A Case for Vanilla SWD: New Perspectives on Informative Slices, Sliced-Wasserstein Distances, and Learning Rates

Anonymous authors
Paper under double-blind review

Abstract

The practical applications of Wasserstein distances (WDs) are constrained by their sample and computational complexities. Sliced-Wasserstein distances (SWDs) provide a workaround by projecting distributions onto one-dimensional subspaces, leveraging the more efficient, closed-form WDs for 1D distributions. However, in high dimensions, most random projections become uninformative due to the concentration of measure phenomenon. Although several SWD variants have been proposed to focus on *informative* slices, they often introduce additional complexity, numerical instability, and compromise desirable theoretical (metric) properties of SWD. Amid the growing literature that focuses on directly modifying the slicing distribution, which often face challenges, we revisit the standard, "vanilla" Sliced-Wasserstein and propose instead to rescale the 1D Wasserstein to make all slices equally informative. Importantly, we show that with an appropriate notion of *slice informativeness*, rescaling for all individual slices simplifies to a single global scaling factor on the SWD. This, in turn, translates to the standard learning rate search for gradient-based learning in common ML workflows. We perform extensive experiments across various machine learning tasks showing that vanilla SWD, when properly configured, can often match or surpass the performance of more complex variants.

1 Introduction

Optimal transport (OT) theory (Villani et al., 2009; Peyré et al., 2019) provides a principled way to compare data distributions by finding an optimal transportation plan that minimizes the expected cost of moving mass between them, leading to the popular Wasserstein distance (WD) central to many learning applications (Khamis et al., 2024). However, the computational complexity of OT solvers poses a significant bottleneck when calculating the WD. In cases of discrete measures or sample-based scenarios, which are common in machine learning, the problem typically reduces to linear programming with time complexity $\mathcal{O}(N^3 \log N)$, space complexity $\mathcal{O}(N^2)$, and sample complexity $\mathcal{O}(N^{-\frac{1}{d}})$, where N is the number of support points and d the data dimensionality. These unfavorable scaling properties, particularly the curse of dimensionality in sample complexity, make WD impractical for many real-world applications. To address these challenges, several approaches have been proposed, including entropic regularized OT (Cuturi, 2013), smooth OT (Blondel et al., 2018; Manole et al., 2024)), and sliced OT (Bonneel et al., 2015).

In particular, the Sliced-Wasserstein distance (SWD) (Rabin et al., 2012; Bonneel et al., 2015) projects high-dimensional distributions onto 1D subspaces and aggregates the closed-form OT solutions in these subspaces. This method is particularly attractive because 1D Wasserstein distances can be computed efficiently with a time complexity of $\mathcal{O}(N \log N)$ and a space complexity of $\mathcal{O}(N)$ for discrete measures. Additionally, SWD provides a metric between probability distributions that retains many desirable properties of the Wasserstein distance (WD), such as being statistically and topologically equivalent to WD, while being more computationally tractable (Nadjahi et al., 2020). Notably, with a dimension-free sample complexity of $\mathcal{O}(N^{-\frac{1}{2}})$, SWD avoids the curse of dimensionality. However, a key drawback of SWD is its projection complexity, which requires exponentially more slices as the data dimensionality increases.

The projection complexity of SWD has motivated several lines of work that aim to enhance the effectiveness of the slicing approach, especially in addressing variance reduction (Nguyen & Ho, 2023), approximation error reduction (Nguyen et al., 2023), and slicing complexity (Kolouri et al., 2019; Deshpande et al., 2019; Nguyen et al., 2020; 2024a; Nguyen & Ho, 2024; Nguyen et al., 2024b). This is particularly relevant in high-dimensional machine learning settings, where data often has supports in low-dimensional subspaces. These SWD variants are data-driven, focusing on identifying the most informative slices for capturing distributional differences in the data. For instance, Max-SW (Deshpande et al., 2019) and DSW (Nguyen et al., 2020) seek to find slices/projections that maximize the differences between the data distributions. GSW (Kolouri et al., 2019), ASW (Chen et al., 2020), and TSW (Tran et al., 2025a) extend SWD by allowing 'non-linear' projections to capture complex data structures. EBSW (Nguyen & Ho, 2024) designs an energy-based slicing distribution that is parameter-free and has the density proportional to an energy function of the projected 1D distance. MSW (Nguyen et al., 2024a) imposes a first-order Markov structure to avoid redundant, independent projections. More recently, RPSW (Nguyen et al., 2024b) proposes using the normalized differences between random samples from the two distributions to ensure that the projections are sampled from the subspace in which the data resides. BOSW (Acharya & Hyde, 2025) uses Bayesian optimization to select informative slices in a sample-efficient manner. These methods improve the performance of SW in various downstream tasks and have significantly expanded the tools at our disposal for both researchers and practitioners alike. Nonetheless, these elegant extensions also come with increased computational cost, numerical instability, complicated design choices, and often lose the metricity of the SWD.

In this paper, we argue that the standard SWD, with proper hyperparameters, can often match or surpass the performance of more complex variants in many learning tasks while retaining its simplicity and theoretical guarantees. Our key insight is that when d-dimensional data have k-dimensional supports, where $k \ll d$, almost all random slices $\theta \sim \mathcal{U}(\mathbb{S}^{d-1})$ can be decomposed into an *informative* component $\theta_D \in \mathbb{R}^k$ within the data subspace and its orthogonal complement $\theta_D^{\perp} \in \mathbb{R}^{d-k}$. This implies most slices still carry relevant information for distinguishing distributions, proportional to $\|\theta_D\|$. By appropriately scaling the distance per slice, we obtain better gradients for learning. In expectation, we show that, with our defined notion of *informativeness*, scaling for all slices (based on their informativeness) simplifies to scaling the SWD by a single scalar factor. In practice, for gradient-based learning, we show empirically that finding an appropriate learning rate is equivalent to getting informative slices **for free**. This allows the classical SWD to adapt to the data's intrinsic dimensionality without explicitly limiting the computation to the subspace. We provide theoretical justification and empirical evidence, offering a fresh perspective on SWD, particularly in high-dimensional settings. Our insights uncover why the standard SWD has remarkably robust performance in distribution-based learning tasks and open up interesting research vectors for investigating other task-specific data assumptions and associated notion of informative slices.

By revisiting the standard SWD with these insights, we elucidate the "performance gap" between the original formulation and recent variants in the existing literature. We emphasize that our work does not diminish the valuable contributions of these variants, which have greatly advanced our understanding of SWD. Rather, we offer a complementary perspective that highlights the potential of the standard SWD when properly integrated into learning tasks. Along that line, we remark that the related body of specialized methods that respects the data geometry (Rabin et al., 2011; Bonet et al., 2022; Martin et al., 2023; Quellmalz et al., 2023; Bonet et al., 2024; Tran et al., 2024; 2025b) remains valuable when the manifold constraint on the data is readily known.

In common ML settings where data is (nearly) supported on a k-dimensional subspace embedded in a d-dimensional space, our findings can be summarized as follows:

- We introduce the ϕ-weighting formulation, unifying various SWD variants. In this framework, we propose reweighting all one-dimensional Wasserstein distances based on *slice informativeness* instead of directly modifying the slicing distribution, as commonly done in the literature. We show that with an appropriate notion of *slice informativeness*, in expectation, this leads to a Subspace Sliced-Wasserstein variant that differs from the standard SWD by a scalar factor.

- Our findings reduce solving the problem of *non-informative slices* to the learning-rate search for the classic SWD, a process that is already standard in ML workflows. In other words, we get *informative slices* for free.
- We perform a comprehensive learning-rate sweep across a range of experiments, including gradient flow (on 3 classic toy datasets, MNIST images, CelebA images), color transfer (3 sets of images), and deep generative modeling on the FFHQ dataset (unconditional generation and unpaired translation with SW). We show that the classic SWD, with appropriate hyperparameters, performs competitively with more advanced methods in these settings.

Notations. Throughout the paper, we let $\mathbb{R}^d$ denote a d-dimensional inner product space, and we denote the unit hypersphere in this space by $\mathbb{S}^{d-1} = \{\theta \in \mathbb{R}^d : \|\theta\|_2 = 1\}$. Additionally, we denote by $\mathcal{P}(\mathbb{R}^d)$ the set of probability measures on $\mathbb{R}^d$ endowed with the σ-algebra of Borel sets, and by $\mathcal{P}_p(\mathbb{R}^d) \subset \mathcal{P}(\mathbb{R}^d)$ the subset of those measures with finite p-th moments. For a measurable function $f : \mathbb{R}^d \to \mathbb{R}$ defined by $f(x) = \theta^\top x$ such that $\theta \in \mathbb{S}^{d-1}$, we denote the pushforward of a measure $\mu \in \mathcal{P}(\mathbb{R}^d)$ through f as $f_\#\mu$. More generally, for any $w \in \mathbb{R}^d$, we write $w_\#\mu$ for the pushforward of μ under the map $x \mapsto w^\top x$. In particular, when $w \in \mathbb{S}^{d-1}$ this coincides with the standard 1D projection onto a unit direction.

2 Background: The Sliced-Wasserstein Distance

Let $\mu \in \mathcal{P}_p(\mathbb{R}^d)$ and $\nu \in \mathcal{P}_p(\mathbb{R}^d)$ be two probability measures of interest.

Wasserstein distance (WD). The p-WD between μ and ν is:

$$W_p^p(\mu, \nu) = \inf_{\pi \in \Pi(\mu,\nu)} \int_{\mathbb{R}^d \times \mathbb{R}^d} \|x - y\|_p^p \, d\pi(x, y), \tag{1}$$

with $\Pi(\mu, \nu) = \{\pi \in \mathcal{P}_p(\mathbb{R}^d \times \mathbb{R}^d) : \pi(A \times \mathbb{R}^d) = \mu(A), \quad \pi(\mathbb{R}^d \times A) = \nu(A)\}$ for all measurable sets $A \subset \mathbb{R}^d$. In one dimension ($d = 1$), the p-WD admits the following closed-form solution:

$$W_p^p(\mu, \nu) = \int_0^1 |F_\mu^{-1}(z) - F_\nu^{-1}(z)|^p \, dz, \tag{2}$$

where F_μ, F_ν are the cumulative distribution functions (CDF) of μ and ν, respectively, and $F_\mu^{-1}(z) := \inf\{t \in \mathbb{R} : F_\mu(t) \geq z\}$ (and similarly for F_ν^{-1}). For empirical measures, Equation 2 becomes a Monte Carlo sum that can be calculated by averaging $|x_{(i)} - y_{(i)}|^p$ between sorted samples. In general, this translates to a highly favorable time complexity of $\mathcal{O}(N \log N)$ and gives rise to the following Sliced-Wasserstein distance.

Sliced-Wasserstein distance (SWD). The SWD between μ and ν is defined as:

$$SW_p(\mu, \nu; \sigma) := \left(\mathbb{E}_{\theta \sim \sigma}\left[W_p^p(\theta_\#\mu, \theta_\#\nu)\right]\right)^{\frac{1}{p}} \tag{3}$$

where $\sigma \in \mathcal{P}(\mathbb{S}^{d-1})$ is the reference measure for slicing vector θ. In the default setting, σ is the uniform distribution, denoted as $\sigma = \mathcal{U}(\mathbb{S}^{d-1})$, and we use $SW_p(\mu, \nu)$ to denote $SW_p(\mu, \nu; \sigma)$ for simplicity. The intractable expectation in Equation 3 admits a Monte Carlo estimator:

$$SW_p(\mu, \nu; \sum_{l=1}^{L} \frac{1}{L}\delta_{\theta_l}) = \left(\frac{1}{L}\sum_{l=1}^{L} W_p^p\big((\theta_l)_\# \, \mu, \, (\theta_l)_\# \, \nu\big)\right)^{\frac{1}{p}}, \tag{4}$$

where $\{\theta_l\}_{l=1}^L \overset{\text{i.i.d.}}{\sim} \sigma$. The Monte Carlo estimator's error decreases as $\frac{1}{\sqrt{L}}$, where L is the number of slices. The main issue becomes how much one can simulate (for large d), which proves to be challenging since most slices are known to be non-informative for data supported in low dimensions. As a result, $SW_p(\mu, \nu; \sum_{l=1}^L \frac{1}{L}\delta_{\theta_l})$ often underestimates the distance between μ and ν in practice. Moreover, L should be sufficiently large compared to d, which is undesirable since the time complexity of SW scales linearly with L.

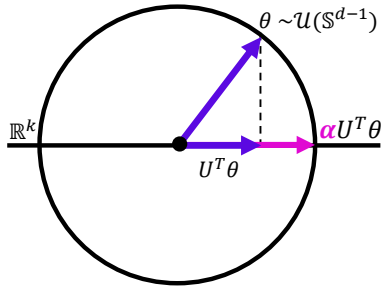


Figure 1: An illustration for rescaling the 1D Wasserstein based on slice *informativeness*. Here, the notion of informativeness can be defined as the slice alignment with the (principal) data subspace.

3 Motivating Related Work

Given this context of SWD, we pause to highlight two sets of works that particularly motivate the present manuscript:

Subspace-constrained OT. Recent works propose computing OT in lower-dimensional subspaces (Paty & Cuturi, 2019; Bonet et al., 2021b; Muzellec & Cuturi, 2019) to improve both efficiency and robustness for high-dimensional data. **1) Subspace Detours** (Bonet et al., 2021b; Muzellec & Cuturi, 2019) constrain transport plans to be optimal when projected onto a subspace. This allows efficient extension of low-dimensional transport solutions to the full space. **2) Subspace Robust** (Paty & Cuturi, 2019) considers the worst-case transport cost over all low-dimensional projections. This can be computed by minimizing $S_k^2(\mu,\nu) = \min_{\pi\in\Pi(\mu,\nu)} \sum_{l=1}^k \lambda_l(V_\pi)$ where $V_\pi := \int (x-y)(x-y)^T d\pi(x,y)$ is the 2nd order displacement matrix for a coupling π, and $\lambda_l(V_\pi)$ its l-th largest eigenvalue.

Gaussian Sliced-Wasserstein. Earlier works ((Sudakov, 1978; Diaconis & Freedman, 1984; Reeves, 2017)) establish several central limit theorems showing that under mild conditions, low-dimensional projections of high-dimensional data converge to Gaussians. Nadjahi et al. (2021) leverages this concentration of measure phenomenon and shows that the sliced-Wasserstein distance with Gaussian projection vectors, defined by $\widetilde{SW}_p^p(\mu,\nu) := \int_{\mathbb{R}^d} W_p^p(\theta_\#\mu, \theta_\#\nu)\, d\gamma_d(\theta)$ with $\gamma_d = \mathcal{N}(0, \frac{1}{d}I_d)$, is proportional to the classical SWD: $\widetilde{SW}_p^p(\mu,\nu) = C_{d,p}\, SW_p^p(\mu,\nu;\mathcal{U}(\mathbb{S}^{d-1}))$, where $C_{d,p} = \left(\frac{2}{d}\right)^{p/2} \frac{\Gamma\left(\frac{d}{2}+\frac{p}{2}\right)}{\Gamma\left(\frac{d}{2}\right)}$, and propose an efficient approximation of the SWD without simulation.

4 A Subspace Perspective on Sliced-Wasserstein Distances

Many machine learning problems involve high-dimensional data that has a low-dimensional structure. Formally, this phenomenon, known as the *manifold hypothesis*, states that for a dataset $X \subset \mathbb{R}^d$, there exists a k-dimensional manifold $\mathcal{M}$ where $k \ll d$ such that X approximately lies on $\mathcal{M}$ (Fefferman et al., 2016). For instance, rigorous dimensionality estimation methods applied to common datasets like MS-COCO (Lin et al., 2014) and ImageNet (Deng et al., 2009) suggest $k < 50$ (Pope et al., 2021), despite their ambient dimension d being orders of magnitude larger. While these manifolds are generally nonlinear, they admit local linear approximations via their tangent spaces. Moreover, in practice, data features typically have strong linear correlations, allowing techniques like Principal Component Analysis (PCA) to identify a principal subspace that captures most of the data variance.

Motivating challenge. This subspace approximation is particularly relevant in the context of SWD. It is known from Kolouri et al. (2019) that when slices θ are sampled uniformly from $\mathbb{S}^{d-1}$, the probability that a random slice is nearly orthogonal to any fixed direction increases exponentially with dimension. Specifically, for a unit vector x_0 representing a principal direction in the data subspace,

$$\Pr\left(|\langle\theta, x_0\rangle| \leq \epsilon\right) > 1 - e^{-d\epsilon^2}, \quad \theta \sim \mathcal{U}(\mathbb{S}^{d-1}). \tag{5}$$

This concentration of measure phenomenon implies that as dimensionality d grows, most random slices become nearly orthogonal to the principal directions of the data subspace. Consequently, the corresponding 1D Wasserstein distances contribute minimally to the SWD. This effect, which we refer to as *slice non-informativeness*, limits the effectiveness of SWD in high-dimensional spaces.

Current approaches: Designing the slicing distribution. Sampling-based methods seek to define a non-uniform slicing distribution that focuses on *discriminative* directions. Optimization-free methods (Nguyen & Ho, 2024; Nguyen et al., 2024b) are objectively faster but do not yield true metrics. Other methods (Nguyen et al., 2020; 2024a) yield proper metrics but are more computationally expensive due to the optimization involved. In the limit, the Max variants use discrete slicing distributions that require global optimality to be metrics, which is generally intractable in practice. Empirically, without careful hyperparameter tuning, the different variants face numerical instability in the larger learning rate regimes, likely because of the overemphasis on directions with large projected distances.

A novel perspective: Rescaling 1D Wasserstein distances. These challenges in directly redefining the slicing distribution motivate us to take a second look at the conventional wisdom of sampling informative slices. We propose an alternative formulation that reweights each 1D Wasserstein based on the *informativeness* of the corresponding slice/projecting direction (see Figure 1 for illustration). By defining the notion of an *informative slice* based on its alignment with the effective data subspace, we demonstrate that it is possible to reweight all slices by *a global constant* on the SWD. This maintains the efficiency and theoretical properties of the classical Sliced-Wasserstein distance; in other words, by simply rescaling by a global constant, one can make SWD often perform comparably to the aforementioned variants of SWD that go to great lengths to design slicing distributions to accelerate SWD estimation[1]. The implications of this finding for using SWD in gradient-based learning will be discussed in subsequent sections.

To formalize this approach, we introduce the following assumption and definitions:

Assumption 4.1 (Effective Subspace Structure)**.** *Let $\mu^d, \nu^d \in \mathcal{P}_p(\mathbb{R}^d)$ be probability measures. We say (μ^d, ν^d) has k-dimensional effective structure if:*

1. *There exists a semi-orthogonal matrix $U \in \mathbb{R}^{d\times k}$ (i.e., $U^TU = I_k$) such that*

$$supp(\mu^d), supp(\nu^d) \subset V_k := col\text{-}span(U).$$

2. *k is minimal, meaning that there does not exist any $U' \in \mathbb{R}^{d\times k'}$ with $k' < k$ s.t (1) holds.*

*We refer to V_k as the **effective subspace** (ES) of μ^d, ν^d, and k as their **effective dimensionality** (ED).*

Note: In Appendix A.9, we also discuss results when this assumption does not hold.

All results below remain valid if $\text{supp}(\mu^d)$ and $\text{supp}(\nu^d)$ lie in a common affine subspace $m + V_k$ for some $m \in \mathbb{R}^d$, since both W_p and SW_p are invariant under a common translation of μ^d and ν^d. For simplicity we state Assumption 4.1 in the linear case.

Informative slices. To unify the diverse slicing strategies in the literature, we introduce the concept of a slice informativeness function $\phi : \mathbb{S}^{d-1} \to \mathbb{R}_+$ that quantifies the relevance of a projection direction $\theta \in \mathbb{S}^{d-1}$ (e.g., by assigning a non-negative value) for distinguishing distributions. Existing variants typically define ϕ as a functional of the input measure μ and ν. For instance, Max-SW (Deshpande et al., 2019), Markovian SW Nguyen et al. (2024a), and EBSW (Nguyen & Ho, 2024) implicitly use

$$\phi_{\mu,\nu}(\theta) = W_p^p(\theta_\#\mu, \theta_\#\nu), \tag{6}$$

to measure the informativeness of θ. On the other hand, RPSW (Nguyen et al., 2024b) implicitly uses

$$\phi_{\mu,\nu}(\theta; \mu, \nu, \gamma_\kappa) = \mathbb{E}_{(X,Y)\sim\mu\times\nu}[\gamma_\kappa(\theta; P_{\mathbb{S}^{d-1}}(X - Y))], \tag{7}$$

[1]We remark that in the world of Bayesian optimization (BO), an analogous discovery has recently been made. Hvarfner et al. (2024) found that a simple global rescaling of the Gaussian prior lengthscale in vanilla BO allows that algorithm to perform well in high dimensions, despite the conventional wisdom that specialized variants of BO for higher dimensions are needed.

where γ_κ is a location-scale distribution (e.g., vMF (Fisher, 1953; Mardia & Jupp, 2009)) and $P_{\mathbb{S}^{d-1}}$ is the projection onto $\mathbb{S}^{d-1}$.

While these data-dependent definitions are expressive, they inherently require evaluating 1D distances or expectations for slices that may effectively be discarded, incurring unnecessary computational overhead (Nguyen et al., 2024a). Furthermore, coupling the slicing distribution to the input measures introduces complex dependencies that complicate the verification of metric properties, such as the triangle inequality (Nguyen & Ho, 2024). To address this limitation, we propose a notion of informativeness grounded in Assumption 4.1 that decouples slice selection from transport cost.

Definition 4.2. *Let $V_k = \mathrm{col}(U)$, where $U \in \mathbb{R}^{d\times k}$ has orthonormal columns (i.e., $U^\top U = I_k$). We define the effective subspace (ES)-aligned informativeness function $\phi_U : \mathbb{S}^{d-1} \to [0, 1]$ by*

$$\phi_U(\theta) = \|U^\top \theta\|_2. \tag{8}$$

Geometrically, $\phi_U(\theta)$ measures the magnitude of the projection of θ onto the effective data subspace V_k. A value of $\phi_U(\theta) \approx 1$ indicates that the slice captures maximum variation within the data support while $\phi_U(\theta) \approx 0$ implies orthogonality to the data. Formally:

Lemma 4.3. *Let $U \in \mathbb{R}^{d\times k}$ satisfy $U^\top U = I_k$ and define $\phi_U(\theta) = \|U^\top\theta\|_2$ for $\theta \in \mathbb{S}^{d-1}$. Then:* ***(a)*** *$0 \le \phi_U(\theta) \le 1$ for all $\theta \in \mathbb{S}^{d-1}$;* ***(b)*** *$\phi_U(\theta) = 1$ iff $\theta \in \mathrm{span}(U) \cap \mathbb{S}^{d-1}$ and $\phi_U(\theta) = 0$ iff $\theta \perp \mathrm{span}(U)$; and* ***(c)*** *for any orthogonal matrix $Q \in \mathbb{R}^{k\times k}$, $\phi_{UQ}(\theta) = \phi_U(\theta)$.*

4.1 The ϕ-weighting formulation

Starting from Equation 3, we propose a general formulation for reweighting slice contributions:

$$\widetilde{SW}_p(\mu, \nu; \sigma, \rho) = \left(\int_{\mathbb{S}^{d-1}} \underbrace{\rho(\phi(\theta)) W_p^p(\theta_\# \mu, \theta_\# \nu)}_{\text{Reweighted contribution}} d\sigma(\theta) \right)^{\frac{1}{p}}, \tag{9}$$

where $\rho : [0, 1] \to \mathbb{R}_+$ is a **ϕ-weighting function** that rescales the contribution of each slice based on a general informativeness function ϕ. This formulation preserves metricity under mild conditions (See Proposition A.5 in the Appendix).

Example 4.4. *If the goal were to reweight all slices to be treated as informative*[2]*, an appropriate choice for ρ could be the multiplicative inverse of $\phi(\theta)$ (more informative slices are scaled to be smaller, and vice versa). That is,*

$$\rho(\phi(\theta)) = \begin{cases} \dfrac{1}{\phi(\theta)^p}, & \text{if } \phi(\theta) > 0, \\ 0, & \text{if } \phi(\theta) = 0. \end{cases} \tag{10}$$

Remark 4.5. *Equation 9 notably does not rely on Assumption 4.1. By defining the appropriate $\rho_\phi(\cdot)$ and $\phi(\cdot)$, the ϕ-weighting formulation can be seen as a unifying formulation that recovers different SW variants.*

- *We set $\rho \equiv 1$ and obtain the classical Sliced-Wasserstein distance.*
- *We set $\phi_{\mu,\nu}(\theta) = W_p^p(\theta_\#\mu, \theta_\#\nu)$ and $\rho \equiv 1$, and take $\sigma = \delta_{\theta^\star}$ where $\theta^\star \in \arg\max_{\theta\in\mathbb{S}^{d-1}} \phi_{\mu,\nu}(\theta)$, and recover Max-SW (Deshpande et al., 2019) (when a maximizer exists).*
- *We set $\phi_{\mu,\nu}(\theta) = W_p^p(\theta_\#\mu, \theta_\#\nu)$, $\rho(r) = \frac{f(r)}{\int_{\mathbb{S}^{d-1}} f(W_p^p(\theta_\#\mu,\theta_\#\nu))\, d\sigma(\theta)}$, $\sigma = \mathcal{U}(\mathbb{S}^{d-1})$ where $f : [0, \infty) \to (0, \infty)$ is an increasing energy function (e.g., $f(x) = e^x$), and recover EBSW (Nguyen & Ho, 2024).*

[2]Conceptually, we reiterate that SWD often underestimates the distance between two distributions because when slices are not aligned with the data subspace, they are implicitly underweighted based on their misalignment (as will be introduced in Section 4.2), even though those slices are in general informative (they contain some component aligned with the data subspace, when they are not totally orthogonal to it; see Figure 1). To rectify this inherent trait of vanilla SWD, one could consider a desirable reweighting to be one that treats all slices as equally informative. Subsequent sections will further motivate this choice.

