# Sliced-Wasserstein Flows: Nonparametric Generative Modeling via Optimal Transport and Diffusions

**Antoine Liutkus** [1] **Umut Şimşekli** [2] **Szymon Majewski** [3] **Alain Durmus** [4] **Fabian-Robert Stöter** [1]

## Abstract

By building upon the recent theory that established the connection between implicit generative modeling (IGM) and optimal transport, in this study, we propose a novel parameter-free algorithm for learning the underlying distributions of complicated datasets and sampling from them. The proposed algorithm is based on a functional optimization problem, which aims at finding a measure that is close to the data distribution as much as possible and also expressive enough for generative modeling purposes. We formulate the problem as a gradient flow in the space of probability measures. The connections between gradient flows and stochastic differential equations let us develop a computationally efficient algorithm for solving the optimization problem. We provide formal theoretical analysis where we prove finite-time error guarantees for the proposed algorithm. To the best of our knowledge, the proposed algorithm is the first nonparametric IGM algorithm with explicit theoretical guarantees. Our experimental results support our theory and show that our algorithm is able to successfully capture the structure of different types of data distributions.

## 1. Introduction

Implicit generative modeling (IGM) (Diggle & Gratton, 1984; Mohamed & Lakshminarayanan, 2016) has become very popular recently and has proven successful in various fields; variational auto-encoders (VAE) (Kingma & Welling, 2013) and generative adversarial networks (GAN) (Goodfellow et al., 2014) being its two well-known examples. The goal in IGM can be briefly described as learning the underlying probability measure of a given dataset, denoted as $\nu \in \mathcal{P}(\Omega)$, where $\mathcal{P}$ is the space of probability measures on the measurable space $(\Omega, \mathcal{A})$, $\Omega \subset \mathbb{R}^d$ is a domain and $\mathcal{A}$ is the associated Borel $\sigma$-field.

Given a set of data points $\{y_1, \dots, y_P\}$ that are assumed to be independent and identically distributed (i.i.d.) samples drawn from $\nu$, the implicit generative framework models them as the output of a measurable map, i.e. $y = T(x)$, with $T : \Omega_\mu \mapsto \Omega$. Here, the inputs $x$ are generated from a known and easy to sample source measure $\mu$ on $\Omega_\mu$ (e.g. Gaussian or uniform measures), and the outputs $T(x)$ should match the unknown target measure $\nu$ on $\Omega$.

Learning generative networks have witnessed several groundbreaking contributions in recent years. Motivated by this fact, there has been an interest in illuminating the theoretical foundations of VAEs and GANs (Bousquet et al., 2017; Liu et al., 2017). It has been shown that these implicit models have close connections with the theory of Optimal Transport (OT) (Villani, 2008). As it turns out, OT brings new light on the generative modeling problem: there have been several extensions of VAEs (Tolstikhin et al., 2017; Kolouri et al., 2018) and GANs (Arjovsky et al., 2017; Gulrajani et al., 2017; Guo et al., 2017; Lei et al., 2017), which exploit the links between OT and IGM.

OT studies whether it is possible to transform samples from a source distribution $\mu$ to a target distribution $\nu$. From this perspective, an ideal generative model is simply a transport map from $\mu$ to $\nu$. This can be written by using some 'push-forward operators': we seek a mapping $T$ that 'pushes $\mu$ onto $\nu$', and is formally defined as $\nu(A) = \mu(T^{-1}(A))$ for all Borel sets $A \subset \mathcal{A}$. If this relation holds, we denote the push-forward operator $T_\#$, such that $T_\#\mu = \nu$. Provided mild conditions on these distributions hold (notably $\mu$ is non-atomic (Villani, 2008)), existence of such a transport map is guaranteed; however, it remains a challenge to construct it in practice.

One common point between VAE and GAN is to adopt an approximate strategy and consider transport maps that

[1]Inria and LIRMM, Univ. of Montpellier, France [2]LTCI, Télécom Paristech, Université Paris-Saclay, Paris, France [3]Institute of Mathematics, Polish Academy of Sciences, Warsaw, Poland [4]CNRS, ENS Paris-Saclay,Université Paris-Saclay, Cachan, France. Correspondence to: Antoine Liutkus <antoine.liutkus@inria.fr>, Umut Şimşekli <umut.simsekli@telecom-paristech.fr>.

*Proceedings of the $36^{th}$ International Conference on Machine Learning*, Long Beach, California, PMLR 97, 2019. 

belong to a *parametric* family $T_\phi$ with $\phi \in \Phi$. Then, they aim at finding the best parameter $\phi^\star$ that would give $T_{\phi^\star}\#\mu \approx \nu$. This is typically achieved by attempting to minimize the following optimization problem: $\phi^\star = \arg\min_{\phi\in\Phi} \mathcal{W}_2(T_{\phi}\#\mu, \nu)$, where $\mathcal{W}_2$ denotes the Wasserstein distance that will be properly defined in Section 2. It has been shown that (Genevay et al., 2017) OT-based GANs (Arjovsky et al., 2017) and VAEs (Tolstikhin et al., 2017) both use this formulation with different parameterizations and different equivalent definitions of $\mathcal{W}_2$. However, their resulting algorithms still lack theoretical understanding.

In this study, we follow a completely different approach for IGM, where we aim at developing an algorithm with explicit theoretical guarantees for estimating a transport map between source $\mu$ and target $\nu$. The generated transport map will be *nonparametric* (in the sense that it does not belong to some family of functions, like a neural network), and it will be iteratively augmented: always increasing the quality of the fit along iterations. Formally, we take $T_t$ as the constructed transport map at time $t \in [0, \infty)$, and define $\mu_t = T_t\#\mu$ as the corresponding output distribution. Our objective is to build the maps so that $\mu_t$ will converge to the solution of a functional optimization problem, defined through a gradient flow in the Wasserstein space. Informally, we will consider a gradient flow that has the following form:

$$\partial_t \mu_t = -\nabla_{\mathcal{W}_2}\Big\{\mathrm{Cost}(\mu_t, \nu) + \mathrm{Reg}(\mu_t)\Big\}, \quad \mu_0 = \mu, \quad (1)$$

where the functional Cost computes a discrepancy between $\mu_t$ and $\nu$, Reg denotes a regularization functional, and $\nabla_{\mathcal{W}_2}$ denotes a notion of gradient with respect to a probability measure in the $\mathcal{W}_2$ metric for probability measures[1]. If this flow can be simulated, one would hope for $\mu_t = (T_t)_\#\mu$ to converge to the minimum of the functional optimization problem: $\min_\mu(\mathrm{Cost}(\mu, \nu) + \mathrm{Reg}(\mu))$ (Ambrosio et al., 2008; Santambrogio, 2017).

