Differentiable Generalized Sliced Wasserstein Plans

Laetitia Chapel*
IRISA - L'Institut Agro Rennes-Angers
Rennes, France.
laetitia.chapel@irisa.fr

Romain Tavenard*
Université de Rennes 2 – IRISA
Rennes, France.
romain.tavenard@univ-rennes2.fr

Samuel Vaiter
CNRS & Université Côte d'Azur
Laboratoire J. A. Dieudonné
Nice, France.
samuel.vaiter@cnrs.fr

Abstract

Optimal Transport (OT) has attracted significant interest in the machine learning community, not only for its ability to define meaningful distances between probability distributions – such as the Wasserstein distance – but also for its formulation of OT plans. Its computational complexity remains a bottleneck, though, and slicing techniques have been developed to scale OT to large datasets. Recently, a novel slicing scheme, dubbed min-SWGG, lifts a single one-dimensional plan back to the original multidimensional space, finally selecting the slice that yields the lowest Wasserstein distance as an approximation of the full OT plan. Despite its computational and theoretical advantages, min-SWGG inherits typical limitations of slicing methods: (i) the number of required slices grows exponentially with the data dimension, and (ii) it is constrained to linear projections. Here, we reformulate min-SWGG as a bilevel optimization problem and propose a differentiable approximation scheme to efficiently identify the optimal slice, even in high-dimensional settings. We furthermore define its generalized extension for accommodating data living on manifolds. Finally, we demonstrate the practical value of our approach in various applications, including gradient flows on manifolds and high-dimensional spaces, as well as a novel sliced OT-based conditional flow matching for image generation – where fast computation of transport plans is essential.

1 Introduction

Optimal Transport (OT) has emerged as a foundational tool in modern machine learning, primarily due to its capacity to provide meaningful comparisons between probability distributions. Rooted in the seminal works of Monge [39] and Kantorovich [28], OT introduces a mathematically rigorous framework that defines distances, such as the Wasserstein distance, that have demonstrated high performance in various learning tasks. Used as a loss function, it is now the workhorse of learning problems ranging from classification, transfer learning or generative modelling, see [40] for a review. One of the key advantages of OT lies in its dual nature: it also constructs an optimal coupling or transport plan between distributions. This coupling reveals explicit correspondences between samples, enabling a wide range of applications. For example, OT has proven valuable in shape matching [11], colour transfer [48, 50], domain adaptation [15], and, more recently, in generative modeling through conditional flow matching [46, 56], where it provides an alignment between the data distribution and samples from a prior.

*equal contribution.

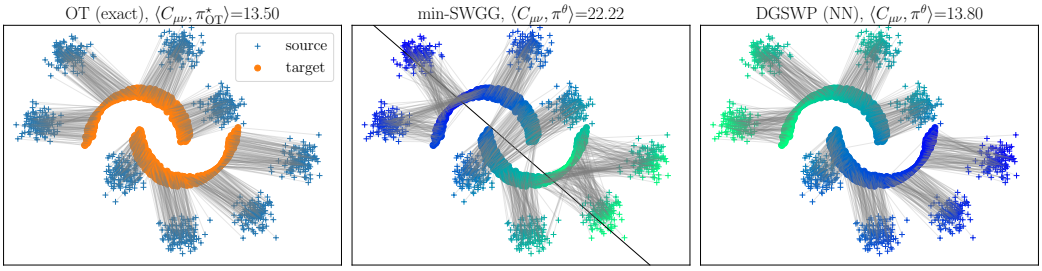


Figure 1: 8Gaussians (source) to Two Moons (target) distributions and associated OT plans in Grey: (Left) exact solution (Middle) min-SWGG, that projects samples on an optimal line determined by random sampling (Right) Differentiable Generalized SW plan, that relies on a neural network to get non linear-based ordering of the samples. Gradient of colours represent the ordering of the samples.

Despite the many successes of optimal transport in machine learning, computing OT plans remains a computationally challenging problem. The most used exact algorithms are drawn from linear programming, typically resulting in a $O(n^3)$ complexity with respect to the number of samples n. To alleviate this issue, several strategies have been developed in the last decade, that can be roughly classified in three families: *i)* regularization-based methods, such as entropic optimal transport [17], *ii)* minibatch-based methods [24, 21], that average the outputs of several smaller optimal transport problems and *iii)* approximation-based through closed-form formulas such as projection-based OT. This paper is concerned with this third class of methods, with the underlying goal to obtain a transport plan. We quickly review them below.

Approximation of OT loss with sliced-OT. The Sliced-Wasserstein Distance (SWD) [49, 12] approximates the Wasserstein distance by projecting onto one-dimensional subspaces, where OT has a closed-form solution obtained in $O(n \log n)$. While SWD averages over multiple projections, max-SWD [19] selects the most informative one, enabling efficient computations on large-scale problems while preserving key properties. Generalized SWD alleviates the inefficiencies caused by the linear projections using polynomial projections or neural networks [31]. They build on generalized Radon transforms to use polynomial projections or neural networks, and provide conditions for which it remains a valid distance: the generalized Radon transform must be injective. Rather than considering non-linear projections, Chen et al. [13] aim at better capturing non-linearities by *augmenting* the input space, using injective neural networks, such that linear projections could better capture them. Generalized SWD has also been defined in the case of tree metrics [57].

Approximation of OT plan with sliced-OT. Addressing the challenge that none of the previous works provide an approximated transport plan, [38] proposes min-SWGG, a slicing scheme that lifts one-dimensional OT plans to the original space; [36] follow the same line but rather define expected plans computed over all the lines instead of retaining the best one.

Differentiable OT. As it is a discrete problem, OT is not differentiable *per se*. Several formulations, that mostly rely on *smoothing* the value function by adding a regularization term, have then been defined, the entropic-regularized version of the OT problem [17] being one of the most striking examples. It can be efficiently solved via Sinkhorn's matrix scaling algorithm, providing an OT approximation that is differentiable. Blondel et al. [7] use a ℓ_2 regularization term to define a smooth and sparse approximation. Regarding the unregularized OT formulation, an approximated gradient can be computed using sub-gradient on the dual formulation (see [23] for instance).

Optimizing on discrete problems. The OT problem is a linear program and the optimal plan belongs to the (potentially rescaled) Birkhoff polytope, *i.e.* the set of doubly-stochastic matrices, which is discrete. It is also the case for a wide range of problems for which the solution is prescribed to belong to such a discrete set. For instance, one can cite the sorting operator, the shortest path or the top-k operator. They break back-propagation along the computational graph, hence preventing their use in deep learning pipelines. Numerous works have been proposed to provide differentiable proxies, based on smoothing the operators: dataSP [33] define a differentiable all-to-all shortest path algorithm, smoothed sorting and ranking [8] can be defined by projection onto the permutahedron, perturbed optimizers [4] have also been defined, relying on perturbing the input data, to name a few.

Contributions. In this paper, we aim at approximating the OT distance and plan. We rely on a slicing scheme, extending the framework of [38], highlighting that it is an instance of a bilevel optimisation problem. We provide two main contributions: a generalized Sliced-Wasserstein that provides approximated OT plans relying on non-linear projections (see Fig. 1 for an illustration), then a GPU-friendly optimization algorithm to find the best projector. We showcase the benefits brought by these contributions in three different experimental contexts: on a 2 dimensional example where a non-linear projection is sought; conducting gradient flow experiment in high dimensional space; introducing a novel sliced OT-based conditional flow matching for image generation.

2 Preliminaries on Optimal Transport

We now give the necessary background on Optimal Transport and on sliced-based approximations. For a reference on computational OT, the reader can refer to [45].

2.1 Discrete Optimal Transport

We consider two point clouds $\{x_i\}_{i=1}^n \in \mathcal{X}^n$ and $\{y_i\}_{i=1}^m \in \mathcal{X}^m$ where $\mathcal{X} \subset \mathbb{R}^d$ is a discrete subset. We denote their associated empirical distributions $\mu = \sum_{i=1}^n a_i \delta_{x_i}$ and $\nu = \sum_{i=1}^m b_i \delta_{y_i}$.

Formulation and Wasserstein distance. We consider a cost function c that will be in our setting $c(u, v) = \|u - v\|_p^p$ for $p > 1$ and $C_{\mu\nu} = (c(x_i, y_j))_{i=1,j=1}^{n,m}$. Traditional Kantorovich formulation of optimal transport is given by the Wasserstein metric on $\mathcal{P}(\mathbb{R}^d)$ defined as

$$W_p^p(\mu, \nu) = \min_{\pi} \sum_{i=1}^{n} \sum_{j=1}^{m} \pi_{i,j} \|x_i - y_j\|_p^p \quad \text{subject to} \quad \pi \in U(a, b), \tag{1}$$

where $U(a, b) = \{\pi \in \mathbb{R}_{\geq 0}^{n \times m} : \pi 1_m = a, \pi^\top 1_n = b\}$ is the set of couplings between μ and ν, Eq. (1) is a convex optimization problem and an optimal transport plan $\pi_{\text{OT}}^\star$ is a solution. Note that, *a priori*, there is no reason for this minimizer to be uniquely defined.

One-dimensional OT. When $d = 1$, and μ, ν are empirical distributions with $a_i = b_i = 1/n$, the optimal transport problem is equivalent to the assignment problem. In this case, the Wasserstein distance can be computed by sorting the empirical samples, resulting in an overall complexity of $O(n \log n)$. Let σ and τ be permutations such that $x_{\sigma(1)} \leq x_{\sigma(2)} \leq \ldots \leq x_{\sigma(n)}$ and $y_{\tau(1)} \leq y_{\tau(2)} \leq \ldots \leq y_{\tau(n)}$. The Wasserstein distance is then given by $W_p^p(\mu, \nu) = \frac{1}{n} \sum_{i=1}^n |x_{\sigma(i)} - y_{\tau(i)}|^p$. The optimal transport is thus monotone and its plan $\pi_{\text{OT}}^\star$ has the form of a permutation matrix. Note that this approach can be easily extended to $n \neq m$ and arbitrary marginals a, b [45].

