Submitted to the Annals of Statistics

NO-REGRET GENERATIVE MODELING VIA PARABOLIC MONGE-AMPÈRE PDE

BY NABARUN DEB [1,a] AND TENGYUAN LIANG [1,b]

[1] *The University of Chicago, Booth School of Business*
[a] *nabarun.deb@chicagobooth.edu*; [b] *tengyuan.liang@chicagobooth.edu*

We introduce a novel generative modeling framework based on a discretized parabolic Monge-Ampère PDE, which emerges as a continuous limit of the Sinkhorn algorithm commonly used in optimal transport. Our method performs iterative refinement in the space of Brenier maps using a mirror gradient descent step. We establish theoretical guarantees for generative modeling through the lens of no-regret analysis, demonstrating that the iterates converge to the optimal Brenier map under a variety of step-size schedules. As a technical contribution, we derive a new Evolution Variational Inequality tailored to the parabolic Monge-Ampère PDE, connecting geometry, transportation cost, and regret. Our framework accommodates non-log-concave target distributions, constructs an optimal sampling process via the Brenier map, and integrates favorable learning techniques from generative adversarial networks and score-based diffusion models. As direct applications, we illustrate how our theory paves new pathways for generative modeling and variational inference.

1. Introduction. Probabilistic generative models have demonstrated impressive empirical performance in modeling complex probability distributions for tasks deemed challenging under conventional statistical frameworks. Over the past decade, many probabilistic generative models have been proposed and analyzed, partly inspired by the theory of optimal transport (OT), including generative adversarial networks (GANs) [4, 34, 41, 51, 56], as well as flow-based and diffusion-based probabilistic models (DPMs) [20, 21, 52, 54, 70, 71], among others.

At a high level, GANs and DPMs aim to model and learn the push-forward map between a *reference distribution* ν, which is easy to sample from, such as a Gaussian distribution, and the *target probability distribution* μ. With the learned push-forward map $\boldsymbol{\tau}$, one can generate new samples X from the target distribution by transforming fresh samples $Y \sim \nu$, specifically $X = \boldsymbol{\tau}(Y)$. Despite abundant empirical evidence demonstrating their effectiveness in practice, these generative models possess certain unfavorable theoretical limitations, which we discuss below.

(i) Ease of sampling: GANs model the push-forward map as a one-step map, namely $\mu = \boldsymbol{\tau}\#\nu$, where sampling is easy; DPMs, on the other hand, represent the push-forward map as a composition of a series of non-linear transformations, expressed as $\mu = (\boldsymbol{\tau}_1 \circ \boldsymbol{\tau}_2 \circ \cdots \circ \boldsymbol{\tau}_T)\#\nu$, where sampling is less transparent since it requires inverting a sequence of non-linear maps known as denoising steps.

(ii) Ease of learning: Brenier's theorem [11] demonstrates that the optimal transport map $\boldsymbol{\tau} = \nabla\psi$ corresponds to a convex potential ψ. However, it is known in GANs that enforcing convexity constraints to regularize learning presents technical challenges. Additionally, learning the one-step map from uncoupled data $X \sim \mu, Z \sim \nu$ proves difficult. In contrast,

MSC2020 subject classifications. Primary 49Q22, 49N99; secondary 65K10, 68Q32.
Keywords and phrases. Bregman divergence, evolution variational inequality, generative models, parabolic Monge-Ampère PDE, regret bounds.

DPMs propose decomposing the task into a sequence of sub-tasks where each map is a small deviation from the identity map, namely $\boldsymbol{\tau}_k \approx \mathrm{id}$, $k = 1, \dots, T$. Furthermore, for each step k, the deviation $\boldsymbol{\tau}_k - \mathrm{id}$ can be learned reliably from coupled data constructed using diffusions.

In this paper, we introduce a novel generative modeling framework inspired by parabolic Monge-Ampère partial differential equation (PDE), a nonlinear PDE that naturally arises in OT as the limit of the Sinkhorn algorithm. Notably, our theoretical framework simultaneously integrates the ease of sampling and the ease of learning, two key features discussed earlier.

The core idea is as follows: we refine the Brenier map $\nabla\psi_k$ iteratively using a discretized version of a mirror gradient flow, thus constructing a chain $\nabla\psi_1, \nabla\psi_2, \dots, \nabla\psi_T \overset{T\to\infty}{\to} \nabla\psi$. In particular, let $f(x) := -\log(\mathrm{d}\mu/\mathrm{d}x)$ be the target density, and define the density at step k as $\rho_k := (\nabla\psi_k)\#e^{-g}$ with $g(x) := -\log(\mathrm{d}\nu/\mathrm{d}x)$. We then study the following vector field iterations

$$\frac{\nabla\psi_{k+1} - \nabla\psi_k}{\eta_k} = -\boldsymbol{\xi}_k, \quad \text{where} \quad \boldsymbol{\xi}_k := \nabla\Big(\log\Big(\rho_k/e^{-f}\Big) \circ \nabla\psi_k\Big) \,.$$

Here, $\boldsymbol{\xi}_k$ resembles a (preconditioned) gradient step under a (local) mirror geometry. Equivalently, the Brenier potential admits an iterative refinement of the form of a (discretized) Monge-Ampère PDE,

$$\frac{\psi_{k+1} - \psi_k}{\eta_k} = -f \circ \nabla\psi_k + g + \log\det\big(\nabla^2\psi_k\big) \,,$$

which is the limit of the Sinkhorn algorithm under certain scalings [8, 25]. We note that explicitly evaluating the $\log\det$ term in the above display can be avoided either by using the classification trick (see Theorem 5.1) or the score-matching trick (see Theorem 5.2).

Two modeling advantages of our idea are: (i) our iterative refinement acts directly on the space of Brenier maps, and thus the limit $(\nabla\psi_T)\#\nu \overset{T\to\infty}{\to} \mu$ provides an optimal one-step sampling of the target measure in the context of Brenier's theorem; (ii) at each iteration, one can learn the gradient $\boldsymbol{\xi}_k$ without worrying about enforcing notions of convexity/cyclic monotonicity. The convexity of ψ_{k+1} will follow as a consequence of the convexity of ψ_k and the small step-size η_k. These speak to the ease of sampling and learning, respectively.

As for theoretical contributions, we provide a refined no-regret analysis for the proposed generative modeling framework, without requiring the target distribution μ to be log-concave. Our regret analysis provides new technical tools to study generative models by introducing a new Evolution Variational Inequality (EVI) [2, 38, 65]—which connects geometry, transportation cost, and regret—tailored to the parabolic Monge-Ampère PDE. Our novel regret analysis significantly expands the theory of generative models to accommodate non-log-concave measures.

1.1. *Organization.* In Section 2, we provide background and motivation for parabolic Monge-Ampère PDE. In Section 3, we introduce and study the properties of Bregman divergence on the space of probability measures, as it serves as the key geometry with respect to which we shall study convergence. Specifically, Section 3.1 features a new three-point identity for Bregman divergences, while Section 3.2 presents a new notion of convexity of the KL divergence with respect to a Wasserstein Bregman divergence (introduced in Example 2 below). In Section 4, we present all our convergence and regret analyses: (i) Section 4.1 provides guarantees for average iterate convergence, (ii) Section 4.2 introduces a new EVI for the KL and Bregman divergences, and (iii) Section 4.3 contains regret bounds and non-asymptotic rates for the last iterate. In Section 5, we discuss how our proposed discretized parabolic PDE can be employed to design novel algorithms for generative modeling (see

Section 5.1) and variational inference (see Section 5.2). In Section 6, we demonstrate the efficacy of our algorithms in sampling from a Gaussian mixture and provide a number of practical guidelines for stability, memory, and running time efficiency. The supplementary file [26] contains the proofs of all our main results and an additional numerical experiment.

1.2. *Notations.* We will write e^{-f} for the density of our target distribution μ supported on some $\mathcal{X} \subseteq \mathbb{R}^d$ and e^{-g} for the density of the reference distribution ν supported on $\mathcal{Y} \subseteq \mathbb{R}^d$. Typically for simple e^{-g} like the Gaussian, we will have $\mathcal{Y} = \mathbb{R}^d$. To help the reader distinguish between the domains, all integrals on $\mathcal{X}$ will be over the variable x and all integrals on $\mathcal{Y}$ will be over the variable y. Only μ, ν are reserved for probability measures, and the other Greek symbols ρ and π (with subscripts) will always refer to probability densities. All vector fields will be denoted using boldface notation, for instance, $\boldsymbol{\xi}, \boldsymbol{\tau} : \mathbb{R}^d \to \mathbb{R}^d$.

The push-forward $\boldsymbol{\tau}\#\mu$ for a vector field $\boldsymbol{\tau}$ denotes the distribution of $\boldsymbol{\tau}(X)$ when $X \sim \mu$. With slight notational abuse, we will sometimes directly apply the $\#$ notation on a probability density, say ρ, in which case $\boldsymbol{\tau}\#\rho$ will also denote a density function. $KL(\rho|\pi)$ and $W_2(\rho, \pi)$ denote the standard KL divergence and the Wasserstein distance between ρ and π, respectively. For densities π and ρ, we will use the standard definitions of Brenier potentials and Brenier maps between π and ρ from [77, Theorem 2.12]. Typically, these potentials will be represented with ϕ or ψ (and the corresponding maps by $\nabla\phi$ and $\nabla\psi$), with appropriate indexing to make the probability measures involved transparent. $\mathcal{P}_2^{\mathrm{ac}}(\mathbb{R}^d)$ will denote the space of probability densities supported on some subset of $\mathbb{R}^d$ with finite second moments, and we will sometimes refer to it as the 2-Wasserstein space.

Given a continuously differentiable function $\psi : \mathbb{R}^d \to \mathbb{R}$, we write its Fenchel dual as $\psi^*(x) = \sup_{y \in \mathcal{Y}}(\langle x, y\rangle - \psi(y))$. We will call a twice differentiable function $\psi : \mathbb{R}^d \to \mathbb{R}$ a m-strongly convex function if $\inf_y \lambda_{\min}(\nabla^2\psi(y)) \geq m$. Similarly, we will call it M-smooth if $\sup_y \lambda_{\max}(\nabla^2\psi(y)) \leq M$. Here $\lambda_{\min}$ and $\lambda_{\max}$ denote the minimum and the maximum eigenvalues of a matrix. The following function spaces will be used: (i) $\mathcal{C}^2$ for twice continuously differentiable functions, (ii) $\mathcal{C}_c^\infty$ for infinitely differentiable functions with compact support, and (iii) $L^2(\rho)$ for square integrable functions with respect to some probability density ρ. Given a functional $\mathcal{F} : \mathcal{U} \to \mathbb{R}$ for some function space $\mathcal{U}$, $\frac{\delta \mathcal{F}}{\delta u}$ will denote the first variation/Gateaux derivative of $\mathcal{F}$ at u. Finally $\mathbb{S}_+^{d\times d}$ will denote the set of $d \times d$ symmetric positive semi-definite matrices; for instance, $\nabla^2\psi(y) \in \mathbb{S}_+^{d\times d}, \forall y \in \mathcal{Y}$.

2. Parabolic Monge-Ampère PDE. Recall that we wish to sample from a complicated *target distribution* with density $e^{-f} \in \mathcal{P}_2^{\mathrm{ac}}(\mathbb{R}^d)$. We also have a *reference distribution* with density $e^{-g} \in \mathcal{P}_2^{\mathrm{ac}}(\mathbb{R}^d)$, which is easy to sample from, for instance, a standard Gaussian. By Brenier's Theorem [11, 58], there exists a Brenier potential $\psi : \mathbb{R}^d \to \mathbb{R}$ such that $\nabla\psi\#e^{-g} = e^{-f}$. The corresponding (static) Monge-Ampère equation [3, 60] thus reads:

$$-f(\nabla\psi(y)) + g(y) + \log\det\big(\nabla^2\psi(y)\big) = 0\,.$$

Learning $\nabla\psi$ is particularly useful for sampling as one can generate samples from e^{-f} by first sampling from $Y \sim e^{-g}$ (say a Gaussian) and then applying the optimal transform $\nabla\psi(Y)$. However, as the above is a non-linear second-order PDE, given e^{-f} and e^{-g}, solving the equation for $\nabla\psi$ is highly computationally intensive. An alternate approach adopted in the PDE literature is to study instead the natural dynamic version, that is, the parabolic PDE

$$\frac{\partial\psi_t}{\partial t}(y) = -f(\nabla\psi_t(y)) + g(y) + \log\det(\nabla^2\psi_t(y))\,. \tag{2.1}$$

The existence of solutions, uniqueness, smoothness, and exponential convergence of $\nabla\psi_t \to \nabla\psi$ as $t \to \infty$ have been studied extensively in the PDE literature; see e.g. [1, 8, 46]. Therefore, $\{\nabla\psi_t\}_{t\geq 0}$ can be viewed as a continuum of iterative refinements of some initial map

$\nabla\psi_0 : \mathbb{R}^d \to \mathbb{R}^d$, eventually leading to the target Brenier map $\nabla\psi$. Furthermore, define $\rho_t := \nabla\psi_t \# e^{-g}$. Then by sampling $Y \sim e^{-g}$, $\nabla\psi_t(Y) \sim \rho_t \approx e^{-f}$ (see [25]). Therefore, $\nabla\psi_t(Y)$ can be viewed as an approximate one-step sample from the target density e^{-f}, for large t. Due to these favorable properties of the parabolic PDE (2.1), it is natural to study an implementable time discretization of (2.1). In this paper, we therefore study the following natural (forward) update rule:

$$\begin{aligned} &\frac{\psi_{k+1}(y) - \psi_k(y)}{\eta_k} \\ &= -f(\nabla\psi_k(y)) + g(y) + \log\det\big(\nabla^2\psi_k(y)\big) \\ &= -f(\nabla\psi_k(y)) - \log\rho_k(\nabla\psi_k(y)) \end{aligned} \tag{2.2}$$

where $\{\eta_k\}_{k\geq 0}$ denotes the set of step-sizes, $\psi_0 : \mathbb{R}^d \to \mathbb{R}$ is some strongly convex function, and

$$\rho_k := (\nabla\psi_k)\# e^{-g}. \tag{2.3}$$

In this paper, we study the choice of step-sizes η_k, and its effect on the convergence of $\rho_k \to e^{-f}$. The implementation of (2.2) based on neural networks is possible using a classification technique as in [35, Proposition 1], or a score matching technique as in [42, Theorem 1]; we will discuss these in detail in Section 5.