We construct a gradient flow where we choose the Cost functional as the *sliced Wasserstein distance* ($\mathcal{SW}_2$) (Rabin et al., 2012; Bonneel et al., 2015) and the Reg functional as the negative entropy. The $\mathcal{SW}_2$ distance is equivalent to the $\mathcal{W}_2$ distance (Bonnotte, 2013) and has important computational implications since it can be expressed as an average of (one-dimensional) projected optimal transportation costs whose analytical expressions are available.

We first show that, with the choice of $\mathcal{SW}_2$ and the negative-entropy functionals as the overall objective, we obtain a valid gradient flow that has a solution path $(\mu_t)_t$, and the probability density functions of this path solve a particular partial differential equation, which has close connections with stochastic differential equations. Even though gradient flows in Wasserstein spaces cannot be solved in general, by exploiting this connection, we are able to develop a practical algorithm that provides approximate solutions to the gradient flow and is algorithmically similar to stochastic gradient Markov Chain Monte Carlo (MCMC) methods[2] (Welling & Teh, 2011; Ma et al., 2015; Durmus et al., 2016; Şimşekli, 2017; Şimşekli et al., 2018). We provide finite-time error guarantees for the proposed algorithm and show explicit dependence of the error to the algorithm parameters.

To the best of our knowledge, the proposed algorithm is the first nonparametric IGM algorithm that has explicit theoretical guarantees. In addition to its nice theoretical properties, the proposed algorithm has also significant practical importance: it has low computational requirements and can be easily run on an everyday laptop CPU.Our experiments on both synthetic and real datasets support our theory and illustrate the advantages of the algorithm in several scenarios.

## 2. Technical Background

### 2.1. Wasserstein distance, optimal transport maps and Kantorovich potentials

For two probability measures $\mu, \nu \in \mathcal{P}_2(\Omega)$, $\mathcal{P}_2(\Omega) = \{\mu \in \mathcal{P}(\Omega) : \int_\Omega \|x\|^2 \mu(\mathrm{d}x) < +\infty\}$, the 2-Wasserstein distance is defined as follows:

$$\mathcal{W}_2(\mu, \nu) \triangleq \Big\{\inf_{\gamma\in\mathcal{C}(\mu,\nu)} \int_{\Omega\times\Omega} \|x - y\|^2 \gamma(dx, dy)\Big\}^{1/2}, \quad (2)$$

where $\mathcal{C}(\mu, \nu)$ is called the set of *transportation plans* and defined as the set of probability measures $\gamma$ on $\Omega\times\Omega$ satisfying for all $A \in \mathcal{A}$, $\gamma(A\times\Omega) = \mu(A)$ and $\gamma(\Omega\times A) = \nu(A)$, i.e. the marginals of $\gamma$ coincide with $\mu$ and $\nu$. From now on, we will assume that $\Omega$ is a compact subset of $\mathbb{R}^d$.

In the case where $\Omega$ is finite, computing the Wasserstein distance between two probability measures turns out to be a linear program with linear constraints, and has therefore a dual formulation. Since $\Omega$ is a Polish space (i.e. a complete and separable metric space), this dual formulation can be generalized as follows (Villani, 2008)[Theorem 5.10]:

$$\mathcal{W}_2(\mu, \nu) = \sup_{\psi\in \mathrm{L}^1(\mu)} \Big\{\int_\Omega \psi(x)\mu(dx) + \int_\Omega \psi^c(x)\nu(dx)\Big\}^{1/2} \quad (3)$$

where $\mathrm{L}^1(\mu)$ denotes the class of functions that are absolutely integrable under $\mu$ and $\psi^c$ denotes the c-conjugate of $\psi$ and is defined as follows: $\psi^c(y) \triangleq \{\inf_{x\in\Omega} \|x -$

[1] This gradient flow is similar to the usual Euclidean gradient flows, i.e. $\partial_t x_t = -\nabla(f(x_t) + r(x_t))$, where $f$ is typically the data-dependent cost function and $r$ is a regularization term. The (explicit) Euler discretization of this flow results in the well-known gradient descent algorithm for solving $\min_x(f(x) + r(x))$.

[2] We note that, despite the algorithmic similarities, the proposed algorithm is not a Bayesian posterior sampling algorithm.

$y\|^2 - \psi(x)\}$. The functions $\psi$ that realize the supremum in (3) are called the Kantorovich potentials between $\mu$ and $\nu$. Provided that $\mu$ satisfies a mild condition, we have the following uniqueness result.

**Theorem 1** ((Santambrogio, 2014)[Theorem 1.4]). *Assume that $\mu \in \mathcal{P}_2(\Omega)$ is absolutely continuous with respect to the Lebesgue measure. Then, there exists a unique optimal transport plan $\gamma^\star$ that realizes the infimum in (2) and it is of the form $(Id \times T)_{\#}\mu$, for a measurable function $T : \Omega \to \Omega$. Furthermore, there exists at least a Kantorovich potential $\psi$ whose gradient $\nabla\psi$ is uniquely determined $\mu$-almost everywhere. The function $T$ and the potential $\psi$ are linked by $T(x) = x - \nabla\psi(x)$.*

The measurable function $T : \Omega \to \Omega$ is referred to as the optimal transport map from $\mu$ to $\nu$. This result implies that there exists a solution for transporting samples from $\mu$ to samples from $\nu$ and this solution is optimal in the sense that it minimizes the $\ell_2$ displacement. However, identifying this solution is highly non-trivial. In the discrete case, effective solutions have been proposed (Cuturi, 2013). However, for continuous and high-dimensional probability measures, constructing an actual transport plan remains a challenge. Even if recent contributions (Genevay et al., 2016) have made it possible to rapidly compute $\mathcal{W}_2$, they do so without constructing the optimal map $T$, which is our objective here.

### 2.2. Wasserstein spaces and gradient flows

By (Ambrosio et al., 2008)[Proposition 7.1.5], $\mathcal{W}_2$ is a distance over $\mathcal{P}(\Omega)$. In addition, if $\Omega \subset \mathbb{R}^d$ is compact, the topology associated with $\mathcal{W}_2$ is equivalent to the weak convergence of probability measures and $(\mathcal{P}(\Omega), \mathcal{W}_2)$[3] is compact. The metric space $(\mathcal{P}_2(\Omega), \mathcal{W}_2)$ is called the *Wasserstein space*.

In this study, we are interested in functional optimization problems in $(\mathcal{P}_2(\Omega), \mathcal{W}_2)$, such as $\min_{\mu\in\mathcal{P}_2(\Omega)} \mathcal{F}(\mu)$, where $\mathcal{F}$ is the functional that we would like to minimize. Similar to Euclidean spaces, one way to formulate this optimization problem is to construct a gradient flow of the form $\partial_t\mu_t = -\nabla_{\mathcal{W}_2}\mathcal{F}(\mu_t)$ (Benamou & Brenier, 2000; Lavenant et al., 2018), where $\nabla_{\mathcal{W}_2}$ denotes a notion of gradient in $(\mathcal{P}_2(\Omega), \mathcal{W}_2)$. If such a flow can be constructed, one can utilize it both for practical algorithms and theoretical analysis.