- *We set* $\phi_{\mu,\nu}(\theta) = \mathbb{E}_{(X,Y)\sim\mu\times\nu}[\gamma_\kappa(\theta; P_{\mathbb{S}^{d-1}}(X-Y))]$, $\rho(r) = \frac{r}{\int_{\mathbb{S}^{d-1}} \phi_{\mu,\nu}(\theta)\, d\sigma(\theta)}$, $\sigma = \mathcal{U}(\mathbb{S}^{d-1})$, *where* γ_κ *is a location-scale distribution with parameter* κ, *and recover RPSW (Nguyen et al., 2024b).*

4.2 Misaligned random projections are implicitly downweighted by a scalar

Under Assumption 4.1, we will show that the 1D Wasserstein corresponding to each random projection is weighted by a scalar related to the (ES-aligned) informativeness of that projection.

The case for 1D effective subspaces. Let $V_1 = \text{span}(u)$ where $u \in \mathbb{S}^{d-1}$, and suppose $\text{supp}(\mu^d), \text{supp}(\nu^d) \subset V_1$. Given $\theta \in \mathbb{S}^{d-1}$, we can decompose it uniquely as $\theta = \theta_{V_1} + \theta_{V_1^\perp}$, where $\theta_{V_1} = (u^\top\theta)u$ and $\theta_{V_1^\perp} \perp V_1$. For any $x \in V_1$, we have $x = (x^\top u)u$, and $\theta^\top x$ can thus be decomposed as:

$$\theta^\top x = (\theta_{V_1} + \theta_{V_1^\perp})^\top x = \theta_{V_1}^\top x = (u^\top\theta)\,(u^\top x). \tag{11}$$

This implies that for any slice θ, the projected distributions $\theta_\#\mu^d$ and $\theta_\#\nu^d$ are equivalent (up to scaling) to the distributions obtained by projecting μ^d and ν^d onto u. Specifically:

$$W_p^p(\theta_\#\mu^d, \theta_\#\nu^d) = |u^\top\theta|^p W_p^p(u_\#\mu^d, u_\#\nu^d). \tag{12}$$

Thus, misaligned slices (projections) θ, with $|u^T\theta| < 1$, implicitly are downweighted: $W_p^p(\theta_\#\mu^d, \theta_\#\nu^d) < W_p^p(u_\#\mu^d, u_\#\nu^d)$.

Generalizing to higher-dimensional effective subspaces. We extend the idea from one dimension to a k-dimensional subspace V_k and investigate how the reweighting function $\rho_\phi(\phi_U(\theta)) = \|U^\top\theta\|^{-p}$ adjusts the contributions of slices in higher dimensions.

Proposition 4.6. *Under Assumption 4.1, let* $\mu^k = (U^\top)_\#\,\mu^d$ *and* $\nu^k = (U^\top)_\#\,\nu^d$ *be the pushforward measures in* $\mathbb{R}^k$. *Then, for any* $\theta^d \in \mathbb{S}^{d-1}$, *we have that:*

$$W_p^p(\theta^d_\#\mu^d, \theta^d_\#\nu^d) = W_p^p((U^\top\theta^d)_\#\mu^k, (U^\top\theta^d)_\#\nu^k) = \|U^\top\theta^d\|^p W_p^p(\theta^k_\#\mu^k, \theta^k_\#\nu^k). \tag{13}$$

where $\theta^k = \frac{U^\top\theta^d}{\|U^\top\theta^d\|}$ *with convention* $\theta^k = 0_k$ *if* $\|U^\top\theta^d\| = 0$.

Furthermore, we have that:

$$SW_p^p\left(\mu^k, \nu^k; \frac{1}{L}\sum_{l=1}^L \delta_{\theta_l^k}\right) = \widetilde{SW}_p^p\left(\mu^d, \nu^d; \frac{1}{L}\sum_{l=1}^L \delta_{\theta_l^d}, \rho\right) \tag{14}$$

$$SW_p^p\left(\mu^k, \nu^k\right) = \widetilde{SW}_p^p\left(\mu^d, \nu^d; \mathcal{U}(\mathbb{S}^{d-1}), \rho\right) \tag{15}$$

We adopt the convention $\frac{1}{0}\cdot 0 = 0$ *in Equation 14 if* $\|U^\top\theta_l^d\| = 0$.

The proof is in Appendix A.4.

Implicit downweighting. Under the conditions of Proposition 4.6, each slice contribution is implicitly downweighted by $\|U^T\theta^d\|^p$. That is, for any $\theta^d \in \mathbb{S}^{d-1}$, we have that $W_p^p(\theta^d_\#\mu^d, \theta^d_\#\nu^d) \leq W_p^p(\mu^k, \nu^k)$. Moreover, the downweighting is maximal if $\theta^d \perp \text{span}(U)$ and vanishes if $\theta^d \in \text{span}(U) \cap \mathbb{S}^{d-1}$.

Rescaling to equalize informativeness. Assumption 4.1 gives rise to the fact that each one-dimensional Wasserstein distance $W_p^p(\theta^d_\#\mu^d, \theta^d_\#\nu^d)$ is implicitly downweighted by $\|U^\top\theta^d\|^p$. This observation naturally fits into the proposed ϕ-weighting formulation, as there is an implicit scaling factor associated with each slice. To counteract it and make all slices equally (ES-aligned) informative, we use the reciprocal weighting function (Equation 10) to compensate for the implicit downweighting of misaligned slices. Then, we have that

$$\tilde{W}_p^p(\theta^d_\#\mu^d, \theta^d_\#\nu^d) = \begin{cases} W_p^p(\theta^k_\#\mu^k, \theta^k_\#\nu^k), & \text{if } \phi_U(\theta^d) > 0, \\ 0, & \text{if } \phi_U(\theta^d) = 0. \end{cases} \tag{16}$$

where $\theta^k = \frac{U^\top\theta^d}{\|U^\top\theta^d\|}$.

4.3 Subspace Sliced-Wasserstein is rescaled Sliced-Wasserstein

In this section, we will show that the generalized notion of informative slices (as defined in Section 4.2) becomes particularly advantageous for equalizing slice informativeness.

Starting from Equation 13, we integrate both sides over $\theta^d \in \mathbb{S}^{d-1}$ with respect to the uniform measure $\sigma(\theta^d)$ and obtain

$$SW_p^p(\mu^d, \nu^d) = \int_{\mathbb{S}^{d-1}} W_p^p(\theta^d_\# \mu^d, \theta^d_\# \nu^d)\, d\sigma(\theta^d) = \int_{\mathbb{S}^{d-1}} \|U^\top \theta^d\|^p W_p^p\left(\theta^k_\# \mu^k, \theta^k_\# \nu^k\right)\, d\sigma(\theta^d). \tag{17}$$

Note that θ^k depends on θ^d, and the distribution of θ^k induced by $\theta^d \sim \sigma$ is uniform over $\mathbb{S}^{k-1}$. We introduce the change of variables from θ^d to θ^k and express the integral in terms of θ^k:

$$SW_p^p(\mu^d, \nu^d) = \int_{\mathbb{S}^{k-1}} dT_\# \sigma(\theta^k) \times W_p^p\left(\theta^k_\# \mu^k, \theta^k_\# \nu^k\right) \times \left(\int_{\theta^d : \frac{U^\top \theta^d}{\|U^\top \theta^d\|} = \theta^k} \|U^\top \theta^d\|^p \, d\sigma(\theta^d | \theta^k) \right), \tag{18}$$

where $\sigma(\cdot|\theta^k)$ is the conditional distribution of θ^d, and $T : \theta \mapsto \frac{U^\top \theta}{\|U^\top \theta\|}$ is the mapping from θ^d to θ^k.

The inner integral over θ^d can be evaluated as a scaling factor $C_{d,k}$ dependent on σ, θ^k, U. When $\sigma = \mathcal{U}(\mathbb{S}^{d-1})$, $C_{d,k}$ is invariant for all θ^k.

Substituting back into Equation 18, and letting $\sigma_k = T_\# \sigma = \mathcal{U}(\mathbb{S}^{k-1})$ denote the distribution of θ^k, we obtain

$$SW_p^p(\mu^d, \nu^d) = C_{d,k} \int_{\mathbb{S}^{k-1}} W_p^p\left(\theta^k_\# \mu^k, \theta^k_\# \nu^k\right)\, d\sigma_k(\theta^k). \tag{19}$$

Since $\sigma_k(\theta^k)$ integrates to 1 over $\mathbb{S}^{k-1}$, and $W_p^p\left(\theta^k_\# \mu^k, \theta^k_\# \nu^k\right)$ is integrated over all θ^k, we can express the right-hand side as $C_{d,k} \cdot SW_p^p(\mu^k, \nu^k; \sigma_k)$. Intuitively speaking, this means the *loss of information*[3] is due to an implicit constant factor on $SW_p^p(\mu^d, \nu^d)$, which we denote as the **Effective Subspace Scaling Factor** (ESSF). Thus, rescaling the one-dimensional Wasserstein for all slices via Equation 16 is mathematically equivalent to simply multiplying the SWD by the reciprocal of the ESSF. We proceed further to make this connection explicit by the following theorem.

Theorem 4.7 (Effective Subspace Scaling Factor)**.** *Let $\mu^d, \nu^d \in \mathcal{P}(\mathbb{R}^d)$ satisfy Assumption 4.1, and define $\mu^k = (U^\top)_\# \mu^d$ and $\nu^k = (U^\top)_\# \nu^d$. Then we have that*

$$SW_p^p(\mu^d, \nu^d) = \frac{C_k}{C_d} \cdot SW_p^p(\mu^k, \nu^k), \tag{20}$$

where $C_d = 2^{p/2} \frac{\Gamma\left(\frac{d}{2} + \frac{p}{2}\right)}{\Gamma\left(\frac{d}{2}\right)}$ and C_k is defined analogously, with Γ denoting the Gamma function.

When $k < d$, assuming $\|U^\top \theta^d_l\| \neq 0$ is reasonable since $\mathcal{U}(\mathbb{S}^{d-1})(\{\theta \in \mathbb{S}^{d-1} : U^\top \theta = 0\}) = 0$.

The proof is in Appendix A.4.

While Theorem 4.7 establishes the exact scaling law in expectation, we now define an empirical estimator $\widehat{ESSF}(L)$ and bound its convergence to show the property holds for the Monte Carlo estimates used in practical settings (finite numbers of slices).

Proposition 4.8. *Let $\mu^d, \nu^d \in \mathcal{P}(\mathbb{R}^d)$ satisfy Assumption 4.1. Consider the empirical estimator $\widehat{ESSF}(L)$ defined as:*

$$\widehat{ESSF}(L) = \frac{1}{L} \sum_{l=1}^{L} \|U^\top \theta^d_l\|^p, \tag{21}$$

[3]By loss of information, we mean the fact that vanilla SWD reduces the weight of a slice that is not aligned with the data subspace; aggregating over all the slices, this leads to the conclusion computing the SWD in the ambient dimension d gives a value that is in general less than computing the SWD in the subspace dimension k (note that $C_{d,k} \leq 1$).

where $\{\theta_l^d\}_{l=1}^L \overset{i.i.d.}{\sim} \mathcal{U}(\mathbb{S}^{d-1})$. *We have that:*

1. $\mathbb{E}[\widehat{ESSF}(L)] = \frac{C_k}{C_d}$ *and* $Var(\widehat{ESSF}(L)) = \mathcal{O}(\frac{1}{L})$.

2. *Let*
$$\epsilon_L = \left| SW_p^p\Big(\mu^d, \nu^d; \frac{1}{L}\sum_{l=1}^L \delta_{\theta_l^d}\Big) - \widehat{ESSF}(L) \cdot SW_p^p\Big(\mu^k, \nu^k; \frac{1}{L}\sum_{l=1}^L \delta_{\theta_l^k}\Big)\right|.$$
Then $\epsilon_L \xrightarrow{a.s.} 0$ *as* $L \to \infty$.

3. *There exists a constant* $K > 0$ *depending only on* μ^d *and* ν^d *such that for any* $\delta > 0$, *we have*
$$\mathbb{P}(\epsilon_L < \delta) \;\geq\; 1 \;-\; \exp\Big(-\frac{\delta^2 L}{K^2}\Big).$$

The proof of this proposition is in Appendix A.6.

In Section 5.1, we provide empirical results showing how the variance of $\widehat{ESSF}(L)$ changes with L.

4.4 Informative slicing via learning rate search in first-order optimization

While Assumption 4.1 is reasonable as an effective model for real-world data, in gradient-based learning, the SWD objective is evaluated on empirical measures supported on finite minibatches. At each iteration, the source and target supports contain at most $2B$ points and therefore lie in a low-dimensional affine span.

Remark 4.9. *Let* $\{x_i\}_{i=1}^{2B} \subset \mathbb{R}^d$ *be a minibatch of* $2B$ *samples (*B *from source,* B *from target). Let* $X = [x_1, \ldots, x_{2B}] \in \mathbb{R}^{d\times 2B}$ *be the corresponding data matrix. Then the samples lie in an affine subspace of dimension at most* $k \leq \min\{2B-1, d\}$*; after centering, they lie in a linear subspace with the same dimension bound.*

The relevance of this observation is that, in first-order optimization such as stochastic gradient descent (SGD), an SWD objective is differentiated with respect to the locations of these empirical support points. Consequently, the gradients driving SGD depend on how the projected one-dimensional Wasserstein distances vary along directions contained in the effective span of the current empirical measures. Under Assumption 4.1, this induces a systematic scaling of per-slice gradients by the alignment factor $\|U^\top\theta\|^p$, which we make explicit below.

Proposition 4.10. *For discrete distributions* $\hat{\mu}_d = \sum_{i=1}^n q_i^1 \delta_{x_i}$ *and* $\hat{\nu}_d = \sum_{j=1}^m q_j^2 \delta_{y_j}$ *satisfying Assumption 4.1, let* $\hat{\mu}_k = (U^\top)_\# \hat{\mu}_d$ *and* $\hat{\nu}_k = (U^\top)_\# \hat{\nu}_d$.

Then for any $\theta \in \mathbb{S}^{d-1}$ *we have:*
$$\nabla_x W_p^p(\theta_\# \hat{\mu}_d, \theta_\# \hat{\nu}_d) = \|U^\top\theta\|^p \nabla_x W_p^p(\theta_\#^k \hat{\mu}_k, \theta_\#^k \hat{\nu}_k), \tag{22}$$

where $\theta^k = U^\top\theta / \|U^\top\theta\|$. *Furthermore, define the empirical gradient error for each* x_i *as*
$$\epsilon_L(x_i) := \left\| \nabla_{x_i} SW_p^p\left(\hat{\mu}_d, \hat{\nu}_d; \frac{1}{L}\sum_{l=1}^L \delta_{\theta_l^d}\right) - \widehat{ESSF}(L) \cdot \nabla_{x_i} SW_p^p\left(\hat{\mu}_k, \hat{\nu}_k; \frac{1}{L}\sum_{l=1}^L \delta_{\theta_l^k}\right)\right\|. \tag{23}$$

Then the following statements hold:

1. $\epsilon_L(x_i) \xrightarrow{\mathbb{P}} 0$ *as* $L \to \infty$.

2. $\mathbb{P}(\|\epsilon_L(x_i)\| \leq \epsilon) \geq 1 - 2e^{-\epsilon^2 L/(pq_i^1 K)^2}$, *where* $K = \max_{x_i, y_j} \|x_i - y_j\|^{p-1} < \infty$.

We refer readers to the Appendix A.7 for the detailed discussion and proofs.

In particular, note that Equation 22 looks like a scalar times the gradient of the objective function (when using SWD as an objective). In SGD, one updates an iterate by the negative of a scalar—the learning rate—times the gradient of the objective. We can fold the scaling factor $||U^T\theta||^p$ into the learning rate. Note that as in Equation 23, aggregating all the slices, this looks folding the global constant effective subspace scaling factor, once, into the learning rate. Thus, *it is mathematically equivalent to rescale the slices by the ESSF or to search for an optimal learning rate*; i.e., standard learning-rate search is sufficient to recover effective optimization behavior.

We emphasize that we do not propose a new learning-rate selection strategy. The choice of how to search over learning rates (e.g., grid or random search (Bergstra & Bengio, 2012), Bayesian optimization (Snoek et al., 2012), bandit-based methods (Li et al., 2018), or standard schedules (Smith, 2017; Loshchilov & Hutter, 2017)) is orthogonal to our contribution and follows standard practice in machine learning. Our (novel) claim and observation is simply that, once learning rate is accounted for, *the apparent performance gap between classical SWD algorithms and more complex variants largely disappears across a broad range of SWD-based learning tasks*. Moreover, since hyperparameter search is already standard in practical workflows, users of SWD in gradient-based optimization pipelines often effectively get informative slices for free.

5 Experiments

Our theoretical analysis (particularly Theorem 4.7 and Proposition 4.10) predicts that the primary consequence of high ambient dimensionality is an (approximately) scalar shrinkage of SWD gradient magnitude, up to a residual term that vanishes in probability as $L \to \infty$. Since (S)GD-style first-order updates take the form $x \leftarrow x - \eta g$ where $\eta > 0$ is the learning rate, multiplying g by a scalar is operationally equivalent to rescaling the learning rate η. Consequently, benchmarks that compare methods at fixed or narrowly tuned learning rates may be insufficient. We therefore design our experiments to reveal the one-dimensional performance basin along η, i.e., the range of learning rates yielding stable effective convergence. If the degradation of standard SWD is primarily a scaling issue rather than a loss of information, we hypothesize that its good basins persist but shift toward larger learning rates as d/k increases. We test this hypothesis in two stages: first on synthetic data where the effective dimension k is controlled, and second on real-world generative tasks where k is unknown and dynamic.

We sweep learning rates for all methods using the same grid; for non-learning-rate hyperparameters we use the default settings from the official implementations. Unless stated otherwise, we fix $L = 50$ slices in learning experiments to match standard SWD practice and to ensure a consistent computational budget across methods. While increasing L reduces Monte Carlo variability in the sliced estimator and its gradient (Propositions 4.8 and 4.10), the population-level scaling effect identified in Theorem 4.7 persists independently of L. As shown in Figure 3, the concentration of $\widehat{\mathrm{ESSF}}(L)$ is already visible in our tested (d, k) regimes, and larger L primarily smooths the curves without altering the basin-shift behavior. We evaluate on representative SW-based learning tasks that are standard in the sliced-OT literature and are commonly used to claim gains over classical SW. We do not claim that SWD is sufficient for every OT application; settings with known geometric constraints or tasks that depend on a small set of discriminative directions may still benefit from specialized slicing schemes.

Further detailed numerical results and additional visualizations are in the Appendix.

5.1 Numerical Validation of Main Results

We first validate the two quantitative predictions that drive the rest of the paper: (i) under Assumption 4.1, increasing the ambient dimension d primarily rescales SW_p^p by a constant factor depending only on (d, k, p) (Theorem 4.7); and (ii) in the finite-slice regime, this factor is well-approximated by the empirical estimator $\widehat{\mathrm{ESSF}}(L)$ with concentration as L increases (Proposition 4.8). These two checks isolate the "scalar shrinkage" effect from unrelated modeling choices and from downstream optimization dynamics.

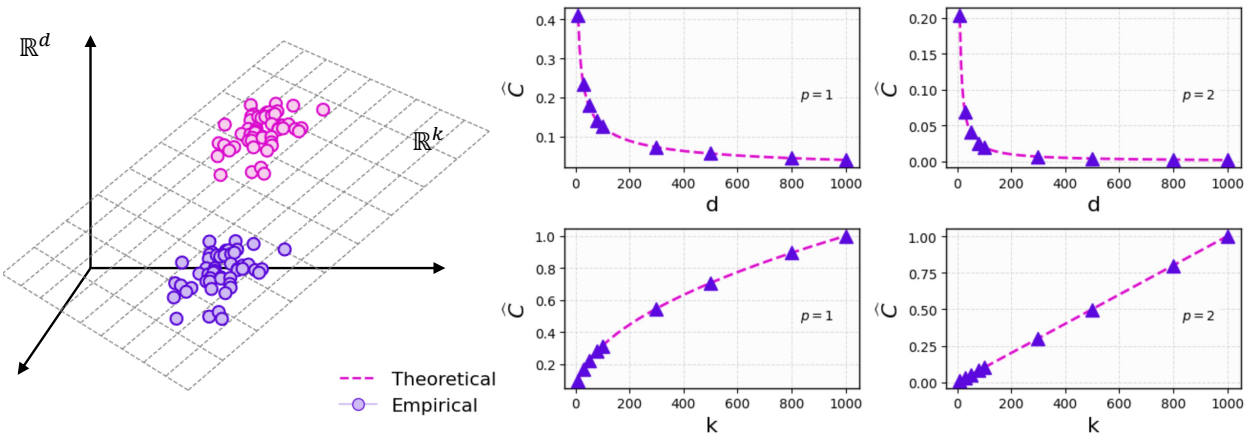


Figure 2: Left: Illustration of two embedded Gaussians. **Top row**: Empirical $\widehat{C}$ for varying d with $k = 2$ and $p = 1, 2$. **Bottom row**: Empirical $\widehat{C}$ for varying k at $d = 1000$ with $p = 1, 2$.

Verifying Theorem 4.7 for $p = 1, 2$. We consider two k-dimensional isotropic Gaussians embedded in $\mathbb{R}^d$ ($d \geq k$), so the effective subspace dimension is controlled by construction. Theorem 4.7 predicts $SW_p^p(\mu^d, \nu^d) = \frac{C_k}{C_d} SW_p^p(\mu^k, \nu^k)$, hence the ratio $\widehat{C} = \frac{\widehat{SW}_p^p(\mu^d,\nu^d)}{\widehat{SW}_p^p(\mu^k,\nu^k)}$ should depend only on (d, k, p) and match $\frac{C_k}{C_d}$ (up to Monte Carlo error). In particular, for fixed k the ratio should decrease as d increases, and for fixed d it should increase with k. We generate 500 samples from each distribution and fix the number of slices to $L = 1000$ to make Monte Carlo error negligible relative to the predicted scaling. We vary d and k as follows: **a)** fix $k = 2$, vary $p \in \{1, 2\}$ and $d \in \{10, 30, 50, 80, 100, 300, 500, 800, 1000\}$; **b)** fix $d = 1000$, vary $p \in \{1, 2\}$ and $k \in \{10, 30, 50, 80, 100, 300, 500, 800, 1000\}$ (averaged over 10 runs). Figure 2 shows that $\widehat{C}$ closely tracks the theoretical $\frac{C_k}{C_d}$ in both regimes, supporting the interpretation that the high-d degradation of classical SWD is captured by a constant downweighting of random slices rather than a change in the underlying 1D transport along the effective subspace.

Verifying Proposition 4.8. Proposition 4.8 states that $\widehat{\text{ESSF}}(L) = \frac{1}{L}\sum_{l=1}^{L} \|U^\top \theta_l^d\|^p$ is an unbiased estimator of $\frac{C_k}{C_d}$ with $\text{Var}(\widehat{\text{ESSF}}(L)) = \mathcal{O}(1/L)$ and exponential concentration. We evaluate $\widehat{\text{ESSF}}(L)$ over 1000 trials for $L \in \{10, 50, 100, 500, 1000, 5000, 10000\}$, using $d \in \{100, 500, 1000\}$ and $k \in \{2, 10, 50\}$ (with $p = 1$). Figure 3 confirms both aspects: the empirical mean matches the theoretical constant and the dispersion shrinks as L increases. This supports treating the loss of informativeness in finite-slice SWD as a stable scalar effect that can be compensated for in optimization (Section 5.2 and Figure 4).

5.2 Gradient Flow

Classic synthetic datasets under ambient embeddings. We generate 300 target particles from three classic 2D datasets (Swiss roll, 8 Gaussians, Knot) and initialize 300 source particles from a 2D isotropic Gaussian. To isolate the effect predicted by Theorem 4.7 and Proposition 4.10, we embed both source and target into $\mathbb{R}^d$ for $d \in \{2, 50, 100\}$ by padding with zeros and applying a random d-dimensional rotation. This procedure increases the ambient dimension while preserving the intrinsic geometry ($k = 2$), so any change in optimization behavior is attributable to the ambient-dimension scaling of the SWD gradient magnitude. We run 10,000 iterations of vanilla gradient descent, reporting results over 3 runs. Learning rates are swept over $\{1, 3, 5, 8\} \times 10^{\{-6,-5,-4,-3,-2,-1,0,1,2\}}$. Consistent with a scalar shrinkage effect, the set of learning rates yielding stable progress for classical SWD does not disappear as d increases; instead it translates toward larger η, as expected from maintaining an approximately constant effective step size $\eta \cdot \widehat{ESSF}(L)$.