2.2 Sliced Wasserstein Distance

Formulation. Sliced-Wasserstein (SWD) relies on a simple idea: disintegrate the original problem onto unidimensional ones, and average over the different solutions. More precisely, SWD [49] approximates the Wasserstein distance by averaging along projection directions $\theta \in \mathbb{S}^{d-1}$ as

$$SWD_p^p(\mu, \nu) := \int_{\mathbb{S}^{d-1}} W_p^p(P_\sharp^\theta \mu, P_\sharp^\theta \nu) \mathrm{d}\lambda(\theta), \tag{2}$$

where $P^\theta : \mathbb{R}^d \to \mathbb{R}$ is the 1D projection onto the unit vector θ, $P^\theta(x) = \langle x, \theta \rangle$, and λ is the uniform distribution on the unit sphere $\mathbb{S}^{d-1}$. Typically, SWD is computed thanks to a Monte-Carlo approximation in which L directions are drawn independently, leading to a computational complexity of $O(dLn + Ln \log n)$. One of the main drawbacks of SWD is that it requires a high number of random projections, which leads to intractability for high dimensional problems. Then, there have been works to perform selective sampling, *e.g.* [42], or to optimize over the directions [19, 38]. The two later works rely on non-convex formulations that can be optimized.

Generalized Sliced WD. The idea of non-linear slicing has been explored in several works, with the aim to improve the projection efficiency, *e.g.* when the data live on non-linear manifolds. In [31], the generalized SWD uses nonlinear projections such as neural network-based ones; the conditions on which it yields a valid metric are also stated: the projection map must be injective. In [42], a generalized projection is also proposed in addition to selective sampling. Augmented Sliced Wasserstein

(ASWD) [13] pursued the same goal but, rather than considering non-linear projections, they use a neural network to *augment* the input space, which leads to a space on which a linear projection better fits the data. All these works lead to significant improvements in a wide set of machine learning scenarios.

2.3 Sliced Wasserstein Plan

As it is defined as an average over 1D OT distances, SWD does not provide inherently a transport plan[2]. Recently, some works have tackled this limitation by lifting one or several one-dimensional plans back to the original multidimensional space [38, 36].

Sliced Wasserstein Generalized Geodesics. Mahey et al. [38] introduced an OT surrogate that lifts the plan from the 1D projection onto the original space. In more detail, when $n = m$, it is defined as:

$$\text{min-SWGG}_p^p(\mu, \nu) = \min_{\theta \in \mathbb{S}^{d-1}} \text{SWGG}_p^p(\mu, \nu, \theta) := \frac{1}{n} \sum_{i=1}^{n} \|x_{\sigma_\theta(i)} - y_{\tau_\theta(i)}\|_p^p \tag{3}$$

where σ_θ and τ_θ are the permutations obtained by sorting $P_\sharp^\theta \mu$ and $P_\sharp^\theta \nu$. Note that it extends naturally to the case where $n \neq m$ which we omit here for brevity. By setting $p = 2$, it hinges on the notion of Wasserstein generalized geodesics [2] with pivot measure supported on a line. This alternative formulation allows deriving an optimization scheme to find the optimal θ that relies on multiple *copies* of the projected samples, which holds only for $p = 2$. For a fixed direction θ, provided that families $(P_\sharp^\theta(x_i))_i$ and $(P_\sharp^\theta(y_i))_i$ are injective, that is to say there is no ambiguity on the orderings σ_θ and τ_θ, $\text{SWGG}(\cdot, \cdot, \theta)$ is itself a metric [37, 54]. Min-SWGG has appealing properties: it yields an upper bound of the Wasserstein distance that still provides an explicit transport map between the input measures. The authors show that min-SWGG metrises weak convergence and is translation-equivariant.

Limitations of min-SWGG. While min-SWGG provides an upper bound on the Wasserstein distance, its tightness is not guaranteed in general. However, two settings are known where the bound is exact: (i) when one of the distributions is supported on a one-dimensional subspace ; and (ii) when the ambient dimension satisfies $d \geq 2n$ [38]. More generally, the number of reachable permutations increases with the ambient dimension [16], making data dimensionality a critical factor in the quality of the approximation. These insights motivate our first contribution: the definition of Generalized Wasserstein plans, which aim to extend the set of reachable permutations and better capture non-linear structures.

3 Generalized Sliced Wasserstein Plan

Generalized Sliced Wasserstein Plan (GSWP) is built upon the idea of generalizing SWGG to an arbitrary scalar field parametrized by some $\theta \in \mathbb{R}^q$. We consider a continuous map $\phi : \mathbb{R}^d \times \mathbb{R}^q \to \mathbb{R}$, typically a one-dimensional projection $\phi(x, \theta) = \langle x, \theta \rangle$ (in which case $q = d$), or a neural network, as in Fig. 1 for example. A Generalized Sliced Wasserstein Plan distance is defined as the one-dimensional Wasserstein distance between the point clouds through the image of $\phi^\theta := \phi(\cdot, \theta)$ for a given $\theta \in \mathbb{R}^q$.

Definition 1. Let $p > 1$, $\theta \in \mathbb{R}^q$ and $\mu, \nu \in \mathcal{P}(\mathcal{X})$. The θ-Generalized Sliced Wasserstein Plan distance between μ and ν is defined as

$$d^\theta(\mu, \nu) = W_p(\phi_\sharp^\theta \mu, \phi_\sharp^\theta \nu).$$

Any element $\pi^\theta(\mu, \nu) \in U(a, b)$ that achieves $d^\theta(\mu, \nu)$ is called a θ-Generalized Sliced Wasserstein Plan (θ-GSWP).

Denoting $\theta \mapsto C_{\mu\nu}^\phi(\theta)$ the cost matrix between μ, ν through the image of ϕ, a θ-GSWP reads

$$\pi^\theta \in \arg\min_{\pi \in U(a,b)} g(\pi, \theta) := \langle C_{\mu\nu}^\phi(\theta), \pi \rangle = \sum_{i=1}^{n} \sum_{j=1}^{m} \pi_{i,j} |\phi(x_i, \theta) - \phi(y_j, \theta)|^p. \tag{4}$$

The $g(x, \cdot)$ function is continuous but not convex, as illustrated in Fig. 2. Note that, *i)* for $p > 1$, the map $C_{\mu\nu}^\phi$ has the same regularity as ϕ and *ii)* a solution π^θ of (4) is a suboptimal point of the standard problem (1). Indeed, since the optimization occurs on the same coupling space $U(a, b)$, π^θ is an admissible point for the original problem. GSWP defines a distance on the space of measures $\mathcal{P}(\mathcal{X})$.

[2]Note that a transport plan can also be inferred when performing SW gradient flows by putting into correspondence the original and final samples when the algorithm has converged.

Proposition 1. *Let $\theta \in \mathbb{R}^q$ and assume ϕ^θ is an injective map on $\mathcal{X}$. Then d^θ is a distance on $\mathcal{P}(\mathcal{X})$.*

For clarity, all our proofs are presented in Sec A.1. Note that the injectivity condition is akin to designing sufficient and necessary conditions for the injectivity of the generalized Radon transform [5] and relates to the problem of reconstructing measures from discrete measurements [53]. We recall that there is no continuous injective map from $\mathbb{R}^d$ to $\mathbb{R}$ for $d > 1$, and we emphasize that we require only injectivity from the (potential) support to $\mathbb{R}$. In practice, using a general ϕ increases expressivity, enabling a wider range of permutations compared to a linear map. This hypothesis is satisfied almost surely in the linear case, and could also be satisfied with an injective neural network [47]. We require this property to prove that we obtain a metric; the object d^θ still makes sense without this hypothesis. From a numerical point of view, we simply rely on the order given by the sort algorithm used by Python that can be interpreted as a selection map from the set of minimizers.

By an application of Rockafellar's envelope theorem, we also have a characterization of the gradient of d^θ as a function of θ. Let us assume that $p > 1$, ϕ is jointly C^1 and (4) has a unique solution at $\theta \in \mathbb{R}^q$. Then we have $\nabla_\theta d^\theta = \frac{\partial C^\phi_{\mu\nu}}{\partial \theta}(\theta)^\top \pi^\theta$. Note that a similar result on the subdifferential of d^θ is true if ϕ is not differentiable but convex, thanks to Danskin's theorem.

Definition 2. Let $p > 1$ and $\mu, \nu \in \mathcal{P}(\mathcal{X})$. The minimal Generalized Sliced Wasserstein Plan semimetric min-GSWP between μ and ν is defined as

$$\text{min-GSWP}^p_p(\mu,\nu) = \min_{\theta\in\mathbb{R}^q} h(\theta) := f(\pi^\theta) := \langle C_{\mu\nu}, \pi^\theta\rangle \quad (5)$$
$$\text{subject to} \quad \pi^\theta \in \arg\min_{\pi\in U(a,b)} g(\pi,\theta).$$

Equation (5) defines a bilevel optimization problem. In the case where $\phi(x,\theta) = \langle x,\theta\rangle$ and if we further constrain θ to live on the unit sphere, (5) is the min-SWGG [38] approximation of OT. Observe that since π^θ is an admissible coupling for the original OT problem (1), min-GSWP(μ,ν) is an upper bound for $W_p(\mu,\nu)$. Unfortunately, the value function $\theta \mapsto h(\theta)$ does not, in general, possess desirable regularity properties, even for the simple choice of a one-dimensional projection $\phi(x,\theta) = \langle x,\theta\rangle$: it is discontinuous, as illustrated in Fig. 2. Moreover, as soon as the maps $\phi(x,\theta)$ and $\phi(y,\theta)$ give rise to the same permutations σ_θ and τ_θ, the value of π^θ remains the same. As a consequence, one cannot use (stochastic) gradient-based bilevel methods such as [44, 18, 3] on (5).

It turns out that even if min-SWGG was introduced as the minimization over the sphere $\mathbb{S}^{d-1}$, it is possible to see it as an unconstrained optimization problem thanks to the following lemma.

Lemma 1. *Assume that $\theta \mapsto \phi(x,\theta)$ is 1-homogeneous for all $x \in \mathbb{R}^d$. Then, $\theta \mapsto d^\theta(\mu,\nu)$ is 1-homogeneous, and h is invariant by scaling: $h(c\theta) = h(\theta)$ for all $c > 0$.*

In particular, for every open set where h is differentiable, if ϕ is 1-homogeneous, the gradient flow $\dot\theta = -\nabla h(\theta)$ has orthogonal level lines $\langle\dot\theta,\theta\rangle = 0$, hence the dynamics occur on the sphere of radius equal to the norm of the initialization (see Lemma 3 in the Appendix). In practice, it means – and we observed – that we do not have to care about the normalization of θ during our gradient descent.

When $\phi(x,\theta) = \langle x,\theta\rangle$, min-SWGG can be smoothed by making *perturbed* copies of the projections $\phi(x,\theta)$; heuristics for determining the number of copies and the scale of the noise are given in [38]. Nevertheless, the continuity of the obtained smooth surrogate cannot be guaranteed, and the formulation is only valid for $p = 2$. The following section proposes a differentiable approximation of min-GSWP rooted in probability theory which is valid for any $p > 1$. The main difference with the aforementioned scheme is that we here perturb the parameters (or direction) θ rather than the samples.