Gradient flows $\partial_t\mu_t = \nabla_{\mathcal{W}_2}\mathcal{F}(\mu_t)$ with respect to a functional $\mathcal{F}$ in $(\mathcal{P}_2(\Omega), \mathcal{W}_2)$ have strong connections with partial differential equations (PDE) that are of the form of a *continuity equation* (Santambrogio, 2017). Indeed, it is shown than under appropriate conditions on $\mathcal{F}$ (see e.g.(Ambrosio et al., 2008)), $(\mu_t)_t$ is a solution of the gradient flow if and only if it admits a density $\rho_t$ with respect to the Lebesgue measure for all $t \geq 0$, and solves the continuity equation given by: $\partial_t\rho_t + \operatorname{div}(v\rho_t) = 0$, where $v$ denotes a vector field and div denotes the divergence operator. Then, for a given gradient flow in $(\mathcal{P}_2(\Omega), \mathcal{W}_2)$, we are interested in the evolution of the densities $\rho_t$, i.e. the PDEs which they solve. Such PDEs are of our particular interest since they have a key role for building practical algorithms.

[3]Note that in that case, $\mathcal{P}_2(\Omega) = \mathcal{P}(\Omega)$

### 2.3. Sliced-Wasserstein distance

In the one-dimensional case, i.e. $\mu, \nu \in \mathcal{P}_2(\mathbb{R})$, $\mathcal{W}_2$ has an analytical form, given as follows: $\mathcal{W}_2(\mu, \nu) = \int_0^1 |F_\mu^{-1}(\tau) - F_\nu^{-1}(\tau)|^2 \, d\tau$, where $F_\mu$ and $F_\nu$ denote the cumulative distribution functions (CDF) of $\mu$ and $\nu$, respectively, and $F_\mu^{-1}, F_\nu^{-1}$ denote the inverse CDFs, also called quantile functions (QF). In this case, the optimal transport map from $\mu$ to $\nu$ has a closed-form formula as well, given as follows: $T(x) = (F_\nu^{-1} \circ F_\mu)(x)$ (Villani, 2008). The optimal map $T$ is also known as the *increasing arrangement*, which maps each quantile of $\mu$ to the same quantile of $\nu$, e.g. minimum to minimum, median to median, maximum to maximum (Villani, 2008). Due to Theorem 1, the derivative of the corresponding Kantorovich potential is given as:

$$\psi'(x) \triangleq \partial_x\psi(x) = x - (F_\nu^{-1} \circ F_\mu)(x).$$

In the multidimensional case $d > 1$, building a transport map is much more difficult. The nice properties of the one-dimensional Wasserstein distance motivate the usage of *sliced-Wasserstein distance* ($\mathcal{SW}_2$) for practical applications. Before formally defining $\mathcal{SW}_2$, let us first define the orthogonal projection $\theta^*(x) \triangleq \langle\theta, x\rangle$ for any direction $\theta \in \mathbb{S}^{d-1}$ and $x \in \mathbb{R}^d$, where $\langle\cdot,\cdot\rangle$ denotes the Euclidean inner-product and $\mathbb{S}^{d-1} \subset \mathbb{R}^d$ denotes the $d$-dimensional unit sphere. Then, the $\mathcal{SW}_2$ distance is formally defined as follows:

$$\mathcal{SW}_2(\mu, \nu) \triangleq \int_{\mathbb{S}^{d-1}} \mathcal{W}_2(\theta^*_{\#}\mu, \theta^*_{\#}\nu) \, d\theta, \qquad (4)$$

where $d\theta$ represents the uniform probability measure on $\mathbb{S}^{d-1}$. As shown in (Bonnotte, 2013), $\mathcal{SW}_2$ is indeed a distance metric and induces the same topology as $\mathcal{W}_2$ for compact domains.

The $\mathcal{SW}_2$ distance has important practical implications: provided that the projected distributions $\theta^*_{\#}\mu$ and $\theta^*_{\#}\nu$ can be computed, then for any $\theta \in \mathbb{S}^{d-1}$, the distance $\mathcal{W}_2(\theta^*_{\#}\mu, \theta^*_{\#}\nu)$, as well as its optimal transport map and the corresponding Kantorovich potential can be analytically computed (since the projected measures are one-dimensional). Therefore, one can easily approximate (4) by using a simple Monte Carlo scheme that draws uniform random samples from $\mathbb{S}^{d-1}$ and replaces the integral in (4) with a finite-sample average. Thanks to its computational benefits, $\mathcal{SW}_2$ was very recently considered for OT-based VAEs and GANs (Deshpande et al., 2018; Wu et al., 2018;

Kolouri et al., 2018), appearing as a stable alternative to the adversarial methods.

## 3. Regularized Sliced-Wasserstein Flows for Generative Modeling

### 3.1. Construction of the gradient flow

In this paper, we propose the following functional minimization problem on $\mathcal{P}_2(\Omega)$ for implicit generative modeling:

$$\min_{\mu}\Big\{\mathcal{F}_\lambda^\nu(\mu) \triangleq \frac{1}{2}\mathcal{SW}_2^2(\mu,\nu) + \lambda\mathcal{H}(\mu)\Big\}, \tag{5}$$

where $\lambda > 0$ is a regularization parameter and $\mathcal{H}$ denotes the negative entropy defined by $\mathcal{H}(\mu) \triangleq \int_\Omega \rho(x)\log\rho(x)dx$ if $\mu$ has density $\rho$ with respect to the Lebesgue measure and $\mathcal{H}(\mu) = +\infty$ otherwise. Note that the case $\lambda = 0$ has been already proposed and studied in (Bonnotte, 2013) in a more general OT context. Here, in order to introduce the necessary noise inherent to generative model, we suggest to penalize the slice-Wasserstein distance using $\mathcal{H}$. In other words, the main idea is to find a measure $\mu^\star$ that is close to $\nu$ as much as possible and also has a certain amount of entropy to make sure that it is sufficiently expressive for generative modeling purposes. The importance of the entropy regularization becomes prominent in practical applications where we have finitely many data samples that are assumed to be drawn from $\nu$. In such a circumstance, the regularization would prevent $\mu^\star$ to collapse on the data points and therefore avoid 'over-fitting' to the data distribution. Note that this regularization is fundamentally different from the one used in Sinkhorn distances (Genevay et al., 2018).