Beyond the natural connections to the Monge-Ampère equation, the continuous and discrete systems (2.1) and (2.2) have several compelling properties that make them useful for applications. We present a few of them below: (i) they can lead to faster flows in the simple Gaussian setting than perhaps the widely studied dynamical system, namely the canonical Fokker-Planck equation [29, 62]; (ii) they are closely related to the dynamics of the celebrated Sinkhorn algorithm [23, 31]; (iii) they can be viewed as steepest descent of the $KL(\cdot|e^{-f})$ functional with respect to a certain weighted local L^2-metric; (iv) they can be used to construct novel generative learning algorithms that combine ease of sampling with ease of learning as mentioned in the introduction; and (v) they provide a new paradigm for variational inference. The first two points will be discussed in this section, while the latter three are deferred to Section 5.

2.1. *Illustrative Example for Gaussians.* We provide a simple example on the rate of convergence of the aforementioned parabolic PDE when both e^{-f} and e^{-g} are densities of centered univariate Gaussians, say $N(0,1)$ and $N(0,\lambda^2)$, $\lambda < 1$. Suppose that $\psi_0(y) = y^2/2$. The continuous time system (2.1), after taking an additional space derivative then reduces to

$$\frac{\partial\psi_t'(y)}{\partial t} = -\psi_t''(y)\psi_t'(y) + \frac{1}{\lambda^2}y + \frac{\psi_t'''(y)}{\psi_t''(y)}\,.$$

As ψ_0' is linear, it is easy to check that all ψ_t's are linear. So, let $\psi_t'(y) = c_t y$ for some c_t. The above differential equation then becomes a Riccati equation $\dot{c}_t = -c_t^2 + \frac{1}{\lambda^2}$ with $c_0 = 1$, which implies $c_t = (1/\lambda)\tanh\big((t/\lambda) + \tanh^{-1}(\lambda)\big)$. Writing $\rho_t = (\psi_t')\# e^{-g}$ gives that ρ_t is the density of the $N(0,\sigma_t^2)$ distribution where $\sigma_t = \tanh\big((t/\lambda) + \tanh^{-1}(\lambda)\big)$. It is easy to check that for the standard Fokker-Planck system, the corresponding solution, say $\rho_F(t)$ is the density of $N(0,\sigma_{F,t}^2)$ where $\sigma_{F,t}^2 = 1 - (1-\lambda^2)e^{-2t}$. As a result,

$$\frac{1-\sigma_{F,t}^2}{1-\sigma_t^2} = \frac{1-\lambda^2}{4}\left(e^{-t(1+\lambda^{-1})-\tanh^{-1}(\lambda)} + e^{t(\lambda^{-1}-1)+\tanh^{-1}(\lambda)}\right)^2 \to \infty \quad \text{as } t \to \infty\,,$$

because $\lambda < 1$. Therefore, σ_t^2 converges to the target variance 1 faster than the Fokker-Planck variance $\sigma_{F,t}^2$. For the discretization scheme (2.2), it can be shown that the ρ_k's (see (2.3)), under appropriate initialization, are all centered Gaussian distributions, and the variance converges to the target variance 1 locally at an exponential rate. For brevity, we defer further details to Section A in the Supplement [26].

In the sequel, we provide in Section 4 global regret analyses of the discretization (2.2) under general assumptions on e^{-f} and e^{-g}, extending far beyond Gaussianity and log-concavity.

2.2. *Connection to Sinkhorn.* The exploding literature on generative modeling in recent years has witnessed state-of-the-art methods that rely on entropy regularized optimal transport (EOT); see e.g. [16, 24, 33, 79]. Given a temperature parameter $\epsilon > 0$, the EOT problem is given by

$$\pi^\epsilon := \underset{\pi \in \Pi(e^{-f}, e^{-g})}{\arg\min} \left\{ \frac{1}{2} \int \|x-y\|^2 \gamma(x,y)\, \mathrm{d}(x,y) + \epsilon KL(\gamma | e^{-f} \otimes e^{-g}) \right\}, \tag{2.4}$$

where $\Pi(e^{-f}, e^{-g})$ denotes the class of probability densities on $\mathbb{R}^d \times \mathbb{R}^d$ with marginal densities e^{-f} and e^{-g}. The optimal π^ϵ from (2.4) is also referred to as the (static) Schrödinger Bridge (see [68]) between e^{-f} and e^{-g}. By [30, 64], it is known that there exist potentials ϕ^ϵ and ψ^ϵ such that $\pi^\epsilon(x,y) = \exp\left(\frac{1}{\epsilon}\langle x, y\rangle - \frac{1}{\epsilon}\phi^\epsilon(x) - \frac{1}{\epsilon}\psi^\epsilon(y) - f(x) - g(y)\right)$. As $\pi^\epsilon \in \Pi(e^{-f}, e^{-g})$, both ϕ^ϵ and ψ^ϵ will satisfy certain fixed point equations. In particular, the equation for ψ^ϵ is given by

$$\psi^\epsilon(y) = \mathcal{V}^\epsilon[\psi^\epsilon](y),$$

where

$$\mathcal{V}^\epsilon[\psi^\epsilon](y) := \epsilon \log \int \frac{\exp\left(\frac{1}{\epsilon}\langle x, y\rangle\right)}{\int \exp\left(\frac{1}{\epsilon}\langle x, y'\rangle - \frac{1}{\epsilon}\psi^\epsilon(y') - g(y')\right) \mathrm{d}y'} e^{-f(x)} \mathrm{d}x\,.$$

Perhaps the most popular algorithm for solving (2.4) is the Sinkhorn algorithm (see [63, 23]) which solves for ϕ^ϵ, ψ^ϵ, π^ϵ using a natural iterative procedure to get $\{\phi_k^\epsilon, \psi_k^\epsilon, \pi_k^\epsilon\}_{k\geq 0}$. In particular, following the $\mathcal{V}^\epsilon$ notation above, the ψ_k^ϵs are updated as

$$\psi_{k+1}^\epsilon = \mathcal{V}^\epsilon[\psi_k^\epsilon](y)\,. \tag{2.5}$$

The parabolic PDE (2.1) can now be viewed as the *scaling limit* of the Sinkhorn algorithm in the low temperature regime $\epsilon \to 0$ when *the number of iterations k scales like t/ϵ for some $t > 0$*. The following proposition (also see [8, Lemma 4.2] and [25, Lemma 4.6]) demonstrates why such a scaling limit is natural. We provide a proof in Section A of the Supplement [26] that is self-contained.

PROPOSITION 2.1. *Suppose $\psi : \mathbb{R}^d \to \mathbb{R}$ is a uniformly strongly convex $\mathcal{C}^2$ function. Let $\tilde{\psi}^\epsilon(y) := \mathcal{V}^\epsilon[\psi](y)$. Then, as $\epsilon \to 0$, we have:*

$$\lim_{\epsilon \to 0} \frac{\tilde{\psi}^\epsilon(y) - \psi(y)}{\epsilon} = -f(\nabla\psi(y)) + g(y) + \log\det\left(\nabla^2\psi(y)\right)\,.$$

Assuming that the Sinkhorn iterates $\{\psi_k^\epsilon\}$'s are $\mathcal{C}^2$ smooth and strongly convex, Theorem 2.1 coupled with the Sinkhorn update rule (2.5) yields the following approximation

$$\frac{\psi_{k+1}^\epsilon(y) - \psi_k^\epsilon(y)}{\epsilon} \approx -f(\nabla\psi_k^\epsilon(y)) + g(y) + \log\det\left(\nabla^2\psi_k^\epsilon(y)\right),$$

which approximately coincides with our discretization scheme (2.2) with constant step-sizes $\eta_k = \epsilon$. Moreover, the above display implies that ψ^{ε}_{k+1} and ψ^{ε}_{k} are $O(\varepsilon)$ apart. As a result, by running the Sinkhorn algorithm for $k = t/\epsilon$ iterations, we expect to get a limiting curve, call it $\{\psi_t\}_{t\geq 0}$. Using similar heuristics as used in the discretization of the Cauchy problem [59, 67], we expect that the LHS $\epsilon^{-1}(\psi^{\epsilon}_{t/\epsilon+1}(y) - \psi^{\epsilon}_{t/\epsilon})$ will converge to the time derivative $\partial_t \psi_t(y)$ whereas the RHS $-f(\nabla\psi^{\epsilon}_{t/\epsilon}(y)) + g(y) + \log\det\Big(\nabla^2\psi^{\epsilon}_{t/\epsilon}(y)\Big)$ should converge to the negative descent direction $-f(\nabla\psi_t(y)) + g(y) + \log\det\big(\nabla^2\psi_t(y)\big)$. This yields the parabolic PDE in (2.1).

3. Bregman Divergences over Probability Measures. Our regret approach to studying the discretized parabolic Monge-Ampère equation (and its application to generative modeling) relies on a novel three-point identity that delineates geometry, transportation cost, and regret. This key analytic tool depends on the concept of Bregman divergences on the 2-Wasserstein space. To motivate it, we recall the definition of Bregman divergence on the standard Euclidean space — For a diffeomorphism $\phi(\cdot)$ and its Fenchel conjugate $\phi^*(\cdot)$, define

$$D_\phi(x|\nabla\phi^*(y)) := \phi(x) - \phi(\nabla\phi^\star(y)) - \langle y, x - \nabla\phi^*(y)\rangle = \phi(x) + \phi^*(y) - \langle x, y\rangle \,. \tag{3.1}$$

It is easy to check that if $\phi(\cdot)$ is strictly convex, then $D_\phi(x|\nabla\phi^*(y)) \geq 0$ with equality if and only if $x = \nabla\phi^*(y)$. The Bregman divergence is a fundamental notion in the celebrated field of Euclidean mirror descent (see [7, 12, 76], and the references therein) and is known to yield nearly dimension-free rates in certain constrained optimization problems. The following definition extends the Bregman divergence (introduced first in [49]; also see [5, 10, 25, 44]) from the finite-dimensional Euclidean space to the infinite-dimensional 2-Wasserstein space.

DEFINITION 1 (Bregman divergence over probability measures). *Given a function $\Gamma : \mathcal{P}_2^{\mathrm{ac}}(\mathbb{R}^d) \to \mathbb{R}$ which admits a well-defined first variation, we define the Bregman divergence functional for Γ as follows:*

$$B_\Gamma(\rho_2|\rho_1) := \Gamma(\rho_2) - \Gamma(\rho_1) - \int \frac{\delta\Gamma}{\delta\rho}(\rho_1)(x)\,(\rho_2 - \rho_1)(x)\,\mathrm{d}x \,,$$

for probability density functions $\rho_1, \rho_2 \in \mathcal{P}_2^{\mathrm{ac}}(\mathbb{R}^d)$, provided the first variation is integrable under ρ_1, ρ_2. We will refer to Γ as the mirror function.

The above definition of Bregman divergence on the 2-Wasserstein space is reminiscent of the usual Euclidean Bregman divergence defined in (3.1). The main difference here is that the role of the Euclidean derivative is played by the first variation. For notational clarity, we use different letters B and D for divergences on the Wasserstein space and on the Euclidean space, respectively. Below, we provide some examples of Bregman divergence over probability measures.

EXAMPLE 1 (Entropy as mirror). *The most popular example of Bregman divergence is when the mirror is the entropy function $H(\rho) := \int \rho(x)\log\rho(x)\mathrm{d}x$, in which case*

$$B_H(\rho_2|\rho_1) = H(\rho_2) - H(\rho_1) - \int \frac{\delta H}{\delta\rho}(\rho_1)(x)\,(\rho_2 - \rho_1)(x)\,\mathrm{d}x = KL(\rho_2|\rho_1) \,,$$

for probability density functions $\rho_1, \rho_2 \in \mathcal{P}_2^{\mathrm{ac}}(\mathbb{R}^d)$, provided the first variation is integrable under ρ_1, ρ_2.

EXAMPLE 2 (Wasserstein distance as mirror). *Another example of Bregman divergence that will be useful in this paper is for the function $G(\rho) := (1/2)W_2^2(\rho, e^{-g})$, where e^{-g} is the reference distribution. To simplify the Bregman divergence for this example, we write ϕ_ρ to be a Brenier potential from ρ to e^{-g}. It is known (see [66, Proposition 7.17] and [2, Corollary 10.2.7]) that*

$$\frac{\delta G}{\delta \rho}(\rho)(x) = \frac{1}{2}\|x\|^2 - \phi_\rho(x)\,.$$

Therefore, the Bregman divergence of G is given by

$$B_G(\rho_2|\rho_1) = G(\rho_2) - G(\rho_1) - \int \left(\frac{1}{2}\|x\|^2 - \phi_{\rho_1}(x)\right)(\rho_2 - \rho_1)(x)\mathrm{d}x\,,$$

for probability densities $\rho_1, \rho_2 \in \mathcal{P}_2^{\mathrm{ac}}(\mathbb{R}^d)$.

