 30$ Graph of ψ_λ at time step $k = 60$ Graph of ψ_λ at time step $k = 150$

Graph of ψ_λ at time step $k = 240$ Graph of ψ_λ at time step $k = 300$ Graph of ψ_λ at time step $k = 360$

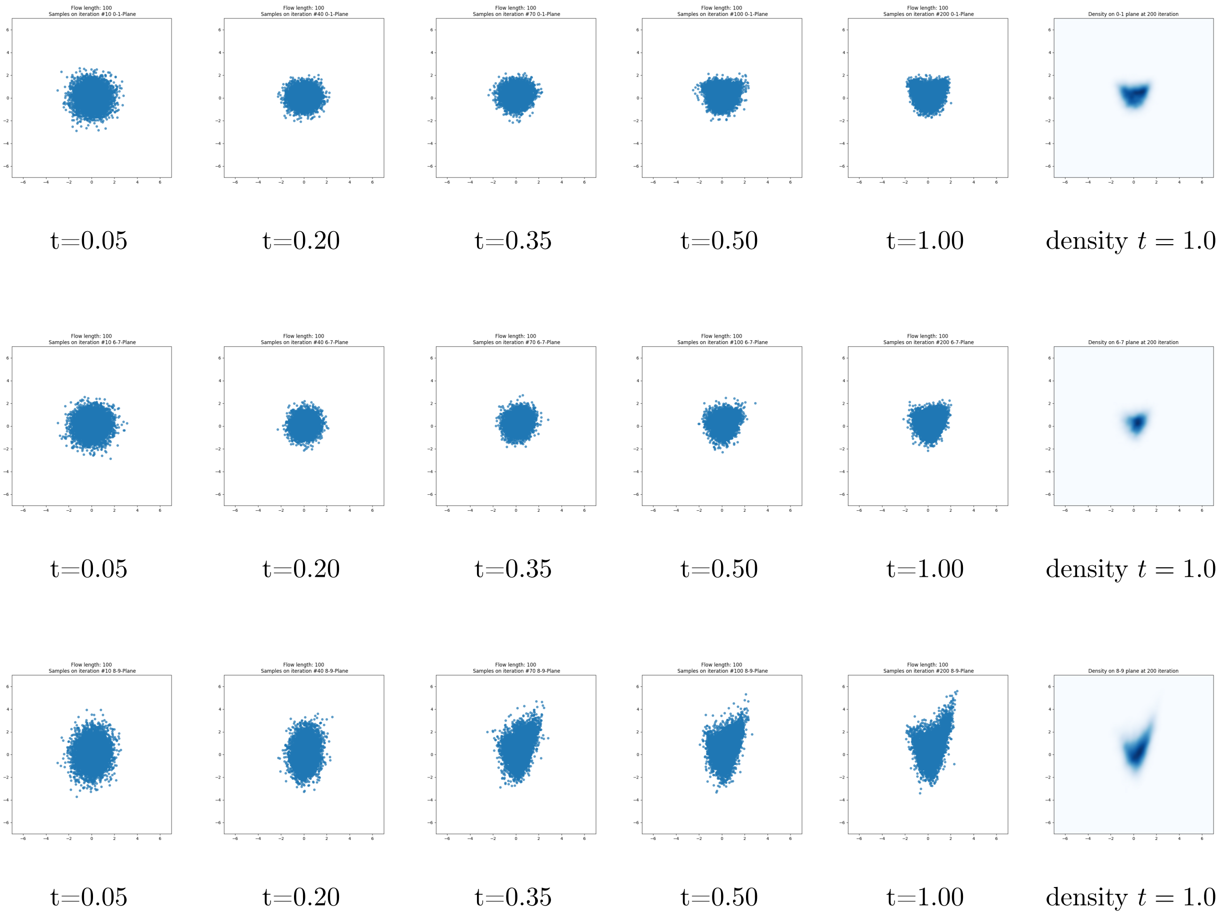
Figure 25: Graph of ψ_λ on $5-15$ plane trained at different time steps

6.2.2 Rosenbrock potential

In the previous example, $V(x)$ is the direct sum of same functions and can be treated as a potential without interactions. Now we consider more general function V involving interaction among its coordinates. In this example, we set dimension $d = 10$. We consider the Rosenbrock typed function [45]:

$$V(x) = \frac{3}{50}\left(\sum_{i=1}^{d-1} 10(x_{k+1} - x_k^2)^2 + (x_k - 1)^2\right).$$