In our first result, we show that there exists a flow $(\mu_t)_{t\geq 0}$ in $(\mathcal{P}(\overline{\mathrm{B}}(0,r)), \mathcal{W}_2)$ which decreases along $\mathcal{F}_\lambda^\nu$, where $\overline{\mathrm{B}}(0,a)$ denotes the closed unit ball centered at 0 and radius $a$. This flow will be referred to as a generalized minimizing movement scheme (see Definition 1 in the supplementary document). In addition, the flow $(\mu_t)_{t\geq 0}$ admits a density $\rho_t$ with respect to the Lebesgue measure for all $t > 0$ and $(\rho_t)_{t\geq 0}$ is solution of a non-linear PDE (in the weak sense).

**Theorem 2.** *Let $\nu$ be a probability measure on $\overline{\mathrm{B}}(0,1)$ with a strictly positive smooth density. Choose a regularization constant $\lambda > 0$ and radius $r > \sqrt{d}$, where $d$ is the data dimension. Assume that $\mu_0 \in \mathcal{P}(\overline{\mathrm{B}}(0,r))$ is absolutely continuous with respect to the Lebesgue measure with density $\rho_0 \in \mathrm{L}^\infty(\overline{\mathrm{B}}(0,r))$. There exists a generalized minimizing movement scheme $(\mu_t)_{t\geq 0}$ associated to (5) and if $\rho_t$ stands for the density of $\mu_t$ for all $t \geq 0$, then $(\rho_t)_t$ satisfies the following continuity equation:*

$$\frac{\partial\rho_t}{\partial t} = -\operatorname{div}(v_t\rho_t) + \lambda\Delta\rho_t, \tag{6}$$

$$v_t(x) \triangleq v(x,\mu_t) = -\int_{\mathbb{S}^{d-1}} \psi'_{t,\theta}(\langle x,\theta\rangle)\theta d\theta \tag{7}$$

*in a weak sense. Here, $\Delta$ denotes the Laplacian operator, $\operatorname{div}$ the divergence operator, and $\psi_{t,\theta}$ denotes the Kantorovich potential between $\theta^*_\#\mu_t$ and $\theta^*_\#\nu$.*

The precise statement of this Theorem, related results and its proof are postponed to the supplementary document. For its proof, we use the technique introduced in (Jordan et al., 1998): we first prove the existence of a generalized minimizing movement scheme by showing that the solution curve $(\mu_t)_t$ is a limit of the solution of a time-discretized problem. Then we prove that the curve $(\rho_t)_t$ solves the PDE given in (6).

### 3.2. Connection with stochastic differential equations

As a consequence of the entropy regularization, we obtain the Laplacian operator $\Delta$ in the PDE given in (6). We therefore observe that the overall PDE is a Fokker-Planck-type equation (Bogachev et al., 2015) that has a well-known probabilistic counterpart, which can be expressed as a stochastic differential equation (SDE). More precisely, let us consider a stochastic process $(X_t)_t$, that is the solution of the following SDE starting at $X_0 \sim \mu_0$:

$$dX_t = v(X_t,\mu_t)dt + \sqrt{2\lambda}dW_t, \tag{8}$$

where $(W_t)_t$ denotes a standard Brownian motion. Then, the probability distribution of $X_t$ at time $t$ solves the PDE given in (6) (Bogachev et al., 2015). This informally means that, if we could simulate (8), then the distribution of $X_t$ would converge to the solution of (5), therefore, we could use the sample paths $(X_t)_t$ as samples drawn from $(\mu_t)_t$. However, in practice this is not possible due to two reasons: (i) the drift $v_t$ cannot be computed analytically since it depends on the probability distribution of $X_t$, (ii) the SDE (8) is a continuous-time process, it needs to be discretized.

We now focus on the first issue. We observe that the SDE (8) is similar to McKean-Vlasov SDEs (Veretennikov, 2006; Mishura & Veretennikov, 2016), a family of SDEs whose drift depends on the distribution of $X_t$. By using this connection, we can borrow tools from the relevant SDE literature (Malrieu, 2003; Cattiaux et al., 2008) for developing an approximate simulation method for (8).

Our approach is based on defining a *particle system* that serves as an approximation to the original SDE (8). The particle system can be written as a collection of SDEs, given as follows (Bossy & Talay, 1997):

$$dX_t^i = v(X_t^i,\mu_t^N)dt + \sqrt{2\lambda}dW_t^i\,, \quad i = 1,\dots,N, \tag{9}$$

where $i$ denotes the particle index, $N \in \mathbb{N}_+$ denotes the total number of particles, and $\mu_t^N = (1/N)\sum_{j=1}^N \delta_{X_t^j}$ denotes the empirical distribution of the particles $\{X_t^j\}_{j=1}^N$. This particle system is particularly interesting, since (i) one

typically has $\lim_{N\to\infty} \mu_t^N = \mu_t$ with a rate of convergence of order $\mathcal{O}(1/\sqrt{N})$ for all $t$ (Malrieu, 2003; Cattiaux et al., 2008), and (ii) each of the particle systems in (9) can be simulated by using an Euler-Maruyama discretization scheme. We note that the existing theoretical results in (Veretennikov, 2006; Mishura & Veretennikov, 2016) do not directly apply to our case due to the non-standard form of our drift. However, we conjecture that a similar result holds for our problem as well. Such a result would be proven by using the techniques given in (Zhang et al., 2018); however, it is out of the scope of this study.

### 3.3. Approximate Euler-Maruyama discretization

In order to be able to simulate the particle SDEs (9) in practice, we propose an approximate Euler-Maruyama discretization for each particle SDE. The algorithm iteratively applies the following update equation: ($\forall i \in \{1,\dots,N\}$)

$$\bar{X}_0^i \overset{\text{i.i.d.}}{\sim} \mu_0, \quad \bar{X}_{k+1}^i = \bar{X}_k^i + h\hat{v}_k(\bar{X}_k^i) + \sqrt{2\lambda h} Z_{k+1}^i, \quad (10)$$

where $k \in \mathbb{N}_+$ denotes the iteration number, $Z_k^i$ is a standard Gaussian random vector in $\mathbb{R}^d$, $h$ denotes the step-size, and $\hat{v}_k$ is a short-hand notation for a computationally tractable estimator of the original drift $v(\cdot, \bar{\mu}_{kh}^N)$, with $\bar{\mu}_{kh}^N = (1/N)\sum_{j=1}^N \delta_{\bar{X}_k^j}$ being the empirical distribution of $\{\bar{X}_k^j\}_{j=1}^N$. A question of fundamental practical importance is how to compute this function $\hat{v}$.