3.1. *A New Three-Point Identity.* The Bregman divergence function $B_G(\cdot|\cdot)$ satisfies non-negativity and can be expressed as the expectation of the standard Bregman divergence with respect to an appropriate probability measure. We defer the reader to Lemmas B.1 and B.2 in the Supplement [26] for further details. To keep the discussion streamlined, we present the most important property of $B_G(\cdot|\cdot)$ here, namely a new three-point identity, that will be instrumental to our regret analysis. The proof is deferred to Section B of the Supplement [26].

LEMMA 3.1 (A new Bregman three-point identity). *Consider probability densities $\rho_1, \rho_2, \pi \in \mathcal{P}_2^{\mathrm{ac}}(\mathbb{R}^d)$, and recall the definitions of $G(\cdot) = (1/2)W_2^2(\cdot, e^{-g})$, its induced $B_G(\cdot|\cdot)$, and the Brenier potentials ϕ_{ρ_i} for $i = 1, 2$ as in Example 2. Define in addition $\pi_i := (\nabla\phi_{\rho_i})\#\pi$ for $i = 1, 2$, and another Bregman divergence $B_{G_\pi}(\cdot|\cdot)$ induced by the function $G_\pi(\cdot) := (1/2)W_2^2(\cdot, \pi)$. Set $\psi_{\rho_i} = \phi_{\rho_i}^*$ for $i = 1, 2$. Then*

$$\int (\psi_{\rho_2} - \psi_{\rho_1})(y)(\pi_1 - e^{-g})(y)\mathrm{d}y = B_G(\pi|\rho_1) - B_G(\pi|\rho_2) + B_{G_\pi}(\pi_1|\pi_2)\,.$$

Let us briefly examine why the above lemma will be useful in our regret bounds. We will use it with $\rho_2 = \rho_{k+1}$ and $\rho_1 = \rho_k$ (see (2.3)). The LHS will be the first-order term in the expansion of the map $\rho \mapsto \mathrm{KL}(\rho|e^{-f})$ around ρ_k. For the RHS, the term $B_G(\pi|\rho_1) - B_G(\pi|\rho_2) \equiv B_G(\pi|\rho_k) - B_G(\pi|\rho_{k+1})$ will be a "telescoping term" as we vary k. The term $B_{G_\pi}(\pi_1|\pi_2)$ is for the "transportation cost" to go from ρ_1 to ρ_2, or equivalently ρ_k to ρ_{k+1}. Curiously, note that the transportation cost does not use the same mirror map G but instead requires a new mirror map G_π. Moreover, it is not directly a Bregman divergence between ρ_1 and ρ_2 but instead between the image of π under $\nabla\phi_{\rho_1}$ and $\nabla\phi_{\rho_2}$. Figure 1 summarizes the maps and probability measures involved in the transportation cost term.

As a concrete example, we can simplify the identity in Lemma 3.1 when all the probability measures involved are Gaussians. Let $\pi \sim N(0, \sigma_\pi^2)$, $e^{-g} \sim N(0, \sigma_g^2)$, $\rho_1 \sim N(0, \sigma_1^2)$ and $\rho_2 \sim N(0, \sigma_2^2)$. Then $\psi'_{\rho_2}(y) = (\sigma_2/\sigma_g)y$ and $\psi'_{\rho_1}(y) = (\sigma_1/\sigma_g)y$. As a result, $\pi_1 \sim N(0, (\sigma_g\sigma_\pi/\sigma_1)^2)$ and $\pi_2 \sim N(0, (\sigma_g\sigma_\pi/\sigma_2)^2)$. It is easy to check that

$$\int (\psi_{\rho_2} - \psi_{\rho_1})(y)(\pi_1 - e^{-g})(y)\mathrm{d}y = \frac{1}{2}\sigma_g(\sigma_1 - \sigma_2)\left(1 - \frac{\sigma_\pi^2}{\sigma_1^2}\right).$$

Further

$$B_G(\pi|\rho_1) = \frac{1}{2}\frac{\sigma_g}{\sigma_1}(\sigma_\pi - \sigma_1)^2, \quad B_G(\pi|\rho_2) = \frac{1}{2}\frac{\sigma_g}{\sigma_2}(\sigma_\pi - \sigma_2)^2,$$

$$B_{G_\pi}(\pi_1|\pi_2) = \frac{1}{2}\sigma_g\sigma_\pi^2\sigma_2\left(\frac{1}{\sigma_1} - \frac{1}{\sigma_2}\right)^2.$$

Direct computations can now be used to verify Lemma 3.1.

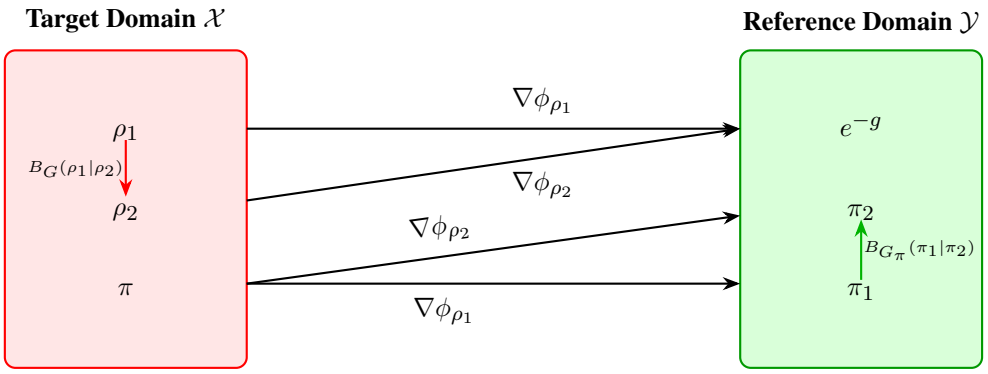


FIG 1. *The probability densities ρ_1, ρ_2, π live on the target domain $\mathcal{X}$ while the densities π_1, π_2, e^{-g} live on the reference domain $\mathcal{Y}$. $\nabla\phi_{\rho_i}$ is the Brenier map from ρ_i to e^{-g}, $i = 1, 2$. As $\pi_i = \nabla\phi_{\rho_i}\#\pi$, Brenier's Theorem [11] implies that $\nabla\phi_{\rho_i}$ is also the Brenier map from π to π_i, for $i = 1, 2$. The telescoping term in Lemma 3.1, $B_G(\pi|\rho_1) - B_G(\pi|\rho_2)$ involves densities ρ_1, ρ_2, π all of which live on the target space. One might naturally expect the transportation cost to be $B_G(\rho_1|\rho_2)$. In sharp contrast, the corresponding term in Lemma 3.1 is $B_{G_\pi}(\pi_1|\pi_2)$ where π_i lives on the reference domain and is the image of π under the Brenier map $\nabla\phi_{\rho_i}$, $i = 1, 2$.*

REMARK 1. In the mirror descent literature on Hilbert spaces, analogs of the above lemma are usually termed three-point inequalities or Bregman proximal inequalities; see [10, Lemma 29], [19, Lemma 3.2], [47, Lemma 1]; also see [32, 55]. For instance, in the Euclidean setting, the Bregman descent update for optimizing a function $F : \mathbb{R}^d \to \mathbb{R}$ with respect to a mirror $\phi : \mathbb{R}^d \to \mathbb{R}$ is given by

$$\nabla\phi(x_{k+1}) = \nabla\phi(x_k) - \eta_k \nabla F(x_k) \,.$$

For any $z \in \mathbb{R}^d$, regret bounds for Euclidean mirror descent rely on the following three-point identity:

$$\langle \nabla\phi(x_k) - \nabla\phi(x_{k+1}), x_k - z\rangle = D_\phi(z|x_k) - D_\phi(z|x_{k+1}) + D_\phi(x_k|x_{k+1}) \,, \tag{3.2}$$

which relates the three points z, x_k and x_{k+1}. Lemma 3.1 can be viewed as an extension of (3.2) to the 2-Wasserstein space which does not possess a Hilbert structure. Note that Lemma 3.1 captures the relationship between five different measures $\pi, \pi_1, \pi_2, \rho_1, \rho_2$, instead of three. As π_i is still a function of π, ρ_i, e^{-g}, we continue to use the term "three-point lemma" for Lemma 3.1; also see Figure 1. Moreover, in contrast to the RHS of (3.2) where all the divergences use the same mirror ϕ, the RHS of Lemma 3.1 requires the use of two different mirrors $G(\cdot) = (1/2)W_2^2(\cdot, e^{-g})$ and $G_\pi(\cdot) = (1/2)W_2^2(\cdot, \pi)$. To the best of our knowledge, the existing three-point equalities/inequalities for other forms of Bregman descent on the 2-Wasserstein space, e.g. [5, Lemma 3], [39, Lemma 8], [48, Theorem 1], also use the same mirror map on both the telescoping and the transportation cost terms. In that sense, Lemma 3.1 is new and is particularly well suited to analyze the discretization (2.2).

3.2. *Bregman and KL: Relative Convexity.* Our next main result in this section introduces a new notion of convexity for the KL divergence with respect to the Bregman divergence $B_G(\cdot|\cdot)$. The proof is deferred to Section B in the Supplement [26].

LEMMA 3.2 (Bregman and KL: relative convexity). *Consider the reference distribution e^{-g} where $g(\cdot)$ is λ-strongly convex. Further, let ρ be a probability density such that the Brenier potential $\psi_\rho(\cdot)$ from e^{-g} to ρ satisfies $\sup_y \lambda_{\max}(\nabla^2\psi_\rho(y)) \le \beta$, for some $\beta > 0$.*

Fix any absolutely continuous probability measure π such that the Brenier potential ψ_π from e^{-g} to π is strictly convex and twice differentiable. We also assume that for any $1 \le i \le d$, it holds that the map

$$y \mapsto e^{-g(y)} \left((\partial_i \psi_\rho^\star)(\nabla \psi_\pi(y)) - y_i \right)$$

vanishes as $|y_i| \to \infty$. Then the following inequality holds:

$$KL(\pi|\rho) \ge \frac{\lambda}{\beta} B_G(\pi|\rho) \, .$$

REMARK 2. Recall $H(\rho) = \int \rho(x) \log \rho(x) \mathrm{d}x$ for $\rho \in \mathcal{P}_2^{\mathrm{ac}}(\mathbb{R}^d)$. The Bregman convexity in Lemma 3.2 can be rewritten as $B_H(\pi|\rho) \ge (\lambda/\beta) B_G(\pi|\rho)$. We believe it introduces a new notion of convexity for the KL divergence. A crucial feature in our Lemma 3.2 is that we do not require strong log-concavity on either π or ρ; we merely assume its Brenier potentials ψ_ρ, ψ_π to be strongly convex, a much milder assumption.

The two most popular existing notions of convexity, namely geodesic convexity (see [2, Chapter 9], [53]) which says $KL(\pi|\rho) \ge (\gamma/2) W_2^2(\pi, \rho)$ and generalized geodesic convexity (see [2, Chapter 11], [65]) which says $KL(\pi|\rho) \ge (\gamma/2) \int \|\nabla \phi_\rho^*(y) - \nabla \phi_\pi^*(y)\|^2 e^{-g(y)} \, \mathrm{d}y$, both require $-\log \rho$ to be γ-strongly convex. However, since we apply Lemma 3.2 with $\rho = \rho_k$, such strong log-concavity is not available to us even if ψ_k and ψ_k^* are uniformly strongly convex.

Finally, we note that the strong convexity assumption of $g(\cdot)$ in Lemma 3.2 can be replaced by the assumption $D_g(y_1|y_2) \ge \lambda D_{\phi_\rho^*}(y_2|y_1)$. This is evident from the proof of Lemma 3.2. Such assumptions directly on divergences instead of metrics are common in the mirror descent literature; see e.g., [22, Definition 10.2.8]. However, for ease of presentation, we stick with the usual notion of strong convexity on g.

4. Regret and Convergence Analysis. In this section, we will discuss regret bounds and convergence rates for ρ_k (see (2.3)) as a function of the total number of iterations T.

4.1. *Average Iterate Convergence.* We start by deriving a result that holds under very weak assumptions and allows for large, constant step-sizes. We prove the KL distance between a weighted mixture of $\{\rho_k\}$'s and e^{-f} converges to 0, at a $O(T^{-1})$ rate. The proof is deferred to Section C of the Supplement [26].

THEOREM 4.1. *Suppose $\{\psi_k\}_{k=0}^T$ are generated according to the discretization (2.2). Then*

$$\mathop{\mathbb{E}}_{Y \sim e^{-g}} [\psi_{k+1}(Y) - \psi_k(Y)] = -\eta_k KL\big(\rho_k | e^{-f}\big) \, . \tag{4.1}$$

Next define $S_T := \sum_{k=0}^{T-1} \eta_k$ and the mixture distribution $\bar{\rho}_T := \sum_{k=0}^{T-1} \frac{\eta_k}{S_T} \rho_k$. We then have:

$$KL(\bar{\rho}_T | e^{-f}) \le S_T^{-1} \mathop{\mathbb{E}}_{Y \sim e^{-g}} [\psi_0(Y) - \psi_T(Y)] \, .$$

REMARK 3. Suppose we use time-invariant step-sizes, i.e., $\eta_k = \eta$ for some $\eta > 0$. Then Theorem 4.1 implies

$$KL(\bar{\rho}_T | e^{-f}) \le \frac{1}{\eta T} \mathop{\mathbb{E}}_{Y \sim e^{-g}} [\psi_0(Y) - \psi_T(Y)] \, ,$$

thereby yielding a $O(1/T)$ rate of convergence for $\bar{\rho}_T$ to e^{-f}, if $\mathbb{E}_{Y \sim e^{-g}} \psi_T(Y)$ is bounded in T. Note that Theorem 4.1 does not impose any convexity restrictions on f and g, nor does it require strong convexity/smoothness of the Brenier potentials ψ_k. This result, inspired by the Polyak-average scheme, also allows for large step-sizes.