4 Differentiable Approximation of min-GSWP

Smoothing estimators by averaging is a popular way to tackle nonsmooth problems. We mention three families of strategies: *i)* smoothing by (infimal) convolution, *ii)* smoothing by using a Gumbel-like trick [1, 4], and *iii)* smoothing by reparameterization, and in particular by using Stein's lemma. Strategy *i)* would be computationally intractable in high dimension, and strategy *ii)* doesn't yield a consistent transport plan due to perturbations in the cost matrix. We then focus on the third option.

We begin by recalling this classical result due to Stein [52], which plays a central role in our analysis. The lemma, restated below in our specific setting, provides an identity for computing gradients

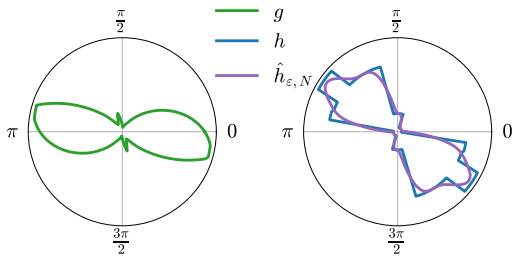


Figure 2: Example g and h (seen as a function of θ on the sphere $\mathbb{S}^1$) for a 2D OT problem. $\hat{h}_{\varepsilon,N}$ is a Monte-Carlo estimate of h_ε with gradient $\hat{\nabla} h_{\varepsilon,N}$. Note that g is continuous whereas h is piecewise constant, hence there is a need for a smoothing mechanism, that results in $\hat{h}_{\varepsilon,N}$.

through expectations involving Gaussian perturbations. Typically, Stein's lemma is stated for "almost differentiable" functions; we require here slightly less regularity.

Assumption 1. There exists a open set $C \subseteq \mathbb{R}^q$ with Hausdorff dimension $\mathcal{H}^{q-1}(\mathbb{R}^q \setminus C) = 0$ (in particular Lebesgue-negligible) such that the mapping $\theta \mapsto h(\theta)$ is continuously differentiable on C.

Under this assumption, Stein's lemma remains true.

Lemma 2 (Stein's lemma)**.** *Let $Z \sim \mathcal{N}(0, \mathrm{Id}_q)$ be a standard multivariate Gaussian random variable, and $\varepsilon > 0$. Suppose Assumption 1 holds and that for all indices j, the partial derivatives satisfy the integrability condition $\mathbb{E}[|\partial_j h(Z)|] < +\infty$. Then, the following identity holds:*

$$\mathbb{E}_Z[\nabla h(\theta + \varepsilon Z)] = \varepsilon^{-1} \mathbb{E}_Z[h(\theta + \varepsilon Z) Z].$$

Note that this version of Stein's lemma is slightly more general than the original of Stein [52] in terms of regularity asked to the value function h (in particular, we do not require it to be absolutely continuous). Assumption 1 may not be satisfied when using a non-smooth network, such as a ReLU network. It is possible to alleviate this issue thanks to conservative calculus [9].

We define the following smoothed value function h_ε, which corresponds to a Gaussian smoothing of the original non-differentiable outer optimization problem:

$$h_\varepsilon(\theta) = \mathbb{E}_Z\left[h(\theta + \varepsilon Z)\right] = \langle C_{\mu\nu}, \pi_\varepsilon^\theta \rangle$$

where the θ-**Differentiable Generalized Sliced Wasserstein Plan (θ-DGSWP)** at smoothing level ε reads

$$\pi_\varepsilon^\theta = \mathbb{E}_Z\left[\arg\min_{\pi \in U(a,b)} g(\pi, \theta + \varepsilon Z)\right]. \tag{6}$$

Due to the nice regularity properties of Stein's approximation, we get the following proposition regarding our approximation of θ-GSWP.

Proposition 2. *Suppose Assumption 1 holds. Let $\mu, \nu \in \mathcal{P}(\mathcal{X})$. The following statements are true:*

1. *(Admissibility.) For any $\varepsilon > 0$, π_ε^θ is admissible (i.e., $\pi_\varepsilon^\theta \in U(a,b)$). Hence, $h_\varepsilon(\theta)$ gives an upper-bound of the Wasserstein metric $h_\varepsilon(\theta) \geq W_p^p(\mu, \nu)$.*
2. *(Differentiability.) For any $\varepsilon > 0$, the map $\theta \mapsto h_\varepsilon(\theta)$ is differentiable. Moreover, we have*
$$\nabla_\theta h_\varepsilon(\theta) = \varepsilon^{-1} \mathbb{E}_Z[h(\theta + \varepsilon Z) Z].$$
3. *(Consistency.) For almost all $\theta \in \mathbb{R}^q$, $\lim_{\varepsilon \to 0} h_\varepsilon(\theta) = h(\theta)$, and if $\nabla h(\theta)$ exists, then $\lim_{\varepsilon \to 0} \nabla h_\varepsilon(\theta) = \nabla h(\theta)$*
4. *(Distance.) Let $\theta \in \mathbb{R}^q$, if ϕ^θ is injective on $\mathcal{X}$ then the map $(\mu, \nu) \mapsto (h(\theta)(\mu, \nu))^{1/p}$ is a distance on $\mathcal{P}(\mathcal{X})$. Assume that for almost all $\theta \in \mathbb{R}^q$, ϕ^θ is injective. Then, $(\mu, \nu) \mapsto (h_\varepsilon(\theta)(\mu, \nu))^{1/p}$ is also a distance on $\mathcal{P}(\mathcal{X})$.*

While this result allows for the unbiased estimation of gradients, it is known that a naive Monte Carlo approximation of the right-hand side tends to suffer from high variance. To address this, one may consider an alternative formulation that often results in a reduced variance estimator. Specifically, using a control variate approach [6], one observes that

$$\mathbb{E}_Z[\nabla h(\theta + \varepsilon Z)] = \varepsilon^{-1} \mathbb{E}_Z[(h(\theta + \varepsilon Z) - h(\theta)) Z], \tag{7}$$

which holds due to the zero mean of the Gaussian distribution and the linearity of the expectation. Equation (7) leads to a Monte-Carlo estimator of the gradient $\nabla h_\varepsilon(\theta)$ defined as

$$\widehat{\nabla} h_{\varepsilon,N}(\theta) = \frac{1}{\varepsilon N} \sum_{k=1}^{N} (h(\theta + \varepsilon z_k) - h(\theta)) z_k = \frac{1}{\varepsilon N} \sum_{k=1}^{N} \langle C_{\mu\nu}, \pi^{\theta+\varepsilon z_k} - \pi^{\theta} \rangle z_k, \tag{8}$$

where the vectors $z_i \sim \mathcal{N}(0, \mathrm{Id}_q)$ are independent standard Gaussian samples. Hence, estimating the Monte-Carlo gradient $\nabla h_{\varepsilon,N}(\pi, \theta)$ requires solving $N + 1$ 1D-optimal transport problems for an overall cost of $O(N(n+m)\log(n+m))$. Algorithm 1 (in Supplementary material) describes a gradient descent method to perform the minimization of h_ε using this Monte-Carlo approximation.

5 Experiments

We evaluate the performance of our Differentiable Generalized Sliced Wasserstein Plans, coined DGSWP, by assessing its ability to provide a meaningful approximated OT plan in several contexts. First, we consider a toy example where a non-linear projection must be considered; we then perform gradient flow experiments on Euclidean and hyperbolic spaces, demonstrating the versatility of our approach. Finally, we integrate sliced-OT plans in an OT-based conditional flow matching in lieu of mini-batch OT. In all the experiments, we use $\varepsilon = 0.05$ and $N = 20$ as DGSWP-specific hyperparameters. Full experimental setups and additional results are provided in App. A.3. Implementation is available online[3]; we also use POT toolbox [22].

5.1 DGSWP as an OT approximation

We begin by examining the illustrative scenario shown in Fig. 1, where the task is to compute an optimal transport (OT) plan between a mixture of eight Gaussians (source) and the Two Moons dataset (target). While some of the Gaussian modes are properly matched by min-SWGG, others are matched to the more distant moon, due to information loss along the direction orthogonal to the projection. To address this limitation, we apply DGSWP with a neural network parametrizing the projection function ϕ, enabling more expressive projections. This improvement is reflected in the quantitative results: the transport cost associated with the DGSWP plan is notably closer to the squared Wasserstein distance than that of the min-SWGG method.

This transport plan above is obtained using the approach outlined in Sec. 4. To further evaluate the impact of the variance reduction technique introduced in Eq. (8), we now repeat the same experiment across 10 different random initializations of the neural network. The average learning curves for the variants with and without variance reduction are shown in Fig. 3. The results clearly indicate that using variance reduction improves both the final transport cost and the stability of the learning process. Based on this evidence, we adopt the variance reduction strategy in all subsequent experiments.

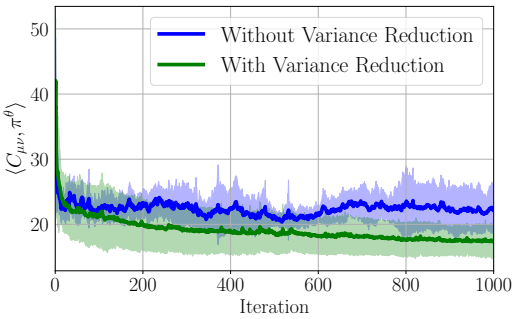


Figure 3: Impact of the variance reduction scheme (first 1,000 iterations).

5.2 Gradient flows

In these gradient flow experiments, our objective is to iteratively transport the particles of a source distribution toward a target distribution by progressively minimizing the DGSWP objective.