We propose to approximate the integral in (7) via a simple Monte Carlo estimate. This is done by first drawing $N_\theta$ uniform i.i.d. samples from the sphere $\mathbb{S}^{d-1}$, $\{\theta_n\}_{n=1}^{N_\theta}$. Then, at each iteration $k$, we compute:

$$\hat{v}_k(x) \triangleq -(1/N_\theta)\sum_{n=1}^{N_\theta} \psi'_{k,\theta_n}(\langle\theta_n, x\rangle)\theta_n, \quad (11)$$

where for any $\theta$, $\psi'_{k,\theta}$ is the derivative of the Kantorovich potential (cf. Section 2) that is applied to the OT problem from $\theta^*_\# \bar{\mu}_{kh}^N$ to $\theta^*_\# \nu$: i.e.

$$\psi'_{k,\theta}(z) = \left[z - (F^{-1}_{\theta^*_\#\nu} \circ F_{\theta^*_\# \bar{\mu}_{kh}^N})(z)\right]. \quad (12)$$

For any particular $\theta \in \mathbb{S}^{d-1}$, the QF, $F^{-1}_{\theta^*_\#\nu}$ for the projection of the target distribution $\nu$ on $\theta$ can be easily computed from the data. This is done by first computing the projections $\langle\theta, y_i\rangle$ for all data points $y_i$, and then computing the empirical quantile function for this set of $P$ scalars. Similarly, $F_{\theta^*_\# \bar{\mu}_{kh}^N}$, the CDF of the particles at iteration $k$, is easy to compute: we first project all particles $\bar{X}_k^i$ to get $\langle\theta, \bar{X}_k^i\rangle$, and then compute the empirical CDF of this set of $N$ scalar values.

In both cases, the true CDF and quantile functions are approximated as a linear interpolation between a set of the computed $Q \in \mathbb{N}_+$ empirical quantiles. Another source of approximation here comes from the fact that the target $\nu$ will in practice be a collection of Dirac measures on the observations $y_i$. Since it is currently common to have a very large dataset, we believe this approximation to be accurate in practice for the target. Finally, yet another source of approximation comes from the error induced by using a finite number of $\theta_n$ instead of a sum over $\mathbb{S}^{d-1}$ in (12).

Even though the error induced by these approximation schemes can be incorporated into our current analysis framework, we choose to neglect it for now, because (i) all of these one-dimensional computations can be done very accurately and (ii) the quantization of the empirical CDF and QF can be modeled as additive Gaussian noise that enters our discretization scheme (10) (Van der Vaart, 1998). Therefore, we will assume that $\hat{v}_k$ is an *unbiased* estimator of $v$, i.e. $\mathbb{E}[\hat{v}(x,\mu)] = v(x,\mu)$, for any $x$ and $\mu$, where the expectation is taken over $\theta_n$.

The overall algorithm is illustrated in Algorithm 1. It is remarkable that the updates of the particles only involves the learning data $\{y_i\}$ through the CDFs of its projections on the many $\theta_n \in \mathbb{S}^{d-1}$. This has a fundamental consequence of high practical interest: these CDF may be computed beforehand in a massively distributed manner that is independent of the sliced Wasserstein flow. This aspect is reminiscent of the *compressive learning* methodology (Gribonval et al., 2017), except we exploit quantiles of random projections here, instead of random generalized moments as done there.

Besides, we can obtain further reductions in the computing time if the CDF, $F_{\theta^*_\#\nu}$ for the target is computed on random

**Algorithm 1:** Sliced-Wasserstein Flow (SWF)

**input** : $\mathcal{D} \equiv \{y_i\}_{i=1}^P$, $\mu_0$, $N$, $N_\theta$, $h$, $\lambda$
**output** : $\{\bar{X}_K^i\}_{i=1}^N$

// Initialize the particles
$\bar{X}_0^i \overset{\text{i.i.d.}}{\sim} \mu_0, \qquad i = 1,\dots,N$
// Generate random directions
$\theta_n \sim \text{Uniform}(\mathbb{S}^{d-1}), \qquad n = 1,\dots,N_\theta$
// Quantiles of projected target
**for** $\theta \in \{\theta_n\}_{n=1}^{N_\theta}$ **do**
  $F^{-1}_{\theta^*_\#\nu} = \text{QF}\{\langle\theta, y_i\rangle\}_{i=1}^P$
// Iterations
**for** $k = 0,\dots K-1$ **do**
  **for** $\theta \in \{\theta_n\}_{n=1}^{N_\theta}$ **do**
    // CDF of projected particles
    $F_{\theta^*_\# \bar{\mu}_{kh}^N} = \text{CDF}\{\langle\theta, \bar{X}_k^i\rangle\}_{i=1}^N$
  // Update the particles
  $\bar{X}_{k+1}^i = \bar{X}_k^i - h\hat{v}_k(\bar{X}_k^i) + \sqrt{2\lambda h} Z_{k+1}^i$
  $i = 1,\dots,N$

mini-batches of the data, instead of the whole dataset of size $P$. This simplified procedure might also have some interesting consequences in privacy-preserving settings: since we can vary the number of projection directions $N_\theta$ for each data point $y_i$, we may guarantee that $y_i$ cannot be recovered via these projections, by picking fewer than necessary for reconstruction using, e.g. compressed sensing (Donoho & Tanner, 2009).

### 3.4. Finite-time analysis for the infinite particle regime

In this section we will analyze the behavior of the proposed algorithm in the asymptotic regime where the number of particles $N \to \infty$. Within this regime, we will assume that the original SDE (8) can be directly simulated by using an approximate Euler-Maruyama scheme, defined starting at $\bar{X}_0 \overset{\text{i.i.d.}}{\sim} \mu_0$ as follows:

$$\bar{X}_{k+1} = \bar{X}_k + h\hat{v}(\bar{X}_k^i, \bar{\mu}_{kh}) + \sqrt{2\lambda h} Z_{k+1}, \tag{13}$$

where $\bar{\mu}_{kh}$ denotes the law of $\bar{X}_k$ with step size $h$ and $\{Z_k\}_k$ denotes a collection of standard Gaussian random variables. Apart from its theoretical significance, this scheme is also practically relevant, since one would expect that it captures the behavior of the particle method (10) with large number of particles.

In practice, we would like to approximate the measure sequence $(\mu_t)_t$ as accurate as possible, where $\mu_t$ denotes the law of $X_t$. Therefore, we are interested in analyzing the distance $\|\bar{\mu}_{Kh} - \mu_T\|_{\text{TV}}$, where $K$ denotes the total number of iterations, $T = Kh$ is called the horizon, and $\|\mu - \nu\|_{\text{TV}}$ denotes the total variation distance between two probability measures $\mu$ and $\nu$: $\|\mu - \nu\|_{\text{TV}} \triangleq \sup_{A \in \mathcal{B}(\Omega)} |\mu(A) - \nu(A)|$.