While Theorem 4.1 provides convergence of the average iterate $\bar{\rho}_T$ to e^{-f}, it is more natural to use the ρ_k (for k large) directly to approximate e^{-f}. This is called the last iterate convergence. In the sequel, we derive last iterate convergence under the additional regularity assumptions on the Brenier potentials $\{\psi_k\}_{k\geq 0}$. A main technical tool is a new EVI that will be used to conduct regret analyses.

4.2. *New Evolution Variational Inequality.* We now adopt a regret analysis approach to study the theoretical guarantees of generative modeling with Monge-Ampère PDE. Our method aligns with the tradition of online learning literature [18, 36, 69] and, in particular, is inspired by [38], where EVIs were utilized to study regret bounds on the Wasserstein space. To establish these regret bounds, the primary tool will be the Bregman divergence on the Wasserstein space introduced in Section 3; specifically, refer to Definition 1.

Recall the Brenier potentials $\{\psi_k\}_{k\geq 0}$ from (2.2) and the corresponding probability densities $\{\rho_k\}_{k\geq 0}$ from (2.3). Our first main result in this section presents a new one-step EVI that will be applied multiple times in deriving our final regret bounds.

LEMMA 4.1. *Consider $\{\psi_k\}_{k=0}^T$ generated from discretization (2.2). Assume that $m_k := \inf_y \lambda_{\min}(\nabla^2\psi_k(y)) > 0$, namely, the potentials $\psi_k(\cdot)$ are m_k-strongly convex. Define $\boldsymbol{\xi}_k(y) := \nabla_y((f+\log\rho_k)(\nabla\psi_k(y)))$. Given any probability density $\pi \in \mathcal{P}_2^{\mathrm{ac}}(\mathbb{R}^d)$, let π_k denote the probability density function of $(\nabla\psi_k^*)\#\pi$. In this setting, the following inequality holds:*

$$\begin{aligned} &KL(\rho_k|e^{-f}) - KL(\pi|e^{-f}) \\ &\leq \frac{1}{\eta_k}(B_G(\pi|\rho_k) - B_G(\pi|\rho_{k+1})) + \frac{1}{2}\frac{\eta_k}{m_{k+1}}\int \|\boldsymbol{\xi}_k(y)\|^2\pi_k(y)\mathrm{d}y - KL(\pi|\rho_k)\,. \end{aligned}$$

PROOF. By a direct computation coupled with (2.2), we have:

$$\begin{aligned} &KL(\rho_k|e^{-f}) - KL(\pi|e^{-f}) \\ &= \int (f+\log\rho_k)(x)(\rho_k-\pi)(x)\mathrm{d}x - KL(\pi|\rho_k) \\ &= -\frac{1}{\eta_k}\int(\psi_{k+1}-\psi_k)(y)e^{-g(y)}\mathrm{d}y + \frac{1}{\eta_k}\int(\psi_{k+1}-\psi_k)(y)\,\pi_k(y)\mathrm{d}y - KL(\pi|\rho_k)\,. \end{aligned}$$

By using Lemma 3.1 with π, $\rho_1 = \rho_k$, and $\rho_2 = \rho_{k+1}$, we note that:

$$\int(\psi_{k+1}-\psi_k)(y)(\pi_k - e^{-g})(y)\mathrm{d}y = B_G(\pi|\rho_k) - B_G(\pi|\rho_{k+1}) + B_{G_\pi}(\pi_k|\pi_{k+1})\,, \tag{4.2}$$

where $G_\pi(\cdot) = (1/2)W_2^2(\cdot,\pi)$. As a result,

$$\begin{aligned} &KL(\rho_k|e^{-f}) - KL(\pi|e^{-f}) \\ &= \frac{1}{\eta_k}\left[B_G(\pi|\rho_k) - B_G(\pi|\rho_{k+1}) + B_{G_\pi}(\pi_k|\pi_{k+1})\right] - KL(\pi|\rho_k)\,. \end{aligned}$$

By Lemma B.2 in the Supplement [26], we further have

$$\begin{aligned} B_{G_\pi}(\pi_k|\pi_{k+1}) &\leq \frac{1}{2}\sup_x \lambda_{\max}(\nabla^2\psi_{k+1}^*(x))\int\|\nabla\psi_{k+1}(y)-\nabla\psi_k(y)\|^2\,\pi_k(y)\mathrm{d}y \\ &= \frac{\eta_k^2}{2}\left(\inf_y \lambda_{\min}(\nabla^2\psi_{k+1}(y))\right)^{-1}\int\|\boldsymbol{\xi}_k(y)\|^2\pi_k(y)\mathrm{d}y \\ &\leq \frac{\eta_k^2}{2m_{k+1}}\int\|\boldsymbol{\xi}_k(y)\|^2\pi_k(y)\mathrm{d}y\,, \end{aligned}$$

where the equality in the second display uses Proposition A.2 in the Supplement [26] and (2.2). By combining the above displays, we have the following:

$$KL(\rho_k|e^{-f}) - KL(\pi|e^{-f})$$
$$\leq \frac{1}{\eta_k}(B_G(\pi|\rho_k) - B_G(\pi|\rho_{k+1})) + \frac{1}{2}\frac{\eta_k}{m_{k+1}}\int \|\boldsymbol{\xi}_k(y)\|^2 \pi_k(y)\mathrm{d}y - KL(\pi|\rho_k)\,.$$

This completes the proof. □

We note that EVIs are popular in the PDE literature and have been used in statistics to address questions on optimization and sampling; see [28, 38, 65]. To the best of our knowledge, none of these results cover EVIs with Bregman divergence over the space of probability measures.

4.3. *Regret Bounds and Last Iterate Convergence.* We now apply Lemma 4.1 to obtain different regret bounds. The proofs are deferred to Section C of the Supplement [26]. We present these bounds in progressive order by imposing slightly stronger regularity conditions on the reference distribution e^{-g} and the Brenier potentials $\{\psi_k\}_{k\geq 0}$. With more regularity conditions, the regret bounds will be stronger.

Our first result provides a $O(\sqrt{T})$-regret bound that requires only a uniform lower bound on the convexity of the Brenier potentials $\{\psi_k\}_{k\geq 0}$.

THEOREM 4.2. *Recall the notation from Lemma 4.1. Suppose $m_k \geq m$ and $\int \|\boldsymbol{\xi}_k(y)\|^2 \pi_k(y)\mathrm{d}y \leq A$ for all $k = 0, 1, \ldots, T+1$, and some $m, A > 0$. Then by choosing $\eta_k = T^{-1/2}$ for all $k = 0, 1, \ldots, T$, we get:*

$$\sum_{k=0}^{T}\big(KL(\rho_k|e^{-f}) - KL(\pi|e^{-f})\big) \leq \sqrt{T}\left(B_G(\pi|\rho_0) + \frac{A}{2m}\right).$$

While Theorem 4.2 shows that $O(\sqrt{T})$-regret is attainable with time-invariant step-sizes, it is natural to ask if the regret can be improved under stronger conditions. To wit, we note that Theorem 4.2 only requires (i) uniform strong convexity of the potentials $\{\psi_k\}$ across iterations $k = 0, 1, 2, \ldots, T$, and (ii) uniform boundedness of the vector fields $\{\boldsymbol{\xi}_k\}$ in the $L^2(\pi_k)$ norm. So far, we have not imposed any smoothness conditions on the $\{\psi_k\}_{k\geq 0}$. In the following result, we show that under additional smoothness conditions on $\{\psi_k\}_{k\geq 0}$ and a strong log-concavity assumption on the reference e^{-g}, logarithmic regret bound is attainable under a carefully chosen sequence of time-varying step-sizes.

THEOREM 4.3. *Recall the notation from Lemma 4.1. Given a probability density $\pi \in \mathcal{P}_2^{\mathrm{ac}}(\mathbb{R}^d)$, let ψ_π denote a Brenier potential from e^{-g} to π. Also suppose $\sup_y \lambda_{\max}(\nabla^2\psi_k(y)) \leq M_k$, for some $M_k > 0$ (i.e., the potentials ψ_k are M_k-smooth) and the map*

$$y \mapsto e^{-g(y)}((\partial_i\psi_k^*)(\nabla\psi_\pi(y)) - y_i)$$

vanishes as $|y_i| \to \infty$, for all $k = 0, 1, \ldots, T$. Finally, let e^{-g} be a strongly log-concave distribution where $g(\cdot)$ is λ-strongly convex.

(i) *Define the following:*

$$S_{k+1} := \lambda\sum_{i=0}^{k}\frac{1}{M_i} \quad \textit{and} \quad \eta_k := S_{k+1}^{-1}.$$

Then we have

$$\sum_{k=0}^{T}(KL(\rho_k|e^{-f}) - KL(\pi|e^{-f})) \leq \frac{1}{2}\sum_{k=0}^{T}\frac{1}{m_{k+1}S_{k+1}}\int \|\boldsymbol{\xi}_k(y)\|^2\pi_k(y)\mathrm{d}y\,.$$

(ii) *Now suppose that* $M_k \leq M$, $m_k \geq m$, *and* $\int \|\boldsymbol{\xi}_k(y)\|^2 \pi_k(y)\mathrm{d}y \leq A$ *for all* $k = 0, 1, \ldots, T$, *and some* $m, M \in (0, \infty)$. *Then by choosing* $\eta_k^{-1} := (k+1)\lambda/M$, *we have:*

$$\sum_{k=0}^{T} \left(KL(\rho_k|e^{-f}) - KL(\pi|e^{-f}) \right) \leq \frac{1}{2} \frac{AM}{\lambda m} (1 + \log(T+1)) .$$

We emphasize that our $O(\log T)$-regret bound does not assume that the target distribution e^{-f} is strongly log-concave or that it satisfies log-Sobolev or Poincaré type inequalities. Instead, the time-independent constant in our bound depends on the ratio of the uniform smoothness parameter M and the uniform strong convexity parameter m on Brenier's potential ψ. This ratio M/m can be viewed as an analog of the condition number (see [57, 81]) in the absence of any functional inequality assumptions on the target e^{-f}. Again, here we merely require strong convexity/smoothness of the Brenier potential ψ, which is known to be convex as a premise even for non-log-concave target distributions e^{-f}. Therefore, our assumptions are much weaker than log-concavity of e^{-f}. While it may be possible to improve the condition number dependence to $\sqrt{M/m}$ by proposing an accelerated version of (2.2), we leave that for future research.

REMARK 4. *Note that the step-sizes chosen in Theorems 4.2 and 4.3 are fundamentally different. While the former uses time-invariant step-sizes, the latter involves time-decaying ones. Therefore, it is instructive to understand the average regret bounds in terms of the* effective time $S_T = \sum_{k=0}^{T-1} \eta_k$ *(defined earlier in Theorem 4.1). For Theorem 4.2,* $S_T = \sqrt{T}$ *and the average regret*

$$\frac{1}{T} \sum_{k=0}^{T} \left(KL(\rho_k|e^{-f}) - KL(\pi|e^{-f}) \right) = O(T^{-1/2}) = O(S_T^{-1}).$$

So the average regret in this case decays polynomially. On the other hand, for Theorem 4.3, $S_T \approx \log T$ *and the average regret*

$$\frac{1}{T} \sum_{k=0}^{T} \left(KL(\rho_k|e^{-f}) - KL(\pi|e^{-f}) \right) = O(T^{-1} \log T) = O(S_T e^{-S_T}).$$

Therefore, in contrast to Theorem 4.2, the average regret in this case decays nearly at an exponential rate in terms of the effective time.

REMARK 5. In [39, Theorem 2], the authors obtain a $O(\sqrt{T})$-regret bound (similar to our Theorem 4.2) for a different mirror-descent-like algorithm, where the mirror map is also a certain KL divergence functional. Unlike us, they impose a uniform logarithmic Sobolev inequality type assumption on the KL divergence functional, which makes their approach very different from ours. The same mirror-descent algorithm was also studied in [5], where the authors assume a stronger version of Lemma 3.2 (both an upper and lower bound). However, the authors do not establish sufficient conditions as we do in Lemma 3.2. These papers use fixed step-sizes and do not recover logarithmic regret bounds like our Theorem 4.3.

The final result in this section establishes the rate of convergence of the last iteration ρ_T to e^{-f} in terms of the Bregman divergence B_G, under the same assumptions that lead to logarithmic regret in Theorem 4.3. The choice of step-sizes $\{\eta_k\}$ is, however, different in the two cases.

THEOREM 4.4. *Consider the same assumptions as in Theorem 4.3, part (ii).*
(i) The update from ρ_k to ρ_{k+1} satisfies the following asymptotic contraction:

$$\limsup_{\eta_k \to 0} \frac{1}{\eta_k} \frac{B_G(e^{-f}|\rho_{k+1}) - B_G(e^{-f}|\rho_k)}{B_G(e^{-f}|\rho_k)} \leq -\frac{\lambda}{M}.$$

(ii) By choosing $\eta_k = (CMB_G(e^{-f}|\rho_0)\log T)/(\lambda T), \forall k \leq T$ with $C > 0$, the following holds for all large enough $T > 0$:

$$B_G(e^{-f}|\rho_T) \leq \left(\frac{1}{T^{CB_G(e^{-f}|\rho_0)}} + \frac{CM^2 A}{2\lambda^2 m} \frac{\log T}{T} \right) B_G(e^{-f}|\rho_0).$$

The first part of Theorem 4.4 demonstrates an "asymptotic contraction" of the sequence of probability measures $\{\rho_k\}$; that is, if $\eta_k \to 0$, then

$$B_G(e^{-f}|\rho_{k+1}) \leq (1 - \eta_k \cdot \lambda/M + o(\eta_k))\, B_G(e^{-f}|\rho_k).$$

This implies that the local contraction rate of ρ_k is governed by the strong convexity parameter of g and the smoothness of the potentials $\{\psi_k\}_{k=0}^T$. Naturally, if the strong convexity parameter of g, specifically λ, is small, or if the smoothness parameter M of the $\{\psi_k\}$s is large, the rate of asymptotic contraction slows down.