MNIST and CelebA particle flows. We further examine this behavior on image-derived empirical measures where the effective dimension k is unknown and may evolve over optimization. For MNIST, we sample 50 images from digit 0 (source) and 50 from digit 1 (target), running gradient flow for 200,000

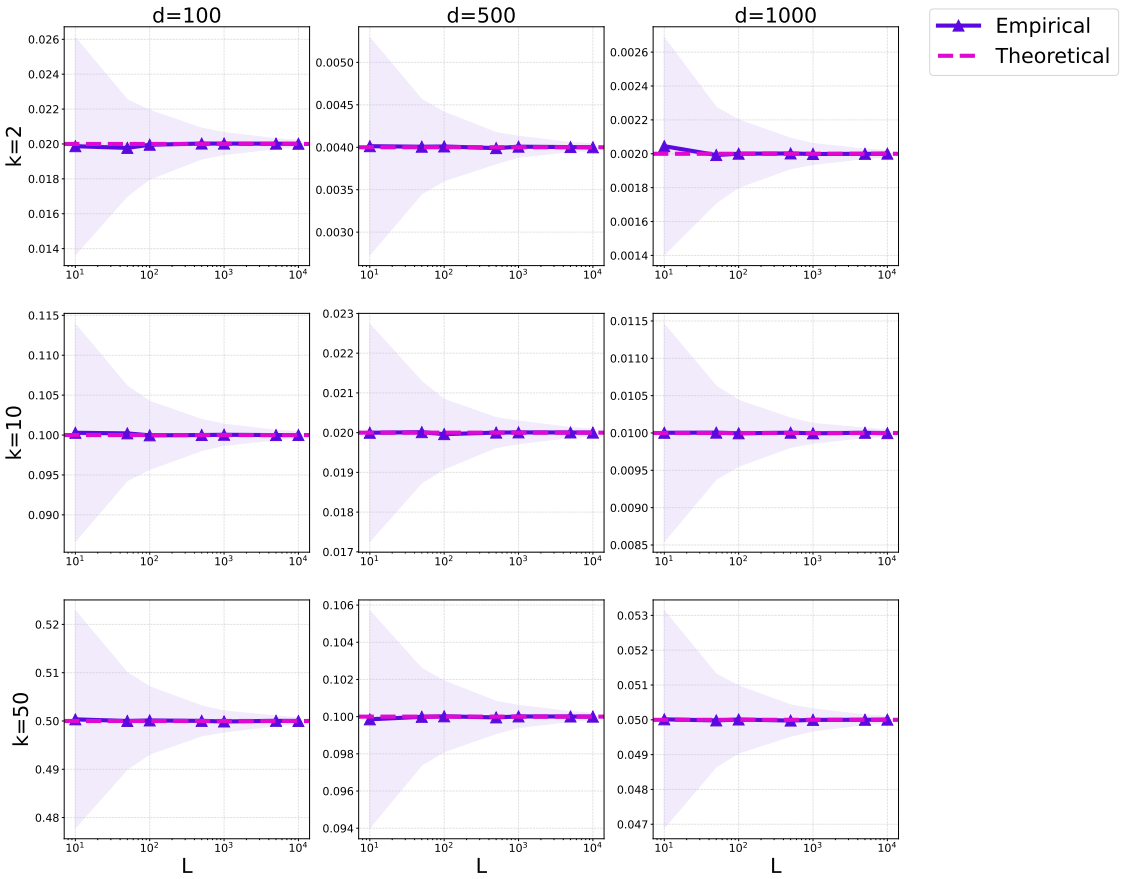


Figure 3: Empirical $\widehat{\mathrm{ESSF}}(L)$ for varying d, k over 1000 runs and with $p = 1$. The dotted line depicts the theoretical value.

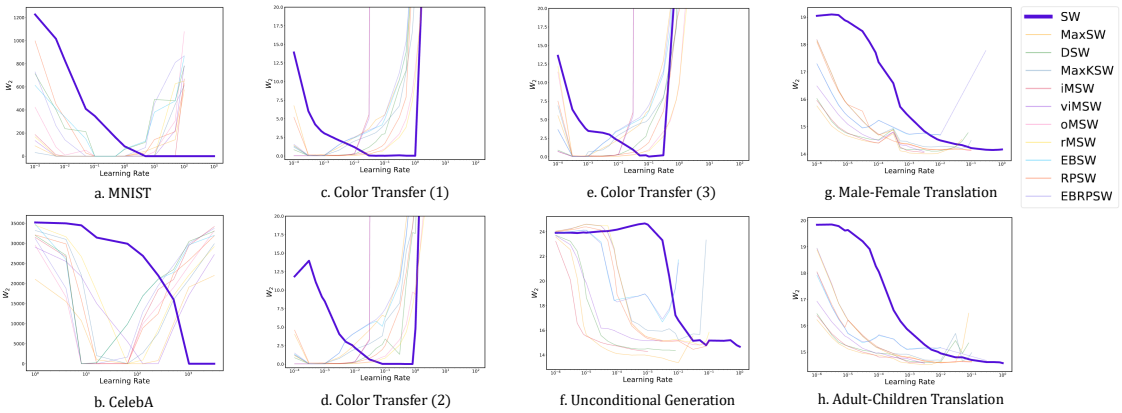


Figure 4: Optimal basin plots for gradient flow (a,b), color transfer (c,d,e), and generative modeling (f,g,h).

Figure 5: Color transfer visualization. Appropriate learning rates are chosen for each method; performing that hyperparameter calibration, all methods are able to perform similarly on the color transfer task.

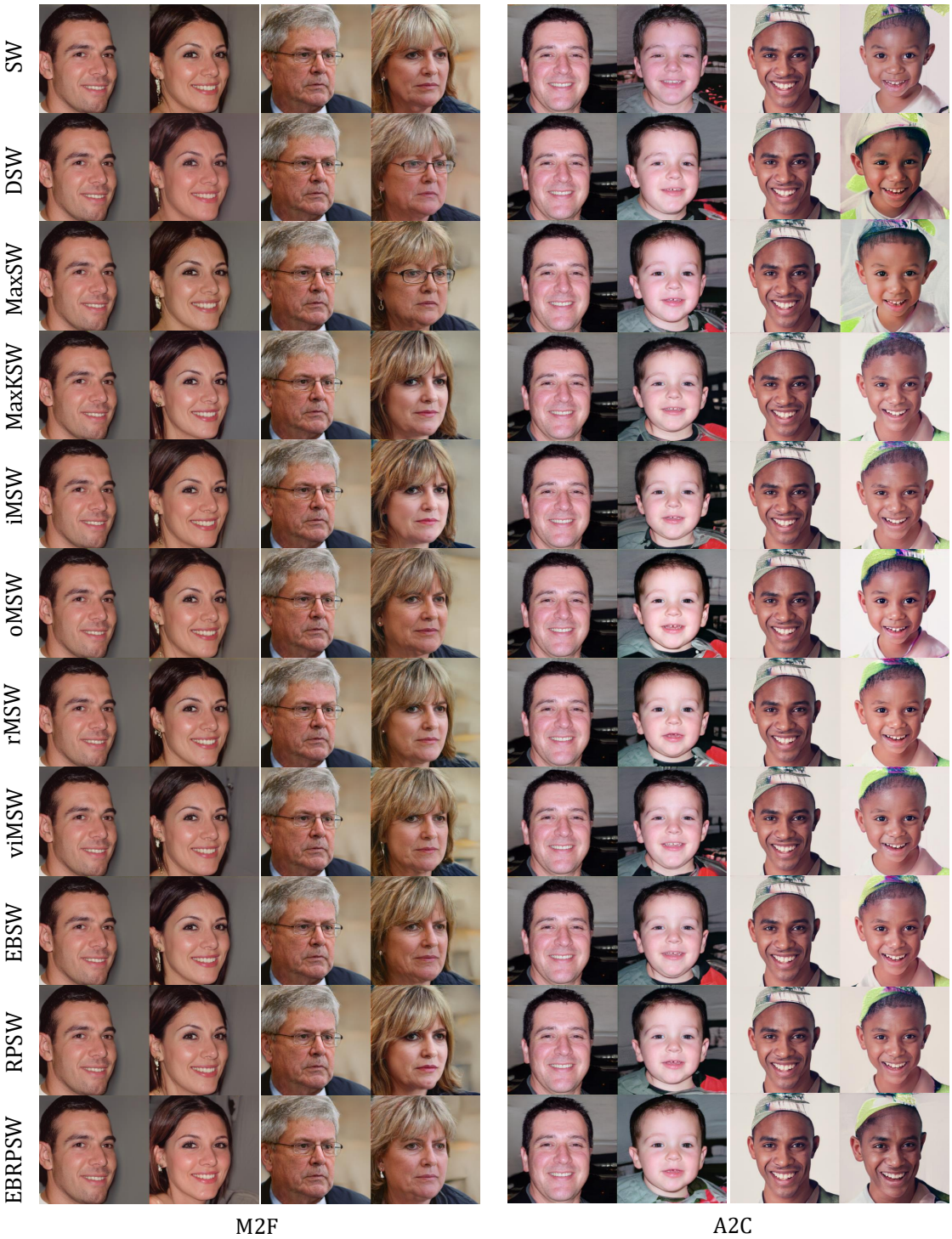


Figure 6: Deep generative modeling example, where samples generated using different SWD variants. The left four columns show male-to-female (M2F) examples, while the right four columns show adult-to-child (A2C) examples. We use appropriate learning rates for each method (guided by the results shown in Figure 4). Because of this, all methods perform similarly, demonstrating that tuning learning rates and using classical SWD can be effective as using customized slicing distributions and more complex variants of SWD.

iterations with learning rates in $\{1, 5\} \times 10^{\{-3,-2,-1,0,1,2,3\}}$. For CelebA, we initialize particles from Gaussian noise and optimize toward 50 target face images for 200,000 iterations, sweeping learning rates up to 3200. Figure 4(a,b) shows the same qualitative signature: classical SWD exhibits a nontrivial basin of effective learning rates, but its basin is centered at substantially larger η than many subspace-aware variants. This matches the interpretation that the dominant gap at default hyperparameters is step-size calibration (through gradient scaling), rather than an absence of informative directions.

5.3 Color Transfer

We follow a similar setup as in (Nguyen et al., 2024a; Nguyen & Ho, 2024; Nguyen et al., 2024b) with different hyperparameters. Our experiments are performed over 3 image sets (see Figure 5). The optimization uses 50,000 iterations. To reduce computational complexity, we optionally apply K-means clustering with 3,000 clusters, reducing the colorspace into an empirical measure with $N = 3{,}000$ particles. **Learning rates:** $\{1, 3, 5, 8\} \times 10^{\{-4,-3,-2,-1,0,1\}}, 100$. Figure 4(c,d,e) makes the role of η explicit: classical SWD attains competitive final objectives, but its effective basin is shifted relative to several variants, and some variants exhibit narrower stability ranges at larger η. Selecting η from the basin therefore separates optimization effects (step-size and stability) from the slicing mechanism itself; with this calibration, the qualitative transfers in Figure 5 are consistent across methods.

5.4 Deep Generative Modeling

There exist various generative modeling setups with Sliced-Wasserstein distances (Kolouri et al., 2018; Deshpande et al., 2018; Wu et al., 2019; Liutkus et al., 2019; Nguyen et al., 2024b). We restrict our setup to the latent space ($d = 512$) of an autoencoder (Pidhorskyi et al., 2020) pretrained on the 1024×1024 FFHQ dataset (Karras et al., 2019). **Learning rates:** $\{1, 3, 5, 8\} \times 10^{\{-6,-5,-4,-3,-2,-1\}}, 1$. In this high-dimensional latent setting, Proposition 4.10 predicts that classical SWD gradients can be substantially smaller in magnitude (via $\widehat{ESSF}(L)$), so recovering comparable optimization dynamics requires larger η.

We evaluate SWD variants on both unconditional generation and unpaired image-to-image translation tasks. For generation, we follow Deshpande et al. (2018)'s SWG setup using a generator $G_\phi(\cdot)$ to transform $z \in \mathbb{R}^8$ to latents $X \in \mathbb{R}^{512}$. For translation, we modify this to use a residual generator transforming source domain X to target domain Y latents. Following Rombach et al. (2022); Korotin et al. (2023), we operate in an autoencoder's latent space to sidestep the known dimensionality challenges of pixel-space SWG (Deshpande et al., 2018; Nadjahi et al., 2021). We train for 10,000 iterations using vanilla gradient descent with batch size 2048. Figure 4(f,g,h) supports the scaling interpretation directly: classical SWD has a clear basin of effective learning rates, but its basin is shifted toward larger η in the $d = 512$ regime. When η is chosen within this basin, classical SWD attains objectives comparable to the variants, and the corresponding samples in Figure 6 are qualitatively similar, which is consistent with the view that much of the apparent advantage of specialized slicing shows up as an implicit step-size change.

Summary. Our experiments show that random slicing does not lack informative directions but instead induces an approximately scalar attenuation of the SWD value and its first-order signal, through factors of the form $\|U^\top \theta\|^p$ that aggregate into $\widehat{ESSF}(L)$ (with a residual that vanishes as L grows). The Gaussian embedding study validates the implied scaling law at the level of distances and shows that $\widehat{ESSF}(L)$ concentrates around its theoretical limit as L increases. The learning experiments then show that the set of learning rates yielding stable effective optimization for classical SWD persists across tasks, but it shifts toward larger η as d/k increases, which is exactly what one expects if the dominant failure mode is step-size miscalibration via gradient shrinkage. Once η is chosen within this basin, classical SWD attains comparable objectives and comparable qualitative outputs to methods that explicitly bias slices toward the data subspace, supporting our hypothesis that much of the apparent high-dimensional gap is primarily a rescaling effect rather than a loss of information.

6 Conclusion

In this paper, we revisit the classical, "vanilla" Sliced-Wasserstein distance and rethink the dominant approach of modifying the slicing distribution to target *informative* directions. We instead view informativeness through a ϕ-weighting formulation that rescales the one-dimensional Wasserstein contributions. Under an effective subspace model, defining informativeness via alignment with the data subspace yields a Subspace SWD variant that is equivalent to standard SWD up to a single scalar factor. In first-order optimization, the same phenomenon appears at the gradient level: the loss of slice informativeness manifests primarily as an approximately scalar shrinkage of gradient magnitude (up to a finite-slice residual), which is operationally equivalent to a learning-rate rescaling. This provides a direct explanation for why vanilla SWD can match or surpass more complex slicing schemes once standard learning-rate calibration is performed, without additional implementation or loss of metric structure. Our experiments across representative SWD-based learning tasks support this view by showing that the "performance gap" is often a shift in the learning-rate basin rather than a disappearance of good solutions. While SWD is not expected to be sufficient for every OT application, especially in settings with known geometric constraints or tasks that rely on a small set of discriminative directions, our results clarify when and why vanilla SWD remains a strong baseline. Future work can use the same framework to study alternative assumptions on data structure and corresponding choices of the rescaling function ρ and informativeness ϕ.

References

Manish Acharya and David Hyde. Efficient sliced wasserstein distance computation via adaptive bayesian optimization. *arXiv preprint arXiv:2509.17405*, 2025. 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. 36

James Bergstra and Yoshua Bengio. Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, 13(10):281–305, 2012. 10

Mathieu Blondel, Vivien Seguy, and Antoine Rolet. Smooth and sparse optimal transport. In *International conference on artificial intelligence and statistics*, pp. 880–889. PMLR, 2018. 1

Clément Bonet, Nicolas Courty, François Septier, and Lucas Drumetz. Sliced-wasserstein gradient flows. *arXiv preprint arXiv:2110.10972*, pp. 1, 2021a. 31

Clément Bonet, Titouan Vayer, Nicolas Courty, François Septier, and Lucas Drumetz. Subspace detours meet gromov–wasserstein. *Algorithms*, 14(12):366, 2021b. 4

Clément Bonet, Paul Berg, Nicolas Courty, François Septier, Lucas Drumetz, and Minh-Tan Pham. Spherical sliced-wasserstein. *arXiv preprint arXiv:2206.08780*, 2022. 2

Clément Bonet, Lucas Drumetz, and Nicolas Courty. Sliced-wasserstein distances and flows on cartan-hadamard manifolds. *arXiv preprint arXiv:2403.06560*, 2024. 2

Nicolas Bonneel and David Coeurjolly. Spot: sliced partial optimal transport. *ACM Transactions on Graphics (TOG)*, 38(4):1–13, 2019. 31

Nicolas Bonneel, Julien Rabin, Gabriel Peyré, and Hanspeter Pfister. Sliced wasserstein distance. *SIAM Journal on Imaging Sciences*, 8(4):2292–2310, 2015. 1

Xiongjie Chen, Yongxin Yang, and Yunpeng Li. Augmented sliced wasserstein distances. *arXiv preprint arXiv:2006.08812*, 2020. 2

Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 26, 2013. 1

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255. IEEE, 2009. 4

Ishan Deshpande, Ziyu Zhang, and Alexander G Schwing. Generative modeling using the sliced wasserstein distance. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3483–3491, 2018. 15

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*, pp. 10648–10656, 2019. 2, 5, 6

Persi Diaconis and David Freedman. Asymptotics of graphical projection pursuit. *The annals of statistics*, pp. 793–815, 1984. 4

Charles Fefferman, Sanjoy Mitter, and Hariharan Narayanan. Testing the manifold hypothesis. *Journal of the American Mathematical Society*, 29(4):983–1049, 2016. 4

Ronald Aylmer Fisher. Dispersion on a sphere. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, 217(1130):295–305, 1953. 6

Carl Hvarfner, Erik Orm Hellsten, and Luigi Nardi. Vanilla bayesian optimization performs great in high dimensions. *arXiv preprint arXiv:2402.02229*, 2024. 5

Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4401–4410, 2019. 15, 40

Abdelwahed Khamis, Russell Tsuchida, Mohamed Tarek, Vivien Rolland, and Lars Petersson. Scalable optimal transport methods in machine learning: A contemporary survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024. 1

Soheil Kolouri, Phillip E Pope, Charles E Martin, and Gustavo K Rohde. Sliced wasserstein auto-encoders. In *International Conference on Learning Representations*, 2018. 15

Soheil Kolouri, Kimia Nadjahi, Umut Simsekli, Roland Badeau, and Gustavo Rohde. Generalized sliced wasserstein distances. *Advances in neural information processing systems*, 32, 2019. 2, 4

Alexander Korotin, Nikita Gushchin, and Evgeny Burnaev. Light schr\" odinger bridge. *arXiv preprint arXiv:2310.01174*, 2023. 15

Lisha Li, Kevin Jamieson, Giulia DeSalvo, Afshin Rostamizadeh, and Ameet Talwalkar. Hyperband: A novel bandit-based approach to hyperparameter optimization. *Journal of Machine Learning Research*, 18(185): 1–52, 2018. 10

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European Conference on Computer Vision*, pp. 740–755. Springer, 2014. 4

Antoine Liutkus, Umut Simsekli, Szymon Majewski, Alain Durmus, and Fabian-Robert Stöter. Sliced-wasserstein flows: Nonparametric generative modeling via optimal transport and diffusions. In *International Conference on Machine Learning*, pp. 4104–4113. PMLR, 2019. 15

Ilya Loshchilov and Frank Hutter. SGDR: Stochastic gradient descent with warm restarts. In *International Conference on Learning Representations (ICLR)*, 2017. 10

Tudor Manole, Sivaraman Balakrishnan, Jonathan Niles-Weed, and Larry Wasserman. Plugin estimation of smooth optimal transport maps. *The Annals of Statistics*, 52(3):966–998, 2024. 1

Kanti V Mardia and Peter E Jupp. *Directional statistics*. John Wiley & Sons, 2009. 6

Rocio Diaz Martin, Ivan Medri, Yikun Bai, Xinran Liu, Kangbai Yan, Gustavo K Rohde, and Soheil Kolouri. Lcot: Linear circular optimal transport. *arXiv preprint arXiv:2310.06002*, 2023. 2

Boris Muzellec and Marco Cuturi. Subspace detours: Building transport plans that are optimal on subspace projections. *Advances in Neural Information Processing Systems*, 32, 2019. 4

Kimia Nadjahi, Alain Durmus, Lénaïc Chizat, Soheil Kolouri, Shahin Shahrampour, and Umut Simsekli. Statistical and topological properties of sliced probability divergences. *Advances in Neural Information Processing Systems*, 33:20802–20812, 2020. 1

Kimia Nadjahi, Alain Durmus, Pierre E Jacob, Roland Badeau, and Umut Simsekli. Fast approximation of the sliced-wasserstein distance using concentration of random projections. *Advances in Neural Information Processing Systems*, 34:12411–12424, 2021. 4, 15, 26

Khai Nguyen and Nhat Ho. Sliced wasserstein estimation with control variates. *arXiv preprint arXiv:2305.00402*, 2023. 2

Khai Nguyen and Nhat Ho. Energy-based sliced wasserstein distance. *Advances in Neural Information Processing Systems*, 36, 2024. 2, 5, 6, 15

Khai Nguyen, Nhat Ho, Tung Pham, and Hung Bui. Distributional sliced-wasserstein and applications to generative modeling. *arXiv preprint arXiv:2002.07367*, 2020. 2, 5

Khai Nguyen, Nicola Bariletto, and Nhat Ho. Quasi-monte carlo for 3d sliced wasserstein. *arXiv preprint arXiv:2309.11713*, 2023. 2

Khai Nguyen, Tongzheng Ren, and Nhat Ho. Markovian sliced wasserstein distances: Beyond independent projections. *Advances in Neural Information Processing Systems*, 36, 2024a. 2, 5, 6, 15

Khai Nguyen, Shujian Zhang, Tam Le, and Nhat Ho. Sliced wasserstein with random-path projecting directions. *arXiv preprint arXiv:2401.15889*, 2024b. 2, 5, 7, 15

François-Pierre Paty and Marco Cuturi. Subspace robust wasserstein distances. In *International conference on machine learning*, pp. 5072–5081. PMLR, 2019. 4

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. 1, 36

Stanislav Pidhorskyi, Donald A Adjeroh, and Gianfranco Doretto. Adversarial latent autoencoders. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14104–14113, 2020. 15

Phillip Pope, Chen Zhu, Ahmed Abdelkader, Micah Goldblum, and Tom Goldstein. The intrinsic dimension of images and its impact on learning. *arXiv preprint arXiv:2104.08894*, 2021. 4

Michael Quellmalz, Robert Beinert, and Gabriele Steidl. Sliced optimal transport on the sphere. *Inverse Problems*, 39(10):105005, 2023. 2

Julien Rabin, Julie Delon, and Yann Gousseau. Transportation distances on the circle. *Journal of Mathematical Imaging and Vision*, 41(1):147–167, 2011. 2

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: Third International Conference, SSVM 2011, Ein-Gedi, Israel, May 29–June 2, 2011, Revised Selected Papers 3*, pp. 435–446. Springer, 2012. 1

Galen Reeves. Conditional central limit theorems for gaussian projections. In *2017 IEEE International Symposium on Information Theory (ISIT)*, pp. 3045–3049. IEEE, 2017. 4

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10684–10695, 2022. 15

Leslie N. Smith. Cyclical learning rates for training neural networks. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2017. 10

Jasper Snoek, Hugo Larochelle, and Ryan P. Adams. Practical Bayesian optimization of machine learning algorithms. In *Advances in Neural Information Processing Systems*, 2012. 10

Vladimir Nikolaevich Sudakov. Typical distributions of linear functionals in finite-dimensional spaces of higher dimension. In *Doklady Akademii Nauk*, volume 243, pp. 1402–1405. Russian Academy of Sciences, 1978. 4

Huy Tran, Yikun Bai, Abihith Kothapalli, Ashkan Shahbazi, Xinran Liu, Rocio P Diaz Martin, and Soheil Kolouri. Stereographic spherical sliced wasserstein distances. In *Forty-first International Conference on Machine Learning*, 2024. 2

Thanh Tran, Viet-Hoang Tran, Thanh Chu, Trang Pham, Laurent El Ghaoui, Tam Le, and Tan M Nguyen. Tree-sliced wasserstein distance with nonlinear projection. *arXiv preprint arXiv:2505.00968*, 2025a. 2

Viet-Hoang Tran, Thanh T Chu, Khoi NM Nguyen, Trang Pham, Tam Le, and Tan M Nguyen. Spherical tree-sliced wasserstein distance. *arXiv preprint arXiv:2503.11249*, 2025b. 2

Cédric Villani et al. *Optimal transport: old and new*, volume 338. Springer, 2009. 1

Jiqing Wu, Zhiwu Huang, Dinesh Acharya, Wen Li, Janine Thoma, Danda Pani Paudel, and Luc Van Gool. Sliced wasserstein generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3713–3722, 2019. 15

A Proofs and Additional Theoretical Results

A.1 Notation

- $\mathbb{R}^d$: d-dimensional Euclidean space, where d is a positive integer.
- $\mathbb{S}^{d-1} := \{x \in \mathbb{R}^d : \|x\| = 1\}$: unit sphere defined in $\mathbb{R}^d$.
- $\mathcal{P}(\mathbb{R}^d)$: set of all probability measures defined on $\mathbb{R}^d$.
- $\mathcal{P}_p(\mathbb{R}^d)$: set of probability measures whose p-th moment is finite, where $p \geq 1$.
- $\mathbb{V}_{k,d}$: set of all $d \times k$ orthogonal matrices, i.e.

  $$\mathbb{V}_{k,d} := \{U \in \mathbb{R}^{d\times k} : U^\top U = I_k\}.$$

  Note, $\mathbb{S}^{d-1} = \mathbb{V}_{1,d}$.
- $U = [U[:,1], U[:,2], \dots U[:,k]] \in \mathbb{V}_{k,d}$: an orthogonal matrix. For each $i \in [1:k]$, $U[:,i] \in \mathbb{R}^d$ is the i-th column of U.