Without manifold assumption. We compare DGSWP with a linear and NN-based ϕ mapping against several baseline methods: Sliced Wasserstein distance (SWD), Augmented Sliced Wasserstein Distance, that has been shown in [13] to outperform GSW methods, and min-SWGG, evaluated in both its random search and optimization-based forms. Fig. 4 presents results across a range of target distributions designed to capture diverse structural and dimensional characteristics. In all experiments, the source distribution is initialized as a uniform distribution within a hypercube. We set the number of

[3] `https://github.com/rtavenar/dgswp`

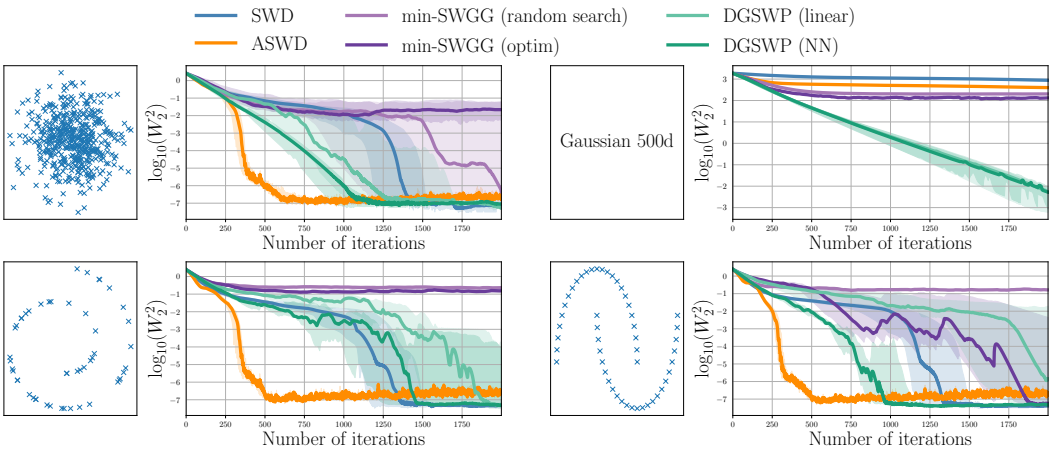


Figure 4: Log of the Wasserstein Distance as a function of the number of iterations of the gradient flow, considering several target distributions. The source distribution is uniform in all cases.

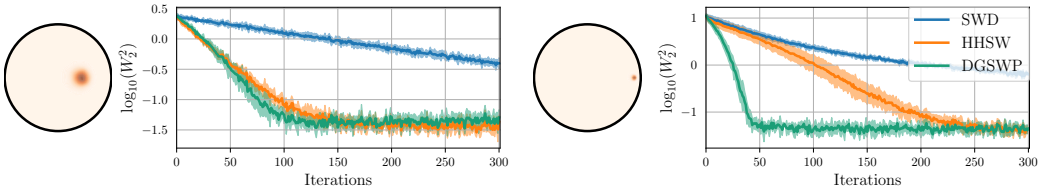


Figure 5: Log of the WD (second and fourth panels) for two different targets (first and third ones) as wrapped normal distributions for HHSW, SWD and DGWSP.

samples at $n = 50$, and repeat each experiment over 10 random seeds, reporting the median transport cost along with the first and third quartiles. We use the same learning rate for all experiments.

In two-dimensional settings, DGSWP consistently converges to a meaningful solution, whereas methods based on the min-SWGG objective sometimes fail to do so, even when using their original optimization scheme. More notably, in the high-dimensional regime, DGSWP stands out as the only slicing-based method capable of producing satisfactory transport plans, underscoring its robustness and scalability in challenging settings. This holds true even though min-SWGG is theoretically equivalent to Wasserstein when the dimension is high enough (here we have $d \geq 2n$); in practice, its optimization often struggles to find informative directions. In contrast, our proposed optimization strategy succeeds even in the linear case (where the formulation effectively reduces to min-SWGG) highlighting the practical benefit of our approach.

On hyperbolic spaces. Hyperbolic spaces are Riemannian manifolds of negative constant curvature [34]. They manifest in several representations and we consider here the Poincaré ball of dimension d, $\mathbb{B}^d$. We aim to construct a manifold feature map for hyperbolic space $\phi : \mathbb{B}^d \times \mathbb{B}^q \to \mathbb{R}$, typically relying on an analogue of hyperplanes. Horospheres [27] generalize hyperplanes in the Poincaré ball and are parametrized by a base point located on the boundary of the space $\theta \in \partial\mathbb{B}^d = \mathbb{S}^{d-1}$. Akin to [29], we consider a map that corresponds to horosphere projection $\phi(x, \theta) = \log(\|x-\theta\|_2^2) - \log(1-\|x\|_2^2)$.

Similarly to [10], we assess the ability of generalized sliced Wasserstein plans to learn distributions that live in the Poincaré disk. We compare with the horospherical sliced-Wasserstein discrepancy (HHSW, [10]) and Sliced Wasserstein computed on the Poincaré ball. For the gradient estimation, we rely on the von Mises-Fisher which is a generalization of a Gaussian distribution from $\mathbb{R}^d$ to $\mathbb{S}^{d-1}$ and perform a Riemannian Gradient Descent using Python toolbox `geoopt` [30] to optimize on θ and the source data (that live on $\mathbb{B}^d$). Figure 5 plots the evolution of the exact log 10-Wasserstein distance between the learned distribution and the target, using the geodesic distance as ground cost. We use wrapped normal distributions as source and set the same learning rate for all methods. DGWSP enables fast convergence towards the target, demonstrating its versatility for hyperbolic manifolds.

Integration method →	Euler		DoPri5	
Algorithm ↓	FID	NFE	FID	NFE
I-CFM	4.61 ± 0.10	100	3.74 ± 0.04	138.75 ± 3.47
OT-CFM	4.75 ± 0.05	100	3.81 ± 0.04	132.56 ± 0.67
DGSWP-CFM (linear)	4.16 ± 0.09	100	4.45 ± 0.08	112.63 ± 2.32
DGSWP-CFM (NN)	$\mathbf{3.62} \pm 0.05$	100	3.95 ± 0.09	120.04 ± 2.30

Table 1: Average FID score ($\pm$ std) and average number of function evaluations (NFE) per batch, computed over 3 runs.

5.3 Sliced-OT based Conditional Flow Matching

We now investigate the use of DGSWP in the context of generative modeling. In the flow matching (FM) framework, a time-dependent velocity field $u_t(x)$ is learned such that it defines a continuous transformation from a prior to the data distribution. In practice, u_t is trained by minimizing a regression loss on synthetic trajectories sampled from a known coupling between source and target distributions that determines the starting and ending points (x_0, x_1) of the trajectory. Several variants of flow matching have been proposed depending on how these pairs are sampled. In Independent CFM (I-CFM, [35]), pairs are sampled independently from the prior and data distributions, respectively. However, this approach ignores any explicit alignment between source and target samples. To address this, OT-CFM [56] proposes to deterministically couple source and target samples using mini-batch OT, so that $(x_0, x_1) \sim \pi$ where π is the OT plan computed between a batch of prior samples and a batch of data samples. This coupling tends to straighten the diffusion trajectories, which leads to improved generation quality in few-step sampling regimes.

Despite its advantages, OT-CFM relies on a trade-off: since computing exact OT is infeasible for large datasets, mini-batch OT is used, which leads to imperfect matchings, especially in high-dimensional spaces involving complex distributions, for which a mini-batch is unlikely to be representative of the whole distribution. This motivates the use of DGSWP as an alternative. By estimating transport plans using DGSWP, we can leverage significantly larger batches, resulting in better couplings, while maintaining low computational cost. In our experiments, we aggregate samples from 10 mini-batches to compute the transport plan, and find that the additional computational cost remains negligible. Our approach is also supported by the findings of Cheng and Schwing [14], who show that increasing the batch size improves performance for OT-CFM in the few-sampling-steps regime.

We conduct experiments on CIFAR-10 [32], reporting FID scores for DGSWP-CFM, OT-CFM, and I-CFM across varying numbers of sampling steps. We use the experimental setup and hyperparameters from Tong et al. [56]. For DGSWP, we evaluate both a linear projector and a more expressive non-linear embedding implemented by a neural network. We compare a fixed-step Euler solver with the adaptive Dormand-Prince method [20] for trajectory integration. The results in Table 1 underscore the importance of learning straight transport trajectories, which translates to more efficient generation: models that induce straighter flows require significantly fewer function evaluations (NFE) to achieve high-quality samples. Notably, even a simple 100-step Euler scheme proves highly effective when used with DGSWP-CFM, demonstrating the practicality of the method for fast generation.

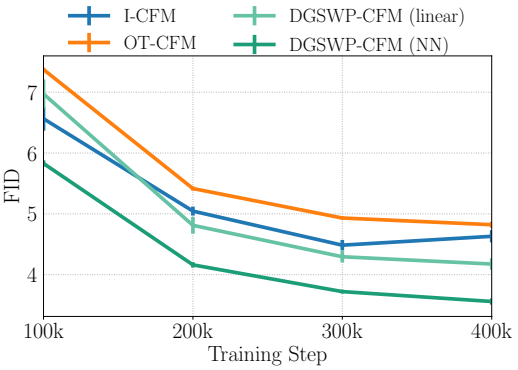


Figure 6: Average FID (over 3 runs) as a function of training iterations for various algorithms using 100-step Euler sampling. Vertical lines represent the standard deviation.

A closer analysis of the results leads to three key observations: (i) Among the projection choices, linear DGSWP underperforms compared to its neural network-based counterpart, highlighting the benefit of extending min-SWGG with non-linear projections; (ii) When using the adaptive Dormand–Prince solver, DGSWP achieves performance comparable to OT-CFM in terms of FID, but with lower

computational cost as indicated by the reduced number of function evaluations; (iii) In the fixed-step regime, DGSWP clearly outperforms all baselines under the Euler solver, offering the best trade-off between sample quality and efficiency, as illustrated in Fig. 6.

6 Conclusion

This paper presents a novel differentiable approach to approximate sliced Wasserstein plans, incorporating non-linear projections. Its differentiable and GPU-efficient formulation enables the definition of optimal projections. The proposed method offers several key advantages: i) efficient computation comparable to that of SWD, ii) ability to provide an approximated transport plan, akin to min-SWGG, iii) improved performance in high-dimensional settings thanks to its improved optimization strategy, iv) the capacity to handle data supported on manifolds through a generalized formulation. To the best of our knowledge, we also provide the first empirical evidence that slicing techniques can be effectively used in conditional flow matching, resulting in better performance and fewer function evaluations.

A key strength of slicing-based approaches for approximating OT lies in their favourable statistical properties, such as improved sample complexity [41]. These advantages also extend to projection-based methods on k-dimensional subspaces [43], as well as to min-SWGG [37]. Future works will explore the conditions that the projection map ϕ must satisfy to preserve these statistical guarantees. Additionally, ensuring the injectivity of ϕ is essential for DGSWP to define a proper distance. Investigating injective neural architectures, such as those proposed in [47], is a promising research direction.

Acknowledgments

SV is supported by ANR PRC MAD ANR-24-CE23-1529. LC and RT are supported by ANR PRCE ANR-25-CE23-6035.

References

[1] J. Abernethy, C. Lee, A. Sinha, and A. Tewari. Online linear optimization via smoothing. In *Conference on Learning Theory*, pages 807–823, 2014.