We solve the corresponding (5) on $[0, 1]$ with step size $h = 0.005$. We set $K_{\text{in}} = K_{\text{out}} = 3000$ and $M_{\text{in}} = 100$, $M_{\text{out}} = 60$.
Here are the sample results, we exhibit the projection of sample points on the $1 - 2$, $7 - 8$ and $9 - 10$ plane in Figure 26. The rightmost figures are plots of estimated densities at $t = 1.0$.



t=0.05 t=0.20 t=0.35 t=0.50 t=1.00 density $t = 1.0$

t=0.05 t=0.20 t=0.35 t=0.50 t=1.00 density $t = 1.0$

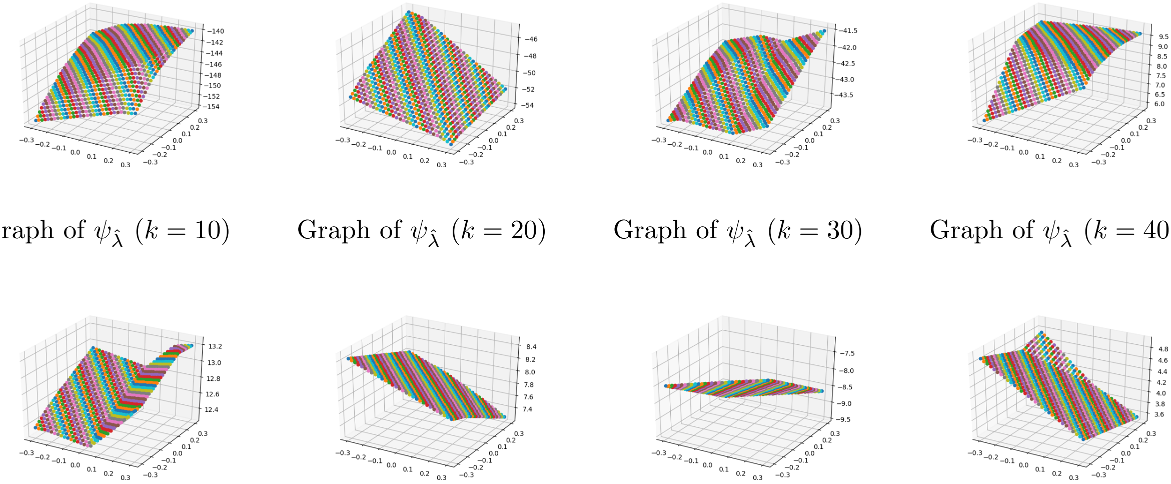
t=0.05 t=0.20 t=0.35 t=0.50 t=1.00 density $t = 1.0$

Figure 26: Plot of samples and estimated densities on different planes

We exhibit the graphs of ψ_λ on $0 - 1$ plane trained at different time steps in Figure 27:

7 Discussion

In this paper, we design and analyze an algorithm for computing high dimensional Fokker-Planck equations. Our approach is based on transport information geometry with probability models arisen in deep learning generative models. We first introduce a set of ODE to approximate the Fokker-Planck equation. This ODE can be viewed as the "spatial discretization" of the PDE from the neural networks. We next propose a



Graph of $\psi_{\hat{\lambda}}$ $(k = 10)$ Graph of $\psi_{\hat{\lambda}}$ $(k = 20)$ Graph of $\psi_{\hat{\lambda}}$ $(k = 30)$ Graph of $\psi_{\hat{\lambda}}$ $(k = 40)$

Graph of $\psi_{\hat{\lambda}}$ $(k = 80)$ Graph of $\psi_{\hat{\lambda}}$ $(k = 120)$ Graph of $\psi_{\hat{\lambda}}$ $(k = 160)$ Graph of $\psi_{\hat{\lambda}}$ $(k = 200)$

Figure 27: Graph of ψ_λ on $0 - 1$ plane trained at different time steps

variational version of the semi-implicit Euler scheme to design a discrete-time update of the proposed ODE. This method has a sampling efficient approach and can be viewed as the JKO scheme in neural networks. We last prove the asymptotic convergence and error analysis results for our proposed schemes.

Our study opens a door for systemically applying the deep neural networks and machine learning approach to compute physical partial differential equations. It is worth mentioning that KL divergence and Wasserstein metric can be naturally formulated in machine learning models. In computational schemes, following the proposed dynamical systems, it will provide a more systemic way of designing sampling efficient algorithms. The other benefit is that our approach does not require any knowledge of the ”data” from the partial differential equation. It is the same as the classical numerical schemes, in which we generate the “data solution" to compute the numerical solution. More importantly, our computation can keep the physical law, such as relative entropy dissipation, in neural network parameters. In numerical analysis, transport information geometry provides a mathematical framework for studying the convergence of algorithms. Here, the asymptotic convergence and error analysis proof of our scheme follows how do the KL divergence and the Wasserstein metric measures the discrepancy between the gradient flow in deep learning generative models and the one in full probability space. We notice that the Wasserstein metric provides a suitable metric structure to analyze the convergence behavior in generative models.