  Note that U induces a linear function from $\mathbb{R}^d$ to $\mathbb{R}^k$, i.e. $x \mapsto U^\top x$. With abuse of notation, we do not distinguish the matrix U and the corresponding linear mapping.
- $\text{Span}(U)$: The linear subspace spanned by U, i.e.

  $$\text{Span}(U) := \text{Span}(\{U[:,1], U[:,2], \dots U[:,k]\}) = \left\{\sum_{i=1}^k \alpha_i U[:,i] : \alpha_i \in \mathbb{R}\right\}.$$
- $V_k \subset \mathbb{R}^d$: a k-dimensional subspace, where k is a positive integer with $k \leq d$. Note, by classical linear algebra theory, we have

  $$V_k = \text{Span}(U)$$

  for some $U \in \mathbb{V}_{d,k}$. **Note, given V_k, U is not uniquely determined.**
- $V_k^\perp$: perpendicular complement of V_k, which is a subspace of dimension $d-k$.
- $\mu^d, \mu, \nu^d, \nu \in \mathcal{P}(\mathbb{R}^d)$: probability measures in d-dimensional space.
- $\mathcal{L}^d$: Lebesgue measure in $\mathbb{R}^d$.
- $C_0(\mathbb{R}^d)$: set of all continuous functions defined on $\mathbb{R}^d$ which vanish at infinity.
- $f_\mu = \frac{d\mu^d}{d\mathcal{L}^d}$: density of μ, that is, for all test functions $\phi \in C_0(\mathbb{R}^d)$:

  $$\int_{\mathbb{R}^d} \phi(x) d\mu^d(x) = \int_{\mathbb{R}^d} f_\mu(x)\phi(x)dx.$$
- $X \sim \mu$: A random variable/vector X following distribution μ. We say X is a **realization** of μ.
- $\mathbb{E}[X] := \mathbb{E}[\mu]$, where $X \sim \mu$: expected value of X, i.e.

  $$\mathbb{E}_\mu[X] = \int_{\mathbb{R}^d} x d\mu(x).$$
- $m_k(\mu)$: k-th moment of measure μ. That is, given realization $X \sim \mu$, $m_k(\mu)$ is defined by

  $$m_k(\mu) := \mathbb{E}[X^k]$$
- $Var(X) := \mathbb{E}[(X - \mathbb{E}(X))^\top (X - \mathbb{E}(X))]$: the covariance matrix of X (or the measure μ).

- $T_{\#}\mu$, where $T : \mathbb{R}^d \to \mathbb{R}^d$ is a function: push-forward measure μ under mapping T. That is, for all Borel sets $A \subset \mathbb{R}^d$, we have
$$T_{\#}\mu(A) = \mu(T^{-1}(A)).$$
Equivalently speaking, suppose $X \sim \mu$ is a realization of μ; then, $T(X) \sim T_{\#}\mu$.
- $\mathcal{N}(e, \Sigma)$: Gaussian distribution, where $e \in \mathbb{R}^d$ is the expected value, $\Sigma \in \mathbb{R}^{d\times d}$ is the covariance matrix.
- 0_d: $d \times 1$ vector where each entry is 0. Similarly, we define 1_d.
- I_d: $d \times d$ identity matrix.
- $\mathcal{U}(\mathbb{S}^{d-1})$: Uniform distribution defined on $\mathbb{S}^{d-1}$.
- $\theta^d \sim \mathcal{U}(\mathbb{S}^{d-1})$: a d-dimensional random vector. We say θ^d is a **realization** of $\mathcal{U}(\mathbb{S}^{d-1})$.
- $\theta, \theta^d, \theta^g$: a d-dimensional vector.
- θ^k: a k-dimensional vector.
- $P_{V_k} := P_U$, where $V_k = \text{Span}(U)$: the projection mapping from $\mathbb{R}^d$ into subspace V_k, i.e.
$$P_{V_k}(x) := P_U(x) = UU^\top x, \forall x \in \mathbb{R}^d.$$
Note, in this case: the mapping $U : \mathbb{R}^d \to \mathbb{R}^k$ with $x \mapsto U^\top x$ is the corresponding parameterization function of projection P_U.
- $\Gamma(\mu, \nu)$: set of joint measures whose marginals are μ, ν respectively:
$$\Gamma(\mu, \nu) := \{\gamma \in \mathcal{P}((\mathbb{R}^d)^2) : (\pi_1)_{\#}\gamma = \mu, (\pi_2)_{\#}\gamma = \nu\},$$
where $\pi_1 : (x, y) \mapsto x, \pi_2 : (x, y) \mapsto y$ are canonical projection mappings.
- $W_p^p(\mu, \nu)$: Wasserstein problem between μ and ν:
$$W_p^p(\mu, \nu) := \inf_{\gamma\in\Gamma(\mu,\nu)} \int_{(\mathbb{R}^d)^2} \|x - y\|^p d\gamma(x, y)$$
- $SW(\mu, \nu; \sigma)$, where $\sigma \in \mathcal{P}(\mathbb{S}^{d-1})$: Sliced Wasserstein problem between μ and ν with respect to reference measure σ:
$$SW_p^p(\mu, \nu; \sigma) := \int_{\mathbb{S}^{d-1}} W_p^p(\theta_{\#}\mu, \theta_{\#}\nu) d\sigma(\theta)$$
- $\phi_U : \mathbb{S}^{d-1} \to \mathbb{R}_+$: ES-informative aligned mapping. A measurable mapping which describes the information of the projected θ on the space spanned by U.
- $\widetilde{SW}(\mu, \nu; \sigma, \rho)$: rescaled sliced Wasserstein distance:
$$\widetilde{SW}(\mu, \nu; \sigma, \rho) := \int_{\mathbb{S}^{d-1}} r(\phi_U(\theta)) W_p^p(\theta_{\#}\mu, \theta_{\#}\nu) d\sigma(\theta)$$
where $\rho : \mathbb{R}_+ \to \mathbb{R}_+$ is a rescaling function. In this paper, we set ρ as the following decreasing function:
$$\rho(x) = \frac{1}{x^p}$$
When $x = 0$, we adopt the convention $\rho(x) = 0$.

Remark A.1. *In this paper, we adopt the following convention.*

We do not distinguish the scalar/vector/matrix and the corresponding induced linear mapping. For example, $\theta \in \mathbb{R}^d$, induces the mapping

$$\mathbb{R}^d \ni x \mapsto \theta^\top x \in \mathbb{R}.$$

- *When θ is a random vector, we refer to it as a "random projection mapping" in both the main text and the appendix. We adopt the same convention for the scalar notation α and the matrix notation U.*
- *We use $\theta_\# \mu$ to denote the push-forward measure induced by mapping $x \mapsto \theta^\top x$. Similarly, $(\theta \times \theta)_\# \gamma$ denotes the push-forward measure of joint measure $\gamma \in \mathcal{P}((\mathbb{R}^d)^2)$ induced by mapping $(x, x') \mapsto (\theta^\top x, \theta^\top x')$. The same convention is adopted for α, U.*

Remark A.2. *For simplicity, in notation $SW(\mu, \nu; \sigma)$, we may relax the restriction that σ is a probability measure. We allow σ to be a finite positive measure in the main text and appendix.*

A.2 Wasserstein distances in $\mathbb{R}^d$ and $\mathbb{R}^k$

In the present manuscript, we assume the probability measures $\mu^d, \nu^d \in \mathcal{P}_p(\mathbb{R}^d)$ are supported in a lower dimensional subspace. We refer to Assumption 4.1 for details.

Let P_U denote the projection mapping from $\mathbb{R}^d$ to V_k:

$$P_U(x) = UU^\top x, \forall x \in \mathbb{R}^d, \tag{24}$$

Then, the corresponding lower-dimensional parameterization mapping is defined as:

$$x \mapsto U^\top x, \forall x \in \mathbb{R}^d. \tag{25}$$

By classical linear algebra theory, it is straightforward to verify the following:

Proposition A.3. *[Basic properties of linear projection] Let P_U, U be defined above, then we have:*

(1) For each $\theta \in \mathbb{R}^d$, θ can be uniquely decomposed into $V_k, V_k^\perp$, i.e. $\theta = \theta_{V_k} + \theta_{V_k^\perp}$, where $\theta_{V_k} = P_U(\theta) \in V_k, \theta_{V_k^\perp} \in V_k^\perp$.

(2) For all $x \in V_k$, $P_U(x) = x$.

(3) If we restrict U to the subspace V_k, denoted as $U\mid_{V_k}$, then $U\mid_{V_k}: V_k \to \mathbb{R}^k$ is a bijection. The inverse is given by

$$\left(U\mid_{V_k}\right)^{-1}(y) = Uy, \quad \forall y \in \mathbb{R}^k.$$

In addition, $\|U^\top x\| = \|x\|$ for all $x \in V_k$.

Proof. It follows directly from the definitions of P_U, U. □

Let $\mu^k = (U^\top)_\# \mu^d$, $\nu^k = (U^\top)_\# \nu^d$, the above proposition directly induces the following relation between the Wasserstein distance between μ^d, ν^d and the Wasserstein distance between μ^k, ν^k.

Proposition A.4. *Under Assumption 4.1, we have the following:*

(1) μ^d can be recovered by the inverse of $U\mid_{V_k}$, i.e.

$$\mu^d = U_\# \mu^k.$$

(2) The mapping

$$\Gamma(\mu^d, \nu^d) \ni \gamma^d \mapsto \gamma^k := (U^\top \times U^\top)_\# \gamma^d \in \Gamma(\mu^k, \nu^k), \tag{26}$$

is a well-defined bijection, where $U^\top \times U^\top$ is defined as

$$\mathbb{R}^d \times \mathbb{R}^d \ni (x, x') \mapsto (U^\top x,\, U^\top x') \in \mathbb{R}^k \times \mathbb{R}^k. \tag{27}$$

(3) The Wasserstein distance is preserved via the lower-dimensional parameterization:

$$W_p^p(\mu^d, \nu^d) = W_p^p((P_U)_\# \mu^d, (P_U)_\# \nu^d) = W_p^p(\mu^k, \nu^k) \tag{28}$$

Proof. Let $X \sim \mu^d$ be a realization.

(1) We have $U^\top X \sim \mu^k$ since $\mu^k = (U^\top)_\# \mu^d$. In addition, by Assumption 4.1, we have $X = UU^\top X$, thus $UU^\top X \sim \mu^d$. That is $U_\# \mu^k = \mu^d$.

(2) Pick $\gamma^d \in \Gamma(\mu^d, \nu^d)$, we have

$$(\pi_1)_\# (U^\top \times U^\top)_\# \gamma^d = (U^\top)_\# ((\pi_1)_\# \gamma^d) = (U^\top)_\# \mu^d = \mu^k$$

Similarly, $(\pi_2)_\# (U^\top \times U^\top)_\# \gamma^d = \nu^k$. Thus the mapping defined in Equation 26 is well-defined. Moreover, from statement (1), we have

$$\Gamma(\mu^k, \nu^k) \ni \gamma^k \mapsto (U^\top \times U^\top)_\# \gamma^k \in \Gamma(\mu^d, \nu^d) \tag{29}$$

is well-defined.

Next, we will show the above mapping is the inverse of Equation 26.

Let $(X, Y) \sim \gamma^d$ be a realization. Then

$$(X, Y) = (UU^\top X, UU^\top Y) \sim (U^\top \times U^\top)_\# (U \times U)_\# \gamma.$$

Thus $(U^\top \times U^\top)_\# (U \times U)_\# \gamma = \gamma$. Thus, the mapping in Equation 29 is the inverse of the mapping in Equation 26. Thus, Equation 26 is invertible (a bijection).

(3) By Proposition A.3 (2), for each $x \in \operatorname{supp}(\mu) \subset V_k$, we have $P_U(x) = x$, thus $(P_U)_\# \mu^d = \mu^d$. Similarly, $(P_U)_\# \nu^d = \nu^d$. Thus we obtain the first equality:

$$W_p^p(\mu^d, \nu^d) = W_p^p((P_U)_\# \mu^d, (P_U)_\# \nu^d).$$

For the second equality, we first pick $\gamma^d \in \Gamma(\mu^d, \nu^d)$ and let $\gamma^k = (U \times U)_\# \gamma^d$. By statement (2), we have $\gamma^k \in \Gamma(\mu^k, \nu^k)$.

$$\begin{aligned}
&\int_{(\mathbb{R}^d)^2} \|x - y\|^p d\gamma^d(x, y) \\
&= \int_{(\mathbb{R}^d)^2} \|U^\top x - U^\top y\|^p \, d\gamma^d(x, y) \\
&= \int_{(\mathbb{R}^k)^2} \|x' - y'\|^p \, d(U^\top \times U^\top)_\# \gamma^d(x', y') \\
&= \int_{(\mathbb{R}^k)^2} \|x' - y'\|^p d\gamma^k(x', y')
\end{aligned}$$

where the first equality follows from Proposition A.3 (3), the second equality follows from the definition of push-forward measure, the third equality holds from statement (2).

Combining the above equality with statement (2), we obtain

$$\begin{aligned} W_p^p(\mu^d, \nu^d) &= \inf_{\gamma^d \in \Gamma(\mu^d, \nu^d)} \int_{\mathbb{R}^d} \|x - y\|^p d\gamma^d(x, y) \\ &= \inf_{\gamma^k \in \Gamma(\mu^k, \nu^k)} \int_{\mathbb{R}^k} \|x' - y'\|^p d\gamma^k(x', y') \\ &= W_p^p(\mu^k, \nu^k) \end{aligned}$$

□

Proposition A.5. *Let $\sigma = \mathcal{U}(\mathbb{S}^{d-1})$ and let $\phi : \mathbb{S}^{d-1} \to \mathbb{R}_+$ and $\rho : \mathbb{R}_+ \to \mathbb{R}_{\geq 0}$ be measurable and define m on $\mathbb{S}^{d-1}$ by*

$$dm(\theta) = (\rho \circ \phi)(\theta) d\sigma(\theta). \tag{30}$$

If $0 < m(\mathbb{S}^{d-1}) < \infty$ and $\rho(\phi(\theta) > 0$ for σ-a.e. θ, then $\widetilde{SW}_p(\cdot, \cdot; \sigma, \rho)$ is a metric on $\mathcal{P}_p(\mathbb{R}^d)$.

Proof. Non-negativity and symmetry are immediate. For the triangle inequality, we note that for each $\theta \in \mathbb{S}^{d-1}$,

$$W_p(\theta_\# \mu, \theta_\# \xi) \leq W_p(\theta_\# \mu, \theta_\# \nu) + W_p(\theta_\# \nu, \theta_\# \xi), \tag{31}$$

and applying Minkowski's inequality to $\theta \mapsto W_p(\theta_\# \mu, \theta_\# \nu)$ in $L^p(\mathbb{S}^{d-1}, m)$ yields the result.

For the identity of indiscernibles, we have that $\widetilde{SW}_p(\mu, \nu) = 0$ implies

$$W_p(\theta_\# \mu, \theta_\# \nu) = 0 \tag{32}$$

for m-a.e. θ.

Since $\rho(\phi(\theta)) > 0$ for σ a.e. θ and $dm = (\rho \circ \phi) d\sigma$, we have $m \sim \sigma$ and hence above holds σ a.e. θ. By Cramer-Wold this implies $\mu = \nu$.

□

Remark A.6. *For the ES-aligned choice $\phi_U(\theta) = \|U^\top \theta\|_2$ and $\rho(x) = x^{-p}$ (with $\rho(0) = 0$), the condition $m(\mathbb{S}^{d-1}) < \infty$ requires $p < k$. Indeed, for $\theta \sim \sigma$ we have $\|U^\top \theta\|_2^2 \sim \mathrm{Beta}(k/2, (d-k)/2)$, and the negative moment exists iff $\frac{p}{2} < \frac{k}{2}$. This does not affect Theorem 4.7 or Proposition 4.10, which relies only on the scaling relationship in expectation.*

A.3 Background: Relationship between the Gaussian and Spherical Uniform Distribution

In this section, we introduce basic properties of multivariate Gaussian and the relation between Gaussian and spherical uniform distribution.

First we consider 1D space $\mathbb{R}$, choose $e \in \mathbb{R}$ and $\sigma > 0$, the Gaussian distribution, denoted as $\mathcal{N}(e, \sigma^2)$, is the probability measure whose density is defined by

$$f(x) := \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-e)^2}{2\sigma^2}},$$

where e, σ^2 are the expected value and variance of X respectively.

When $e = 0, \sigma^2 = 1$, the induced measure is called standard (1D) Gaussian distribution, whose density is given by

$$f(x) := \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}} \tag{33}$$

In space $\mathbb{R}^d$, the above density function can be generalized as:

$$f(x) := \frac{1}{(2\pi)^{d/2}} e^{-\frac{\|x\|^2}{2}} \tag{34}$$

and the induced distribution is called **d-dimensional Standard Gaussian distribution**.

Given $e \in \mathbb{R}^d$ and positive definite $d \times d$ matrix, $\Sigma = AA^T$ where $A \in \mathbb{R}^{d\times k}$, the Gaussian distribution is denoted as $\mathcal{N}(e, \Sigma)$, can be defined by the following well-known proposition:

Proposition A.7 (Definition of Gaussian distribution)**.** *Let $X \sim \mathcal{N}(e, \Sigma)$ be a realization, then the following are equivalent:*

- *$\mathcal{N}(e, \Sigma)$ is Gaussian distribution, with expected value e and covariance matrix Σ.*
- *$X = AG + e$, where $G \sim \mathcal{N}(0, I_d)$, whose density is defined by Equation 34.*
- *$\forall \theta \in \mathbb{R}^d$, $\theta^\top X$ is a 1D Gaussian variable:*

$$\theta^\top X \sim \mathcal{N}(\theta^\top e, (\theta^\top A)^\top (\theta^\top A)).$$

From the proposition, it is straightforward to verify the following:

Proposition A.8 (Basic properties of Gaussian distribution)**.** *Suppose $X \sim \mathcal{N}(e, \Sigma)$, then we have:*

(1) If rank$(\Sigma) = d$, then $\mathcal{N}(e, \Sigma)$ admits the density function:

$$f(x) = \frac{1}{(2\pi)^{d/2} det(\Sigma)^{1/2}} e^{-\frac{(x-e)^T \Sigma^{-1}(x-e)}{2}}$$

(2) Choose $B \in \mathbb{R}^{d\times k}, \beta \in \mathbb{R}^k$, and let $T_{B,e,\beta}(x) := B(x - e) + \beta$, then we have

$$B(X - e) + \beta \sim (T_{B,e,\beta})_\# \mathcal{N}(e, \Sigma) = \mathcal{N}(\beta, B^\top \Sigma B).$$

(3) Suppose $Z \sim \mathcal{N}(0, I_d)$, then the absolute p-th power of Z is given by

$$\mathbb{E}[\|Z\|^p] = 2^{p/2} \frac{\Gamma(\frac{p+d}{2})}{\Gamma(d/2)}.$$

(4) Suppose $Z \sim \mathcal{N}(0, I_d)$, then $r = \|Z\|, \theta = \frac{Z}{\|Z\|}$ are independent.

At the end of this section, we introduce the following relation between the Gaussian distribution and the spherical uniform distribution.

Proposition A.9. *We define the following function f with*

$$\mathbb{R}^d \setminus \{0\} \ni x \mapsto f(x) = \frac{x}{\|x\|}.$$

Suppose $\Sigma = AA^\top$ is a full rank positive-semi-definite matrix, then we have

$$f_\# \mathcal{N}(0_d, \Sigma) = \mathcal{U}(\mathbb{S}^{d-1}).$$

Proof. Let $X \sim \mathcal{N}(0_d, \Sigma)$ be a realization of the d-dimensional Gaussian, $\Theta = f(X) = \frac{X}{\|X\|}$. Note that Θ is well defined $\mathcal{N}(0_d, \Sigma)$-a.s.

Step 1. Suppose $\Sigma = I_d$, it is equivalent to the following:

Suppose $X_1, \dots X_d \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0,1)$ and $\Theta = [\frac{X_1}{\sqrt{\sum_{i=1}^d X_i^2}}, \dots, \frac{X_d}{\sqrt{\sum_{i=1}^d X_i^2}}]^T$, then $\Theta \sim \text{Unif}(\mathbb{S}^{d-1})$. It is a standard result in probability theory. In particular, choose test function $\phi \in C_0(\mathbb{S}^{d-1})$, we have:

$$\begin{aligned}
\mathbb{E}[\phi(\Theta)] &= \int_{\mathbb{R}^d} \phi(\frac{x}{\|x\|}) f_X(x) dx \\
&= \frac{1}{(2\pi)^{d/2}} \int_{\mathbb{R}^d} \phi\left(\frac{x}{\|x\|}\right) e^{-\frac{\|x\|^2}{2}} dx \\
&= \frac{1}{(2\pi)^{d/2}} \int_{\mathbb{S}^{d-1}} \int_{\mathbb{R}_+} \phi(\theta) e^{-r^2/2} r^{d-1} d\theta dr \qquad\qquad r, \theta \text{ are spherical coordinates} \\
&= \int_{\mathbb{S}^{d-1}} \phi(\theta) d\theta \cdot \underbrace{\frac{1}{(2\pi)^{d/2}} \int_{\mathbb{R}_+} e^{-r^2/2} r^{d-1} dr}_{1/\|\mathbb{S}^{d-1}\|}
\end{aligned}$$

Thus, $\Theta \sim \text{Unif}(\mathbb{S}^{d-1})$.

Step 2. Suppose $\Sigma = \text{diag}(\sigma_1, \dots \sigma_d)$ where $\sigma_1, \dots \sigma_d > 0$, we have

$$\Theta = \frac{X}{\|X\|} = \frac{\Sigma^{-1/2} X}{\|\Sigma^{-1/2} X\|},$$

where $\Sigma^{-1/2} X \sim \mathcal{N}(0, I_d)$. Thus, by step 1, we have $\Theta \sim \mathcal{U}(\mathbb{S}^{d-1})$.

Step 3. We consider the general positive definite Σ. We have $\Sigma = U \Lambda U^\top$ where $U \in \mathbb{V}_{d,d}$ is orthonormal matrix.

We have

$$U^\top \Theta = \frac{U^\top X}{\|X\|} = \frac{U^\top X}{\|U^\top X\|}$$

Since $U^\top X \sim \mathcal{N}(0, \Lambda)$ and Λ is a positive diagonal matrix, then from step 2, we have $U^\top \Theta \sim \mathcal{U}(\mathbb{S}^{d-1})$. Thus, $\Theta = U(U^\top \Theta) \sim \mathcal{U}(\mathbb{S}^{d-1})$.

□

Remark A.10. *Note that the above statement (especially the statement in Step 1) is a well-known result, and that is why isotropic Gaussian distribution is called a "rotationally invariant distribution." We do not claim this proposition or its proof as contributions of this article; we present the proof merely for completeness.*

A.4 Relationship between the SWD in $\mathbb{R}^d$ and $\mathbb{R}^k$

In this section, we discuss the proof of Theorem 4.7. We first introduce some intermediate results in the following subsection.

A.4.1 Relationship between $SW_p^p(\mu^d, \nu^d; \mathcal{U}(\mathbb{S}^{d-1}))$ and $SW_p^p(\mu^d, \nu^d; \mathcal{N}(0, I_d))$

The main result in this section is the following proposition

Proposition A.11. *Choose $\mu, \nu \in \mathcal{P}(\mathbb{R}^d)$, we have*

$$2^{p/2} \frac{\Gamma(\frac{p+d}{2})}{\Gamma(d/2)} SW_p^p(\mu, \nu; \mathcal{U}(\mathbb{S}^{d-1})) = SW_p^p(\mu, \nu; \mathcal{N}(0, I_d)) \tag{35}$$

Remark A.12. *If we replace $\mathcal{N}(0, I_d)$ by $\mathcal{N}(0, \frac{1}{d} I_d)$, the corresponding conclusion has been proved by (Nadjahi et al., 2021, Proposition 1). Thus, we do not claim the above statement and related proof as part of the contribution in this paper. We present this statement and the related proof for the readers' convenience.*

To prove the above statement, first it is straightforward to verify the following:

Lemma A.13. *Given $\alpha \in \mathbb{R}$, with abuse of notations, we let $\alpha_{\#}\mu$ denote the pushforward measure of μ under mapping $x \mapsto \alpha x$, then we have*

$$|\alpha|^p W_p^p(\mu, \nu) = W_p^p(\alpha_{\#}\mu, \alpha_{\#}\nu) \tag{36}$$

Proof. If $\alpha = 0$, then both sides are zero, and we've done.

If $\alpha \neq 0$, it is straightforward to verify the following is a well-defined bijection:

$$\Gamma(\mu, \nu) \ni \gamma \mapsto (\alpha \times \alpha)_{\#}\gamma \in \Gamma(\alpha_{\#}\mu, \alpha_{\#}\nu) \tag{37}$$

where $(\alpha \times \alpha)$ denotes the mapping

$$\mathbb{R}^2 \ni (x, x') \mapsto (\alpha x, \alpha x') \in \mathbb{R}^2.$$

Pick $\gamma \in \Gamma(\mu, \nu)$, we have

$$\begin{aligned}
&|\alpha|^p \int_{\mathbb{R}^2} |x - y|^p d\gamma(x, y) \\
&= \int_{\mathbb{R}^2} |\alpha x - \alpha y|^p d\gamma \\
&= \int_{\mathbb{R}^2} |x - y|^p d(\alpha \times \alpha)_{\#}\gamma(x, y)
\end{aligned}$$

Take the infimum for both sides over $\Gamma(\mu, \nu)$, combine it with the fact that Equation 37 is a bijection. We obtain Equation 36. □

Now we introduce the proof of Proposition A.11.