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

[3] Michael Arbel and Julien Mairal. Amortized implicit differentiation for stochastic bilevel optimization. In *ICLR*, 2022.

[4] Quentin Berthet, Mathieu Blondel, Olivier Teboul, Marco Cuturi, Jean-Philippe Vert, and Francis Bach. Learning with differentiable pertubed optimizers. In *NeurIPS*, volume 33, pages 9508–9519, 2020.

[5] Gregory Beylkin. The inversion problem and applications of the generalized Radon transform. *Communications on Pure and Applied Mathematics*, 37(5):579–599, 1984.

[6] Mathieu Blondel and Vincent Roulet. The elements of differentiable programming, 2024. URL `https://arxiv.org/abs/2403.14606`.

[7] Mathieu Blondel, Vivien Seguy, and Antoine Rolet. Smooth and sparse optimal transport. In *AISTATS*, pages 880–889. PMLR, 2018.

[8] Mathieu Blondel, Olivier Teboul, Quentin Berthet, and Josip Djolonga. Fast differentiable sorting and ranking. In *ICML*, pages 950–959, 2020.

[9] Jérôme Bolte and Edouard Pauwels. Conservative set valued fields, automatic differentiation, stochastic gradient methods and deep learning. *Mathematical Programming*, 188(1):19–51, 2021.

[10] Clément Bonet, Laetitia Chapel, Lucas Drumetz, and Nicolas Courty. Hyperbolic sliced-wasserstein via geodesic and horospherical projections. In *Proceedings of 2nd Annual Workshop on Topology, Algebra, and Geometry in Machine Learning (TAG-ML)*. PMLR, 2023.

[11] Nicolas Bonneel and David Coeurjolly. Spot: sliced partial optimal transport. *ACM Transactions on Graphics (TOG)*, 38(4):1–13, 2019.

[12] Nicolas Bonneel, Julien Rabin, Gabriel Peyré, and Hanspeter Pfister. Sliced and Radon Wasserstein barycenters of measures. *Journal of Mathematical Imaging and Vision*, 51:22–45, 2015.

[13] Xiongjie Chen, Yongxin Yang, and Yunpeng Li. Augmented sliced Wasserstein distances. In *ICLR*, 2022.

[14] Ho Kei Cheng and Alexander Schwing. The curse of conditions: Analyzing and improving optimal transport for conditional flow-based generation. *arXiv preprint arXiv:2503.10636*, 2025.

[15] Nicolas Courty, Rémi Flamary, Devis Tuia, and Alain Rakotomamonjy. Optimal transport for domain adaptation. *IEEE transactions on pattern analysis and machine intelligence*, 39(9): 1853–1865, 2016.

[16] Thomas M Cover. The number of linearly inducible orderings of points in d-space. *SIAM Journal on Applied Mathematics*, 15(2):434–439, 1967.

[17] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. *NeurIPS*, 2013.

[18] Mathieu Dagréou, Pierre Ablin, Samuel Vaiter, and Thomas Moreau. A framework for bilevel optimization that enables stochastic and global variance reduction algorithms. *NeurIPS*, 2022.

[19] Ishan Deshpande, Yuan-Ting Hu, Ruoyu Sun, Ayis Pyrros, Nasir Siddiqui, Sanmi Koyejo, Zhizhen Zhao, David Forsyth, and Alexander G Schwing. Max-sliced Wasserstein distance and its use for GANs. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10648–10656, 2019.

[20] John R Dormand and Peter J Prince. A family of embedded runge-kutta formulae. *Journal of computational and applied mathematics*, 6(1):19–26, 1980.

[21] Kilian Fatras, Younes Zine, Rémi Flamary, Remi Gribonval, and Nicolas Courty. Learning with minibatch wasserstein: asymptotic and gradient properties. In *AISTATS*, 2020.

[22] Rémi Flamary, Nicolas Courty, Alexandre Gramfort, Mokhtar Z Alaya, Aurélie Boisbunon, Stanislas Chambon, Laetitia Chapel, Adrien Corenflos, Kilian Fatras, Nemo Fournier, et al. POT: Python optimal transport. *Journal of Machine Learning Research*, 22(78):1–8, 2021.

[23] Aude Genevay, Marco Cuturi, Gabriel Peyré, and Francis Bach. Stochastic optimization for large-scale optimal transport. *NeurIPS*, 2016.

[24] Aude Genevay, Gabriel Peyré, and Marco Cuturi. Learning generative models with Sinkhorn divergences. In *AISTATS*, pages 1608–1617. PMLR, 2018.

[25] Niels Richard Hansen. On stein's unbiased risk estimate for reduced rank estimators. *Statistics & Probability Letters*, 135:76–82, 2018.

[26] Niels Richard Hansen and Alexander Sokol. Degrees of freedom for nonlinear least squares estimation. *arXiv preprint arXiv:1402.2997*, 2014.

[27] Ernst Heintze and Hans-Christoph Im Hof. Geometry of horospheres. *Journal of Differential geometry*, 12(4):481–491, 1977.

[28] Leonid V Kantorovich. On the translocation of masses. In *Dokl. Akad. Nauk. USSR (NS)*, volume 37, pages 199–201, 1942.

[29] Isay Katsman, Eric Chen, Sidhanth Holalkere, Anna Asch, Aaron Lou, Ser Nam Lim, and Christopher M De Sa. Riemannian residual neural networks. *NeurIPS*, 2023.

[30] Max Kochurov, Rasul Karimov, and Serge Kozlukov. Geoopt: Riemannian optimization in pytorch, 2020.

[31] Soheil Kolouri, Kimia Nadjahi, Umut Simsekli, Roland Badeau, and Gustavo Rohde. Generalized sliced wasserstein distances. *NeurIPS*, 32, 2019.

[32] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[33] Alan Lahoud, Erik Schaffernicht, and Johannes Stork. Datasp: A differential all-to-all shortest path algorithm for learning costs and predicting paths with context. In *Uncertainty in Artificial Intelligence*, pages 2094–2112. PMLR, 2024.

[34] John M Lee. *Riemannian manifolds: an introduction to curvature*, volume 176. Springer Science & Business Media, 2006.

[35] Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *ICLR*, 2023.

[36] Xinran Liu, Rocio Diaz Martin, Yikun Bai, Ashkan Shahbazi, Matthew Thorpe, Akram Aldroubi, and Soheil Kolouri. Expected sliced transport plans. In *ICLR*, 2024.

[37] Guillaume Mahey. *Unbalanced and linear optimal transport for reliable estimation of the Wasserstein distance*. PhD thesis, INSA Rouen Normandie, 2024.

[38] Guillaume Mahey, Laetitia Chapel, Gilles Gasso, Clément Bonet, and Nicolas Courty. Fast optimal transport through sliced generalized wasserstein geodesics. *NeurIPS*, 36, 2024.

[39] Gaspard Monge. Mémoire sur la théorie des déblais et des remblais. *Mem. Math. Phys. Acad. Royale Sci.*, pages 666–704, 1781.

[40] Eduardo Fernandes Montesuma, Fred Maurice Ngole Mboula, and Antoine Souloumiac. Recent advances in optimal transport for machine learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

[41] Kimia Nadjahi, Alain Durmus, Lénaïc Chizat, Soheil Kolouri, Shahin Shahrampour, and Umut Simsekli. Statistical and topological properties of sliced probability divergences. *NeurIPS*, 2020.

[42] Khai Nguyen, Nhat Ho, Tung Pham, and Hung Bui. Distributional sliced-wasserstein and applications to generative modeling. In *ICLR*, 2021.

[43] François-Pierre Paty and Marco Cuturi. Subspace robust wasserstein distances. In *ICML*, pages 5072–5081. PMLR, 2019.

[44] Fabian Pedregosa. Hyperparameter optimization with approximate gradient. In *ICML*, pages 737–746, 2016.

[45] Gabriel Peyré, Marco Cuturi, et al. Computational optimal transport: With applications to data science. *Foundations and Trends® in Machine Learning*, 11(5-6):355–607, 2019.

[46] Aram-Alexandre Pooladian, Heli Ben-Hamu, Carles Domingo-Enrich, Brandon Amos, Yaron Lipman, and Ricky TQ Chen. Multisample flow matching: Straightening flows with minibatch couplings. In *ICML*, 2023.

[47] Michael Puthawala, Konik Kothari, Matti Lassas, Ivan Dokmanić, and Maarten De Hoop. Globally injective relu networks. *Journal of Machine Learning Research*, 23(105):1–55, 2022.

[48] Julien Rabin, Julie Delon, and Yann Gousseau. Regularization of transportation maps for color and contrast transfer. In *2010 IEEE International Conference on Image Processing*, 2010.

[49] Julien Rabin, Gabriel Peyré, Julie Delon, and Marc Bernot. Wasserstein barycenter and its application to texture mixing. In *Scale Space and Variational Methods in Computer Vision*, 2012.

[50] Julien Rabin, Sira Ferradans, and Nicolas Papadakis. Adaptive color transfer with relaxed optimal transport. In *IEEE international conference on image processing*, 2014.

[51] Filippo Santambrogio. *Optimal transport for applied mathematicians*, volume 87. Springer, 2015.

[52] Charles Stein. Estimation of the mean of a multivariate normal distribution. *The Annals of Statistics*, 9(6):1135–1151, 1981.

[53] Eloi Tanguy, Rémi Flamary, and Julie Delon. Reconstructing discrete measures from projections. Consequences on the empirical Sliced Wasserstein Distance. *Comptes Rendus. Mathématique*, 362:1121–1129, 2024.

[54] Eloi Tanguy, Laetitia Chapel, and Julie Delon. Sliced optimal transport plans. *arXiv preprint arXiv:2508.01243*, 2025.

[55] Ryan J Tibshirani. Degrees of freedom and model search. *Statistica Sinica*, pages 1265–1296, 2015.

[56] Alexander Tong, Kilian Fatras, Nikolay Malkin, Guillaume Huguet, Yanlei Zhang, Jarrid Rector-Brooks, Guy Wolf, and Yoshua Bengio. Improving and generalizing flow-based generative models with minibatch optimal transport. *Transactions on Machine Learning Research*, 2024.

[57] Thanh Tran, Hoang V Tran, Thanh Chu, Huyen Trang Pham, Laurent Ghaoui, Tam Le, and Tan Minh Nguyen. Tree-sliced wasserstein distance with nonlinear projection. In *ICML*, 2025.