In order to analyze this distance, we exploit the algorithmic similarities between (13) and the stochastic gradient Langevin dynamics (SGLD) algorithm (Welling & Teh, 2011), which is a Bayesian posterior sampling method having a completely different goal, and is obtained as a discretization of an SDE whose drift has a much simpler form. We then bound the distance by extending the recent results on SGLD (Raginsky et al., 2017) to time- and measure-dependent drifts, that are of our interest in the paper.

We now present our second main theoretical result. We present all our assumptions and the explicit forms of the constants in the supplementary document.

**Theorem 3.** *Assume that the conditions given in the supplementary document hold. Then, the following bound holds for $T = Kh$:*

$$\|\bar{\mu}_{Kh} - \mu_T\|_{\text{TV}}^2 \leq \delta_\lambda \left\{ \frac{L^2 K}{2\lambda} \left( \frac{C_1 h^3}{3} + 3\lambda d h^2 \right) + \frac{C_2 \delta K h}{4\lambda} \right\}, \tag{14}$$

*for some $C_1, C_2, L > 0$, $\delta \in (0, 1)$, and $\delta_\lambda > 1$.*

Here, the constants $C_1$, $C_2$, $L$ are related to the regularity and smoothness of the functions $v$ and $\hat{v}$; $\delta$ is directly proportional to the variance of $\hat{v}$, and $\delta_\lambda$ is inversely proportional to $\lambda$. The theorem shows that if we choose $h$ small enough, we can have a non-asymptotic error guarantee, which is formally shown in the following corollary.

**Corollary 1.** *Assume that the conditions of Theorem 3 hold. Then for all $\varepsilon > 0$, $K \in \mathbb{N}_+$, setting*

$$h = (3/C_1) \wedge \left( \frac{2\varepsilon^2 \lambda}{\delta_\lambda L^2 T} (1 + 3\lambda d)^{-1} \right)^{1/2}, \tag{15}$$

*we have*

$$\|\bar{\mu}_{Kh} - \mu_T\|_{\text{TV}} \leq \varepsilon + \left( \frac{C_2 \delta_\lambda \delta T}{4\lambda} \right)^{1/2} \tag{16}$$

*for $T = Kh$.*

This corollary shows that for a large horizon $T$, the approximate drift $\hat{v}$ should have a small variance in order to obtain accurate estimations. This result is similar to (Raginsky et al., 2017) and (Nguyen et al., 2019): for small $\varepsilon$ the variance of the approximate drift should be small as well. On the other hand, we observe that the error decreases as $\lambda$ increases. This behavior is expected since for large $\lambda$, the Brownian term in (8) dominates the drift, which makes the simulation easier.

We note that these results establish the explicit dependency of the error with respect to the algorithm parameters (e.g. step-size, gradient noise) for a fixed number of iterations, rather than explaining the asymptotic behavior of the algorithm when $K$ goes to infinity.

## 4. Experiments

In this section, we evaluate the SWF algorithm on a synthetic and a real data setting. Our primary goal is to validate our theory and illustrate the behavior of our non-standard approach, rather than to obtain the state-of-the-art results in IGM. In all our experiments, the initial distribution $\mu_0$ is selected as the standard Gaussian distribution on $\mathbb{R}^d$, we take $Q = 100$ quantiles and $N = 5000$ particles, which proved sufficient to approximate the quantile functions accurately.

### 4.1. Gaussian Mixture Model

We perform the first set of experiments on synthetic data where we consider a standard Gaussian mixture model (GMM) with 10 components and random parameters. Centroids are taken as sufficiently distant from each other to make the problem more challenging. We generate $P = 50000$ data samples in each experiment.

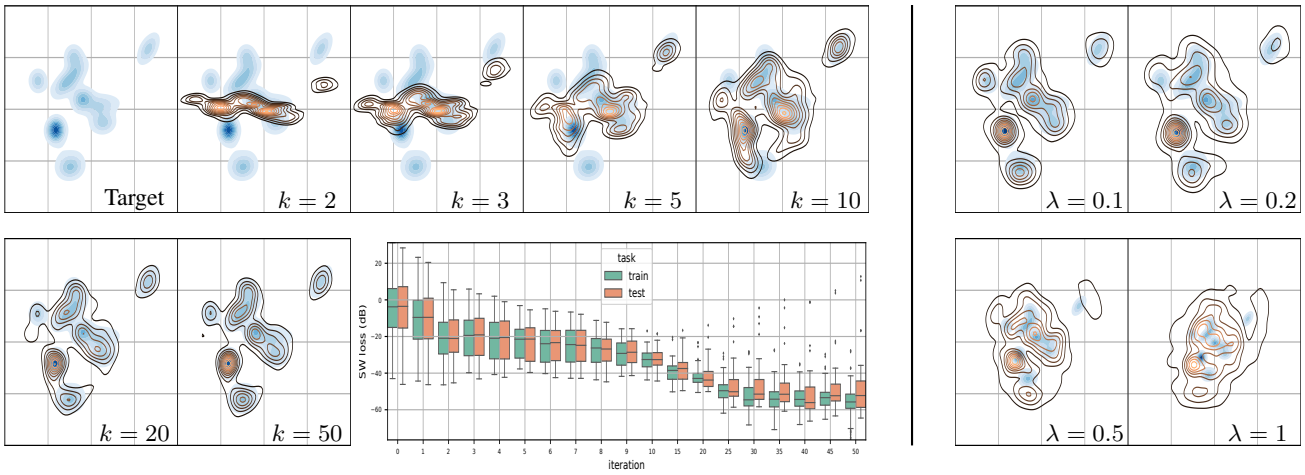


*Figure 1.* SWF on toy 2D data. **Left:** Target distribution (shaded contour plot) and distribution of particles (lines) during SWF. (bottom) SW cost over iterations during training (left) and test (right) stages. **Right:** Influence of the regularization parameter $\lambda$.

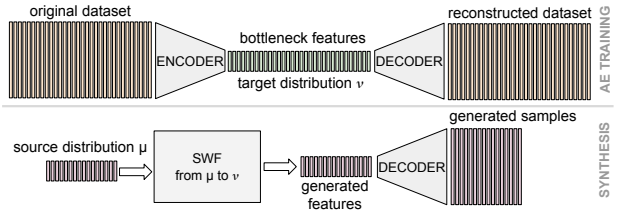


*Figure 2.* First, we learn an autoencoder (AE). Then, we use SWF to transport random vectors to the distribution of the bottleneck features of the training set. The trained decoder is used for visualization.

In our first experiment, we set $d = 2$ for visualization purposes and illustrate the general behavior of the algorithm. Figure 1 shows the evolution of the particles through the iterations. Here, we set $N_\theta = 30$, $h = 1$ and $\lambda = 10^{-4}$. We first observe that the SW cost between the empirical distributions of training data and particles is steadily decreasing along the SW flow. Furthermore, we see that the QFs, $F^{-1}_{\theta^*_\# \bar{\mu}^N_{kh}}$ that are computed with the initial set of particles (the *training* stage) can be perfectly re-used for new unseen particles in a subsequent *test* stage, yielding similar — yet slightly higher — SW cost.