Proof. Suppose $\theta^g \sim \mathcal{N}(0, I_d)$ and let $\theta = \frac{\theta^g}{\|\theta^g\|}$, we have $\theta \sim \mathcal{U}(\mathbb{S}^{d-1})$ by Proposition A.9. Then we have:

$$\begin{aligned}
&SW_p^p(\mu, \nu; \mathcal{N}(0, I_d)) \\
&= \mathbb{E}_{\theta^g \sim \mathcal{N}(0, I_d)}[W_p^p(\theta^g_{\#}\mu, \theta^g_{\#}\nu)] \\
&= \mathbb{E}_{\theta^g \sim \mathcal{N}(0, I_d)}[\|\theta^g\|^p W_p^p(\theta_{\#}\mu, \theta_{\#}\nu)] && \text{by Lemma A.13} \\
&= \mathbb{E}_{\theta^g \sim \mathcal{N}(0, I_d)}[\|\theta^g\|^p] \cdot \mathbb{E}_{\theta \sim \mathcal{U}(\mathbb{S}^{d-1})}[W_p^p(\theta_{\#}\mu, \theta_{\#}\nu)] && \text{by Proposition A.8 (4)} \\
&= 2^{p/2} \frac{\Gamma(\frac{p+d}{2})}{\Gamma(d/2)} \cdot SW_p^p(\mu, \nu; \mathcal{U}(\mathbb{S}^{d-1})) && \text{by Proposition A.8 (3).}
\end{aligned}$$

□

A.5 Proof of Proposition A.4

We adapt notations V_k, U in previous subsection.

Lemma A.14. *Suppose μ^d, ν^d satisfy Assumption 4.1, pick $\theta^d \in \mathbb{R}^d$ and let $\hat{\theta}^k = U^\top \theta^d$ then we have:*

$$\theta_{\#}\mu^d = \hat{\theta}^k_{\#}\mu^k, \theta_{\#}\nu^d = \hat{\theta}^k_{\#}\nu^k.$$

Proof. For each $x \in \text{Span}(U) = V_k$, we have

$$\begin{aligned}
\theta^\top x &= P_U(\theta)^\top x + (\theta - P_U(\theta))^\top x \\
&= P_U(\theta)^\top x + 0 && \text{Since } \theta - P_U(\theta) \in V_k^\perp \\
&= (UU^\top \theta)^\top x \\
&= (U^\top \theta)^\top (U^\top x)
\end{aligned}$$

Thus,

$$\theta^d_{\#}\,\mu^d \;=\; (U^\top\theta^d)_{\#}\,\mu^k \;=\; \hat{\theta}^k_{\#}\,\mu^k$$

Similarly, we have $\theta^d_{\#}\nu^d = \hat{\theta}^k_{\#}\nu^k$ and we complete the proof. □

Lemma A.15. *Suppose* $\theta^d_1, \dots \theta^d_L \overset{i.i.d.}{\sim} \mathcal{U}(\mathbb{S}^{d-1})$ *and let* $\theta^k_l = \frac{U^\top\theta}{\|U^\top\theta\|}, \forall l \in [1:L]$, *then* $\theta^k_1, \dots \theta^k_L \overset{i.i.d}{\sim} \mathcal{U}(\mathbb{S}^{k-1})$.

Proof. First, since $k < d$, we have

$$\mathcal{U}(\theta^d \in \mathbb{S}^{d-1} : U^\top\theta^d = 0_k) = 0.$$

Thus, with probability 1, θ^k_l is well-defined.

By Proposition A.9, with probability 1, we can redefine $\theta^d_1, \dots \theta^d_N$ by the following way:

Suppose $X_1, \dots X_n \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, I_d)$, $\theta^d_l = \frac{X_l}{\|X_l\|}$.

Then

$$\theta^k_l = \frac{U^\top\theta^d_l}{\|U^\top\theta^d_l\|} = \frac{U^\top X_l/\|X_l\|}{\|U^\top X_l/\|X_l\|\|} = \frac{U^\top X_l}{\|U^\top X_l\|}$$

Since $U^\top X_l \sim \mathcal{N}(0, I_k)$, we have $\theta^k_l \sim \mathcal{U}(\mathbb{S}^{k-1})$.

Furthermore, since $X_1, \dots X_N$ are independent, we have $\theta^k_1, \dots \theta^k_N$ are independent. Thus, $\theta^1_1, \dots \theta^k_N \overset{\text{i.i.d.}}{\sim} \mathcal{U}(\mathbb{S}^{k-1})$. □

Now we discuss the proof of Proposition 4.6.

Proof of Proposition . Pick $\theta^d \in \mathbb{S}^{d-1}$.

We have

$$\begin{aligned} W(\theta^d_{\#}\mu^d, \theta^d_{\#}\nu^d) &= W((U^\top\theta)_{\#}\mu^k, (U^\top\theta)_{\#}\nu^k) && \text{By Lemma A.14} \\ &= \|U^\top\theta^d\|^p W(\theta^k_{\#}\mu^k, \theta^k_{\#}\nu^k) && \text{By Lemma A.13} \end{aligned}$$

Thus we prove Equation 13.

Now, we pick $\theta^d_1, \dots \theta^d_N \in \mathbb{S}^{d-1}$, and thus we have:

$$\begin{aligned} & SW^p_p(\mu^k, \nu^k; \frac{1}{L}\sum_{l=1}^{L}\delta_{\theta^k_l}) \\ &= \frac{1}{L}\sum_{l=1}^{L} W^p_p((\theta^k_l)_{\#}\mu^k, (\theta^k_l)_{\#}\nu^k) && \theta^k_l = 0_k \text{ if } \|U^\top\theta^d_l\| = 0 \\ &= \frac{1}{L}\sum_{l=1}^{L} \frac{1}{\|U^\top\theta^d_l\|^p} W^p_p((U^\top\theta^d_l)_{\#}\mu^k, (U^\top\theta^d_l)_{\#}\nu^k) && \text{By convention } 0\cdot\frac{1}{0} = 0 \\ &= \frac{1}{L}\sum_{i=1}^{N} \frac{1}{\|U^\top\theta^d_l\|^p} W^p_p((\theta^d_l)_{\#}\mu^d, (\theta^d_l)_{\#}\nu^d) && \text{by Equation 13} \\ &= \widetilde{SW}^p_p\left(\mu^d, \nu^d; \frac{1}{N}\sum_{i=1}^{N}\delta_{\theta^d_l}, \rho\right) \end{aligned}$$

And we prove Equation 14.

Similarly, we obtain the last equation,

$$
\begin{aligned}
\widetilde{SW}_p^p(\mu^d, \nu^d; \mathcal{U}(\mathbb{S}^{d-1}), h) &= \mathbb{E}_{\theta^d \sim \mathcal{U}(\mathbb{S}^{d-1})} \left[\frac{1}{\|U^\top \theta^d\|^p} W_p^p((\theta^d)_\# \mu^d, (\theta^d)_\# \nu^d) \right] \\
&= \mathbb{E}_{\theta^d \sim \mathcal{U}(\mathbb{S}^{d-1})} \left[W_p^p((\theta^k)_\# \mu^k, (\theta^k)_\# \nu^k) \right] && \text{By Equation 13} \\
&= \mathbb{E}_{\theta^k \sim \mathcal{U}(\mathbb{S}^{k-1})} [W_p^p((\theta^k)_\# \mu^k, (\theta^k)_\# \nu^k)] && \text{By Lemma A.15} \\
&= SW_p^p(\mu^k, \nu^k)
\end{aligned}
$$

□

A.5.1 Proof of Theorem 4.7

In this section, we first discuss the relation between $SW_p^p(\mu^d, \nu^d; \mathcal{N}(0, I_d))$ and $SW_p^p(\mu^k, \nu^k; \mathcal{N}(0, I_k))$ under Assumption 4.1. Next, we present the proof of Theorem 4.7.

Based on the above lemma, we can derive the following relation between $SW_p^p(\mu^d, \nu^d; \mathcal{N}(0, I_d))$ and $SW_p^p(\mu^k, \nu^k; \mathcal{N}(0, I_k))$.

Lemma A.16. *Under Assumption 4.1, we have*

$$
SW_p^p(\mu^d, \nu^d; \mathcal{N}(0, I_d)) = SW_p^p(\mu^k, \nu^k; \mathcal{N}(0, I_k)) \tag{38}
$$

Proof. Suppose $\theta^d \sim \mathcal{N}(0, I_d)$ and let $\theta^k = U^\top \theta^d$. Then by Proposition A.8 (1), we have $\theta^k \sim \mathcal{N}(0, U^\top I_d U) = \mathcal{N}(0, I_k)$. Therefore,

$$
\begin{aligned}
&SW_p^p(\mu^d, \nu^d; \mathcal{N}(0, I_d)) \\
&= \mathbb{E}_{\theta^d \sim \mathcal{N}(0, I_d)} [W_p^p(\theta^d_\# \mu^d, \theta^d_\# \nu^d)] \\
&= \mathbb{E}_{\theta^d \sim \mathcal{N}(0, I_d)} [W_p^p(\theta^k_\# \mu^k, \theta^k_\# \nu^k)] && \text{By Lemma A.14, where } \theta^k = U^\top \theta^d \\
&= \mathbb{E}_{\theta^k \sim \mathcal{N}(0, I_k)} [W_p^p(\theta^k_\# \mu^k, \theta^k_\# \nu^k)] \\
&= SW_p^p(\mu^k, \nu^k; \mathcal{N}(0, I_k))
\end{aligned}
$$

and we complete the proof. □

Combining the above lemma and Proposition A.11, we can prove Theorem 4.7.

Proof of Theorem 4.7. For the first equality, we have

$$
\begin{aligned}
&SW_p^p(\mu^d, \nu^d; \mathcal{U}(\mathbb{S}^{d-1})) \\
&= \frac{1}{C_d} SW_p^p(\mu^d, \nu^d; \mathcal{N}(0_d, I_d)) && \text{By Proposition A.11} && (39) \\
&= \frac{1}{C_d} SW_p^p(\mu^k, \nu^k; \mathcal{N}(0_k, I_k)) && \text{By Lemma A.16} \\
&= \frac{C_k}{C_d} SW_p^p(\mu^k, \nu^k; \mathcal{U}(\mathbb{S}^{k-1})) && \text{By Proposition A.11} && (40)
\end{aligned}
$$

where $C_d = 2^{p/2} \frac{\Gamma\left(\frac{d}{2} + \frac{p}{2}\right)}{\Gamma\left(\frac{d}{2}\right)}$ and C_k is defined similarly.

□

A.6 Proof of Proposition 4.8

We first introduce the following lemma:

Lemma A.17. *Let $I_{d\times k}$ denote the matrix $\begin{bmatrix} I_{k\times k} \\ 0_{(d-k)\times k} \end{bmatrix}$, and suppose $\theta^d \sim \mathcal{U}(\mathbb{S}^{d-1})$, then $\|U^\top \theta^d\|$ and $\|I_{d\times k}^\top \theta^d\|$ have the same distribution.*

Proof. Since U has orthonormal columns, there exist orthogonal matrices $V_1 \in \mathbb{R}^{d\times d}$ and $V_2 \in \mathbb{R}^{k\times k}$ such that $U = V_1 I_{d\times k} V_2$.

Then we have

$$\|U^\top \theta^d\| = \|V_2^\top I_{d\times k}^\top V_1^\top \theta^d\| = \|I_{d\times k}^\top V_1^\top \theta^d\|$$

Since $\theta^d \sim \mathcal{U}(\mathbb{S}^{d-1})$, then $V_1^\top \theta^d \sim \mathcal{U}(\mathbb{S}^{d-1})$.

Thus, $I_{d\times k}^\top \theta^d, I_{d\times k}^\top V_1^\top \theta^d$ have the same distribution. Thus $\|I_{d\times k}^\top \theta^d\|, \|I_{d\times k}^\top V_1^\top \theta^d\| = \|U^\top \theta^d\|$ have the same distribution. □

Based on this, we can prove the statement (1) in Proposition 4.8.

Proof of Proposition 4.8 (1). By the above lemma, it is sufficient to consider $U = I_{d\times k}$.

Let $\theta^{d,g} \sim \mathcal{N}(0, I_d)$, and let $\theta^{d,g}[i], i \in [1:d]$ denote each component of $\theta^{d,g}$. Thus $\theta^{d,g}[1], \dots \theta^{d,g}[d] \overset{i.i.d}{\sim} \mathcal{N}(0,1)$. We can redefine θ^d as $\theta^d = \frac{\theta^{d,g}}{\|\theta^{d,g}\|}$, thus,

$$\begin{aligned} \|U^\top \theta^d\|^2 &= \frac{\|U^\top \theta^{d,g}\|^2}{\|\theta^{d,g}\|^2} \\ &= \frac{\sum_{i=1}^k \theta^{d,g}[i]^2}{\sum_{i=1}^d \theta^{d,g}[i]^2} \sim \text{Beta}(\frac{k}{2}, \frac{d-k}{2}) \end{aligned}$$

Thus, we have

$$\mathbb{E}[\|U^\top \theta^d\|^p] = \mathbb{E}[(\|U^\top \theta^d\|^2)^{p/2}] = \frac{\Gamma(k/2+p/2)\Gamma(d/2)}{\Gamma(k/2)\Gamma(d/2+p/2)} = \frac{C_k}{C_d}.$$

Note, $\|U^\top \theta_1^d\|, \dots \|U^\top \theta_L^d\|$ are i.i.d. random variables, thus, we have

$$\mathbb{E}[\widehat{ESSF}(L)] = \frac{1}{L}\mathbb{E}[\sum_{l=1}^L \|U^\top \theta_l^d\|^p] = \frac{C_k}{C_d}.$$

Similarly,

$$\text{Var}[\widehat{ESSF}(L)] = \frac{1}{L}\text{Var}[\|U^\top \theta_l^d\|^p]$$

where $\text{Var}[\|U^\top \theta_l^d\|^p] > 0$, is the variance of the $p/2-$th power of a $\text{Beta}(k/2, (d-k)/2)$ variable, which is a constant only depends on (d, k, p).

□

Proof of Proposition 4.8(2)–(3). Write $Z_l := \|U^\top \theta_l^d\|^p \in [0,1]$ and $\bar{Z} := \frac{1}{L}\sum_{l=1}^L Z_l = \widehat{ESSF}(L)$. From Proposition 4.6 and the discrete gradient formula, each per-sample gradient satisfies

$$\nabla_{x_i} W_p^p(\theta_\# \hat{\mu}^d, \theta_\# \hat{\nu}^d) \;=\; Z\, \nabla_{x_i} W_p^p(\theta_\#^k \hat{\mu}_k, \theta_\#^k \hat{\nu}_k),$$

with $\|\nabla_{x_i} W_p^p(\theta_\#^k \hat{\mu}_k, \theta_\#^k \hat{\nu}_k)\| \leq K$ for some $K < \infty$ depending only on the (compact) supports. Hence

$$\|\epsilon_L(x_i)\| \leq \frac{pq_i^1}{L} \sum_{l=1}^{L} |Z_l - \bar{Z}| \left\|\nabla_{x_i} W_p^p(\theta_l^k{}_\# \hat{\mu}_k, \theta_l^k{}_\# \hat{\nu}_k)\right\| \leq pq_i^1 K \frac{1}{L} \sum_{l=1}^{L} |Z_l - \bar{Z}|.$$

Using $|Z_l - \bar{Z}| \leq |Z_l - \mathbb{E}Z| + |\bar{Z} - \mathbb{E}Z|$ and averaging, we obtain

$$\|\epsilon_L(x_i)\| \;\leq\; 2pq_i^1 K\, |\bar{Z} - \mathbb{E}Z|.$$

Since $Z_l \in [0,1]$ are i.i.d., Hoeffding's inequality gives for any $\delta > 0$,

$$\mathbb{P}\Big(\, |\bar{Z} - \mathbb{E}Z| \geq \delta \,\Big) \;\leq\; 2\exp\!\big(-2\delta^2 L\big).$$

This yields (2) by the strong law of large numbers (a.s. convergence), and (3) by setting $\delta \mapsto \delta/(2pq_i^1 K)$ above.

□

A.6.1 Proof of Proposition 4.8

A.7 Special case: Learning rate bound for the SWD Gradient Flow problem

In this section, we consider the following sliced gradient flow problem Bonet et al. (2021a):

$$\begin{aligned}
&\mu_{t+1} \leftarrow \arg\min_{\mu \in \mathcal{P}_2(\mathbb{R}^k)} \frac{1}{2\tau} SW_2^2(\mu, \mu_t) + F(\mu) \\
&s.t. \mu_0 = \mu^k \\
&\text{where } F(\mu) := SW_2^2(\mu, \nu^k), \text{for some } \nu^k, \tau > 0
\end{aligned}$$

In the discrete setting, $\mu^k = \sum_{i=1}^n q_i^1 \delta_{x_i}, \nu^k = \sum_{j=1}^m q_j^2 \delta_{y_j}$. Furthermore, we assume that the pmf of μ_t is fixed. Then the above problem can be transferred to the following:

$$X^{t+1} \leftarrow X^t - h_t \odot \nabla_X SW_2^2(\mu_t, \nu^k), \text{ where } \mu_t = \sum_{i=1}^n q_i^1 \delta_{x_i^t}, X^t = [x_1, \ldots, x_n] \tag{41}$$

where $\odot$ denote the element-wise product operator, and $h_t \in \mathbb{R}_+^n$.

We will discuss how to select the appropriate learning rate h_t.

Gradient and Hessian of Sliced Wasserstein distance. First, we discuss the gradient and Hessian matrix of the function $X \mapsto SW_2^2(\mu, \nu^k)$:

Pick $\theta \in \mathbb{S}^{d-1}$ and suppose that γ_θ is an optimal transportation plan for $W_2^2(\theta_\# \mu, \theta_\# \nu^k)$.

Then by Bonneel & Coeurjolly (2019), we have:

$$\nabla_{x_i} W_2^2(\theta_\# \mu, \theta_\# \nu^k) = 2\theta\theta^\top (q_i^1 x_i - \sum_{j=1}^m y_j \gamma_{i,j}^\theta), \forall x_i$$

Note, when $W_2^2(\theta_\# \mu, \theta_\# \nu^k)$ is induced by a Monge mapping, the above formulation can be simplified to $q_i^1 \theta\theta^\top (x_i - T(x_i))$.

Thus the Hessian matrix is

$$\left[\frac{\partial^2 W_2^2(\theta_\# \mu, \theta_\# \nu^k)}{\partial x_i[l] \partial x_i[l']}\right]_{l,l' \in [1:d]} = 2q_i^1 \theta\theta^\top.$$

Therefore, the gradient for mapping $X \mapsto SW_2^2(\mu, \nu^k)$ with respect to each x_i is given by:

$$g(x_i) := \nabla_{x_i} SW_2^2(\mu, \nu^k) = 2\int_{\mathbb{S}^{d-1}} \theta\theta^\top (q_i^1 x_i - \sum_{i=1}^m y_j \gamma_{i,j}^\theta) d\mathcal{U}(\mathbb{S}^{d-1})(\theta)$$
$$\approx \frac{2}{N}\sum_{l=1}^L \theta_l \theta_l^\top (q_i^1 x_i - \sum_{j=1}^m y_j \gamma_{i,j}^{\theta_l})$$

where the second line is the Monte carlo approximation.

Similarly, the Hession matrix and the Monte carlo approximation are given by

$$H(x_i) := H_{x_i}(SW_2^2(\mu, \nu^k)) = 2q_i^1 \int \theta\theta^\top d\mathcal{U}(\mathbb{S}^{d-1})(\theta) = 2q_i^1 \frac{1}{k} I_k$$
$$\approx \frac{2q_i^1}{N}\sum_{i=1}^N \theta\theta^\top$$

By classical machine learning theory, the optimal learning rate for x_i, is given by

$$(h_t)_i = \frac{g(x_i)^\top g(x_i)}{g(x_i)^\top H g(x_i)} = \frac{k}{2q_i^1}, \forall i \in [1:n] \tag{42}$$

Remark A.18. *We consider a simplified case to intuitively understand the above learning rate. Suppose $\mu^k = q_i^1 \delta_{x_i}$ and $\nu^k = q_i^1 \delta_{y_j}$ (relaxing the assumption that μ^k and ν^k are probability measures). Then, we have:*

$$\begin{aligned}
&SW_2^2(\mu^d, \nu^d) \\
&= q_i^1 \mathbb{E}_{\theta\sim\mathcal{U}(\mathbb{S}^{d-1})}[(\theta^\top x_i - \theta^\top y_j)^2] \\
&= q_i^1 \mathbb{E}_{\theta\sim\mathcal{U}(\mathbb{S}^{d-1})}[\|\theta\theta^\top x_i - \theta\theta^\top y_j\|^2] \\
&= q_i^1 (x_i - y_j)^\top \mathbb{E}[\theta\theta^\top](x_i - y_j) \\
&= \frac{1}{k} q_i^1 \|x_i - y_j\|_2^2 \\
&= \frac{1}{k} W_2^2(\mu, \nu).
\end{aligned}$$

Thus the gradient with respect to x_i becomes

$$g(x_i) = 2\frac{q_i^1}{k}(x_i - y_j).$$

Letting $t = 0$, we plug the learning rate from Equation 42 and the gradient into Equation 41, obtaining:

$$x_i^{t+1} \leftarrow x_i^t - (y_j - x_i) = y_j.$$

Intuitively, the learning rate $(h_t)_i$ for x_i is chosen such that the (negative) gradient becomes the displacement given by the classical OT transportation plan, i.e.,

$$-g(x_i) \approx y_j - x_i.$$

That is, when θ is sufficiently large (i.e., $\frac{1}{L}\sum_{i=1}^L \theta\theta^\top \approx \frac{1}{k} I_k$), μ_t^k will converge to ν^k in one step.

A.8 Proof of Proposition 4.10

Pick x_i from $\{x_1, \dots, x_n\}$. Note, based on Assumption 4.1, $x_i = UU^\top x_i = Ux_i^k, \forall i \in [1:n]$. Thus, we have

$$\begin{aligned}
&\nabla_{x_i} SW_p^p(\hat{\mu}, \hat{\nu}; \frac{1}{L}\sum_{l=1}^{L}\delta_{\theta_l}) \\
&= U\nabla_{x_i^k} SW_p^p(\hat{\mu}, \hat{\nu}; \frac{1}{L}\sum_{l=1}^{L}\delta_{\theta_l}) \\
&= U\nabla_{x_i^k} \sum_{l=1}^{L}\frac{1}{L}\sum_{i,j}(q_i^1\|U^\top\theta_l\|(\theta_l^k)^\top(x_i^k - \frac{1}{q_i^1}y_j^k\gamma_{i,j}^{\theta_l}))^p \\
&= q_i^1 pU\frac{1}{L}\sum_{l=1}^{L}\|U^\top\theta_l\|(\theta_l^k)^\top(x_i^k - \frac{1}{q_i^1}y_j^k\gamma_{i,j}^{\theta_l})^{p-1}.
\end{aligned}$$

Similarly,

$$\begin{aligned}
&\nabla_{x_i} SW_p^p(\hat{\mu}^k, \hat{\nu}^k; \frac{1}{L}\sum_{l=1}^{L}\delta_{\theta_l^k}) \\
&= q_i^1 pU\frac{1}{L}\sum_{l=1}^{L}(\theta_l^k)^\top(x_i^k - \frac{1}{q_i^1}y_j^k\gamma_{i,j}^{\theta_l})^{p-1}
\end{aligned} \tag{43}$$

where γ^{θ_l} is the optimal transportation plan for 1D problem $W_p^p((\theta_l)_\#\hat{\mu}, (\theta_l)_\#\hat{\nu}) = W_p^p((\theta_l^k)_\#\hat{\mu}^k, (\theta_l^k)_\#\hat{\nu}^k)$. Thus,

$$\begin{aligned}
\epsilon_L(x_i) &= \nabla_{x_i} SW_p^p(\hat{\mu}^d, \hat{\nu}^d; \sum_{l=1}^{L}\delta_{\theta_l^d}) - \widehat{ESSF}(L)\cdot\nabla_{x_i} SW_p^p(\hat{\mu}_k, \hat{\nu}_k; \sum_{l=1}^{L}\delta_{\theta_l^k}) \\
&= \frac{pq_i^1}{L}U\sum_{l=1}^{L}(\|U^\top\theta_l\|^p - \frac{1}{L}\sum_{l'=1}^{L}\|U^\top\theta_{l'}\|^p)\underbrace{\theta_l^k((\theta_l^k)^\top(x_i^k - \frac{1}{q_i^1}\sum_{j=1}^{m}y_j^k\gamma_{i,j}^{\theta_l}))^{p-1}}_{A(\theta_l^k)}.
\end{aligned}$$

where $A(\theta_l^k)$ is a vector function from $\mathbb{S}^{k-1}$ to $\mathbb{R}^k$. By Cauchy–Schwarz inequality, and the fact $\|\theta_l^k\| = 1$, we have

$$\|A(\theta_l^k)\| \le \max_{x_i, y_j}\|x_i^k - y_j^k\|^{p-1} = \max_{x_i, y_j}\|x_i - y_j\|^{p-1}$$

Then we have:

$$\begin{aligned}
\|\epsilon_L(x_i)\| &= pq_i^1\left|\underbrace{\sum_{l=1}^{L}\frac{1}{L}\left(\|U^\top\theta_l\|^p - \frac{1}{L}\sum_{l'=1}^{L}\|U^\top\theta_{l'}\|^p\right)}_{B_L}\right|\|UA(\theta_l^k)\| \\
&= pq_i^1\|A(\theta_l^k)\||B_L| \\
&\le pq_i^1 K|B_L|.
\end{aligned}$$

By law of large number, with probability 1, $B_L \to 0$, thus $\|(\epsilon_L)\| \to 0$, that is $\epsilon_L \to 0_d$.