A Appendix / supplemental material

A.1 Proofs

We start by the proof of Lemma 1, concerned with 0-homogeneity of our target $\theta \mapsto h(\theta)$.

Proof of Lemma 1. We prove in fact that for all $c > 0$, $\pi^\theta = \pi^{c\theta}$. This is a consequence of the fact that if $\theta \mapsto \phi(x, \theta)$ is 1-homogeneous, then so is $\theta \mapsto C^\phi_{\mu\nu}(\theta)$. Thus, the level sets of $\pi \mapsto g(\pi, c\theta)$ are the level-sets of $\pi \mapsto g(\pi, \theta)$ dilated by a factor c. Hence, they share the same minimizers, and in consequence $h(c\theta) = h(\theta)$. □

For the sake of completeness, we also prove the comment stating that a gradient flow on h preserves the norm of the initialization with the following lemma.

Lemma 3. *Let $U \subseteq \mathbb{R}^q$ an open set, assume that $h : U \mapsto \mathbb{R}$ is differentiable and h is 0-homogeneous, i.e., $h(c\theta) = h(\theta)$. Consider the gradient flow dynamics*

$$\begin{cases} \theta(0) &= \theta_0 \in U \\ \dot\theta(t) &= -\nabla h(\theta(t)). \end{cases} \tag{9}$$

Then, there exists an interval $I \subseteq \mathbb{R}_{\geq 0}$ and a unique solution $t \mapsto \theta(t)$ such that for all $t \in I$, $\langle \dot\theta(t), \theta(t)\rangle = 0$ and $\|\theta(t)\| = \theta_0$.

Proof. Orthogonal gradient of 0-homogeneous function. Consider the real function $\psi : \mathbb{R} \to \mathbb{R}$ defined by $\psi(c) = h(c\theta)$. By 0-homogeneity, $\psi(c) = h(\theta) = \psi(1)$. Hence, ψ is constant on $\mathbb{R}^*$, thus $\psi'(c) = 0$ for all $c \neq 0$. Using the chain rule for real functions, we have that for all $c \neq 0$

$$\psi'(c) = \langle (c \mapsto c\theta)'(c), \nabla h(c\theta)\rangle = \langle \theta, \nabla h(c\theta)\rangle = 0.$$

Using this fact for $c = 1$, we conclude that

$$\forall \theta \in U, \quad \langle \theta, \nabla h(\theta)\rangle = 0. \tag{10}$$

Orthogonal dynamics. The existence and uniqueness of the Cauchy problem (9) comes from the Cauchy-Lipschitz theorem. Consider this solution $t \mapsto \theta(t)$ defined over I. Then,

$$\langle \dot\theta(t), \theta(t)\rangle = -\langle \nabla h(\theta(t)), \theta(t)\rangle = 0,$$

using (10). Hence, $\dot\theta(t) \perp \theta(t)$ for all $t \in I$.

Conservation of the norm. Consider $r(t) = \|\theta(t)\|^2$. The chain rule tells us that for all $t \in I$,

$$r'(t) = 2\langle \dot\theta(t), \theta(t)\rangle,$$

hence $r' = 0$ and thus r is a constant function. □

The proof of Proposition 1 relies on the fact that the p-Wasserstein distance is a metric on 1D measures, and that ϕ^θ is conveniently supposed to be injective over the reference set $\mathcal{X}$.

Proof of Proposition 1. Let $p > 1$, $\theta \in \mathbb{R}^q$, and assume that ϕ^θ is an injective map from $\mathbb{X}$ $\mathcal{X}$? to $\mathbb{R}$. Let μ, ν, ξ three discrete distributions in $\mathcal{P}(\mathcal{X})$. Recall that

$$d^\theta(\mu, \nu) = W_p(\phi^\theta_\sharp \mu, \phi^\theta_\sharp \nu).$$

Using the fact that W_p is a metric on $\mathcal{P}_p(\mathbb{R})$, we also obtain that W_p is a metric on $\mathcal{P}(\mathcal{X})$ as a restriction.

Well-posedness. Since $d^\theta(\mu, \mu) = W_p(\phi^\theta_\sharp \mu, \phi^\theta_\sharp \mu)$, and that W_p is a metric, we have that $d^\theta(\mu, \mu) = 0$.

Symmetry. The symmetry comes directly from the one of W_p.

Positivity. Suppose that $\mu \neq \nu$. Since W_p is a metric, we only need to prove that $\phi^\theta_\sharp \mu \neq \phi^\theta_\sharp \nu$. Assuming that, where $x_i \neq x_{i'}$ for all $i \neq i'$ and $y_j \neq y_{j'}$ for all $j \neq j'$,

$$\mu = \sum_{i=1}^n a_i \delta_{x_i} \quad \text{and} \quad \nu = \sum_{j=1}^m b_j \delta_{y_j},$$

we have that

$$\phi^\theta_\sharp \mu = \sum_{i=1}^n a_i \delta_{\phi^\theta(x_i)} \quad \text{and} \quad \phi^\theta_\sharp \nu = \sum_{j=1}^m b_j \delta_{\phi^\theta(y_j)}.$$

Using the injectivity of ϕ^θ, we have that $\phi^\theta(x_i) \neq \phi^\theta(x_{i'})$ for all $i \neq i'$ and $\phi^\theta(y_j) \neq \phi^\theta(y_{j'})$ for all $j \neq j'$. Hence, $\phi^\theta_\sharp \mu = \phi^\theta_\sharp \nu$ if and only if $n = m$ and there exists a permutation $\sigma : \{1, \dots, n\} \to \{1, \dots, n\}$ such that

$$a_i = b_{\sigma(i)} \quad \text{and} \quad \phi^\theta(x_i) = \phi^\theta(y_{\sigma(i)}), \text{ for all } i.$$

But then, using the injectivity of ϕ^θ, we have $\{x_i\}_{i=1}^n = \{y_i\}_{i=1}^n$. Hence, $\mu = \nu$ which is a contradiction.

Triangle inequality. We have that $d^\theta(\mu, \nu) = W_p(\phi^\theta_\sharp \mu, \phi^\theta_\sharp \nu)$. Using the triangle inequality on W_p, we have that $d^\theta(\mu, \nu) \leq W_p(\phi^\theta_\sharp \mu, \phi^\theta_\sharp \xi) + W_p(\phi^\theta_\sharp \xi, \phi^\theta_\sharp \nu) = d^\theta(\mu, \xi) + d^\theta(\xi, \nu)$. □

We now turn to Stein's lemma. The proof of Stein's lemma under a (weak) differentiability criterion [52] is classic, and relies on integration by parts and the properties of the normal distribution. Nevertheless, we are concerned with a function $\theta \mapsto h(\theta)$ that typically will have discontinuities, breaking the classical proof. Note that one cannot expect Stein's lemma to hold true for any kind of discontinuities, even with almost everywhere differentiability. The celebrated example is the Heaviside function $h(\theta) = 1_{\theta \geq 0}$ in 1D where Stein's lemma needs a correction term if there is a non-negligible number of them in the sense of the $\mathcal{H}^{q-1}$ Hausdorff dimension. This setting was studied by [26, 55, 25] for various applications in statistics and signal processing, in particular for Stein's unbiased risk estimation.

Proof of Lemma 2. The proof of this result is mostly contained in [25, Proposition 1]. We outline the strategy. Assumption 1 requires having $\mathcal{H}^{q-1}(\mathbb{R}^q \setminus C) = 0$. Hence, for all $i \in \{1, \dots, q\}$ and Lebesgue almost all $(\theta_1, \dots, \theta_{i-1}, \theta_{i+1}, \dots, \theta_q) \in \mathbb{R}^{q-1}$, the map

$$t \mapsto h(\theta_1, \dots, \theta_{i-1}, t, \theta_{i+1}, \dots, \theta_q)$$

is absolutely continuous on every compact interval of $\mathbb{R}$. So, in turn, $h \in W^{1,1}_{\text{loc}}(\mathbb{R}^q)$ which in turn shows that h is almost differentiable in the sense of Stein [52] and we can thus apply [52, Lemma 1]. □

We now turn to the proof of Proposition 2, regarding the properties of the smoothed version h_ε of h.

Proof of Proposition 2. (*Admissibility.*) Let $\varepsilon > 0$ and $\theta \in \mathbb{R}^q$. Recall that

$$\pi^\theta_\varepsilon = \mathbb{E}_{Z \sim \mathcal{N}(0, I_q)} \left[\arg \min_{\pi \in U(a,b)} g(\pi, \theta + \varepsilon Z) \right].$$

Denote by $u(z) = \arg\min_{\pi \in U(a,b)} g(\pi, \theta + \varepsilon z)$, hence $\pi^\theta_\varepsilon = \mathbb{E}_{Z \sim \mathcal{N}(0, I_q)}[u(Z)]$. For all $z \in \mathbb{R}^q$, $u(z) \in U(a, b)$ by definition of the minimization problem. Hence,

$$\pi^\theta_\varepsilon = \int_{\mathbb{R}^q} u(z) \rho(z) \mathrm{d}z,$$

where $\rho(z) = (2\pi)^{-q/2} \exp(-\frac{1}{2}\|x\|^2)$ is the probability density function of the multivariate normal law and $\mathrm{d}z$ is the Lebesgue measure on $\mathbb{R}^q$. Since $U(a, b)$ is convex and $u(z) \in U(a, b)$, then $\int_{\mathbb{R}^q} u(z)\rho(z)\mathrm{d}z \in U(a, b)$ also. In turn, since $\pi^\theta_\varepsilon \in U(a, b)$, then given the true solution $\pi^\star_{\text{OT}}$ of (1), we have $\sum_{i=1}^n \sum_{j=1}^m \pi^\star_{\text{OT},i,j} \|x_i - y_j\|_p^p \leq \sum_{i=1}^n \sum_{j=1}^m \pi^\theta_{i,j} \|x_i - y_j\|_p^p$

(*Differentiability.*) This is a direct consequence of Lemma 2 and Lebesgue dominated convergence theorem to invert expectation and derivative.