The second part concerns the last iterate convergence of rate $O(T^{-1}\log T)$. It presents a non-asymptotic convergence rate of ρ_T to e^{-f} under the Bregman divergence $B_G(e^{-f}|\rho_T)$. When $B_G(e^{-f}|\rho_0) > 0$, it indicates a nearly $O(T^{-1})$ rate up to logarithmic factors, as the first term in the bound can be made arbitrarily small by selecting C large enough. Note that this bound also captures the fact that if ρ_0 and e^{-f} are close, then so are ρ_T and e^{-f}.

5. Applications.

5.1. *Generative Modeling and Neural-PDE.* In this section, we demonstrate how the discretized parabolic Monge-Ampère PDE (2.2) can be used to design new generative modeling algorithms, that is, learning to sample from a target measure μ with density $e^{-f} := \mathrm{d}\mu/\mathrm{d}x$. It turns out our discretized parabolic Monge-Ampère PDE integrates well with techniques established in current generative modeling literature: (i) learning density-ratio through logistic regression as in GANs [34]; (ii) learning the score function via score matching as in DPMs [42, 71]. Additionally, our discretized parabolic Monge-Ampère PDE can be implemented as a neural PDE using a residual neural network architecture [40] with standard auto-differentiation libraries.

Recall the discretized Monge-Ampère PDE (2.2); one crucial step for implementation is to learn

(i) the log density-ratio function $h_k(x) := \log\big(\rho_k(x)/e^{-f(x)}\big)$, or
(ii) its derivative, namely the score functions $\boldsymbol{m}_k(x) := \nabla \log\big(\rho_k(x)/e^{-f(x)}\big)$,

given data $X \sim e^{-f}$ and $\tilde{X} \sim \rho_k$ drawn i.i.d. from the target distribution and the simulated distribution respectively.

The following two propositions directly address how to estimate each term, with the first taken from [35, Proposition 1], and the second from [42, Theorem 1]. We shall design new generative modeling algorithms with neural networks based on these two propositions, inspired by the discretized parabolic Monge-Ampère PDE.

Neural-PDE via Logistic Regression.

PROPOSITION 5.1 (Log density-ratio via logistic regression). *Let $\rho, \pi \in \mathcal{P}_2^{\mathrm{ac}}(\mathbb{R}^d)$ and consider the following data-generating process $(X, L) \sim \gamma$: first, sample labels $L = 0$ or 1 with equal probability, and then generate data given the label as $X|L = 0 \sim \pi$ and $X|L = 1 \sim \rho$. With the data $(X, L) \sim \gamma$, define the logistic loss functional*

$$\mathcal{L}(h) := \mathop{\mathbb{E}}_{(X,L)\sim\gamma} \left[L \log\left(1 + e^{-h(X)}\right) + (1 - L) \log\left(1 + e^{h(X)}\right) \right] .$$

Assume $(\rho + \pi)(x) > 0, \forall x \in \mathbb{R}^d$, then the unique minimizer $h^* : \mathbb{R}^d \to \mathbb{R}$ of $\mathcal{L}(h)$ satisfies $h^*(x) = \log(\rho(x)/\pi(x))$.

Proposition 5.1 provides a procedure to estimate the log-density-ratio: given i.i.d. data $X_1, \cdots, X_n \sim e^{-f}$ and $\tilde{X}_1, \cdots, \tilde{X}_n \sim \rho_k = (\nabla\psi_k)\#e^{-g}$, augment to labeled data $\{(X_i, L_i = 0)\}_{i=1}^n \cup \{(\tilde{X}_i, \tilde{L}_i = 1)\}_{i=1}^n$ and denote its empirical distribution as $\widehat{\gamma}_k$; specify a neural network $h(\cdot;\theta) : \mathbb{R}^d \to \mathbb{R}$ parametrized by θ and solve

(5.1)
$$\widehat{h}_k(\cdot) := h(\cdot;\hat{\theta}) \text{ , where } \hat{\theta} \in \arg\min_{\theta} \mathop{\mathbb{E}}_{(X,L)\sim\widehat{\gamma}_k} \left[L \log\left(1 + e^{-h(X;\theta)}\right) + (1 - L) \log\left(1 + e^{h(X;\theta)}\right) \right] .$$

The minimizer $\widehat{h}_k$ approximates h_k, the true log density-ratio function.

In practice, one can implement $\widehat{h}_k(\cdot)$ with neural networks. This motivates a new generative modeling algorithm by iteratively refining the Brenier potential ψ_k with a new neural network $\widehat{h}_k$, coupled with auto-differentiation, composition, and residual connections; See Algorithm 1 below for details.

Neural-PDE via Score Matching.

PROPOSITION 5.2 (Score matching via Fisher divergence). *Consider $\rho \in \mathcal{P}_2^{\mathrm{ac}}(\mathbb{R}^d)$ and the corresponding data score function $\nabla \log \rho$. Let $x \in \mathbb{R}^d$ and $\boldsymbol{\sigma}_i(x)$ denote the i-th component of a vector-valued function. Consider the Fisher divergence functional $\mathcal{J}(\boldsymbol{\sigma})$*

$$\mathcal{J}(\boldsymbol{\sigma}) := \frac{1}{2} \mathop{\mathbb{E}}_{X\sim\rho} [\|\nabla \log \rho(X) - \boldsymbol{\sigma}(X)\|^2] .$$

Assume $\boldsymbol{\sigma}$ is differentiable and that, $\forall i$, $\rho(x)\boldsymbol{\sigma}_i(x) \to 0$ as $|x_i| \to \infty$. Then

$$\mathcal{J}(\boldsymbol{\sigma}) = \mathop{\mathbb{E}}_{X\sim\rho} \left[\sum_{i=1}^d \partial_i \boldsymbol{\sigma}_i(X) + \frac{1}{2} \boldsymbol{\sigma}_i(X)^2 \right] + const \ .$$

Proposition 5.2 also sets forth the following procedure to estimate the score functions: given i.i.d. $\tilde{X}_1, \cdots, \tilde{X}_n \sim \rho_k = (\nabla\psi_k)\#e^{-g}$, denote its empirical distribution $\widehat{\rho}_k$, specify a neural network $\boldsymbol{\sigma}(\cdot;\omega) : \mathbb{R}^d \to \mathbb{R}^d$ parametrized by ω and solve

$$\widehat{\boldsymbol{\sigma}}_k(\cdot) := \boldsymbol{\sigma}(\cdot;\hat{\omega}) \text{ , where } \hat{\omega} \in \arg\min_{\omega} \mathop{\mathbb{E}}_{X\sim\widehat{\rho}_k} \left[\sum_{i=1}^d \partial_i \boldsymbol{\sigma}_i(X;\omega) + \frac{1}{2} \boldsymbol{\sigma}_i(X;\omega)^2 \right] . \tag{5.2}$$

Algorithm 1: Monge-Ampère Neural-PDE via Logistic Regression

Input : T, total number of steps; $\{\eta_k\}_{k\le T}$, the step-sizes; initialize a neural network function $\psi_0 : \mathbb{R}^d \to \mathbb{R}$, and set $k = 0$.

Output: A neural network function $\psi_T : \mathbb{R}^d \to \mathbb{R}$, and samples from $(\nabla\psi_T)\#e^{-g}$.

while $k < T$ **do**

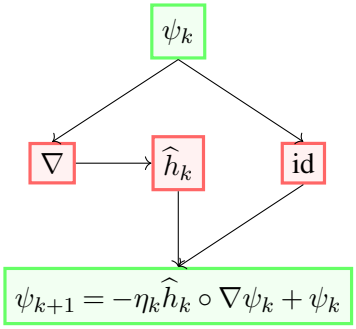
Sampling step: Given i.i.d. data $X_1, \cdots, X_n \sim e^{-f}$, sample $\tilde{X}_1, \cdots, \tilde{X}_n \sim \rho_k = (\nabla\psi_k)\#e^{-g}$, augment to $\{(X_i, L_i = 0)\}_{i=1}^n \cup \{(\tilde{X}_i, \tilde{L}_i = 1)\}_{i=1}^n$, and form its empirical distribution as $(X, L) \sim \widehat{\gamma}_k$;

Learning step: Estimate a neural network discriminator function $\widehat{h}_k$ as in (5.1) ;

Neural Network update: Define a new neural network with the residual architecture shown on the right $\psi_{k+1} = -\eta_k \widehat{h}_k \circ \nabla\psi_k + \psi_k$;

Set $k \leftarrow k + 1$;

end



The minimizer $\widehat{\boldsymbol{\sigma}}_k$ approximates the score function $\nabla \log \rho_k$; similarly, one can solve for $\widehat{\boldsymbol{\sigma}}$ that approximates the data score function $\nabla \log e^{-f}$, and define

$$\widehat{\boldsymbol{m}}_k := \widehat{\boldsymbol{\sigma}}_k - \widehat{\boldsymbol{\sigma}} \,. \tag{5.3}$$

This motivates a new generative modeling algorithm to refine the Brenier map $\boldsymbol{n}_k := \nabla\psi_k$ by iteratively adding neural network maps $\widehat{\boldsymbol{m}}_k$ using a residual connection, coupled with auto-differentiation and composition; we outline in Algorithm 2 below.

REMARK 6 (Parabolic Monge-Ampère discretizations in the literature). *A discretization similar to (2.2) was studied in [73, 74] with a time-invariant step-size sequence. However, the authors do not establish any convergence bounds. To the best of our knowledge, the integration of neural learning techniques from generative networks and score matching, as described in this section, is also novel within the parabolic PDE literature. This approach is practically useful because it avoids the otherwise cumbersome Hessian computation in (2.2). Moreover, the use of parabolic PDEs in existing literature is primarily for adaptive mesh generation (see [72, 14, 13]) to approximate solutions of other differential equations. In that sense, the connection between parabolic Monge-Ampère and generative modeling is also new.*

5.2. *Variational Formulation of Parabolic PDE.* We first present a variational objective that leads to the discretization (2.2). We will then use this variational objective to provide a new perspective on variational inference and also propose a new closed form iterative scheme for Gaussian variational inference.

Steepest Descent Interpretation. In Euclidean space, the forward gradient descent can be viewed as the solution of a minimizing movement scheme involving an objective function and a regularizer (the squared Euclidean distance). In this section, we show that the forward discretization (2.2) can also be viewed as the solution of a variational problem that involves

(i) an objective function, which will be the KL divergence to e^{-f} and (ii) a regularizer which will be a divergence on the space of probability measures (to be defined below). This is in sharp contrast to usual Wasserstein gradient flows where the regularizer is the usual 2-Wasserstein distance. To recast (2.2) as a variational problem, we will define the divergence on the tangent set $\mathrm{Tan}_{e^{-g}} := \overline{\{\nabla\psi : \psi \in \mathcal{C}_c^\infty\}} \subseteq L^2(e^{-g})$ where the closure is taken with respect to the $L^2(e^{-g})$ norm.

Algorithm 2: Monge-Ampère Neural-PDE via Score Matching

Input : T, total number of steps; $\{\eta_k\}_{k\leq T}$, the step-sizes; initialize a vector-valued neural network $\boldsymbol{n}_0 : \mathbb{R}^d \to \mathbb{R}^d$, and set $k = 0$.

Output: A vector-valued neural network function $\boldsymbol{n}_T : \mathbb{R}^d \to \mathbb{R}^d$, and samples from $(\boldsymbol{n}_T)\#e^{-g}$.

while $k < T$ **do**

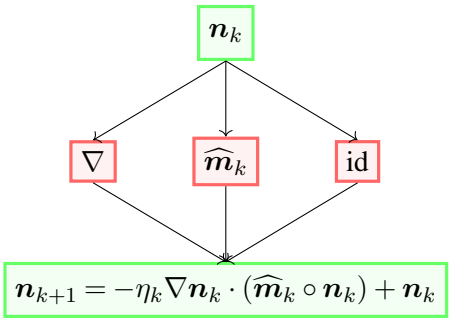
Sampling step: Obtain i.i.d. samples from $\rho_k = (\boldsymbol{n}_k)\#e^{-g}$, form the empirical measure $\widehat{\rho}_k$;

Learning step: Estimate a neural network score $\widehat{\boldsymbol{\sigma}}_k$ as in (5.2) and define the corresponding $\widehat{\boldsymbol{m}}_k$ as in (5.3) ;

Neural Network update: Define a new neural network with the residual architecture shown on the right $\boldsymbol{n}_{k+1} = -\eta_k \nabla\boldsymbol{n}_k \cdot (\widehat{\boldsymbol{m}}_k \circ \boldsymbol{n}_k) + \boldsymbol{n}_k$, here $\nabla\boldsymbol{n}_k : \mathbb{R}^d \to \mathbb{S}_+^{d\times d}$, and $\widehat{\boldsymbol{m}}_k \circ \boldsymbol{n}_k : \mathbb{R}^d \to \mathbb{R}^d$ and '$\cdot$' denotes the matrix-vector product ;

Set $k \leftarrow k + 1$;

end



DEFINITION 2 (Linearized Wasserstein divergence). *Let $\nabla\psi_1, \nabla\psi_2 \in \mathrm{Tan}_{e^{-g}}$ such that $\nabla\psi_1$ is twice differentiable and strictly convex. Then the linearized Wasserstein divergence*

$$LD_W(\nabla\psi_2; \nabla\psi_1) := \int (\nabla\psi_2(y) - \nabla\psi_1(y))^\top (\nabla^2\psi_1(y))^{-1} (\nabla\psi_2(y) - \nabla\psi_1(y)) e^{-g(y)} \mathrm{d}y.$$

We used the term linearized because without the inverse Hessian term, LD_W is exactly the popular linearized optimal transport distance in the literature; see [15, 61, 80]. On the other hand, we call it a divergence because if $\nabla\psi_2$ and $\nabla\psi_1$ are "close", then $LD_W(\nabla\psi_2; \nabla\psi_1)$ is the second-order approximation for the following expected Bregman divergence $\mathbb{E}_{Y\sim e^{-g}}[\psi_2^*(\nabla\psi_2(Y)) - \psi_2^*(\nabla\psi_1(Y)) - \langle \nabla\psi_2(Y) - \nabla\psi_1(Y), \nabla\psi_2^*(\nabla\psi_1(Y))\rangle]$ (see Section 3 for more details on Bregman divergences). As a final step to defining the variational problem for (2.2), we reparametrize the KL divergence as a functional on $\mathrm{Tan}_{e^{-g}}$, i.e., define $\tilde{F}(\nabla\psi) := KL((\nabla\psi)\#e^{-g} | e^{-f})$.