In our second experiment on Figure 1, we investigate the effect of the level of the regularization $\lambda$. The distribution of the particles becomes more spread with increasing $\lambda$. This is due to the increment of the entropy, as expected.

### 4.2. Experiments on real data

In the second set of experiments, we test the SWF algorithm on two real datasets. (i) The traditional MNIST dataset that contains 70K binary images corresponding to different digits. (ii) The popular CelebA dataset (Liu et al., 2015), that contains 202K color-scale images. This dataset is advocated as more challenging than MNIST. Images were interpolated as $32 \times 32$ for MNIST, and $64 \times 64$ for CelebA.

*Figure 3.* Samples generated after 200 iterations of SWF to match the distribution of bottleneck features for the training dataset. Visualization is done with the pre-trained decoder.

In experiments reported in the supplementary document, we found out that directly applying SWF to such high-dimensional data yielded noisy results, possibly due to the insufficient sampling of $\mathbb{S}^{d-1}$. To reduce the dimensionality, we trained a standard convolutional autoencoder (AE) on the training set of both datasets (see Figure 2 and the supplementary document), and the target distribution $\nu$ considered becomes the distribution of the resulting bottleneck features, with dimension $d$. Particles can be visualized with the pre-trained decoder. Our goal is to show that SWF permits to directly sample from the distribution of bottleneck features, as an alternative to enforcing this distribution to

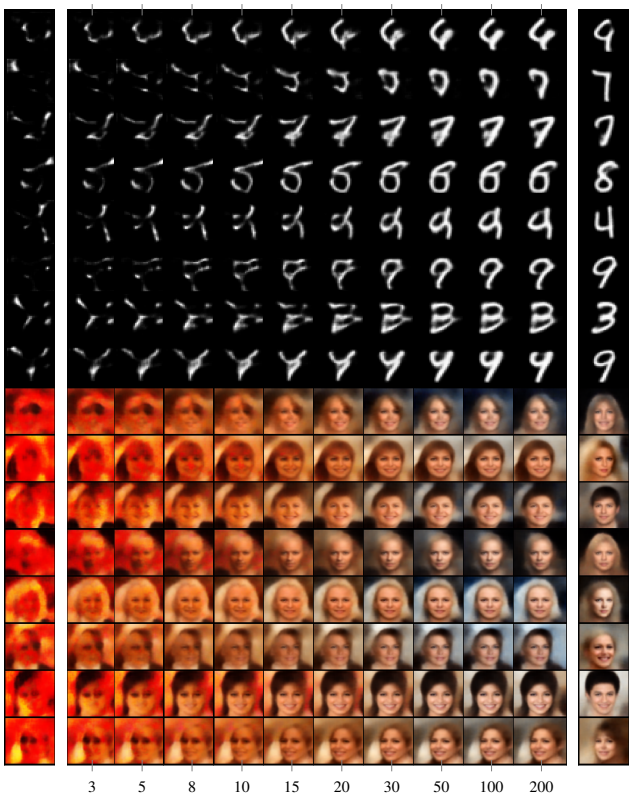


*Figure 4.* Initial random particles (left), particles through iterations (middle, from 1 to 200 iterations) and closest sample from the training dataset (right), for both MNIST and CelebA.

match some prior, as in VAE. In the following, we set $\lambda = 0$, $N_\theta = 40000$, $d = 32$ for MNIST and $d = 64$ for CelebA.

Assessing the validity of IGM algorithms is generally done by visualizing the generated samples. Figure 3 shows some particles after $500$ iterations of SWF. We can observe they are considerably accurate. Interestingly, the generated samples gradually take the form of either digits or faces along the iterations, as seen on Figure 4. In this figure, we also display the closest sample from the original database to check we are not just reproducing training data.

For a visual comparison, we provide the results presented in (Deshpande et al., 2018) in Figure 5. These results are obtained by running different IGM approaches on the MNIST dataset, namely GAN (Goodfellow et al., 2014), Wasserstein GAN (W-GAN) (Arjovsky et al., 2017) and the Sliced-Wasserstein Generator (SWG) (Deshpande et al., 2018). The visual comparison suggests that the samples generated by SWF are of slightly better quality than those, although research must still be undertaken to scale up to high dimensions without an AE.

*Figure 5.* Performance of GAN (left), W-GAN (middle), SWG (right) on MNIST. (The figure is directly taken from (Deshpande et al., 2018).)

*Figure 6.* Applying a pre-trained SWF on new samples located in-between the ones used for training. Visualization is done with the pre-trained decoder.

We also provide the outcome of the pre-trained SWF with samples that are regularly spaced in between those used for training. The result is shown in Figure 4.2. This plot suggests that SWF is a way to interpolate non-parametrically in between latent spaces of regular AE.

## 5. Conclusion and Future Directions

In this study, we proposed SWF, an efficient, nonparametric IGM algorithm. SWF is based on formulating IGM as a functional optimization problem in Wasserstein spaces, where the aim is to find a probability measure that is close to the data distribution as much as possible while maintaining the expressiveness at a certain level. SWF lies in the intersection of OT, gradient flows, and SDEs, which allowed us to convert the IGM problem to an SDE simulation problem. We provided finite-time bounds for the infinite-particle regime and established explicit links between the algorithm parameters and the overall error. We conducted several experiments, where we showed that the results support our theory: SWF is able to generate samples from non-trivial distributions with low computational requirements.

The SWF algorithm opens up interesting future directions: (i) extension to differentially private settings (Dwork & Roth, 2014) by exploiting the fact that it only requires random projections of the data, (ii) showing the convergence scheme of the particle system (9) to the original SDE (8), (iii) providing bounds directly for the particle scheme (10).

## Acknowledgments

This work is partly supported by the French National Research Agency (ANR) as a part of the FBIMATRIX (ANR-16-CE23-0014) and KAMoulox (ANR-15-CE38-0003-01) projects. Szymon Majewski is partially supported by Polish National Science Center grant number 2016/23/B/ST1/00454.

## References

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

Arjovsky, M., Chintala, S., and Bottou, L. Wasserstein generative adversarial networks. In *International Conference on Machine Learning*, pp. 214–223, 2017.

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

Bogachev, V. I., Krylov, N. V., Röckner, M., and Shaposhnikov, S. V. *Fokker-Planck-Kolmogorov Equations*, volume 207. American Mathematical Soc., 2015.

Bonneel, N., Rabin, J., Peyré, G., and Pfister, H. Sliced and Radon Wasserstein barycenters of measures. *Journal of Mathematical Imaging and Vision*, 51(1):22–45, 2015.

Bonnotte, N. *Unidimensional and evolution methods for optimal transportation*. PhD thesis, Paris 11, 2013.