(*Consistency.*) The first fact is a consequence of Lebesgue dominated convergence theorem. The second one use the expression of the gradient through the variance reduction expression:

$$\nabla_\theta h_\varepsilon(\theta) = \varepsilon^{-1}\mathbb{E}_Z[(h(\theta+\varepsilon Z) - h(\theta))Z] = \mathbb{E}_Z\left[\frac{h(\theta+\varepsilon Z) - h(\theta)}{\varepsilon} Z\right].$$

Hence, again using Lebesgue dominated convergence theorem, we have

$$\lim_{\varepsilon\to 0} \nabla_\theta h_\varepsilon(\theta) = \lim_{\varepsilon\to 0} \mathbb{E}_Z\left[\frac{h(\theta+\varepsilon Z) - h(\theta)}{\varepsilon} Z\right] = \mathbb{E}_Z\left[\lim_{\varepsilon\to 0} \frac{h(\theta+\varepsilon Z) - h(\theta)}{\varepsilon} Z\right].$$

Recognizing the directional derivative of $h(\theta)$ if h is differentiable at θ, we get that

$$\lim_{\varepsilon\to 0} \nabla_\theta h_\varepsilon(\theta) = \mathbb{E}_Z\left[\langle \nabla h(\theta), Z\rangle Z\right] = \nabla h(\theta).$$

(*Distance.*) We assume here that ϕ^θ is injective on $\mathcal{X}$. We split the proof for h (**1.**) and h_ε (**2.**).

1. Proof that $(\mu,\nu) \mapsto h(\theta)(\mu,\nu)$ is a distance over $\mathcal{P}(\mathcal{X})$. The *positivity* comes from the suboptimality of $\pi^\theta(\mu,\nu)$, that is $h(\theta)(\mu,\nu) \geq W_p^p(\mu,\nu) > 0$ if $\mu \neq \nu$ (as W_p is a metric itself). The *symmetry* comes from the fact that $C_{\mu\nu}$ is symmetric and that $\pi^\theta(\mu,\nu) = (\pi^\theta(\nu,\mu))^\top$. Regarding the *well-posedness*, since ϕ^θ is injective, then $W_p(\phi^\theta_\sharp\mu, \phi^\theta_\sharp\mu) = 0$ and $\pi^\theta(\mu,\mu)$ is the identity matrix. Hence,

$$h(\theta)(\mu,\mu) = \langle C_{\mu\mu}, \pi^\theta(\mu,\mu)\rangle = \sum_{i=1}^n\sum_{j=1}^n (C_{\mu\mu})_{ij}\pi^\theta_{ij}(\mu,\mu) = \sum_{i=1}^n (C_{\mu\mu})_{ii} = 0,$$

since for all i, $(C_{\mu\mu})_{ii} = \|x_i - x_i\|_p^p = 0$. Concerning the *triangle inequality*, let $\mu_1,\mu_2,\mu_3 \in \mathcal{P}(\mathcal{X})$. Let us denote

$$\pi^{12} = \pi^\theta(\mu_1,\mu_2) \in \mathbb{R}^{n_1\times n_2}, \quad \pi^{13} = \pi^\theta(\mu_1,\mu_3) \in \mathbb{R}^{n_1\times n_3}, \quad \pi^{23} = \pi^\theta(\mu_2,\mu_3) \in \mathbb{R}^{n_2\times n_3}$$

Using the specific structure of the 1D optimal transport [51], there exists a tensor $\Pi \in \mathbb{R}^{n_1\times n_2\times n_3}$ of order 3 such that admits π^{12}, π^{13} and π^{23} as marginals, that is

$$\begin{array}{lll} \forall i,j, & \pi^{12}_{i,j} & = \sum_{k=1}^{n_3} \Pi_{i,j,k} \\ \forall i,k, & \pi^{13}_{i,k} & = \sum_{j=1}^{n_2} \Pi_{i,j,k} \\ \forall j,k, & \pi^{23}_{j,k} & = \sum_{i=1}^{n_1} \Pi_{i,j,k}. \end{array}$$

Since this structure provides us a "gluing lemma", we continue the proof similarly to the standard proof of the triangular inequality of the Wasserstein distance.

$$\begin{aligned} (h(\theta)(\mu_1,\mu_3))^{1/p} &= \left(\langle C_{\mu_1\mu_3}, \pi^{13}\rangle\right)^{1/p} \\ &= \left(\sum_{i=1}^{n_1}\sum_{k=1}^{n_3} \pi^{13}_{ik}\|x_i - z_k\|_p^p\right)^{1/p} && \text{by definition} \\ &= \left(\sum_{i=1}^{n_1}\sum_{j=1}^{n_2}\sum_{k=1}^{n_3} \Pi_{ijk}\|x_i - z_k\|_p^p\right)^{1/p} && \text{as glue.} \end{aligned}$$

Using that $\|x_i - z_k\|_p^p \leq \|x_i - y_j\|_p^p + \|y_j - z_k\|_p^p$, we get that

$$(h(\theta)(\mu_1,\mu_3))^{1/p} \leq \left(\sum_{i=1}^{n_1}\sum_{j=1}^{n_2}\sum_{k=1}^{n_3} \Pi_{ijk}(\|x_i - y_j\|_p^p + \|y_j - z_k\|_p^p)\right)^{1/p}.$$

Applying now the Minkowski inequality, we obtain that

$$(h(\theta)(\mu_1,\mu_3))^{1/p} \leq \left(\sum_{i=1}^{n_1}\sum_{j=1}^{n_2}\sum_{k=1}^{n_3} \Pi_{ijk}\|x_i - y_j\|_p^p\right)^{1/p} + \left(\sum_{i=1}^{n_1}\sum_{j=1}^{n_2}\sum_{k=1}^{n_3} \Pi_{ijk}\|y_j - z_k\|_p^p\right)^{1/p}.$$

Using the fact that Π has marginals π^{12} and π^{23}, we get that

$$(h(\theta)(\mu_1,\mu_3))^{1/p} \leq \left(\sum_{i=1}^{n_1}\sum_{j=1}^{n_2}\pi_{ij}^{12}\|x_i-y_j\|_p^p\right)^{1/p} + \left(\sum_{j=1}^{n_2}\sum_{k=1}^{n_3}\pi_{jk}^{23}\|y_j-z_k\|_p^p\right)^{1/p}.$$

Hence,

$$(h(\theta)(\mu_1,\mu_3))^{1/p} \leq (h(\theta)(\mu_1,\mu_2))^{1/p} + (h(\theta)(\mu_1,\mu_2))^{1/p}.$$

2. Proof that $(\mu,\nu) \mapsto h_\varepsilon(\theta)(\mu,\nu)$ is a distance over $\mathcal{P}(\mathcal{X})$. The *well-posedness* comes from the fact that $h_\varepsilon(\theta)(\mu,\mu) = \mathbb{E}_Z[h(\theta+\varepsilon Z)(\mu,\mu)] = \mathbb{E}_Z[0] = 0$. The symmetry is also a direct consequence of the *symmetry* of $h(\theta)(\mu,\nu)$ and the *positivity* comes from the fact that the expectation of a positive quantity is positive (understood almost surely on $\mathbb{R}^q$). For the *triangle inequality*, we use the linearity of the expectation: let $\mu_1,\mu_2,\mu_3 \in \mathcal{P}(\mathcal{X})$. Then, for all $\theta \in \mathbb{R}^q$, and for all $z \in \mathbb{R}^q$, using the fact that $h(\theta+\varepsilon z)$ is a distance

$$h(\theta+\varepsilon z)(\mu_1,\mu_3)^{1/p} \leq h(\theta+\varepsilon z)(\mu_1,\mu_2)^{1/p} + h(\theta+\varepsilon z)(\mu_2,\mu_3)^{1/p}.$$

Hence, taking the expectation and using linearity gives that

$$\mathbb{E}_Z[h(\theta+\varepsilon Z)(\mu_1,\mu_3)^{1/p}] \leq \mathbb{E}_Z[h(\theta+\varepsilon Z)(\mu_1,\mu_2)^{1/p}] + \mathbb{E}_Z[h(\theta+\varepsilon Z)(\mu_2,\mu_3)^{1/p}].$$

□

A.2 Algorithm

Algorithm 1 describes a gradient descent method to perform the minimization of h_ε using the Monte-Carlo approximation from Eq. (8). Thus, the overall cost of DGSWP is $O(TN(n+m)\log(n+m) + TN(n+m)d)$

Algorithm 1 Monte-Carlo gradient descent of $h_\varepsilon(\theta)$

Require: $\theta_0 \in \mathbb{R}^q$, step size policy η_t, smoothing parameter $\varepsilon > 0$, number of Monte Carlo samples N, number of iterations T

for $t = 0$ to $T-1$ **do**
  Sample i.i.d. perturbation vectors $z_1,\dots,z_N \sim \mathcal{N}(0,\mathrm{Id}_q)$
  for $k = 1$ to N **do**
    Solve OT problems to obtain $\pi^{\theta_t+\varepsilon z_k}$ and π^{θ_t} ▷ using 1D OT solver
    $g_k \leftarrow \langle C_{\mu\nu}, \pi^{\theta_t+\varepsilon z_k} - \pi^{\theta_t}\rangle$
  $\widehat{\nabla} h_{\varepsilon,N}(\theta_t) \leftarrow \frac{1}{\varepsilon N}\sum_{k=1}^N g_k z_k$ ▷ approximate gradient
  $\theta_{t+1} \leftarrow \theta_t - \eta_t \widehat{\nabla} h_{\varepsilon,N}(\theta_t)$ ▷ update parameter
return θ_T

A.3 Additional results and experiment details

All experiments except the Conditional Flow Matching (CFM) were run on a MacBook Pro M2 Max with 32 GB of RAM. On this machine, Fig. 1 took approximately 3 minutes per run (10,000 iterations), Fig. 3 about 6 minutes for 10 runs (with two models trained sequentially, 1,000 iterations), Fig. 4 required roughly 30 minutes, and Fig. 5 took around 10 minutes in total (all models considered, 10 repetitions). The CFM experiments were dispatched over a GPU cluster composed of GPU-A100 80G, GPU-A6000 48 Go, with a total runtime of 130h for training and inference of all presented models. We observe that I-CFM is approximately 25% faster than the other methods. Meanwhile, OT-CFM, min-DGSWP-CFM (linear), and min-DGSWP-CFM (NN) exhibit comparable runtimes. In practice, the DGSWP batch size (the number of minibatches gathered together to compute DGSWP mappings) serves as a hyperparameter of our method, and we set it to 10 specifically to align the runtime of our approach with that of OT-CFM. We estimate that the total compute time over the course of the project—including experimentation, debugging, and hyperparameter tuning—is approximately two orders of magnitude larger than the reported runtimes for CPU-based experiments, and one order of magnitude larger for the GPU-based experiments.