In the future, we shall study the computation of gradient flows raised in transport information geometry. Examples include Poros media equation and aggregation equations etc. Besides, we shall extend the current study to compute Hamiltonian flows in transport information geometry. There are several examples, such as Schrödinger equation, Schrödinger bridge system, and compressible Euler equation, etc.

References

[1] S Amari. Natural Gradient Works Efficiently in Learning. *Neural Computation*, 10(2):251–276, 1998.

[2] S Amari. *Information Geometry and Its Applications.* Number volume 194 in Applied Mathematical Sciences. Springer, Japan, 2016.

[3] Luigi Ambrosio, Nicola Gigli, and Giuseppe Savaré. *Gradient flows: in metric spaces and in the space of probability measures.* Springer Science & Business Media, 2008.

[4] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *International Conference on Machine Learning*, pages 214–223, 2017.

[5] Nihat Ay, Jürgen Jost, Hông Vân Lê, and Lorenz Johannes Schwachhöfer. *Information Geometry*. Ergebnisse Der Mathematik Und Ihrer Grenzgebiete A @series of Modern Surveys in Mathematics$l3. Folge, Volume 64. Springer, Cham, 2017.

[6] Tamer Başar and Pierre Bernhard. *H-infinity optimal control and related minimax design problems: a dynamic game approach*. Springer Science & Business Media, 2008.

[7] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*, 2018.

[8] Daniel A Cervantes Cabrera, Pedro Gonzalez-Casanova, Christian Gout, L Héctor Juárez, and L Rafael Reséndiz. Vector field approximation using radial basis functions. *Journal of Computational and Applied Mathematics*, 240:163–173, 2013.

[9] Jose A Carrillo, Katy Craig, Li Wang, and Chaozhen Wei. Primal dual methods for wasserstein gradient flows. *arXiv preprint arXiv:1901.08081*, 2019.

[10] JS Chang and G Cooper. A practical difference scheme for fokker-planck equations. *Journal of Computational Physics*, 6(1):1–16, 1970.

[11] Joseph L Doob. The brownian movement and stochastic equations. *Annals of Mathematics*, pages 351–369, 1942.

[12] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Deep sparse rectifier neural networks. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 315–323, 2011.

[13] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[14] Ulf Grenander and Michael I Miller. Representations of knowledge in complex systems. *Journal of the Royal Statistical Society: Series B (Methodological)*, 56(4):549–581, 1994.

[15] Richard Holley and Daniel Stroock. Logarithmic sobolev inequalities and stochastic ising models. *Journal of statistical physics*, 46(5):1159–1194, 1987.

[16] R. Jordan, D. Kinderlehrer, and F. Otto. The Variational Formulation of the Fokker–Planck Equation. *SIAM Journal on Mathematical Analysis*, 29(1):1–17, 1998.

[17] Richard Jordan, David Kinderlehrer, and Felix Otto. The variational formulation of the fokker–planck equation. *SIAM journal on mathematical analysis*, 29(1):1–17, 1998.

[18] Yuehaw Khoo, Jianfeng Lu, and Lexing Ying. Solving parametric pde problems with artificial neural networks. *arXiv preprint arXiv:1707.03351*, 2017.

[19] Yuehaw Khoo, Jianfeng Lu, and Lexing Ying. Solving for high-dimensional committor functions using artificial neural networks. *Research in the Mathematical Sciences*, 6(1):1, 2019.

[20] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[21] Pankaj Kumar and S Narayanan. Solution of fokker-planck equation by finite element and finite difference methods for nonlinear systems. *Sadhana*, 31(4):445–461, 2006.

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

[23] Wuchen Li. Geometry of probability simplex via optimal transport. *arXiv:1803.06360 [math]*, 2018.

[24] Wuchen Li, Alex Tong Lin, and Guido Montúfar. Affine natural proximal learning. 2019.

[25] Wuchen Li, Shu Liu, Hongyuan Zha, and Haomin Zhou. Parametric fokker-planck equation. In Frank Nielsen and Frédéric Barbaresco, editors, *Geometric Science of Information*, pages 715–724, Cham, 2019. Springer International Publishing.

[26] Wuchen Li and Guido Montufar. Natural gradient via optimal transport. *arXiv:1803.07033 [cs, math]*, 2018.