Bossy, M. and Talay, D. A stochastic particle method for the McKean-Vlasov and the Burgers equation. *Mathematics of Computation of the American Mathematical Society*, 66(217):157–192, 1997.

Bousquet, O., Gelly, S., Tolstikhin, I., Simon-Gabriel, C.-J., and Schoelkopf, B. From optimal transport to generative modeling: the vegan cookbook. *arXiv preprint arXiv:1705.07642*, 2017.

Cattiaux, P., Guillin, A., and Malrieu, F. Probabilistic approach for granular media equations in the non uniformly convex case. *Prob. Theor. Rel. Fields*, 140(1-2):19–40, 2008.

Şimşekli, U., Yildiz, C., Nguyen, T. H., Cemgil, A. T., and Richard, G. Asynchronous stochastic quasi-Newton MCMC for non-convex optimization. In *ICML*, pp. 4674–4683, 2018.

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

Deshpande, I., Zhang, Z., and Schwing, A. Generative modeling using the sliced wasserstein distance. *arXiv preprint arXiv:1803.11188*, 2018.

Diggle, P. J. and Gratton, R. J. Monte carlo methods of inference for implicit statistical models. *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 193–227, 1984.

Donoho, D. and Tanner, J. Observed universality of phase transitions in high-dimensional geometry, with implications for modern data analysis and signal processing. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 367(1906):4273–4293, 2009.

Durmus, A., Şimşekli, U., Moulines, E., Badeau, R., and Richard, G. Stochastic gradient Richardson-Romberg Markov Chain Monte Carlo. In *NIPS*, 2016.

Dwork, C. and Roth, A. The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science*, 9(3–4):211–407, 2014.

Genevay, A., Cuturi, M., Peyré, G., and Bach, F. Stochastic optimization for large-scale optimal transport. In *Advances in Neural Information Processing Systems*, pp. 3440–3448, 2016.

Genevay, A., Peyré, G., and Cuturi, M. Gan and vae from an optimal transport point of view. *arXiv preprint arXiv:1706.01807*, 2017.

Genevay, A., Peyré, G., and Cuturi, M. Learning generative models with Sinkhorn divergences. In *International Conference on Artificial Intelligence and Statistics*, pp. 1608–1617, 2018.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial nets. In *Advances in neural information processing systems*, pp. 2672–2680, 2014.

Gribonval, R., Blanchard, G., Keriven, N., and Traonmilin, Y. Compressive statistical learning with random feature moments. *arXiv preprint arXiv:1706.07180*, 2017.

Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., and Courville, A. C. Improved training of Wasserstein GANs. In *Advances in Neural Information Processing Systems*, pp. 5769–5779, 2017.

Guo, X., Hong, J., Lin, T., and Yang, N. Relaxed Wasserstein with applications to GANs. *arXiv preprint arXiv:1705.07164*, 2017.

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

Kingma, D. P. and Welling, M. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Kolouri, S., Martin, C. E., and Rohde, G. K. Sliced-wasserstein autoencoder: An embarrassingly simple generative model. *arXiv preprint arXiv:1804.01947*, 2018.

Lavenant, H., Claici, S., Chien, E., and Solomon, J. Dynamical optimal transport on discrete surfaces. In *SIGGRAPH Asia 2018 Technical Papers*, pp. 250. ACM, 2018.

Lei, N., Su, K., Cui, L., Yau, S.-T., and Gu, D. X. A geometric view of optimal transportation and generative model. *arXiv preprint arXiv:1710.05488*, 2017.

Liu, S., Bousquet, O., and Chaudhuri, K. Approximation and convergence properties of generative adversarial learning. In *Advances in Neural Information Processing Systems*, pp. 5551–5559, 2017.

Liu, Z., Luo, P., Wang, X., and Tang, X. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, 2015.

Ma, Y. A., Chen, T., and Fox, E. A complete recipe for stochastic gradient MCMC. In *Advances in Neural Information Processing Systems*, pp. 2899–2907, 2015.

Malrieu, F. Convergence to equilibrium for granular media equations and their Euler schemes. *Ann. Appl. Probab.*, 13(2):540–560, 2003.

Mishura, Y. S. and Veretennikov, A. Y. Existence and uniqueness theorems for solutions of McKean–Vlasov stochastic equations. *arXiv preprint arXiv:1603.02212*, 2016.

Mohamed, S. and Lakshminarayanan, B. Learning in implicit generative models. *arXiv preprint arXiv:1610.03483*, 2016.

Nguyen, T. H., Şimşekli, U., , and Richard, G. Non-asymptotic analysis of fractional Langevin Monte Carlo for non-convex optimization. In *ICML*, 2019.

Rabin, J., Peyré, G., Delon, J., and Bernot, M. Wasserstein barycenter and its application to texture mixing. In Bruckstein, A. M., ter Haar Romeny, B. M., Bronstein, A. M., and Bronstein, M. M. (eds.), *Scale Space and Variational Methods in Computer Vision*, pp. 435–446, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg. ISBN 978-3-642-24785-9.

Raginsky, M., Rakhlin, A., and Telgarsky, M. Non-convex learning via stochastic gradient Langevin dynamics: a nonasymptotic analysis. In *Proceedings of the 2017 Conference on Learning Theory*, volume 65, pp. 1674–1703, 2017.

Santambrogio, F. Introduction to optimal transport theory. In Pajot, H., Ollivier, Y., and Villani, C. (eds.), *Optimal Transportation: Theory and Applications*, chapter 1. Cambridge University Press, 2014.

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

Şimşekli, U. Fractional Langevin Monte Carlo: Exploring Lévy Driven Stochastic Differential Equations for Markov Chain Monte Carlo. In *International Conference on Machine Learning*, 2017.

Tolstikhin, I., Bousquet, O., Gelly, S., and Schoelkopf, B. Wasserstein auto-encoders. *arXiv preprint arXiv:1711.01558*, 2017.

Van der Vaart, A. W. *Asymptotic statistics*, volume 3. Cambridge university press, 1998.

Veretennikov, A. Y. On ergodic measures for McKean-Vlasov stochastic equations. In *Monte Carlo and Quasi-Monte Carlo Methods 2004*, pp. 471–486. Springer, 2006.

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

Welling, M. and Teh, Y. W. Bayesian learning via stochastic gradient Langevin dynamics. In *International Conference on Machine Learning*, pp. 681–688, 2011.

Wu, J., Huang, Z., Li, W., and Gool, L. V. Sliced wasserstein generative models. *arXiv preprint arXiv:1706.02631*, abs/1706.02631, 2018.

Zhang, J., Zhang, R., and Chen, C. Stochastic particle-optimization sampling and the non-asymptotic convergence theory. *arXiv preprint arXiv:1809.01293*, 2018.