A.3.1 Hyperparameter settings

We report here the hyperparameter configurations used across the main experiments. Figures 1 and 3 correspond to the same experiment—Fig. 3 highlights early training dynamics, while Fig. 1 depicts results at convergence. The projection network used is a 3-layer MLP with ReLU activations: (with dimensions $2 \to 64 \to 16 \to 1$). Optimization is done using SGD with a learning rate of 0.2; for the variant without variance reduction, a lower learning rate of 0.0002 is used to ensure convergence (cf. Fig. 8 in which the same learning rate is used for both variants). In Figure 4 (gradient flow experiments), we perform 2000 outer flow steps using SGD with a learning rate of 0.01. At each flow step, we execute 20 projection steps (or inner optimization updates when using learnable projectors). For the latter, we use Adam with a learning rate of 0.01. The neural projector for our method is a single-hidden-layer MLP with ReLU activations and He initialization. In Figure 5, which investigates gradient flows on hyperbolic manifolds, we vary the outer learning rate across methods to account for differences in convergence speed: the base learning rate is 2.5, used for HHSW; SW uses a scaled learning rate of 17.5, and DGSWP uses a reduced rate of 0.83. Each flow step is composed of 100 projection or inner optimization steps. For the Conditional Flow Matching (CFM) experiment shown in Figures 6, 9, 10 and Table 1, we adopt the same training hyperparameters as in Tong et al. [56]. For our method specifically, the projection model is a 3-layer fully connected network with SELU activations: $3 \times 32 \times 32 \to 256 \to 256 \to 1$. Its parameters are optimized using Adam with a learning rate of 0.01. We perform 1000 optimization steps for the projection model at initialization, followed by 1 step per CFM training iteration.

A.3.2 Benchmarking min-SWGG and DGSWP (linear) at comparable time budget

Method	Iterations	$\log_{10} W_2^2$	Time (s)
min-SWGG (optim)	500	**-0.81**	3.34
DGSWP (linear)	173	-0.49	3.34
min-SWGG (optim)	1000	**-0.89**	6.67
DGSWP (linear)	345	-0.84	6.67
min-SWGG (optim)	1500	-0.80	10.00
DGSWP (linear)	516	**-1.07**	9.99
min-SWGG (optim)	2000	-0.79	13.33
DGSWP (linear)	687	**-1.28**	13.32

Table 2: Convergence comparison of min-SWGG and DGSWP (linear) methods under comparable time budgets for the gradient flow experiment on the Swiss Roll dataset. Bold values indicate the best performance at each time budget.

In the gradient flow experiments presented in Section 5, methods are compared in terms of convergence as a function of the number of gradient steps. However, when comparing min-SWGG and DGSWP (linear), one might question whether the gradient descent procedure in DGSWP (linear) introduces additional computational overhead.

To investigate this phenomenon, we present in Table 2 a comparison of these methods under fixed time budgets. Specifically, we select the number of iterations for DGSWP (linear) such that the total runtime is less than or equal to the corresponding min-SWGG run. Our results demonstrate that min-SWGG achieves better performance in the early stages of optimization, while DGSWP (linear) becomes more accurate as the computational budget increases.

A.3.3 Additional comparison to Expected Sliced Transport Plans

In the core of the paper, we have considered min-SWGG, SWD and ASWD as our main competitors. We include in this section an additional comparison to Expected Sliced Transport Plans (EST) [36]. For the sake of the comparison, we vary their temperature hyperparameter in the set $\{0, 1, 10\}$ and observe performance both in the Gaussians $\to$ Moons transport problem (Table 3) and the gradient flow experiments (Fig. 7). We observe that larger temperature settings seem to improve EST performance, yet EST remain consistently outperformed by DGSWP.

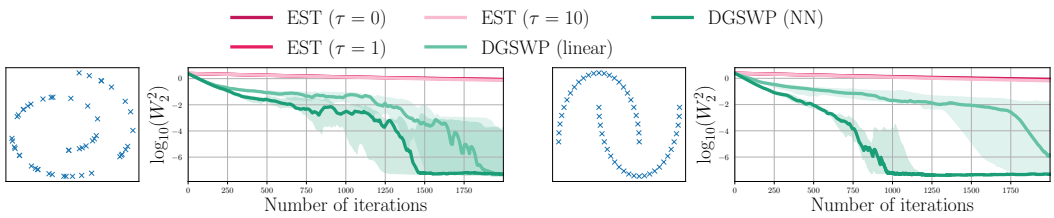


Figure 7: Comparison between DGSWP variants and EST on the gradient flow experiment.

Method	**Cost**
Exact OT	13.50
min-SWGG	22.22
Expected Sliced Transport Plans ($\tau = 0$)	32.28
Expected Sliced Transport Plans ($\tau = 1$)	21.30
Expected Sliced Transport Plans ($\tau = 10$)	20.09
DGSWP (NN)	13.80

Table 3: Comparison of methods by transportation cost. The OT problem is the 8Gaussians to Two Moons from Fig. 1. The cost reported in this table is the associated transportation cost, ie the cost $\langle C_{\mu\nu}, \pi \rangle$ where π is provided by the method at stake.

A.3.4 Additional results for the CFM experiment

Fig.9 presents a set of generated images using I-CFM, OT-CFM and DGSWP (NN) after 400k iterations using the 100-step Euler integrator.

Fig. 10 presents the evolution of the image generation quality during training when using the adaptive-step Dormand-Prince strategy for the integration.

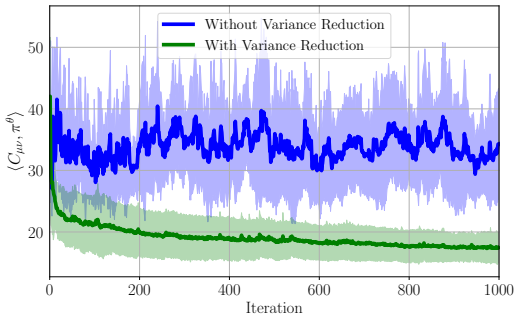
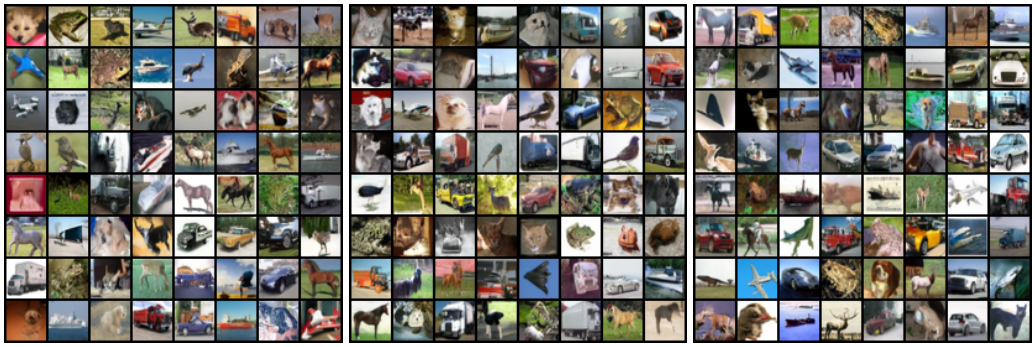


Figure 8: Impact of the variance reduction scheme from Eq. 8. Here, the same learning rate is used for both variants, in which case the variant without variance reduction does not even converge.



(a) I-CFM (b) OT-CFM (c) DGSWP-CFM

Figure 9: Example of generated images after 400k steps.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We provide theoretical but also experimental results that support the claims made in the abstract and introduction. That is to say we "reformulate min-SWGG as a bilevel optimization problem and propose a differentiable approximation scheme to efficiently identify the optimal slice, even in13high-dimensional settings. We furthermore define its generalized extension for accommodating to data living on manifolds. Finally, we demonstrate the practical value of our approach in various applications".

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

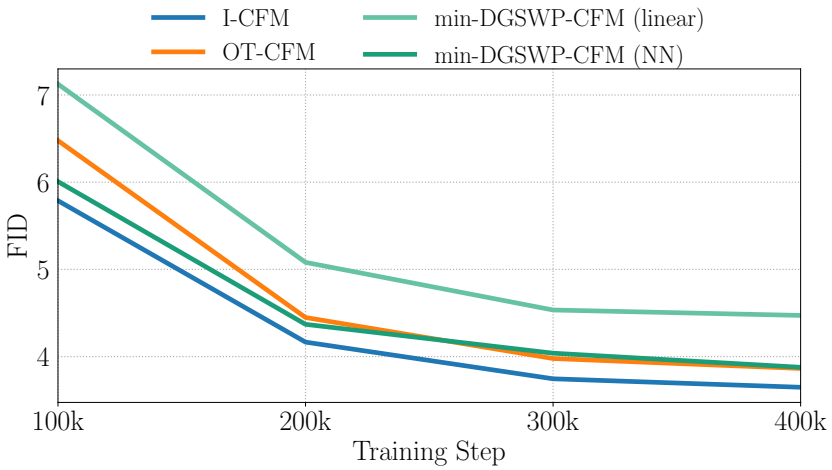


Figure 10: FID as a function of training iterations for various algorithms using adaptive-step DoPri5 sampling.

Justification: We address the main limitations of our work as future works in the conclusion: "Future works will explore the conditions that the projection map ϕ must satisfy to preserve [...] statistical guarantees. Additionally, ensuring the injectivity of ϕ is essential for DGSWP to define a proper distance. Investigating injective neural architectures [...] is a promising direction for future research." We also discuss the computational efficiency into the appendix and describe the full experimental setup.

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: The proofs are given in the supplementary material.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: We provide the entire implementation, that allows running the all set of experiments and reproduce the illustrative figures of the paper, in the supplementary material.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: We provide the entire implementation, that allows running the all set of experiments and reproduce the illustrative figures of the paper, in the supplementary material. We provide the entire commands or notebooks for ease of reproductibility. Datasets used for the experiments are public datasets; for toy examples, code for producing the dataset is also provided.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We present the experimental setting in the core of the paper, and describe thoroughly the parameter values in the supplementary material. The parameters are also given into the code.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: For most of our results, we provide several runs to assess the variability of our results. For the OT-CFM experiments, we run the experiments only once, as in [56], to avoid extensive use of computational resources. Note that, in this case, we do not claim any statistical difference but rather propose a qualitative evaluation of the results.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The type of compute workers that have been used is described into the supplementary material.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We have conducted the research conform in every respect with the NeurIPS code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [NA]

Justification: There is no clear societal impact of the work performed.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: We rely on CIFAR10 and synthetic datasets, which limits the risk of misuse.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: We mainly use the POT library [22] (under MIT licence) and torchCFM [56] (MIT licence). We also use the code of [10] for which no information of licence is provided. All these resources are cited in the paper. We also use the CIFAR10 dataset [32], for which the official page does not mention any licence.

Guidelines:

- The answer NA means that the paper does not use existing assets.

- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: We provide the implementation of the method.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.