PROPOSITION 5.3. *Suppose $\nabla\psi_1 \in \mathrm{Tan}_{e^{-g}}$ is strictly convex and twice differentiable. Also, assume that both f and g are twice differentiable and the map*

$$y \mapsto \sum_{j=1}^d \int \frac{\partial^2}{\partial y_i \partial y_j} \psi^*(\nabla\psi(y)) \frac{\partial}{\partial y_j} \Theta(y) e^{-g(y)} \mathrm{d}y$$

vanishes as $|y_i| \to \infty$ *for all* $i = 1, 2, \ldots, d$, *and all* $\Theta \in \mathcal{C}_c^\infty$. *Let*

(5.4)
$$\nabla\tilde{\psi} \in \underset{\nabla\psi \in L^2(e^{-g})}{\arg\min} \left[\tilde{F}(\nabla\psi_1) + \left\langle \frac{\delta}{\delta(\nabla\psi)} \tilde{F}(\nabla\psi)\big|_{\nabla\psi=\nabla\psi_1}, \nabla\psi - \nabla\psi_1 \right\rangle_{e^{-g}} + \frac{1}{2\tau} L D_W(\nabla\psi; \nabla\psi_1) \right].$$

Then any optimizer $\nabla\tilde{\psi}$ *of the above variational problem satisfies the following stationary condition:*

$$\nabla\tilde{\psi}(y) - \nabla\psi_1(y) = -\tau\nabla\left(f(\nabla\psi_1(y)) - g(y) - \log\det\left(\nabla^2\psi_1(y)\right)\right)$$

We defer the proof of Theorem 5.3 to Section D of the Supplement [26]. Note that the above variational problem is reminiscent of preconditioned gradient descent [10, 50] and Newton iterations [6, 17]. However, to the best of our knowledge, the above variational problem is new in the literature, which makes our parabolic PDE-based approach different from contemporary mirror descent schemes on the 2-Wasserstein space, e.g., [5, 39, 44, 48].

Bayesian Variational Inference. Our discretized parabolic PDE (2.2) and the variational interpretation in Theorem 5.3 can be leveraged to provide an alternate perspective on variational inference. A common computational challenge in Bayesian Statistics is to compute integrals with respect to a complicated high-dimensional probability distribution, say our target e^{-f}. Variational inference [9] has emerged as a computationally viable approximation. At the core of variational inference is to find

$$\hat{\rho}_{\mathcal{V}} := \underset{\rho \in \mathcal{V}}{\arg\min}\, KL(\rho | e^{-f}),$$

where $\mathcal{V}$ denotes a candidate variational family of distributions. These families are chosen so that integrals against measures in $\mathcal{V}$ are easy to compute. Popular examples of variational families include:

(i) Gaussian variational inference [27, 45] - Here $\mathcal{V}$ is the family of d-dimensional Gaussian distributions, i.e., $\mathcal{V} := \{N(m, \Sigma) : m \in \mathbb{R}^d,\ \Sigma \in \mathbb{S}_+^{d\times d}$ is symmetric positive definite$\}$.
(ii) Mean-field variational inference [43, 78] - Here $\mathcal{V}$ is the family of d-dimensional product measures, i.e., $\mathcal{V} = \{\pi_1 \otimes \ldots \otimes \pi_d : \pi_i$ is a probability measure on $\mathbb{R}$, $1 \le i \le d\}$.

An alternate way of looking at the above variational inference problem is to formulate it in terms of transport maps from a simple reference distribution e^{-g}, say $N(0, \lambda^2 Id)$ for some $\lambda > 0$. So, instead of parameterizing with a family of distributions, we now parameterize with a family of transport maps. We define

(5.5)
$$\nabla\hat{\psi}_{\boldsymbol{\mathcal{S}}} := \underset{\nabla\psi \in \mathcal{S}}{\arg\min}\, KL\big(\nabla\psi \# e^{-g} | e^{-f}\big),$$

where $\boldsymbol{\mathcal{S}} \subseteq L^2(e^{-g})$ is now a *variational family of functions*. Some natural examples would include:

(i) Gaussian variational inference - Here

$$\boldsymbol{\mathcal{S}} := \{Ay + m : m \in \mathbb{R}^d,\ A \in \mathbb{S}_+^{d\times d} \text{ is symmetric positive definite}\}.$$

(ii) Mean-field variational inference - Here

$$\boldsymbol{\mathcal{S}} := \{(S_1(y_1), \ldots, S_d(y_d)) : S_i : \mathbb{R} \to \mathbb{R} \text{ is monotone increasing}\}.$$

The above formulation in (5.5) should immediately remind us of the variational form introduced in Theorem 5.3 as $KL(\nabla\psi\#e^{-g}|e^{-f})$ is exactly $\tilde{F}(\nabla\psi)$ from Theorem 5.3. Therefore, we can iteratively solve (5.4) over a potentially nonparametric class of vector fields $\boldsymbol{\mathcal{S}} \subseteq L^2(e^{-g})$, i.e.,

(5.6)
$$\nabla\psi_{k+1} \in \operatorname*{arg\,min}_{\nabla\psi\in\boldsymbol{\mathcal{S}}} \left[\tilde{F}(\nabla\psi_k) + \left\langle \frac{\delta}{\delta(\nabla\psi)}\tilde{F}(\nabla\psi_k), \nabla\psi - \nabla\psi_k \right\rangle_{e^{-g}} + \frac{1}{2\tau} LD_W(\nabla\psi;\nabla\psi_k) \right].$$

Here the first variation $\frac{\delta}{\delta(\nabla\psi)}\tilde{F}(\nabla\psi_k)$ is the variation restricted to the class $\boldsymbol{\mathcal{S}}$, i.e., for any $\nabla\Theta \in \boldsymbol{\mathcal{S}}$, $\langle \frac{\delta}{\delta(\nabla\psi)}\tilde{F}(\nabla\psi_k), \nabla\Theta\rangle_{e^{-g}} = \lim_{\epsilon\to 0} \frac{\tilde{F}(\nabla\psi_k+\epsilon\nabla\Theta)-\tilde{F}(\nabla\psi_k)}{\epsilon}$. For parametric classes of $\boldsymbol{\mathcal{S}}$ such as in Gaussian variational inference, one could directly solve (5.6) over the natural parameters μ and A. In the univariate setting, the new Gaussian variational inference updates are summarized in Algorithm 3.

Algorithm 3: Gaussian Variational Inference

Input : T, total number of steps; $\{\eta_k\}_{k\le T}$, the step-sizes; e^{-g}, density of $N(0,\lambda^2)$, $\lambda > 0$, initializer (m_0, σ_0)

Output: Final mean and standard deviation (m_T, σ_T).

while $k < T$ **do**

Update variational parameters:

$$(m_{k+1}, \sigma_{k+1}) \leftarrow G_{\eta_k}(m_k, \sigma_k) \qquad \text{where}$$

$$G_\eta(t,s) := \begin{pmatrix} t - \frac{\eta s}{\lambda}\,\mathbb{E}_{Y\sim N(0,\lambda^2)}\, f'((s/\lambda)Y + t)) \\ s - \frac{\eta}{\lambda^2}(s^2\,\mathbb{E}_{Y\sim N(0,\lambda^2)}\, f''((s/\lambda)Y + t) - 1). \end{pmatrix}$$

Set $k \leftarrow k+1$

end

To heuristically see the validity of Algorithm 3, consider the case $\eta_k \equiv \eta$ in which case the updates can be written as the fixed point iteration $(m_{k+1}, \sigma_{k+1}) = G_\eta(m_k, \sigma_k)$ where $G_\eta(\cdot,\cdot)$ is defined in Algorithm 3. Therefore, the limit (m_k, σ_k) as $k \to \infty$, say (m, σ) if it exists, would satisfy $(m, \sigma) = G_\eta(m, \sigma)$. This simplifies to

$$\mathop{\mathbb{E}}_{X\sim N(m,\sigma^2)} f'(X) = 0, \quad \text{and} \quad \mathop{\mathbb{E}}_{X\sim N(m,\sigma^2)} f''(X) = \sigma^{-2},$$

which are exactly the stationary conditions for (m, σ^2) arising as a solution of the Gaussian variational inference problem; see [45, Equation 1.9]. In Section E of the Supplement [26], we provide a numerical experiment that illustrates the performance of Algorithm 3.

6. Numerical Experiments. In this section, we provide a numerical example to demonstrate the efficacy of our proposed algorithms toward sampling from a non-log-concave target. To wit, we will use a Gaussian location mixture as our target distribution

$$e^{-f(x)} = \frac{1}{2}\phi(x-2) + \frac{1}{2}\phi(x+2), \tag{6.1}$$

where $\phi(\cdot)$ is the standard normal density. We initialize $\psi_0(y) = y^2/2$ and choose the reference $e^{-g(y)}$ to be the standard normal density. Note that the optimal transport from the standard normal to the mixture (6.1) is given by

$$\psi_\star'(y) = F^{-1} \circ \Phi(y), \tag{6.2}$$

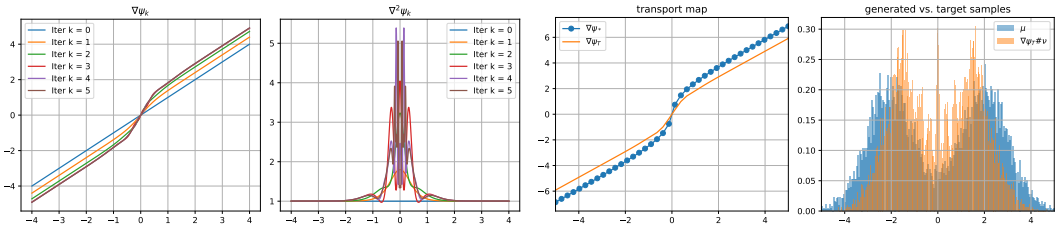


FIG 2. *Oracle updates — The first and second plots track the evolution of the gradient and the Hessian, respectively. The third plot compares the learned transport map at the final iterate with the target optimal transport map from* $N(0,1)$ *to the Gaussian mixture* (6.1)*. The fourth plot compares the histograms between* 10000 *samples from the mixture in* (6.1) *and* 10000 *samples from the push-forward measure* $\nabla\psi_T \# N(0,1)$.

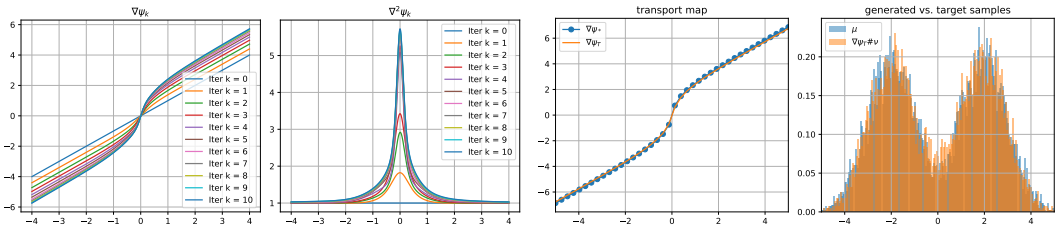


FIG 3. *Same as Figure 2 for oracle updates with distillation.*

where $\Phi(\cdot)$ is the standard normal cumulative distribution function and $F(x) = (1/2)(\Phi(x-2) + \Phi(x+2))$. The discrete updates from (2.2) can now be written as

$$\psi_{k+1}(y) = \psi_k(y) + \eta_k \Delta_k(y)\,, \tag{6.3}$$
$$\Delta_k(y) := \log\left(\frac{1}{2}\phi(\psi_k'(y)-2) + \frac{1}{2}\phi(\psi_k'(y)+2)\right) + \frac{1}{2}y^2 + \log\left(\psi_k''(y)\right)\,,$$

under an appropriately chosen sequence of step-sizes which will be described below. We will describe our experiments in four steps that highlight the practical challenges in discretization and our proposed strategies for addressing them.

Oracle computation of (2.2) *with batch auto-differentiation.* In the first approach, we assume access to the functional form of the target e^{-f} and we update the potentials analytically from (6.3). All gradients and higher-order derivatives were computed using the automatic differentiation module available in `PyTorch` (through the `torch.autograd` function). The step-sizes are chosen adaptively as follows:

- We partition the range $[-3,3]$ into 1000 equally spaced parts. Suppose $\{a_1, a_2, \ldots, a_{1000}\}$ denotes the boundary points of the partition.
- Suppose that we have constructed up to ψ_k, we then choose

$$\eta_k = \frac{1}{2}\max\left\{\max\left\{-\frac{\psi_k''(a_i)}{\Delta_k''(a_i)} : \Delta_k''(a_i) < 0, 1 \le i \le 1000\right\}, 0.4\right\}. \tag{6.4}$$

The above choice ensures that the double derivative of ψ_{k+1} is strictly positive at the grid points. The performance of the oracle procedure is illustrated in Figure 2. There are two major challenges which arise in the above implementation:

(a) The higher-order derivatives are numerically unstable. This instability accumulates in the Hessians of the potentials as can be seen in the second panel of Figure 2.