It remains to bound the convergence rate of $\|\epsilon_L\|$.

By Hoeffding inequality and the fact $\|U^\top \theta_l\|^2 \in [0,1]$, we have

$$\mathbb{P}(|B_L| \geq \epsilon) \leq 2e^{-\epsilon^2 L}.$$

Replacing ϵ by $\epsilon/(pq_i^1 K)$, we obtain:

$$\mathbb{P}(\|\epsilon_L(x_i)\| \leq \epsilon) \geq 1 - 2e^{-\epsilon^2 L/(pq_i^1 K)^2}.$$

and we complete the proof.

A.9 Discussion when Assumption 4.1 is not satisfied

In this section, we briefly discuss the context when Assumption 4.1 is not satisfied. In particular, we aim to show the following:

Proposition A.19. *Let U, V_k be defined in Assumption 4.1, choose $\mu^d, \nu^d \in \mathcal{P}_p(\mathbb{R}^d)$, and let μ^k, ν^k be defined by $(U^\top)_\# \mu^d, (U^\top)_\# \nu^d$, we claim the following:*

$$W_2^2(\mu^k, \nu^k) \leq W_2^2(\mu^d, \nu^d) \leq W_2^2(\mu^k, \nu^k) + 2(m_2(U_\#^\perp \mu^d) + m_2(U_\#^\perp \nu^d)) \tag{44}$$

where $m_2(U_\#^\perp \mu^d)$ denotes the second moment of the measure $U_\#^\perp \mu^d$.

Proof. For each pair $(x, y) \in (\mathbb{R}^d)^2$, we have

$$\begin{aligned}
&\|P_U(x) - P_U(y)\|^2 \\
&\leq \|x - y\|^2 && \text{By definition of projection} && (45)\\
&= \|P_U(x) - P_U(y)\|^2 + \|P_{U^\perp}(x) - P_{U^\perp}(y)\|^2 && \text{Pythagorean theorem} \\
&\leq \|P_U(x) - P_U(y)\|^2 + 2\|(U^\perp)^\top x\|^2 + 2\|(U^\perp)^\top y\|^2 && && (46)
\end{aligned}$$

From Equation 45, we have

$$W_2^2((P_U)_\# \mu^d, (P_U)_\# \nu^d) \leq W_2^2(\mu^d, \nu^d).$$

Combining it with Proposition A.4, we have:

$$W_2^2(\mu^k, \nu^k) = W_2^2((P_U)_\# \mu^d, (P_U)_\# \nu^d) \leq W_2^2(\mu^d, \nu^d),$$

and we prove the first inequality in Equation 44.

Similarly, let $\gamma \in \Gamma(\mu, \nu)$ be the optimal transportation plan for $W_2^2((P_U)_\# \mu^d, (P_U)_\# \nu^d)$. From Equation 46, we have:

$$\begin{aligned}
&W_2^2(\mu^d, \nu^d) \\
&\leq \mathbb{E}_{(X,Y)\sim\gamma}[\|X - Y\|^2] \\
&\leq \mathbb{E}_{(X,Y)\sim\gamma}[\|P_U(X) - P_U(Y)\|^2 + 2\|(U^\perp)^\top X\|^2 + 2\|(U^\perp)^\top Y\|^2] && \text{By Equation 46} \\
&= W_2^2((P_U)_\# \mu, (P_U)_\# \nu) + 2(m_2((U^\perp)_\# \mu^d) + m_2((U^\perp)_\# \nu^d))
\end{aligned}$$

Thus, we prove the second inequality of Equation 44. □

Proposition A.20. *Based on the same notations of Proposition A.19, we have:*

$$\frac{k}{d} SW_2^2(\mu^k, \nu^k) \leq SW_2^2(\mu^d, \nu^d) \leq \frac{k}{d} SW_2^2(\mu^k, \nu^k) + 2\frac{d-k}{d}(m_2(U_\#^\perp \mu^d) + m_2(U_\#^\perp \nu^d)) \tag{47}$$

Proof. Pick $\theta \in \mathbb{S}^{d-1}$ and $x, y \in \mathbb{R}^d$. We have:

$$
\begin{aligned}
&\|\theta^\top P_U(x) - \theta^\top P_U(y)\|^2 \\
&\leq \|\theta^\top x - \theta^\top y\|^2 && (48)\\
&= \|P_U(\theta)^\top P_U(x) - P_U(\theta)^\top P_U(y)\|^2 + \|P_{U^\perp}(\theta)^\top P_{U^\perp}(x) - P_{U^\perp}(\theta)^\top P_{U^\perp}(y)\|^2 \\
&\leq \|\theta^\top P_U(x) - \theta^\top P_U(y)\|^2 + \|(U^\perp)^\top \theta\|^2 \|(U^\perp)^\top x - (U^\perp)^\top y\|^2 && \text{Cauchy–Schwarz inequality}\\
&\leq \|\theta^\top P_U(x) - \theta^\top P_U(y)\|^2 + 2\|(U^\perp)^\top \theta\|^2 (\|(U^\perp)^\top x\|^2 + \|(U^\perp)^\top y\|^2) && (49)
\end{aligned}
$$

Choose $\theta \in \mathbb{S}^{d-1}$. From Proposition A.19, we have

$$W_2^2(\theta_\#(P_U)_\#\mu^d, \theta_\#(P_U)_\#\nu^d) \leq W_2^2(\theta_\#\mu^d, \theta_\#\nu^d)$$

Take expected value with respect to θ, we have

$$SW_2^2((P_U)_\#\mu^d, (P_U)_\#\nu^d) \leq SW_2^2(\mu^d, \nu^d)$$

Combining this with Theorem 4.7, we prove the first inequality in Equation 47.

Similarly, from Equation 49, we have

$$W_2^2(\theta_\#\mu^d, \theta_\#\nu^d) \leq W_2^2(\theta_\#(P_U)_\#\mu^d, \theta_\#(P_U)_\#\nu^d) + 2\|(U^\perp)^\top \theta\|^2 (m_2(U^\perp_\#\mu^d) + m_2(U^\perp_\#\nu^d)).$$

Take expected value with respect to θ, we obtain:

$$
\begin{aligned}
&SW_2^2(\mu^d, \nu^d) \\
&\leq SW_2^2((P_U)_\#\mu^d, (P_U)_\#\nu^d) + 2\mathbb{E}_\theta[\|U^\perp\theta\|^2](m_2(U^\perp_\#\mu^d) + m_2(U^\perp_\#\nu^d)) \\
&= \frac{k}{d} SW_2^2(\mu^k, \nu^k) + 2\frac{d-k}{d}(m_2(U^\perp_\#\mu^d) + m_2(U^\perp_\#\nu^d))
\end{aligned}
$$

there the last equality holds from Theorem 4.7 and the fact $\|U^\perp\theta\|^2 \sim \text{Beta}(\frac{d-k}{2}, \frac{k}{2})$.

□

B Additional Details for the Numerical Experiments

B.1 Gradient Flow

B.1.1 Background Overview

Let $\mathcal{P}(\mathbb{R}^d)$ denote the space of probability measures on $\mathbb{R}^d$. For $\mu, \nu \in \mathcal{P}(\mathbb{R}^d)$, the gradient flow of the SWD distance in the space of probability measures evolves according to the continuity equation

$$\frac{\partial \mu_t}{\partial t} + \nabla \cdot (v_t \mu_t) = 0, \tag{50}$$

where μ_t is a time-dependent probability measure and v_t the velocity field $v_t = -\nabla \frac{\delta SW_2^2(\mu_t, \nu)}{\delta \mu_t}$. This describes the transport of measure μ_t in the Wasserstein space $\mathcal{P}_2(\mathbb{R}^d)$, commonly referred to as **Wasserstein Gradient Flows** (WGF, Ambrosio et al. (2008))

For numerical simulation in practice, one discretizes this dynamic using a particle approximation. We let $\{x_t^t\}_{i=1}^N$ denote a system of N particles evolving according to the following system of ODEs:

$$\frac{dx_i^t}{dt} = -\nabla_{x_i} SW_2^2(\mu_t^N, \nu), \tag{51}$$

where $\mu_t^N = \frac{1}{L}\sum_{i=1}^L \delta_{x_i^t}$ is the empirical measure based on the particle positions x_i^t.

These WGF particle-based approaches preserve key features of continuous systems and have been widely adopted, especially machine learning applications (Peyré et al. (2019).

B.1.2 Experiments

On classic synthetic datasets

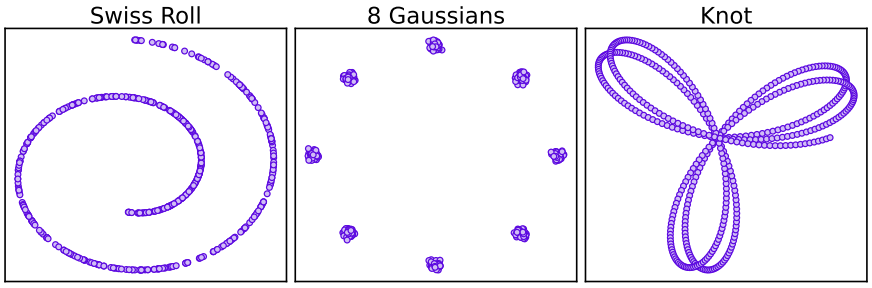


Figure 7: Classic synthetic 2D datasets (shown) embedded in spaces of different target dimensions.

Table 1: Quantitative comparison of the best final converged $W_2(\downarrow)$ and runtime $(\downarrow)$ between different variants for Gradient Flow with (embedded) classic synthetic datasets.

Met.	Swiss			8 Gauss.			Knot			RT(s)↓
	$d=2$	$d=50$	$d=100$	$d=2$	$d=50$	$d=100$	$d=2$	$d=50$	$d=100$	
SWD	$0.0001^{\pm 0.0000}$	$0.0004^{\pm 0.0000}$	$0.0004^{\pm 0.0000}$	$0.0002^{\pm 0.0000}$	$0.0002^{\pm 0.0001}$	$0.0006^{\pm 0.0001}$	$0.0002^{\pm 0.0000}$	$0.0004^{\pm 0.0000}$	$0.0004^{\pm 0.0000}$	$8.62^{\pm 0.04}$
MaxSW	$0.0000^{\pm 0.0000}$	$0.0219^{\pm 0.0051}$	$0.0342^{\pm 0.0022}$	$0.0005^{\pm 0.0000}$	$0.0171^{\pm 0.0004}$	$0.0385^{\pm 0.0006}$	$0.0005^{\pm 0.0000}$	$0.0246^{\pm 0.0009}$	$0.0303^{\pm 0.0009}$	$74.02^{\pm 1.61}$
DSW	$0.0002^{\pm 0.0001}$	$0.0004^{\pm 0.0000}$	$0.0004^{\pm 0.0000}$	$0.0002^{\pm 0.0001}$	$0.0004^{\pm 0.0001}$	$0.0006^{\pm 0.0001}$	$0.0003^{\pm 0.0000}$	$0.0004^{\pm 0.0001}$	$0.0004^{\pm 0.0000}$	$162.25^{\pm 0.20}$
MaxKSW	$0.0002^{\pm 0.0000}$	$0.0124^{\pm 0.0082}$	$0.0122^{\pm 0.0010}$	$0.0002^{\pm 0.0000}$	$0.0154^{\pm 0.0001}$	$0.0216^{\pm 0.0007}$	$0.0002^{\pm 0.0000}$	$0.0165^{\pm 0.0048}$	$0.0171^{\pm 0.0048}$	$125.23^{\pm 0.54}$
iMSW	$0.0001^{\pm 0.0000}$	$0.0021^{\pm 0.0001}$	$0.0050^{\pm 0.0001}$	$0.0001^{\pm 0.0000}$	$0.0021^{\pm 0.0001}$	$0.0059^{\pm 0.0001}$	$0.0002^{\pm 0.0001}$	$0.0034^{\pm 0.0001}$	$0.0054^{\pm 0.0001}$	$74.45^{\pm 0.03}$
viMSW	$0.0002^{\pm 0.0001}$	$0.0003^{\pm 0.0000}$	$0.0005^{\pm 0.0000}$	$0.0003^{\pm 0.0001}$	$0.0003^{\pm 0.0001}$	$0.0008^{\pm 0.0000}$	$0.0003^{\pm 0.0001}$	$0.0005^{\pm 0.0000}$	$0.0005^{\pm 0.0000}$	$255.76^{\pm 0.28}$
oMSW	$0.0001^{\pm 0.0000}$	$0.0002^{\pm 0.0000}$	$0.0005^{\pm 0.0000}$	$0.0002^{\pm 0.0001}$	$0.0002^{\pm 0.0000}$	$0.0006^{\pm 0.0000}$	$0.0002^{\pm 0.0001}$	$0.0004^{\pm 0.0000}$	$0.0004^{\pm 0.0000}$	$16.55^{\pm 0.01}$
rMSW	$0.0002^{\pm 0.0000}$	$0.0003^{\pm 0.0000}$	$0.0005^{\pm 0.0000}$	$0.0003^{\pm 0.0001}$	$0.0003^{\pm 0.0000}$	$0.0008^{\pm 0.0000}$	$0.0003^{\pm 0.0001}$	$0.0006^{\pm 0.0000}$	$0.0005^{\pm 0.0000}$	$179.70^{\pm 1.08}$
EBSW	$0.0002^{\pm 0.0001}$	$0.0002^{\pm 0.0000}$	$0.0005^{\pm 0.0000}$	$0.0001^{\pm 0.0000}$	$0.0002^{\pm 0.0000}$	$0.0006^{\pm 0.0000}$	$0.0003^{\pm 0.0001}$	$0.0004^{\pm 0.0000}$	$0.0002^{\pm 0.0000}$	$9.66^{\pm 1.15}$
RPSW	$0.0001^{\pm 0.0000}$	$0.0001^{\pm 0.0000}$	$0.0004^{\pm 0.0000}$	$0.0002^{\pm 0.0000}$	$0.0001^{\pm 0.0000}$	$0.0010^{\pm 0.0000}$	$0.0002^{\pm 0.0000}$	$0.0001^{\pm 0.0000}$	$0.0004^{\pm 0.0000}$	$19.47^{\pm 0.03}$
EBRPSW	$0.0007^{\pm 0.0002}$	$0.0002^{\pm 0.0001}$	$0.0003^{\pm 0.0000}$	$0.0002^{\pm 0.0001}$	$0.0002^{\pm 0.0001}$	$0.0006^{\pm 0.0000}$	$0.0002^{\pm 0.0001}$	$0.0004^{\pm 0.0000}$	$0.0002^{\pm 0.0000}$	$20.30^{\pm 0.05}$

On the MNIST and CelebA images

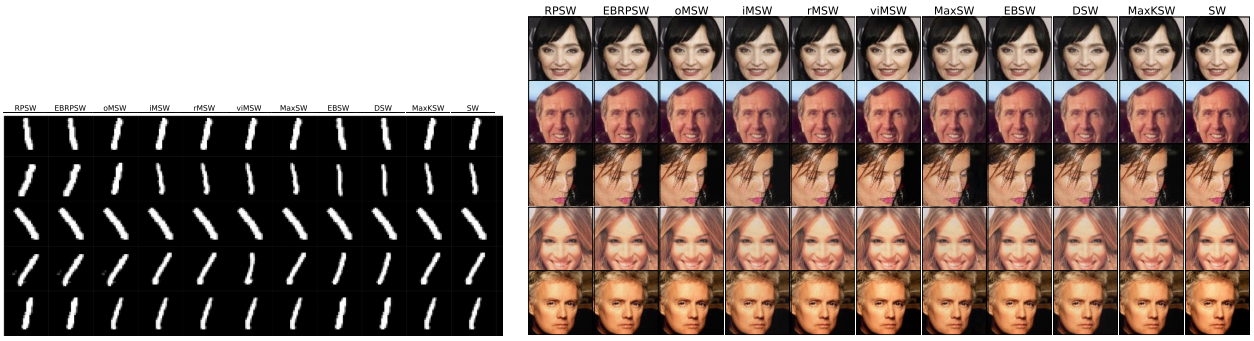


Figure 8: Gradient Flow visualization for images from the MNIST dataset (left) and the CelebA dataset (right).

Method	**MNIST (s) ↓**	**CelebA (s)↓**
DSW	$12500.00^{\pm 0.00}$	$126054.85^{\pm 745.89}$
EBSW	$686.18^{\pm 45.31}$	$6694.50^{\pm 148.08}$
RPSW	$800.36^{\pm 5.97}$	$6038.05^{\pm 86.32}$
EBRPSW	$699.33^{\pm 9.70}$	$3171.20^{\pm 582.31}$
oMSW	$482.29^{\pm 8.46}$	$3808.34^{\pm 475.66}$
iMSW	$1359.97^{\pm 10.19}$	$3601.99^{\pm 11.01}$
rMSW	$1115.98^{\pm 150.49}$	$100358.26^{\pm 1002.95}$
viMSW	$4161.11^{\pm 16.05}$	$96007.74^{\pm 937.50}$
MaxSW	$7231.97^{\pm 70.28}$	$9780.51^{\pm 485.71}$
MaxKSW	$6891.43^{\pm 35.52}$	$65560.25^{\pm 332.10}$
SWD	$441.41^{\pm 36.85}$	$3335.51^{\pm 76.52}$

Table 2: Runtime comparison for all methods in the MNIST/CelebA setups

B.2 Color Transfer

Table 3: Quantitative comparison of the best final converged $W_2 \downarrow$ and runtime $\downarrow$ between different variants for Color Transfer.

Method	**Best $W_2 \downarrow$ (LR)**			**Runtime(s) ↓**
	Set 1	**Set 2**	**Set 3**	
SWD	$0.01^{\pm 0.00}$ (1e-1)	$0.01^{\pm 0.00}$ (8e-1)	$0.00^{\pm 0.00}$ (1e0)	$8.62^{\pm 0.04}$
MaxSW	$0.03^{\pm 0.00}$ (1e-3)	$0.03^{\pm 0.00}$ (3e-4)	$0.03^{\pm 0.00}$ (3e-4)	$74.02^{\pm 1.61}$
DSW	$0.03^{\pm 0.00}$ (1e-3)	$0.03^{\pm 0.00}$ (1e-3)	$0.03^{\pm 0.00}$ (8e-4)	$162.25^{\pm 0.20}$
MaxKSW	$0.03^{\pm 0.00}$ (5e-4)	$0.03^{\pm 0.00}$ (5e-4)	$0.03^{\pm 0.00}$ (5e-4)	$125.23^{\pm 0.54}$
iMSW	$0.03^{\pm 0.00}$ (1e-3)	$0.03^{\pm 0.00}$ (1e-4)	$0.03^{\pm 0.00}$ (1e-3)	$74.45^{\pm 0.03}$
viMSW	$0.03^{\pm 0.00}$ (1e-3)	$0.03^{\pm 0.00}$ (1e-4)	$0.03^{\pm 0.00}$ (1e-3)	$255.76^{\pm 0.28}$
oMSW	$0.03^{\pm 0.00}$ (1e-3)	$0.03^{\pm 0.00}$ (1e-3)	$0.03^{\pm 0.00}$ (1e-3)	$16.55^{\pm 0.01}$
rMSW	$0.03^{\pm 0.00}$ (1e-3)	$0.03^{\pm 0.00}$ (1e-3)	$0.03^{\pm 0.00}$ (1e-3)	$179.70^{\pm 1.08}$
EBSW	$0.03^{\pm 0.00}$ (1e-3)	$0.03^{\pm 0.00}$ (1e-3)	$0.03^{\pm 0.00}$ (1e-3)	$9.66^{\pm 1.15}$
RPSW	$0.10^{\pm 0.00}$ (3e-3)	$0.10^{\pm 0.00}$ (1e-2)	$0.10^{\pm 0.09}$ (1e-3)	$19.47^{\pm 0.03}$
EBRPSW	$0.03^{\pm 0.00}$ (8e-4)	$0.03^{\pm 0.00}$ (5e-4)	$0.03^{\pm 0.00}$ (5e-4)	$20.30^{\pm 0.05}$

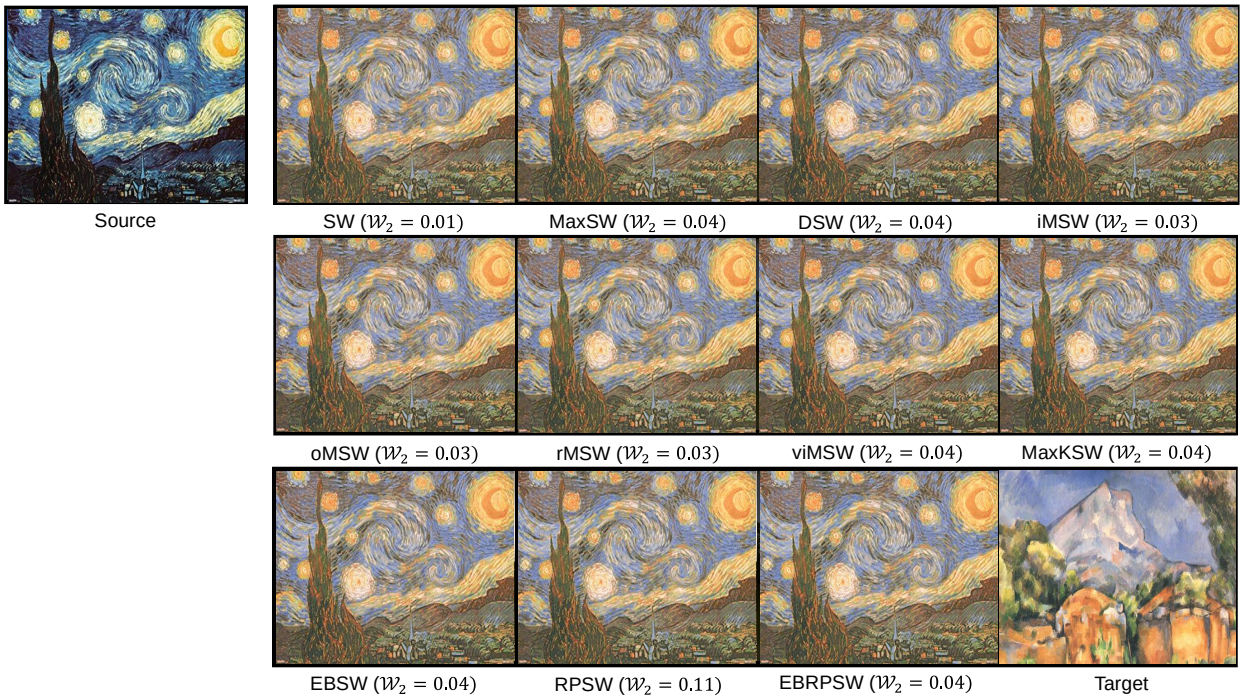


Figure 9: Additional color transfer visualization

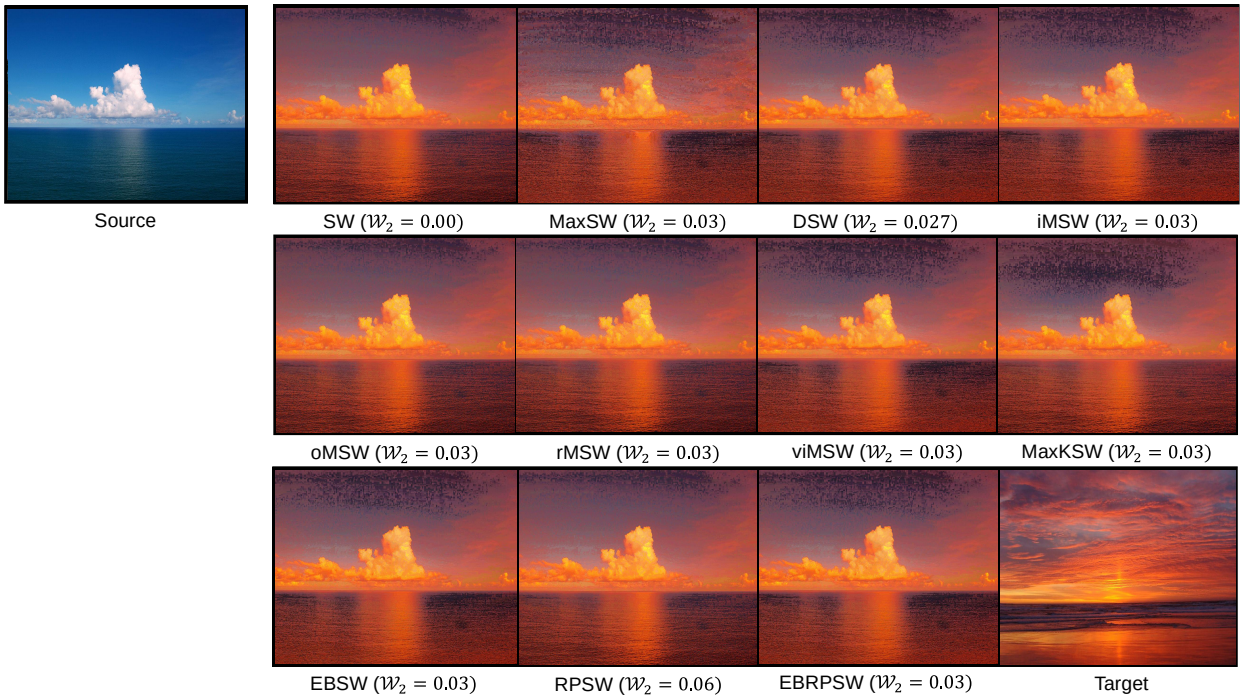


Figure 10: Additional color transfer visualization

B.3 Deep Generative Modeling

In this section, we provide additional details on the model architecture and numerical results for the deep generative modeling tasks. We present both qualitative and quantitative results, including the W_2 metric, to evaluate the model's performance across different SWD variants. The experiments were conducted using the FFHQ dataset.