[27] Wuchen Li and Guido Montufar. Ricci curvature for parametric statistics via optimal transport. *arXiv preprint arXiv:1807.07095*, 2018.

[28] Alex Tong Lin, Wuchen Li, Stanley Osher, and Guido Montufar. Wasserstein proximal of GANs, 2019.

[29] Qiang Liu and Dilin Wang. Stein Variational Gradient Descent: A General Purpose Bayesian Inference Algorithm. *arXiv:1608.04471 [cs, stat]*, 2016.

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

[31] James Martens and Roger Grosse. Optimizing neural networks with kronecker-factored approximate curvature. In *International conference on machine learning*, pages 2408–2417, 2015.

[32] Dominic Masters and Carlo Luschi. Revisiting small batch training for deep neural networks. *arXiv preprint arXiv:1804.07612*, 2018.

[33] Edward Nelson. *Quantum Fluctuations*. Princeton Series in Physics. Princeton University Press, Princeton, N.J, 1985.

[34] Felix Otto. The Geometry of Dissipative Evolution Equations: The Porous Medium Equation. *Communications in Partial Differential Equations*, 26(1-2):101–174, 2001.

[35] Felix Otto and Cédric Villani. Generalization of an inequality by talagrand and links with the logarithmic sobolev inequality. *Journal of Functional Analysis*, 173(2):361–400, 2000.

[36] Michele Pavon, Esteban G. Tabak, and Giulio Trigila. The data-driven Schroedinger bridge. *arXiv:1806.01364 [math]*, 2018.

[37] Philipp Petersen and Felix Voigtlaender. Optimal approximation of piecewise smooth functions using deep relu neural networks. *Neural Networks*, 108:296–330, 2018.

[38] Lukas Pichler, Arif Masud, and Lawrence A Bergman. Numerical solution of the fokker–planck equation by finite difference and finite element methods—a comparative study. In *Computational Methods in Stochastic Dynamics*, pages 69–85. Springer, 2013.

[39] Di Qi and Andrew J. Majda. Low-dimensional reduced-order models for statistical response and uncertainty quantification: Barotropic turbulence with topography. *Physica D: Nonlinear Phenomena*, 343:7–27, 2017.

[40] Maziar Raissi, Paris Perdikaris, and George E Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707, 2019.

[41] Danilo Jimenez Rezende and Shakir Mohamed. Variational Inference with Normalizing Flows. *arXiv:1505.05770 [cs, stat]*, 2015.

[42] Danilo Jimenez Rezende and Shakir Mohamed. Variational inference with normalizing flows. *arXiv preprint arXiv:1505.05770*, 2015.

[43] Hannes Risken. *The Fokker-Planck Equation*, volume 18 of *Springer Series in Synergetics*. Springer Berlin Heidelberg, Berlin, Heidelberg, 1989.

[44] Gareth O Roberts, Richard L Tweedie, et al. Exponential convergence of langevin distributions and their discrete approximations. *Bernoulli*, 2(4):341–363, 1996.

[45] HoHo Rosenbrock. An automatic method for finding the greatest or least value of a function. *The Computer Journal*, 3(3):175–184, 1960.

[46] Sebastian Ruder. An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*, 2016.

[47] Tamar Schlick. *Molecular modeling and simulation: an interdisciplinary guide: an interdisciplinary guide*, volume 21. Springer Science & Business Media, 2010.

[48] Justin Sirignano and Konstantinos Spiliopoulos. Mean field analysis of neural networks. *arXiv preprint arXiv:1805.01053*, 2018.

[49] S. Surjanovic and D. Bingham. Virtual library of simulation experiments: Test functions and datasets. Retrieved February 8, 2020, from http://www.sfu.ca/~ssurjano.

[50] Cédric Villani. *Topics in optimal transportation*. Number 58. American Mathematical Soc., 2003.

[51] Cédric Villani. *Optimal transport: old and new*, volume 338. Springer Science & Business Media, 2008.

[52] E Weinan, Jiequn Han, and Arnulf Jentzen. Deep learning-based numerical methods for high-dimensional parabolic partial differential equations and backward stochastic differential equations. *Communications in Mathematics and Statistics*, 5(4):349–380, 2017.

[53] Max Welling and Yee W Teh. Bayesian learning via stochastic gradient langevin dynamics. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, pages 681–688, 2011.

[54] Yaohua Zang, Gang Bao, Xiaojing Ye, and Haomin Zhou. Weak adversarial networks for high-dimensional partial differential equations. *arXiv preprint arXiv:1907.08272*, 2019.