(b) Computing the gradients and Hessians via backpropagation requires storing the full computational graph which is memory intensive. In our experiments, we found it hard to go beyond $T = 5$ or 6 iterations. While the learned transport map after 5 iterations captures a bimodal structure, the third and fourth panels of Figure 2 show that it is not reasonably close to $\psi'_\star$ (see (6.2)).

Oracle computation of (2.2) *with distillation.* To address the numerical instabilities and memory allocation issues with the oracle computation, we use a *distillation* step at every iteration. The distillation decouples the computational graph and regularizes the instabilities. It proceeds as follows: with the k-th iteration potential ψ_k, we calculate the update direction $\Delta_k(y)$ according to (6.3). Then we train a simple *student* network by minimizing

$$\Delta_k^{\text{stud}} \in \operatorname*{arg\,min}_{\mathrm{nn}(\cdot;\theta)} \frac{1}{M} \sum_{j=1}^{M} (\Delta'_k(Z_j) - \mathrm{nn}'(Z_j;\theta))^2, \tag{6.5}$$

where the above minimization is over $\mathrm{nn}(\cdot;\theta)$, a three-layer neural network with two hidden layers each of width 32 with weights θ and $Z_1, \ldots, Z_M \overset{i.i.d.}{\sim} \mathrm{Unif}[-3,3]$, with $M = 500$. The above distillation step initializes the student network from a cold start and uses the `Adam` optimizer for training. To guarantee smoothness, we choose the softplus activation function ($x \mapsto \log(1+e^x)$) for the student network. We then update $\psi_{k+1} = \psi_k + \eta_k \Delta_k^{\text{stud}}$ with adaptive step-sizes from (6.4). This process is repeated recursively. It ensures that the gradients and Hessians of ψ_k are stable. It also avoids the need to store a large computational graph (growing exponentially with number of iterations), hence overcoming the memory allocation issues. This allows us to run more iterations $T = 10$ with much less computational time. The performance of this procedure is presented in Figure 3. In contrast to Figure 2, the gradients and Hessians in Figure 3 are much more stable. Within 10 iterations, we observe very close agreement between ψ'_k and $\psi'_\star$ in the third and fourth panels of Figure 3.

Algorithm 2 with distillation. Both of the above approaches rely on the knowledge of the functional form of the target e^{-f}. In practice, we only have access to samples, say $X_1, \ldots, X_n \overset{i.i.d.}{\sim} e^{-f}$. In such a scenario, let us first discuss the performance of Algorithm 2 with $n = 10000$. In (5.2), we use a two hidden-layer neural network of width 32, and the softplus activation function, to optimize the score-matching objective. We use the same adaptive step-sizes as in (6.4) and distillation approach similar to (6.5). Figure 4 shows that the iterates are much more stable than the analytic oracle updates. After 10 iterations, the learned transport map ψ'_k is close to the optimal map $\psi'_\star$ in the interval $[-3,3]$, beyond which the approximation deteriorates slightly (particularly for $|y| > 3$). A potential reason for this is that the score matching approach tries to learn the score for the target (6.1) and the current iterate distribution $\rho_k = (\psi'_k)\#N(0,1)$ separately, without leveraging the fact that the two scores are likely to be closer as the number of iterates increases. This limitation is overcome by our next approach.

Algorithm 1 with distillation. Let us now discuss the performance of Algorithm 1. In (5.1), we again use a two-hidden-layer neural network of width 32, and the softplus activation function, to optimize the logistic loss. We use the same adaptive step-sizes as in (6.4) and distillation approach as in (6.5). This time the distillation learns an estimate of the log density-ratio via a simple student network, say Δ_k^{stud}. Figure 5 shows that the iterates (their gradients as well as hessians) are stable. After 10 iterations, the learned transport map ψ'_k nicely approximates the true optimal map $\psi'_\star$, better than in the previous score matching based approach, particularly outside $[-3,3]$. We believe this is because Algorithm 1 learns the logarithm of the density ratio between the current distribution $\rho_k = (\psi'_k)\#N(0,1)$ and the target (6.1)

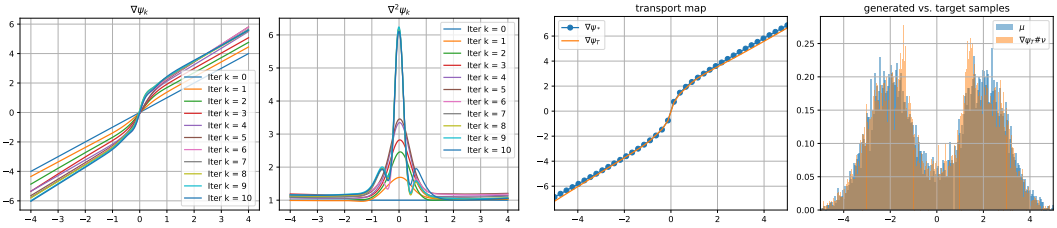


FIG 4. *Same as Figure 2 for score matching based Algorithm 2 with distillation.*

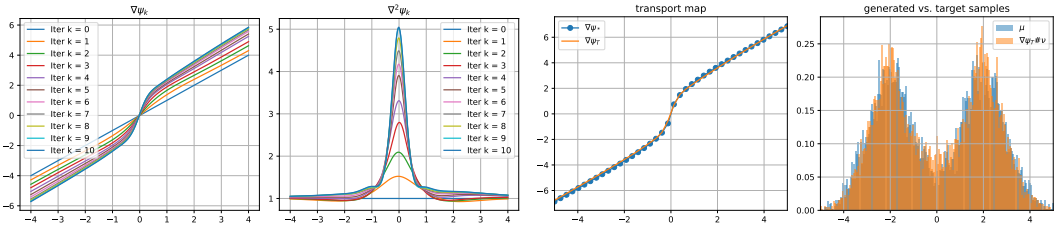


FIG 5. *Same as Figure 2 for logistic regression based Algorithm 1 with distillation.*

directly (unlike in Algorithm 2 where the score functions are learned separately). Since this log density-ratio gets closer to 0 as the number of iterations grows, it can be approximated accurately using a simple neural network.

REMARK 7 (Effect of distillation). *Let us consider our approach based on logistic regression. Note that the distillation step attempts to approximate the log density-ratio* $f + \log \rho_k$ *with a simpler student network, say* Δ_k^{stud}*. This approximation error can be transferred easily to our current regret bounds. In particular, if*

$$\int (f + \log \rho_k - \Delta_k^{stud})(x)(\rho_k - e^{-f})(x)\, dx = O(\eta_k),$$

where η_k *denotes the step-size, then all our regret bounds and last iterate convergence bounds continue to hold after adjusting for constants. In Section E of the Supplement [26], we discuss another method to avoid long backpropagation chains that do not require distillation.*

REMARK 8 (On termination criteria). *A practical question is how to determine when to terminate the proposed algorithms. One natural strategy is to assess whether the push forward samples* $\nabla\psi_k(Y_i)$*, where* $Y_i \sim \mathcal{N}(0, I)$ *(for instance), are statistically indistinguishable from samples drawn from the target distribution. This can be done by performing a two-sample hypothesis test (for example, an energy [75] or kernel-based test [37]) at a prescribed significance level* α*. Alternatively, one may compute a discrepancy measure such as the maximum mean discrepancy (see [37, Equation 3]) between the generated and target samples, and monitor its decay across iterations. The iteration can then be stopped once the maximum mean discrepancy falls below a user-defined tolerance, providing a simple and data-driven convergence criterion.*

Discussion and Future work. In this paper, we have introduced a new generative modeling framework which draws a novel connection between a parabolic PDE ((2.1)) and probabilistic sampling. We have proposed a time discretization for (2.1) and established different

regimes of convergence rates to the target distribution, depending on smoothness assumptions and appropriate step-size choices. Our theoretical results include bounds on average iterate (see Theorem 4.1), average regret (see Theorems 4.2 and 4.3) and last iterate (see Theorem 4.4). In addition, we show how the discretization integrates nicely with neural network based learning techniques from generative networks and score-based diffusion models (see Section 5.1). Finally, we illustrate how the proposal naturally yields a new family of variational inference algorithms. Overall, our proposal combines the ease of one-pass sampling typically associated with GANs (generative adversarial networks) alongside the ease of distribution learning associated with DPMs (diffusion-based probabilistic models).

While our paper is an early theoretical step towards bridging the gap between parabolic PDE discretization and sampling, a number of interesting theoretical and practical questions remain open. A few of them are discussed below.

1. **Sample complexity bounds**: Given data $X_1, \ldots, X_n \overset{i.i.d.}{\sim} e^{-f}$, suppose the initial distribution ρ_0 is chosen to be the empirical measure $n^{-1}\sum_{i=1}^n \delta_{X_i}$. A natural question would be to establish a sample complexity bound, i.e., given an $\epsilon > 0$, one must find $n \equiv n(\epsilon, e^{-f})$, $T \equiv T(\epsilon, e^{-f})$, and appropriate step-sizes $\{\eta_k\}_{k=0}^T$ such that $KL(\rho_T|e^{-f}) \leq \epsilon$.
2. **Structural estimation in high dimension**: Empirically it is often observed that structural assumptions on e^{-f} can lead to faster convergence of generative models, particularly when the dimension d is large. It would be theoretically challenging and practically useful to establish such results for our proposed parabolic Monge-Ampère discretization. Two natural structures one can impose are: (a) Low intrinsic dimension — Here the target distribution is supported on some low-dimensional manifold. While this means the target does not admit a Lebesgue density, we note that Algorithm 1 from Section 5.1 does not require the existence of a density, and (b) Markov random field: Here the target has a density e^{-f} but the (conditional) dependence structure is governed by an underlying sparse graph. A canonical example would be a standard Markov chain.
3. **Optimal transport map estimation**: There has been extensive research in recent years towards constructing computationally tractable estimators of optimal transport maps. While the Sinkhorn algorithm shows tremendous potential, establishing statistical convergence rates for a Sinkhorn-based estimator, under smoothness/structural conditions, remains open. Since the parabolic PDE targets the optimal transport map from e^{-g} to e^{-f}, our proposed discretizations yield an estimator (say $\nabla\hat{\psi}_T$) of this optimal transport map. It would be interesting to explore whether one can establish sample complexity bounds under the popular metric $\|\nabla\hat{\psi}_T - \nabla\psi_\infty\|_{L^2(e^{-g})}$, and show that it offers fast convergence under structural/smoothness assumptions on e^{-f}.
4. **Empirical validation on high-resolution image data**: Diffusion based generative models are popularly used to generate high-resolution images. It would be interesting to explore empirically how well our proposed algorithms perform in such image generation experiments. There are interesting practical challenges that arise while working with images, such as (a) potential violation of smoothness assumptions required for Theorems 4.2, 4.3, and 4.4, (b) designing structured neural networks that exploit potential low-dimensional nature of images, and (c) tuning the step-size sequence adaptively.

Acknowledgment. Liang acknowledges the generous support from the NSF Career Award (DMS-2042473) and the Wallman Society of Fellows at the University of Chicago. The authors thank Takuya Koriyama for implementing a distillation module, which we use in the numerical experiment section.

SUPPLEMENTARY MATERIAL

Supplement to "No-Regret Generative Modeling via Parabolic Monge-Ampère PDE"
In the Supplementary file [26], we provide proof of all results in this paper, certain important auxiliary results, and a numerical experiment.

REFERENCES

[1] ABEDIN, F. and KITAGAWA, J. (2020). Exponential convergence of parabolic optimal transport on bounded domains. *Analysis & PDE* **13** 2183–2204.

[2] AMBROSIO, L., GIGLI, N. and SAVARÉ, G. (2008). *Gradient flows in metric spaces and in the space of probability measures*, second ed. *Lectures in Mathematics ETH Zürich*. Birkhäuser Verlag, Basel. MR2401600

[3] AMPÈRE, A.-M. (1819). *Mémoire contenant l'application de la théorie exposée dans le XVII. e Cahier du Journal de l'École polytechnique, à l'intégration des équations aux différentielles partielles du premier et du second ordre*. De l'Imprimerie royale.

[4] ARJOVSKY, M., CHINTALA, S. and BOTTOU, L. (2017). Wasserstein generative adversarial networks. In *International conference on machine learning* 214–223. PMLR.

[5] AUBIN-FRANKOWSKI, P.-C., KORBA, A. and LÉGER, F. (2022). Mirror descent with relative smoothness in measure spaces, with application to sinkhorn and em. *Advances in Neural Information Processing Systems* **35** 17263–17275.

[6] BATTITI, R. (1992). First-and second-order methods for learning: between steepest descent and Newton's method. *Neural computation* **4** 141–166.

[7] BECK, A. and TEBOULLE, M. (2003). Mirror descent and nonlinear projected subgradient methods for convex optimization. *Oper. Res. Lett.* **31** 167–175. MR1967286

[8] BERMAN, R. J. (2020). The Sinkhorn algorithm, parabolic optimal transport and geometric Monge–Ampère equations. *Numerische Mathematik* **145** 771–836.

[9] BLEI, D. M., KUCUKELBIR, A. and MCAULIFFE, J. D. (2017). Variational inference: A review for statisticians. *Journal of the American statistical Association* **112** 859–877.

[10] BONET, C., USCIDDA, T., DAVID, A., AUBIN-FRANKOWSKI, P.-C. and KORBA, A. (2024). Mirror and preconditioned gradient descent in wasserstein space. *arXiv preprint arXiv:2406.08938*.