Model architecture:

For unconditional generation:

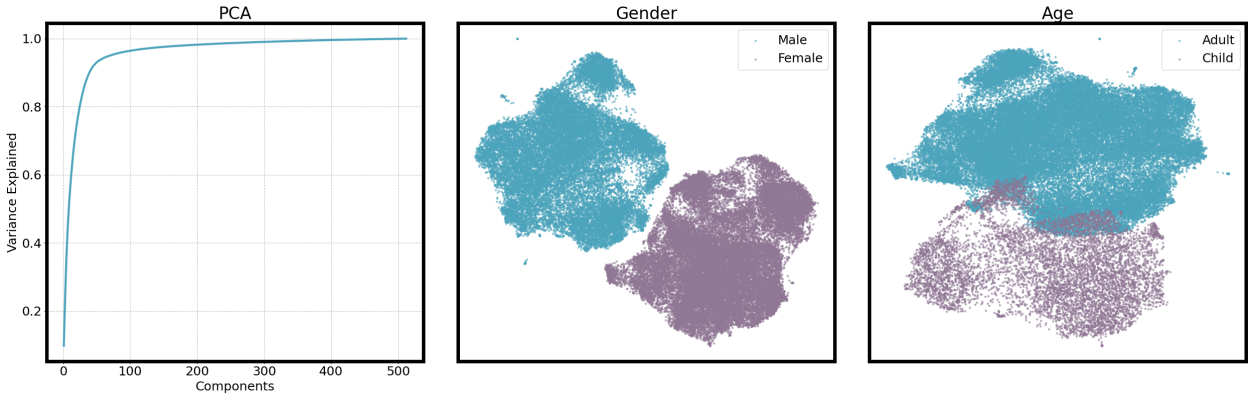
$$\begin{aligned} G(z) = \mathrm{FC}_d &\circ \mathrm{LeakyReLU}_{0.2} \circ \mathrm{BN} \circ \mathrm{FC}_{1024} \\ &\circ \mathrm{LeakyReLU}_{0.2} \circ \mathrm{BN} \circ \mathrm{FC}_{512} \\ &\circ \mathrm{LeakyReLU}_{0.2} \circ \mathrm{BN} \circ \mathrm{FC}_{256}(z), \quad z \in \mathbb{R}^8, G(z) \in \mathbb{R}^{512} \end{aligned}$$

For unpaired translation:

$$\begin{aligned} T(z) = z + \mathrm{FC}_d &\circ \mathrm{LeakyReLU}_{0.2} \circ \mathrm{BN} \circ \mathrm{FC}_{1024} \\ &\circ \mathrm{LeakyReLU}_{0.2} \circ \mathrm{BN} \circ \mathrm{FC}_{1024}(z), \quad z \in \mathbb{R}^{512} \end{aligned} \tag{52}$$

Table 4: Quantitative comparison between different variants for Deep Generative Modeling 5.4.

Method	**Unconditional Gen.**			**Unpaired Translation**			
	$W_2 \downarrow$	(LR)	RT(s) $\downarrow$	M2F: $W_2 \downarrow$ (LR)	RT(s)$\downarrow$	A2C: $W_2 \downarrow$ (LR)	RT(s)$\downarrow$
SWD	$14.67^{\pm.01}$	(1e0)	$26.69^{\pm.23}$	$14.15^{\pm.02}$ (5e-1)	$25.47^{\pm.09}$	$14.58^{\pm.03}$ (1e0)	$27.94^{\pm.14}$
MaxSW	$13.38^{\pm.17}$	(1e-2)	$95.83^{\pm.24}$	$14.01^{\pm.02}$ (5e-3)	$102.68^{\pm.03}$	$14.52^{\pm.02}$ (3e-3)	$103.29^{\pm2.98}$
DSW	$14.35^{\pm.06}$	(8e-3)	$197.70^{\pm.14}$	$14.11^{\pm.02}$ (5e-3)	$198.42^{\pm.38}$	$14.60^{\pm.04}$ (5e-3)	$198.08^{\pm.32}$
MaxKSW	$15.22^{\pm.03}$	(1e-2)	$53.38^{\pm3.16}$	$14.20^{\pm.01}$ (5e-2)	$45.90^{\pm.04}$	$14.65^{\pm.02}$ (5e-2)	$45.73^{\pm.08}$
iMSW	$14.27^{\pm.01}$	(1e-3)	$90.27^{\pm.23}$	$14.06^{\pm.01}$ (3e-3)	$92.70^{\pm.12}$	$14.59^{\pm.01}$ (1e-3)	$92.65^{\pm.17}$
viMSW	$15.16^{\pm.03}$	(1e-3)	$49.26^{\pm.05}$	$14.09^{\pm.01}$ (3e-3)	$271.10^{\pm.42}$	$14.57^{\pm.01}$ (3e-3)	$271.09^{\pm.84}$
oMSW	$14.81^{\pm.04}$	(5e-2)	$29.34^{\pm.18}$	$14.12^{\pm.01}$ (8e-2)	$31.39^{\pm.05}$	$14.58^{\pm.03}$ (1e-2)	$31.23^{\pm.04}$
rMSW	$14.80^{\pm.07}$	(5e-2)	$193.77^{\pm.63}$	$14.16^{\pm.01}$ (8e-2)	$195.07^{\pm.12}$	$14.60^{\pm.02}$ (8e-3)	$195.81^{\pm.27}$
EBSW	$16.68^{\pm.19}$	(3e-3)	$22.16^{\pm.07}$	$14.71^{\pm.02}$ (1e-2)	$26.67^{\pm.00}$	$15.09^{\pm.04}$ (8e-4)	$26.68^{\pm.02}$
RPSW	$14.46^{\pm.06}$	(3e-2)	$33.35^{\pm.07}$	$14.14^{\pm.00}$ (1e-1)	$37.11^{\pm.10}$	$14.60^{\pm.01}$ (1e-2)	$36.99^{\pm.12}$
EBRPSW	$16.90^{\pm.22}$	(3e-3)	$34.40^{\pm.07}$	$14.69^{\pm.02}$ (1e-2)	$38.15^{\pm.05}$	$15.10^{\pm.05}$ (1e-3)	$38.14^{\pm.11}$



(a) Left: Cumulative Explaning Variance plot for the FFHQ latents. **Middle/Right:** UMAP visualization of the Gender and Age splits.

FFHQ Subset	**Train size**	**Test size**
Adults (≥ 18)	48786	8104
Children (< 10)	8345	1405
Male	26732	4351
Female	32816	5572

(b) Subset size.

Figure 11: FFHQ dataset (Karras et al. (2019))

B.4 Detailed numerical results

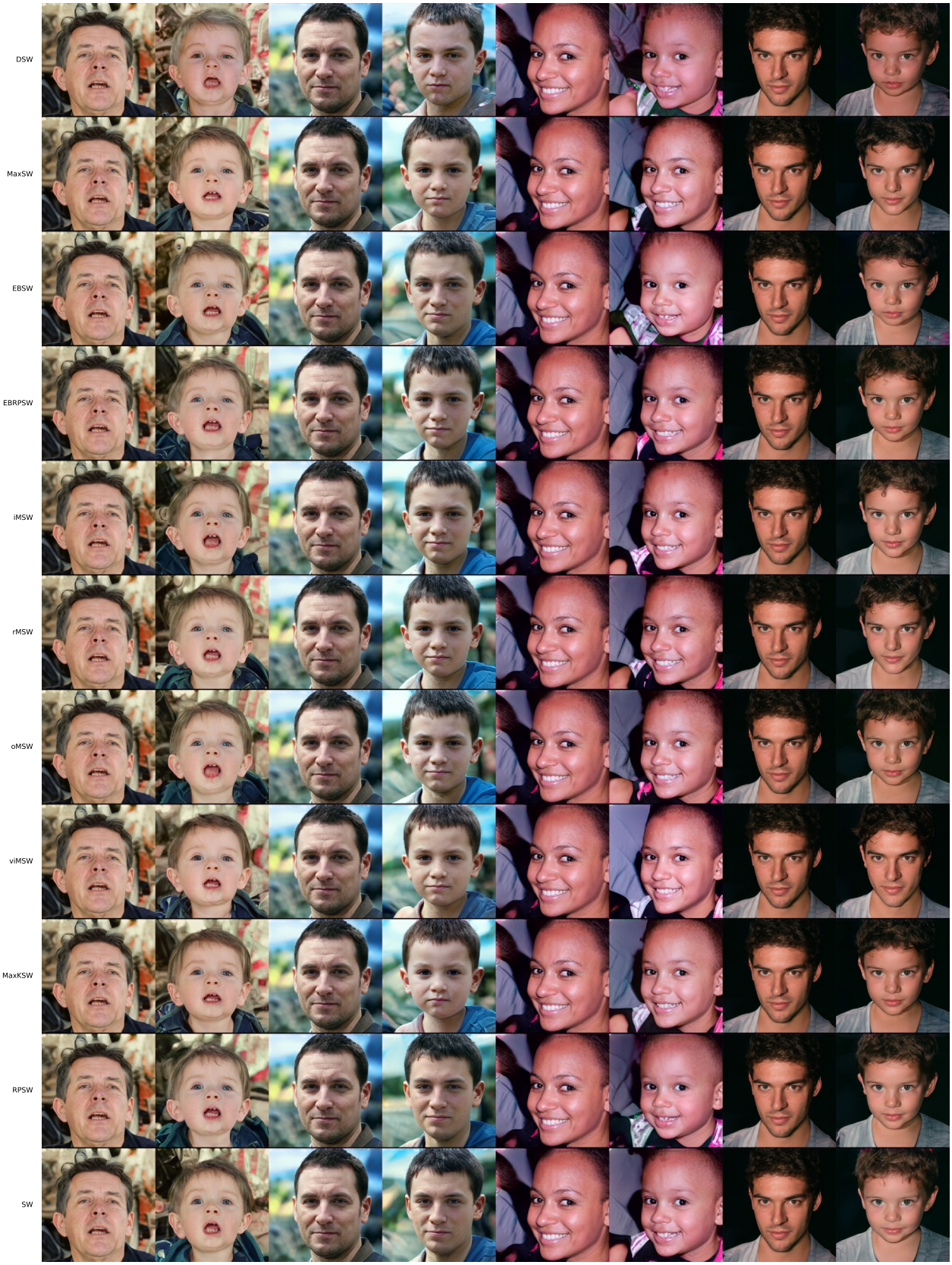


Figure 12: Visualization for the A2C translation task (using the model with the lowest W_2 for each method).

Figure 13: Visualization for the M2F translation task (using the model with the lowest W_2 for each method).

LR	SW	MaxSW	DSW	MaxKSW	iMSW	viMSW	oMSW	rMSW	EBSW	RPSW	EBRPSW
100	$2.3681^{\pm 0.3508}$	$0.5442^{\pm 0.0195}$	$1.8232^{\pm 1.2533}$	$1.2128^{\pm 0.0685}$	$3.8247^{\pm 0.6643}$	$3.1542^{\pm 0.3530}$	$1.2212^{\pm 0.0565}$	$1.8336^{\pm 0.2363}$	$2.9758^{\pm 0.0588}$	$2.7564^{\pm 0.1211}$	$3.0755^{\pm 0.1628}$
80	$1.7578^{\pm 0.4630}$	$0.5171^{\pm 0.0098}$	$0.9759^{\pm 0.5058}$	$1.3729^{\pm 0.0088}$	$2.3046^{\pm 0.5586}$	$1.8896^{\pm 0.9034}$	$1.3554^{\pm 0.1909}$	$1.8872^{\pm 0.4225}$	$2.4539^{\pm 0.0358}$	$2.1423^{\pm 0.0344}$	$2.2756^{\pm 0.0813}$
50	$0.8872^{\pm 0.3438}$	$1.8463^{\pm 0.9464}$	$0.9317^{\pm 0.5872}$	$1.6624^{\pm 0.0214}$	$0.6639^{\pm 0.1253}$	$1.2846^{\pm 0.0949}$	$1.4466^{\pm 0.1613}$	$1.4180^{\pm 0.1092}$	$1.4591^{\pm 0.0569}$	$1.1420^{\pm 0.0967}$	$1.4347^{\pm 0.1352}$
30	$0.6367^{\pm 0.2107}$	$0.9107^{\pm 0.3422}$	$0.5040^{\pm 0.2023}$	$0.9698^{\pm 0.0011}$	$0.7317^{\pm 0.2280}$	$0.4797^{\pm 0.2114}$	$0.7519^{\pm 0.0310}$	$0.9445^{\pm 0.2815}$	$1.1241^{\pm 0.3024}$	$0.7251^{\pm 0.0681}$	$0.9207^{\pm 0.2912}$
10	$0.2152^{\pm 0.0336}$	$0.4321^{\pm 0.0154}$	$0.2632^{\pm 0.1356}$	$0.1576^{\pm 0.0045}$	$0.2419^{\pm 0.0488}$	$0.1563^{\pm 0.0275}$	$0.1995^{\pm 0.0240}$	$0.2076^{\pm 0.0414}$	$0.3243^{\pm 0.0648}$	$0.2212^{\pm 0.0339}$	$0.3707^{\pm 0.0406}$
8	$0.2267^{\pm 0.0385}$	$0.3502^{\pm 0.0581}$	$0.1917^{\pm 0.0742}$	$0.2063^{\pm 0.0025}$	$0.2307^{\pm 0.0693}$	$0.2052^{\pm 0.0742}$	$0.1746^{\pm 0.0061}$	$0.1845^{\pm 0.0750}$	$0.1792^{\pm 0.1090}$	$0.1613^{\pm 0.0202}$	$0.3136^{\pm 0.0649}$
5	$0.1355^{\pm 0.0168}$	$0.2353^{\pm 0.0056}$	$0.1060^{\pm 0.0565}$	$0.1113^{\pm 0.0150}$	$0.1258^{\pm 0.0381}$	$0.0990^{\pm 0.0052}$	$0.1097^{\pm 0.0109}$	$0.1307^{\pm 0.0414}$	$0.1225^{\pm 0.0701}$	$0.1417^{\pm 0.0255}$	$0.2239^{\pm 0.0023}$
3	$0.0612^{\pm 0.0111}$	$0.1480^{\pm 0.0087}$	$0.0575^{\pm 0.0198}$	$0.0793^{\pm 0.0036}$	$0.0774^{\pm 0.0143}$	$0.0599^{\pm 0.0219}$	$0.0644^{\pm 0.0023}$	$0.0596^{\pm 0.0094}$	$0.1042^{\pm 0.0027}$	$0.0392^{\pm 0.0146}$	$0.1115^{\pm 0.0030}$
1	$0.0286^{\pm 0.0079}$	$0.0715^{\pm 0.0020}$	$0.0221^{\pm 0.0119}$	$0.0473^{\pm 0.0016}$	$0.0224^{\pm 0.0038}$	$0.0268^{\pm 0.0093}$	$0.0208^{\pm 0.0004}$	$0.0245^{\pm 0.0041}$	$0.0210^{\pm 0.0021}$	$0.0176^{\pm 0.0076}$	$0.0216^{\pm 0.0022}$
8×10^{-1}	$0.0182^{\pm 0.0067}$	$0.0631^{\pm 0.0046}$	$0.0165^{\pm 0.0049}$	$0.0429^{\pm 0.0012}$	$0.0197^{\pm 0.0031}$	$0.0267^{\pm 0.0039}$	$0.0181^{\pm 0.0007}$	$0.0205^{\pm 0.0026}$	$0.0178^{\pm 0.0048}$	$0.0233^{\pm 0.0054}$	$0.0151^{\pm 0.0021}$
5×10^{-1}	$0.0144^{\pm 0.0058}$	$0.0492^{\pm 0.0012}$	$0.0100^{\pm 0.0046}$	$0.0353^{\pm 0.0012}$	$0.0123^{\pm 0.0008}$	$0.0128^{\pm 0.0066}$	$0.0118^{\pm 0.0009}$	$0.0121^{\pm 0.0051}$	$0.0109^{\pm 0.0001}$	$0.0117^{\pm 0.0049}$	$0.0091^{\pm 0.0014}$
3×10^{-1}	$0.0063^{\pm 0.0025}$	$0.0365^{\pm 0.0014}$	$0.0040^{\pm 0.0026}$	$0.0238^{\pm 0.0014}$	$0.0079^{\pm 0.0014}$	$0.0078^{\pm 0.0012}$	$0.0063^{\pm 0.0003}$	$0.0062^{\pm 0.0020}$	$0.0051^{\pm 0.0010}$	$0.0083^{\pm 0.0029}$	$0.0052^{\pm 0.0019}$
1×10^{-1}	$0.0014^{\pm 0.0005}$	$0.0099^{\pm 0.0001}$	$0.0023^{\pm 0.0003}$	$0.0041^{\pm 0.0008}$	$0.0019^{\pm 0.0001}$	$0.0023^{\pm 0.0008}$	$0.0019^{\pm 0.0002}$	$0.0025^{\pm 0.0010}$	$0.0019^{\pm 0.0008}$	$0.0015^{\pm 0.0007}$	$0.0025^{\pm 0.0005}$
8×10^{-2}	$0.0019^{\pm 0.0005}$	$0.0070^{\pm 0.0005}$	$0.0016^{\pm 0.0008}$	$0.0022^{\pm 0.0000}$	$0.0017^{\pm 0.0001}$	$0.0029^{\pm 0.0003}$	$0.0016^{\pm 0.0002}$	$0.0017^{\pm 0.0008}$	$0.0023^{\pm 0.0002}$	$0.0020^{\pm 0.0003}$	$0.0012^{\pm 0.0001}$
5×10^{-2}	$0.0013^{\pm 0.0005}$	$0.0036^{\pm 0.0006}$	$0.0016^{\pm 0.0001}$	$0.0013^{\pm 0.0001}$	$0.0015^{\pm 0.0002}$	$0.0011^{\pm 0.0001}$	$0.0009^{\pm 0.0002}$	$0.0014^{\pm 0.0008}$	$0.0012^{\pm 0.0002}$	$0.0013^{\pm 0.0003}$	$0.0012^{\pm 0.0002}$
3×10^{-2}	$0.0005^{\pm 0.0001}$	$0.0014^{\pm 0.0001}$	$0.0007^{\pm 0.0004}$	$0.0006^{\pm 0.0001}$	$0.0006^{\pm 0.0002}$	$0.0007^{\pm 0.0004}$	$0.0007^{\pm 0.0000}$	$0.0006^{\pm 0.0001}$	$0.0004^{\pm 0.0003}$	$0.0009^{\pm 0.0001}$	$0.0007^{\pm 0.0002}$
1×10^{-2}	$0.0002^{\pm 0.0000}$	$0.0005^{\pm 0.0000}$	$0.0003^{\pm 0.0000}$	$0.0002^{\pm 0.0000}$	$0.0002^{\pm 0.0001}$	$0.0003^{\pm 0.0001}$	$0.0002^{\pm 0.0001}$	$0.0003^{\pm 0.0001}$	$0.0003^{\pm 0.0001}$	$0.0002^{\pm 0.0000}$	$0.0002^{\pm 0.0001}$
8×10^{-3}	$0.0002^{\pm 0.0001}$	$0.0004^{\pm 0.0000}$	$0.0002^{\pm 0.0001}$	$0.0002^{\pm 0.0000}$	$0.0002^{\pm 0.0000}$	$0.0002^{\pm 0.0000}$	$0.0001^{\pm 0.0001}$	$0.0003^{\pm 0.0000}$	$0.0002^{\pm 0.0000}$	$0.0002^{\pm 0.0001}$	$0.0002^{\pm 0.0001}$
5×10^{-3}	$0.0046^{\pm 0.0063}$	$0.0145^{\pm 0.0083}$	$0.0001^{\pm 0.0000}$	$0.0056^{\pm 0.0045}$	$0.0176^{\pm 0.0004}$	$0.0001^{\pm 0.0000}$	$0.0108^{\pm 0.0023}$	$0.0001^{\pm 0.0000}$	$0.0001^{\pm 0.0000}$	$0.0139^{\pm 0.0109}$	$0.0040^{\pm 0.0045}$
3×10^{-3}	$0.0744^{\pm 0.0031}$	$0.0547^{\pm 0.0049}$	$0.0584^{\pm 0.0035}$	$0.0675^{\pm 0.0070}$	$0.0652^{\pm 0.0027}$	$0.0578^{\pm 0.0032}$	$0.0693^{\pm 0.0050}$	$0.0571^{\pm 0.0098}$	$0.0490^{\pm 0.0053}$	$0.0720^{\pm 0.0118}$	$0.0430^{\pm 0.0022}$
1×10^{-3}	$0.4724^{\pm 0.0245}$	$0.4962^{\pm 0.0489}$	$0.3954^{\pm 0.0311}$	$0.4584^{\pm 0.0498}$	$0.4294^{\pm 0.0182}$	$0.3809^{\pm 0.0095}$	$0.4875^{\pm 0.0300}$	$0.4454^{\pm 0.0413}$	$0.4598^{\pm 0.0035}$	$0.4550^{\pm 0.0257}$	$0.4569^{\pm 0.0212}$
8×10^{-4}	$0.5596^{\pm 0.0618}$	$0.5197^{\pm 0.0412}$	$0.5045^{\pm 0.0217}$	$0.5379^{\pm 0.0168}$	$0.5398^{\pm 0.0372}$	$0.5325^{\pm 0.0290}$	$0.5704^{\pm 0.0030}$	$0.4655^{\pm 0.0322}$	$0.5102^{\pm 0.0100}$	$0.5269^{\pm 0.0288}$	$0.5842^{\pm 0.0629}$
5×10^{-4}	$0.6497^{\pm 0.0230}$	$0.6422^{\pm 0.0430}$	$0.5670^{\pm 0.0304}$	$0.6304^{\pm 0.0142}$	$0.6935^{\pm 0.0272}$	$0.5945^{\pm 0.0148}$	$0.6689^{\pm 0.0122}$	$0.6006^{\pm 0.0077}$	$0.6022^{\pm 0.0140}$	$0.6490^{\pm 0.0217}$	$0.6022^{\pm 0.0103}$
3×10^{-4}	$0.7475^{\pm 0.0404}$	$0.7277^{\pm 0.0425}$	$0.7061^{\pm 0.0107}$	$0.7534^{\pm 0.0072}$	$0.7213^{\pm 0.0118}$	$0.6642^{\pm 0.0273}$	$0.7026^{\pm 0.0120}$	$0.6609^{\pm 0.0067}$	$0.7100^{\pm 0.0204}$	$0.7137^{\pm 0.0168}$	$0.7379^{\pm 0.0714}$
1×10^{-4}	$0.8183^{\pm 0.0169}$	$0.8544^{\pm 0.0166}$	$0.7573^{\pm 0.0481}$	$0.7824^{\pm 0.0234}$	$0.7953^{\pm 0.0529}$	$0.7755^{\pm 0.0233}$	$0.8008^{\pm 0.0592}$	$0.7258^{\pm 0.0020}$	$0.8298^{\pm 0.0434}$	$0.8379^{\pm 0.0426}$	$0.8268^{\pm 0.0378}$
8×10^{-5}	$0.8061^{\pm 0.0603}$	$0.8473^{\pm 0.0326}$	$0.7661^{\pm 0.0200}$	$0.8084^{\pm 0.0587}$	$0.8328^{\pm 0.0445}$	$0.7158^{\pm 0.0079}$	$0.8242^{\pm 0.0773}$	$0.7232^{\pm 0.0083}$	$0.8122^{\pm 0.0442}$	$0.8063^{\pm 0.0226}$	$0.7910^{\pm 0.0183}$
5×10^{-5}	$0.8857^{\pm 0.0193}$	$0.7754^{\pm 0.0065}$	$0.7568^{\pm 0.0355}$	$0.8341^{\pm 0.0581}$	$0.7777^{\pm 0.0325}$	$0.7548^{\pm 0.0466}$	$0.8440^{\pm 0.0168}$	$0.7539^{\pm 0.0457}$	$0.8688^{\pm 0.0086}$	$0.8262^{\pm 0.0200}$	$0.8144^{\pm 0.0535}$
3×10^{-5}	$0.8197^{\pm 0.0106}$	$0.8329^{\pm 0.0499}$	$0.7885^{\pm 0.0524}$	$0.8745^{\pm 0.0060}$	$0.8467^{\pm 0.0079}$	$0.7556^{\pm 0.0234}$	$0.8248^{\pm 0.0403}$	$0.7488^{\pm 0.0444}$	$0.8240^{\pm 0.0296}$	$0.8314^{\pm 0.0190}$	$0.8560^{\pm 0.0302}$
1×10^{-5}	$0.8319^{\pm 0.0335}$	$0.8280^{\pm 0.0646}$	$0.7332^{\pm 0.0098}$	$0.8436^{\pm 0.0053}$	$0.8606^{\pm 0.0325}$	$0.7585^{\pm 0.0207}$	$0.7989^{\pm 0.0125}$	$0.8057^{\pm 0.0343}$	$0.8514^{\pm 0.0213}$	$0.8401^{\pm 0.0153}$	$0.8115^{\pm 0.0153}$
8×10^{-6}	$0.7844^{\pm 0.0210}$	$0.8536^{\pm 0.0094}$	$0.7587^{\pm 0.0282}$	$0.7918^{\pm 0.0159}$	$0.8102^{\pm 0.0273}$	$0.8835^{\pm 0.0241}$	$0.8531^{\pm 0.0217}$	$0.8103^{\pm 0.0076}$	$0.8061^{\pm 0.0339}$	$0.8325^{\pm 0.0247}$	$0.8034^{\pm 0.0514}$
5×10^{-6}	$0.8850^{\pm 0.0634}$	$0.8597^{\pm 0.0217}$	$0.7783^{\pm 0.0245}$	$0.8346^{\pm 0.0281}$	$0.8473^{\pm 0.0400}$	$0.8492^{\pm 0.0526}$	$0.8505^{\pm 0.0519}$	$0.8617^{\pm 0.0261}$	$0.8431^{\pm 0.0268}$	$0.8187^{\pm 0.0314}$	$0.8304^{\pm 0.0387}$
3×10^{-6}	$0.8508^{\pm 0.0199}$	$0.8491^{\pm 0.0034}$	$0.7657^{\pm 0.0149}$	$0.8293^{\pm 0.0317}$	$0.8062^{\pm 0.0639}$	$0.8466^{\pm 0.0131}$	$0.8537^{\pm 0.0494}$	$0.8266^{\pm 0.0428}$	$0.8254^{\pm 0.0214}$	$0.8292^{\pm 0.0522}$	$0.8453^{\pm 0.0027}$
1×10^{-6}	$0.8197^{\pm 0.0568}$	$0.8836^{\pm 0.0216}$	$0.8065^{\pm 0.0506}$	$0.8372^{\pm 0.0384}$	$0.8665^{\pm 0.0376}$	$0.8543^{\pm 0.0218}$	$0.8868^{\pm 0.0447}$	$0.8319^{\pm 0.0460}$	$0.8134^{\pm 0.0318}$	$0.8402^{\pm 0.0557}$	$0.8057^{\pm 0.0303}$