[11] BRENIER, Y. (1991). Polar factorization and monotone rearrangement of vector-valued functions. *Communications on pure and applied mathematics* **44** 375–417.

[12] BUBECK, S., COHEN, M. B., LEE, J. R. and LEE, Y. T. (2021). Metrical task systems on trees via mirror descent and unfair gluing. *SIAM J. Comput.* **50** 909–923. MR4264651

[13] BUDD, C. J., HUANG, W. and RUSSELL, R. D. (2009). Adaptivity with moving grids. *Acta Numerica* **18** 111–241.

[14] BUDD, C. J. and WILLIAMS, J. (2009). Moving mesh generation using the parabolic Monge–Ampère equation. *SIAM Journal on Scientific Computing* **31** 3438–3465.

[15] CAI, T., CHENG, J., CRAIG, N. and CRAIG, K. (2020). Linearized optimal transport for collider events. *Physical Review D* **102** 116019.

[16] CAO, T., BIE, A., VAHDAT, A., FIDLER, S. and KREIS, K. (2021). Don't generate me: Training differentially private generative models with sinkhorn divergence. *Advances in Neural Information Processing Systems* **34** 12480–12492.

[17] CARTIS, C., GOULD, N. I. and TOINT, P. L. (2010). On the complexity of steepest descent, Newton's and regularized Newton's methods for nonconvex unconstrained optimization problems. *Siam journal on optimization* **20** 2833–2852.

[18] CESA-BIANCHI, N. and LUGOSI, G. (2006). *Prediction, learning, and games*. Cambridge University Press, Cambridge. MR2409394

[19] CHEN, G. and TEBOULLE, M. (1993). Convergence analysis of a proximal-like minimization algorithm using Bregman functions. *SIAM Journal on Optimization* **3** 538–543.

[20] CHEN, S., CHEWI, S., LEE, H., LI, Y., LU, J. and SALIM, A. (2024). The probability flow ode is provably fast. *Advances in Neural Information Processing Systems* **36**.

[21] CHEN, S., CHEWI, S., LI, J., LI, Y., SALIM, A. and ZHANG, A. R. (2022). Sampling is as easy as learning the score: theory for diffusion models with minimal data assumptions. *arXiv preprint arXiv:2209.11215*.

[22] CHEWI, S. (2023). Log-concave sampling. *Book draft*.

[23] CUTURI, M. (2013). Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems* **26**.

[24] DE BORTOLI, V., THORNTON, J., HENG, J. and DOUCET, A. (2021). Diffusion schrödinger bridge with applications to score-based generative modeling. *Advances in Neural Information Processing Systems* **34** 17695–17709.

[25] DEB, N., KIM, Y.-H., PAL, S. and SCHIEBINGER, G. (2023). Wasserstein mirror gradient flow as the limit of the sinkhorn algorithm. *arXiv preprint arXiv:2307.16421*.

[26] DEB, N. and LIANG, T. (2026). Supplement to "No-Regret Generative Modeling via Parabolic Monge-Ampère PDE".

[27] DIAO, M. Z., BALASUBRAMANIAN, K., CHEWI, S. and SALIM, A. (2023). Forward-backward Gaussian variational inference via JKO in the Bures-Wasserstein space. In *International Conference on Machine Learning* 7960–7991. PMLR.

[28] DIEULEVEUT, A., DURMUS, A. and BACH, F. (2020). Bridging the gap between constant step size stochastic gradient descent and Markov chains. *Ann. Statist.* **48** 1348–1382. MR4124326

[29] FOKKER, A. D. (1914). Die mittlere Energie rotierender elektrischer Dipole im Strahlungsfeld. *Annalen der Physik* **348** 810–820.

[30] FORTET, R. (1940). Résolution d'un système d'équations de M. Schrödinger. *Journal de mathématiques pures et appliquées* **19** 83–105.

[31] FRANKLIN, J. and LORENZ, J. (1989). On the scaling of multidimensional matrices. *Linear Algebra and its applications* **114** 717–735.

[32] FRIGYIK, B. A., SRIVASTAVA, S. and GUPTA, M. R. (2008). Functional Bregman divergence and Bayesian estimation of distributions. *IEEE Transactions on Information Theory* **54** 5130–5139.

[33] GENEVAY, A., PEYRÉ, G. and CUTURI, M. (2018). Learning generative models with sinkhorn divergences. In *International Conference on Artificial Intelligence and Statistics* 1608–1617. PMLR.

[34] GOODFELLOW, I., POUGET-ABADIE, J., MIRZA, M., XU, B., WARDE-FARLEY, D., OZAIR, S., COURVILLE, A. and BENGIO, Y. (2020). Generative adversarial networks. *Communications of the ACM* **63** 139–144.

[35] GOODFELLOW, I. J., POUGET-ABADIE, J., MIRZA, M., XU, B., WARDE-FARLEY, D., OZAIR, S., COURVILLE, A. and BENGIO, Y. (2014). Generative adversarial nets. *Advances in neural information processing systems* **27**.

[36] GORDON, G. J. (1999). Regret bounds for prediction problems. In *Proceedings of the Twelfth Annual Conference on Computational Learning Theory (Santa Cruz, CA, 1999)* 29–40. ACM, New York. MR1811599

[37] GRETTON, A., BORGWARDT, K. M., RASCH, M. J., SCHÖLKOPF, B. and SMOLA, A. (2012). A kernel two-sample test. *The journal of machine learning research* **13** 723–773.

[38] GUO, W., HUR, Y., LIANG, T. and RYAN, C. (2022). Online learning to transport via the minimal selection principle. In *Conference on Learning Theory* 4085–4109. PMLR.

[39] HAN, D.-S., KIM, J., YOO, H. B. and ZHANG, B.-T. (2025). Variational Mirror Descent for Robust Learning in Schrödinger Bridge.

[40] HE, K., ZHANG, X., REN, S. and SUN, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* 770–778.

[41] HUR, Y., GUO, W. and LIANG, T. (2024). Reversible Gromov–Monge sampler for simulation-based inference. *SIAM Journal on Mathematics of Data Science* **6** 283–310.

[42] HYVÄRINEN, A. (2005). Estimation of non-normalized statistical models by score matching. *J. Mach. Learn. Res.* **6** 695–709. MR2249836

[43] JIANG, Y., CHEWI, S. and POOLADIAN, A.-A. (2024). Algorithms for mean-field variational inference via polyhedral optimization in the Wasserstein space. In *The Thirty Seventh Annual Conference on Learning Theory* 2720–2721. PMLR.

[44] KARIMI, M. R., HSIEH, Y.-P. and KRAUSE, A. (2024). Sinkhorn flow as mirror flow: A continuous-time framework for generalizing the sinkhorn algorithm. In *International Conference on Artificial Intelligence and Statistics* 4186–4194. PMLR.

[45] KATSEVICH, A. and RIGOLLET, P. (2024). On the approximation accuracy of Gaussian variational inference. *Ann. Statist.* **52** 1384–1409. MR4804813

[46] KIM, Y.-H., STREETS, J. and WARREN, M. (2012). Parabolic optimal transport equations on manifolds. *International Mathematics Research Notices* **2012** 4325–4350.

[47] LAN, G., LU, Z. and MONTEIRO, R. D. (2011). Primal-dual first-order methods with iteration-complexity for cone programming. *Mathematical Programming* **126** 1–29.

[48] LÉGER, F. (2021). A gradient descent perspective on Sinkhorn. *Applied Mathematics & Optimization* **84** 1843–1855.

[49] LI, W. (2021). Transport information Bregman divergences. *Information Geometry* **4** 435–470.

[50] LI, X.-L. (2017). Preconditioned stochastic gradient descent. *IEEE transactions on neural networks and learning systems* **29** 1454–1466.

[51] LIANG, T. (2021). How well generative adversarial networks learn distributions. *Journal of Machine Learning Research* **22** 1–41.
[52] LIANG, T., DHARMAKEERTHI, K. and KORIYAMA, T. (2026). Denoising Diffusions with Optimal Transport: Localization, Curvature, and Multi-Scale Complexity. *Transactions on Machine Learning Research*.
[53] LIERO, M., MIELKE, A. and SAVARÉ, G. (2023). Fine properties of geodesics and geodesic λ-convexity for the Hellinger–Kantorovich distance. *Archive for Rational Mechanics and Analysis* **247** 112.
[54] LIM, S., YOON, E., BYUN, T., KANG, T., KIM, S., LEE, K. and CHOI, S. (2023). Score-based Generative Modeling through Stochastic Evolution Equations in Hilbert Spaces.
[55] LU, H., FREUND, R. M. and NESTEROV, Y. (2018). Relatively smooth convex optimization by first-order methods, and applications. *SIAM Journal on Optimization* **28** 333–354.
[56] LÜBECK, F., BUNNE, C., GUT, G., DEL CASTILLO, J. S., PELKMANS, L. and ALVAREZ-MELIS, D. (2022). Neural unbalanced optimal transport via cycle-consistent semi-couplings. *arXiv preprint arXiv:2209.15621*.
[57] MA, Y.-A., CHATTERJI, N. S., CHENG, X., FLAMMARION, N., BARTLETT, P. L. and JORDAN, M. I. (2021). Is there an analog of Nesterov acceleration for gradient-based MCMC? *Bernoulli* **27** 1942 – 1992.
[58] MCCANN, R. J. (1995). Existence and uniqueness of monotone measure-preserving maps. *Duke Math. J.* **80** 309–323. MR1369395
[59] MÉRIGOT, Q. (2016). Discretization of Euler's equations using optimal transport: Cauchy and boundary value problems. *Séminaire Laurent Schwartz—EDP et applications* 1–12.
[60] MONGE, G. (1784). *Mémoire sur le calcul intégral des équations aux différences partielles*. Imprimerie royale.
[61] MOOSMÜLLER, C. and CLONINGER, A. (2023). Linear optimal transport embedding: provable Wasserstein classification for certain rigid transformations and perturbations. *Information and Inference: A Journal of the IMA* **12** 363–389.
[62] PLANCK, V. (1917). Über einen Satz der statistischen Dynamik und seine Erweiterung in der Quantentheorie. *Sitzungberichte der*.
[63] RÜSCHENDORF, L. (1995). Convergence of the iterative proportional fitting procedure. *The Annals of Statistics* 1160–1174.
[64] RÜSCHENDORF, L. and THOMSEN, W. (1993). Note on the Schrödinger equation and I-projections. *Statistics & probability letters* **17** 369–375.
[65] SALIM, A., KORBA, A. and LUISE, G. (2020). The Wasserstein proximal gradient algorithm. *Advances in Neural Information Processing Systems* **33** 12356–12366.
[66] SANTAMBROGIO, F. (2015). *Optimal transport for applied mathematicians*. *Progress in Nonlinear Differential Equations and their Applications* **87**. Birkhäuser/Springer, Cham Calculus of variations, PDEs, and modeling. MR3409718
[67] SANTAMBROGIO, F. (2017). Euclidean, metric, and Wasserstein gradient flows: an overview. *Bulletin of Mathematical Sciences* **7** 87–154.
[68] SCHRÖDINGER, E. (1935). The present status of quantum mechanics. *Die Naturwissenschaften* **23** 1–26.
[69] SHALEV-SHWARTZ, S. (2012). Online learning and online convex optimization. *Foundations and Trends® in Machine Learning* **4** 107–194.
[70] SONG, Y., DURKAN, C., MURRAY, I. and ERMON, S. (2021). Maximum likelihood training of score-based diffusion models. *Advances in Neural Information Processing Systems* **34** 1415–1428.
[71] SONG, Y. and ERMON, S. (2019). Generative modeling by estimating gradients of the data distribution. *Advances in Neural Information Processing Systems* **32**.
[72] SULMAN, M., WILLIAMS, J. and RUSSELL, R. D. (2011). Optimal mass transport for higher dimensional adaptive grid generation. *Journal of computational physics* **230** 3302–3330.
[73] SULMAN, M. H., NGUYEN, T. B., HAYNES, R. D. and HUANG, W. (2021). Domain decomposition parabolic Monge–Ampère approach for fast generation of adaptive moving meshes. *Computers & Mathematics with Applications* **84** 97–111.
[74] SULMAN, M. M., WILLIAMS, J. and RUSSELL, R. D. (2011). An efficient approach for the numerical solution of the Monge–Ampère equation. *Applied Numerical Mathematics* **61** 298–307.
[75] SZÉKELY, G. J. and RIZZO, M. L. (2013). Energy statistics: A class of statistics based on distances. *Journal of statistical planning and inference* **143** 1249–1272.
[76] TZEN, B., RAJ, A., RAGINSKY, M. and BACH, F. (2023). Variational principles for mirror descent and mirror Langevin dynamics. *IEEE Control Syst. Lett.* **7** 1542–1547. MR4600662
[77] VILLANI, C. (2009). *Optimal transport: old and new* **338**. Springer.
[78] WAINWRIGHT, M. J. and JORDAN, M. I. (2008). Graphical models, exponential families, and variational inference. *Foundations and Trends® in Machine Learning* **1** 1–305.

[79] WANG, G., JIAO, Y., XU, Q., WANG, Y. and YANG, C. (2021). Deep generative learning via schrödinger bridge. In *International conference on machine learning* 10794–10804. PMLR.

[80] WANG, W., SLEPČEV, D., BASU, S., OZOLEK, J. A. and ROHDE, G. K. (2013). A linear optimal transportation framework for quantifying and visualizing variations in sets of images. *International journal of computer vision* **101** 254–269.

[81] ZHANG, S., CHEWI, S., LI, M., BALASUBRAMANIAN, K. and ERDOGDU, M. A. (2023). Improved discretization analysis for underdamped Langevin Monte Carlo. In *The Thirty Sixth Annual Conference on Learning Theory* 36–71. PMLR.