Table 5: Numerical results for Gradient Flow with the Knot dataset ($d = 2$)

LR	SW	MaxSW	DSW	MaxKSW	iMSW	viMSW	oMSW	rMSW	EBSW	RPSW	EBRPSW
100	$2.7116^{\pm 0.6616}$	$0.6408^{\pm 0.0079}$	$3.2280^{\pm 0.5094}$	$1.3326^{\pm 0.0411}$	$2.2337^{\pm 0.8727}$	$2.5000^{\pm 0.8335}$	$1.3862^{\pm 0.2603}$	$2.5671^{\pm 0.8620}$	$2.8073^{\pm 0.1416}$	$2.6577^{\pm 0.0282}$	$2.7618^{\pm 0.1329}$
80	$2.1367^{\pm 0.6087}$	$0.5710^{\pm 0.0041}$	$1.0384^{\pm 0.3640}$	$1.3753^{\pm 0.1106}$	$1.7140^{\pm 0.1703}$	$1.8681^{\pm 0.6268}$	$1.5932^{\pm 0.1041}$	$2.5953^{\pm 0.6861}$	$2.3348^{\pm 0.0828}$	$2.2001^{\pm 0.0631}$	$2.2090^{\pm 0.1610}$
50	$0.6877^{\pm 0.0723}$	$1.4925^{\pm 0.8598}$	$1.4048^{\pm 0.1910}$	$1.5581^{\pm 0.0122}$	$1.1232^{\pm 0.1551}$	$0.8977^{\pm 0.2941}$	$1.2540^{\pm 0.0568}$	$1.4821^{\pm 0.1792}$	$1.4024^{\pm 0.1124}$	$1.1371^{\pm 0.0850}$	$1.4613^{\pm 0.0400}$
30	$0.5389^{\pm 0.2452}$	$0.8486^{\pm 0.0496}$	$0.8765^{\pm 0.1254}$	$0.9592^{\pm 0.0019}$	$0.8816^{\pm 0.0522}$	$0.6488^{\pm 0.3089}$	$0.8072^{\pm 0.0214}$	$0.9189^{\pm 0.1387}$	$0.9178^{\pm 0.3697}$	$0.6777^{\pm 0.0658}$	$0.9473^{\pm 0.1249}$
10	$0.1721^{\pm 0.0402}$	$0.4116^{\pm 0.0909}$	$0.1965^{\pm 0.0915}$	$0.1744^{\pm 0.0095}$	$0.2179^{\pm 0.1150}$	$0.3502^{\pm 0.0193}$	$0.1614^{\pm 0.0207}$	$0.1752^{\pm 0.0400}$	$0.3453^{\pm 0.0457}$	$0.1866^{\pm 0.1160}$	$0.3806^{\pm 0.0842}$
8	$0.1892^{\pm 0.0560}$	$0.4166^{\pm 0.0334}$	$0.2441^{\pm 0.0545}$	$0.2037^{\pm 0.0043}$	$0.1355^{\pm 0.0865}$	$0.1707^{\pm 0.0453}$	$0.1800^{\pm 0.0074}$	$0.2334^{\pm 0.0587}$	$0.2948^{\pm 0.1139}$	$0.1311^{\pm 0.0399}$	$0.2549^{\pm 0.1194}$
5	$0.0907^{\pm 0.0390}$	$0.2481^{\pm 0.0296}$	$0.1453^{\pm 0.0272}$	$0.1270^{\pm 0.0113}$	$0.0916^{\pm 0.0514}$	$0.1663^{\pm 0.0083}$	$0.0956^{\pm 0.0204}$	$0.0990^{\pm 0.0220}$	$0.1724^{\pm 0.0669}$	$0.1171^{\pm 0.0207}$	$0.1779^{\pm 0.0662}$
3	$0.0738^{\pm 0.0312}$	$0.1655^{\pm 0.0088}$	$0.0793^{\pm 0.0224}$	$0.1074^{\pm 0.0030}$	$0.0767^{\pm 0.0161}$	$0.0300^{\pm 0.0172}$	$0.0645^{\pm 0.0015}$	$0.0593^{\pm 0.0474}$	$0.0931^{\pm 0.0083}$	$0.0656^{\pm 0.0181}$	$0.0988^{\pm 0.0052}$
1	$0.0167^{\pm 0.0020}$	$0.0788^{\pm 0.0019}$	$0.0264^{\pm 0.0057}$	$0.0489^{\pm 0.0069}$	$0.0205^{\pm 0.0067}$	$0.0167^{\pm 0.0011}$	$0.0191^{\pm 0.0021}$	$0.0148^{\pm 0.0076}$	$0.0234^{\pm 0.0030}$	$0.0230^{\pm 0.0114}$	$0.0208^{\pm 0.0013}$
8×10^{-1}	$0.0225^{\pm 0.0040}$	$0.0648^{\pm 0.0041}$	$0.0244^{\pm 0.0036}$	$0.0404^{\pm 0.0040}$	$0.0166^{\pm 0.0036}$	$0.0094^{\pm 0.0042}$	$0.0159^{\pm 0.0007}$	$0.0277^{\pm 0.0018}$	$0.0176^{\pm 0.0034}$	$0.0178^{\pm 0.0031}$	$0.0189^{\pm 0.0066}$
5×10^{-1}	$0.0090^{\pm 0.0039}$	$0.0482^{\pm 0.0017}$	$0.0147^{\pm 0.0036}$	$0.0251^{\pm 0.0038}$	$0.0078^{\pm 0.0023}$	$0.0125^{\pm 0.0050}$	$0.0106^{\pm 0.0006}$	$0.0099^{\pm 0.0044}$	$0.0095^{\pm 0.0022}$	$0.0101^{\pm 0.0049}$	$0.0086^{\pm 0.0035}$
3×10^{-1}	$0.0057^{\pm 0.0014}$	$0.0405^{\pm 0.0016}$	$0.0134^{\pm 0.0078}$	$0.0131^{\pm 0.0008}$	$0.0060^{\pm 0.0010}$	$0.0092^{\pm 0.0013}$	$0.0066^{\pm 0.0003}$	$0.0039^{\pm 0.0021}$	$0.0062^{\pm 0.0012}$	$0.0047^{\pm 0.0011}$	$0.0055^{\pm 0.0015}$
1×10^{-1}	$0.0018^{\pm 0.0009}$	$0.0138^{\pm 0.0012}$	$0.0019^{\pm 0.0005}$	$0.0031^{\pm 0.0003}$	$0.0078^{\pm 0.0088}$	$0.0021^{\pm 0.0011}$	$0.0020^{\pm 0.0001}$	$0.0013^{\pm 0.0003}$	$0.0018^{\pm 0.0004}$	$0.0020^{\pm 0.0003}$	$0.0019^{\pm 0.0002}$
8×10^{-2}	$0.0018^{\pm 0.0004}$	$0.0076^{\pm 0.0007}$	$0.0025^{\pm 0.0001}$	$0.0020^{\pm 0.0002}$	$0.0088^{\pm 0.0099}$	$0.0012^{\pm 0.0002}$	$0.0018^{\pm 0.0001}$	$0.0020^{\pm 0.0001}$	$0.0020^{\pm 0.0002}$	$0.0016^{\pm 0.0005}$	$0.0015^{\pm 0.0004}$
5×10^{-2}	$0.0010^{\pm 0.0001}$	$0.0038^{\pm 0.0004}$	$0.0017^{\pm 0.0005}$	$0.0016^{\pm 0.0003}$	$0.0015^{\pm 0.0004}$	$0.0009^{\pm 0.0004}$	$0.0010^{\pm 0.0000}$	$0.0010^{\pm 0.0001}$	$0.0009^{\pm 0.0001}$	$0.0010^{\pm 0.0001}$	$0.0007^{\pm 0.0002}$
3×10^{-2}	$0.0007^{\pm 0.0004}$	$0.0020^{\pm 0.0003}$	$0.0081^{\pm 0.0104}$	$0.0008^{\pm 0.0000}$	$0.0008^{\pm 0.0001}$	$0.0009^{\pm 0.0002}$	$0.0005^{\pm 0.0001}$	$0.0005^{\pm 0.0001}$	$0.0031^{\pm 0.0030}$	$0.0005^{\pm 0.0001}$	$0.0007^{\pm 0.0002}$
1×10^{-2}	$0.0002^{\pm 0.0001}$	$0.0004^{\pm 0.0000}$	$0.0003^{\pm 0.0001}$	$0.0002^{\pm 0.0000}$	$0.0001^{\pm 0.0000}$	$0.0071^{\pm 0.0098}$	$0.0001^{\pm 0.0000}$	$0.0002^{\pm 0.0000}$	$0.0002^{\pm 0.0001}$	$0.0002^{\pm 0.0000}$	$0.0003^{\pm 0.0001}$
8×10^{-3}	$0.0002^{\pm 0.0001}$	$0.0000^{\pm 0.0000}$	$0.0002^{\pm 0.0001}$	$0.0002^{\pm 0.0001}$	$0.0002^{\pm 0.0000}$	$0.0002^{\pm 0.0001}$	$0.0001^{\pm 0.0001}$	$0.0002^{\pm 0.0001}$	$0.0003^{\pm 0.0001}$	$0.0001^{\pm 0.0001}$	$0.0028^{\pm 0.0037}$
5×10^{-3}	$0.0001^{\pm 0.0000}$	$0.0113^{\pm 0.0080}$	$0.0023^{\pm 0.0031}$	$0.0006^{\pm 0.0007}$	$0.0074^{\pm 0.0104}$	$0.0166^{\pm 0.0130}$	$0.0014^{\pm 0.0018}$	$0.0012^{\pm 0.0016}$	$0.0018^{\pm 0.0024}$	$0.0001^{\pm 0.0000}$	$0.0101^{\pm 0.0102}$
3×10^{-3}	$0.0462^{\pm 0.0025}$	$0.0576^{\pm 0.0029}$	$0.0463^{\pm 0.0037}$	$0.0524^{\pm 0.0029}$	$0.0520^{\pm 0.0014}$	$0.0573^{\pm 0.0045}$	$0.0440^{\pm 0.0008}$	$0.0443^{\pm 0.0063}$	$0.0388^{\pm 0.0036}$	$0.0537^{\pm 0.0011}$	$0.0461^{\pm 0.0020}$
1×10^{-3}	$0.4562^{\pm 0.0128}$	$0.4431^{\pm 0.0294}$	$0.4015^{\pm 0.0061}$	$0.4037^{\pm 0.0474}$	$0.4144^{\pm 0.0245}$	$0.4502^{\pm 0.0148}$	$0.4378^{\pm 0.0449}$	$0.3841^{\pm 0.0371}$	$0.4561^{\pm 0.0102}$	$0.4135^{\pm 0.0113}$	$0.4053^{\pm 0.0156}$
8×10^{-4}	$0.5542^{\pm 0.0443}$	$0.5135^{\pm 0.0036}$	$0.4889^{\pm 0.0376}$	$0.5049^{\pm 0.0621}$	$0.4779^{\pm 0.0132}$	$0.4414^{\pm 0.0209}$	$0.5153^{\pm 0.0144}$	$0.4962^{\pm 0.0564}$	$0.5377^{\pm 0.0322}$	$0.5181^{\pm 0.0430}$	$0.4708^{\pm 0.0153}$
5×10^{-4}	$0.6267^{\pm 0.0115}$	$0.6978^{\pm 0.0145}$	$0.6018^{\pm 0.0523}$	$0.5962^{\pm 0.0185}$	$0.6326^{\pm 0.0183}$	$0.6380^{\pm 0.0633}$	$0.5696^{\pm 0.0220}$	$0.6054^{\pm 0.0356}$	$0.5978^{\pm 0.0170}$	$0.6307^{\pm 0.0299}$	$0.6033^{\pm 0.0117}$
3×10^{-4}	$0.6768^{\pm 0.0396}$	$0.7378^{\pm 0.0884}$	$0.6948^{\pm 0.0818}$	$0.6501^{\pm 0.0650}$	$0.7257^{\pm 0.0189}$	$0.6678^{\pm 0.0099}$	$0.7080^{\pm 0.0066}$	$0.6992^{\pm 0.0195}$	$0.6705^{\pm 0.0717}$	$0.7056^{\pm 0.0626}$	$0.6892^{\pm 0.0496}$
1×10^{-4}	$0.8010^{\pm 0.0609}$	$0.8032^{\pm 0.0078}$	$0.7756^{\pm 0.0116}$	$0.8050^{\pm 0.0386}$	$0.7569^{\pm 0.0699}$	$0.7831^{\pm 0.0213}$	$0.7689^{\pm 0.0297}$	$0.7545^{\pm 0.0447}$	$0.7353^{\pm 0.0105}$	$0.8001^{\pm 0.0219}$	$0.7609^{\pm 0.0142}$
8×10^{-5}	$0.7912^{\pm 0.0136}$	$0.8318^{\pm 0.0096}$	$0.7750^{\pm 0.0291}$	$0.7513^{\pm 0.0187}$	$0.7757^{\pm 0.0432}$	$0.7533^{\pm 0.0463}$	$0.7707^{\pm 0.0304}$	$0.7996^{\pm 0.0452}$	$0.7798^{\pm 0.0275}$	$0.8173^{\pm 0.0162}$	$0.7751^{\pm 0.0186}$
5×10^{-5}	$0.7931^{\pm 0.0177}$	$0.8609^{\pm 0.0384}$	$0.7861^{\pm 0.0270}$	$0.7883^{\pm 0.0147}$	$0.7286^{\pm 0.0337}$	$0.7751^{\pm 0.0454}$	$0.7664^{\pm 0.0712}$	$0.7524^{\pm 0.0513}$	$0.7911^{\pm 0.0444}$	$0.7857^{\pm 0.0189}$	$0.7777^{\pm 0.0174}$
3×10^{-5}	$0.7866^{\pm 0.0157}$	$0.8463^{\pm 0.0257}$	$0.8185^{\pm 0.0083}$	$0.8197^{\pm 0.0474}$	$0.8045^{\pm 0.0219}$	$0.7678^{\pm 0.0227}$	$0.8187^{\pm 0.0423}$	$0.7756^{\pm 0.0405}$	$0.8055^{\pm 0.0188}$	$0.7912^{\pm 0.0439}$	$0.7578^{\pm 0.0359}$
1×10^{-5}	$0.7728^{\pm 0.0195}$	$0.8348^{\pm 0.0121}$	$0.8120^{\pm 0.0219}$	$0.7983^{\pm 0.0566}$	$0.7745^{\pm 0.0238}$	$0.7827^{\pm 0.0089}$	$0.8054^{\pm 0.0241}$	$0.8240^{\pm 0.0307}$	$0.8172^{\pm 0.0239}$	$0.7912^{\pm 0.0288}$	$0.7982^{\pm 0.0576}$
8×10^{-6}	$0.7996^{\pm 0.0623}$	$0.8318^{\pm 0.0096}$	$0.7549^{\pm 0.0266}$	$0.8504^{\pm 0.0217}$	$0.8288^{\pm 0.0260}$	$0.8287^{\pm 0.0296}$	$0.8424^{\pm 0.0903}$	$0.8012^{\pm 0.0562}$	$0.7854^{\pm 0.0054}$	$0.8035^{\pm 0.0243}$	$0.7890^{\pm 0.0228}$
5×10^{-6}	$0.8437^{\pm 0.0315}$	$0.8609^{\pm 0.0384}$	$0.8357^{\pm 0.0442}$	$0.8353^{\pm 0.0145}$	$0.7918^{\pm 0.0215}$	$0.7905^{\pm 0.0248}$	$0.7750^{\pm 0.0088}$	$0.8025^{\pm 0.0390}$	$0.8363^{\pm 0.0332}$	$0.7982^{\pm 0.0254}$	$0.8205^{\pm 0.0078}$
3×10^{-6}	$0.7935^{\pm 0.0268}$	$0.8463^{\pm 0.0257}$	$0.8084^{\pm 0.0411}$	$0.8457^{\pm 0.0174}$	$0.7783^{\pm 0.0195}$	$0.7867^{\pm 0.0324}$	$0.8515^{\pm 0.0514}$	$0.8093^{\pm 0.0442}$	$0.8079^{\pm 0.0208}$	$0.8266^{\pm 0.0051}$	$0.8228^{\pm 0.0168}$
1×10^{-6}	$0.8332^{\pm 0.0122}$	$0.8348^{\pm 0.0121}$	$0.8042^{\pm 0.0282}$	$0.8091^{\pm 0.0190}$	$0.7770^{\pm 0.0248}$	$0.7748^{\pm 0.0164}$	$0.7932^{\pm 0.0170}$	$0.8373^{\pm 0.0212}$	$0.8203^{\pm 0.0194}$	$0.8121^{\pm 0.0115}$	$0.8057^{\pm 0.0246}$

Table 6: Numerical results for Gradient Flow with the Swiss dataset ($d = 2$)

LR	SW	MaxSW	DSW	MaxKSW	iMSW	viMSW	oMSW	rMSW	EBSW	RPSW	EBRPSW
100	$1.8754^{\pm 1.2183}$	$3.6719^{\pm 1.6669}$	$2.2167^{\pm 1.3910}$	$1.1766^{\pm 0.0901}$	$2.0996^{\pm 0.5058}$	$2.3956^{\pm 0.7993}$	$3.0252^{\pm 0.1706}$	$2.6436^{\pm 1.0882}$	$3.0252^{\pm 0.1706}$	$2.6426^{\pm 0.0669}$	$2.8145^{\pm 0.1491}$
80	$1.6682^{\pm 0.9727}$	$1.8203^{\pm 0.9088}$	$2.2781^{\pm 0.4102}$	$1.1566^{\pm 0.0318}$	$2.0056^{\pm 0.7764}$	$1.8520^{\pm 0.5824}$	$2.2930^{\pm 0.0903}$	$1.8364^{\pm 1.0702}$	$2.2930^{\pm 0.0903}$	$2.0552^{\pm 0.1070}$	$2.3738^{\pm 0.0394}$
50	$0.7122^{\pm 0.1076}$	$1.5041^{\pm 0.2527}$	$0.7793^{\pm 0.4214}$	$0.9272^{\pm 0.0260}$	$1.1239^{\pm 0.3555}$	$1.3310^{\pm 0.4241}$	$1.7172^{\pm 0.1023}$	$1.4756^{\pm 0.3155}$	$1.7172^{\pm 0.1023}$	$1.1673^{\pm 0.1447}$	$1.4905^{\pm 0.0416}$
30	$0.6045^{\pm 0.3404}$	$1.0704^{\pm 0.0587}$	$0.6909^{\pm 0.3766}$	$0.7364^{\pm 0.0218}$	$0.7664^{\pm 0.0791}$	$0.4753^{\pm 0.1999}$	$1.0015^{\pm 0.0625}$	$0.8430^{\pm 0.2975}$	$1.0015^{\pm 0.0625}$	$0.6435^{\pm 0.3303}$	$0.9820^{\pm 0.1278}$
10	$0.1544^{\pm 0.0464}$	$0.4782^{\pm 0.0479}$	$0.2348^{\pm 0.0887}$	$0.1933^{\pm 0.0312}$	$0.1795^{\pm 0.0326}$	$0.3404^{\pm 0.0127}$	$0.2363^{\pm 0.0632}$	$0.2242^{\pm 0.0245}$	$0.2363^{\pm 0.0632}$	$0.2729^{\pm 0.0176}$	$0.3536^{\pm 0.0355}$
8	$0.2014^{\pm 0.0251}$	$0.3462^{\pm 0.0410}$	$0.2254^{\pm 0.0901}$	$0.1854^{\pm 0.0010}$	$0.1386^{\pm 0.0738}$	$0.1293^{\pm 0.0563}$	$0.3805^{\pm 0.0112}$	$0.1638^{\pm 0.0318}$	$0.3805^{\pm 0.0112}$	$0.2027^{\pm 0.0718}$	$0.0854^{\pm 0.0351}$
5	$0.0995^{\pm 0.0458}$	$0.2004^{\pm 0.0136}$	$0.0772^{\pm 0.0469}$	$0.0998^{\pm 0.0078}$	$0.1063^{\pm 0.0017}$	$0.1894^{\pm 0.0047}$	$0.0780^{\pm 0.0013}$	$0.2197^{\pm 0.0233}$	$0.0780^{\pm 0.0013}$	$0.1016^{\pm 0.0034}$	$0.1692^{\pm 0.0604}$
3	$0.0868^{\pm 0.0193}$	$0.1177^{\pm 0.0058}$	$0.0640^{\pm 0.0260}$	$0.0613^{\pm 0.0033}$	$0.0885^{\pm 0.0218}$	$0.4035^{\pm 0.0251}$	$0.1012^{\pm 0.0058}$	$0.4207^{\pm 0.0243}$	$0.1012^{\pm 0.0058}$	$0.0676^{\pm 0.0091}$	$0.0958^{\pm 0.0013}$
1	$0.0230^{\pm 0.0086}$	$0.0436^{\pm 0.0010}$	$0.0245^{\pm 0.0084}$	$0.0230^{\pm 0.0008}$	$0.0183^{\pm 0.0091}$	$0.7765^{\pm 0.0445}$	$0.0187^{\pm 0.0014}$	$0.8306^{\pm 0.0391}$	$0.0171^{\pm 0.0014}$	$0.0250^{\pm 0.0009}$	$0.0209^{\pm 0.0019}$
8×10^{-1}	$0.0246^{\pm 0.0058}$	$0.0342^{\pm 0.0012}$	$0.0168^{\pm 0.0050}$	$0.0189^{\pm 0.0005}$	$0.0171^{\pm 0.0011}$	$0.8516^{\pm 0.0207}$	$0.0139^{\pm 0.0012}$	$0.8743^{\pm 0.0473}$	$0.0177^{\pm 0.0050}$	$0.0217^{\pm 0.0016}$	$0.0151^{\pm 0.0036}$
5×10^{-1}	$0.0091^{\pm 0.0027}$	$0.0233^{\pm 0.0006}$	$0.0103^{\pm 0.0042}$	$0.0146^{\pm 0.0006}$	$0.0096^{\pm 0.0027}$	$0.9355^{\pm 0.0290}$	$0.0090^{\pm 0.0007}$	$0.9269^{\pm 0.0258}$	$0.0092^{\pm 0.0024}$	$0.0142^{\pm 0.0033}$	$0.0099^{\pm 0.0037}$
3×10^{-1}	$0.0070^{\pm 0.0006}$	$0.0158^{\pm 0.0001}$	$0.0064^{\pm 0.0027}$	$0.0119^{\pm 0.0005}$	$0.0078^{\pm 0.0019}$	$0.9861^{\pm 0.0259}$	$0.0062^{\pm 0.0002}$	$1.0561^{\pm 0.0192}$	$0.0041^{\pm 0.0019}$	$0.0073^{\pm 0.0002}$	$0.0059^{\pm 0.0017}$
1×10^{-1}	$0.0020^{\pm 0.0002}$	$0.0097^{\pm 0.0002}$	$0.0018^{\pm 0.0010}$	$0.0059^{\pm 0.0009}$	$0.0025^{\pm 0.0007}$	$1.0240^{\pm 0.0293}$	$0.0017^{\pm 0.0001}$	$1.0638^{\pm 0.0370}$	$0.0015^{\pm 0.0005}$	$0.0027^{\pm 0.0008}$	$0.0021^{\pm 0.0007}$
8×10^{-2}	$0.0019^{\pm 0.0003}$	$0.0076^{\pm 0.0002}$	$0.0021^{\pm 0.0003}$	$0.0040^{\pm 0.0002}$	$0.0020^{\pm 0.0001}$	$1.0657^{\pm 0.0399}$	$0.0017^{\pm 0.0001}$	$1.0979^{\pm 0.0350}$	$0.0018^{\pm 0.0003}$	$0.0019^{\pm 0.0003}$	$0.0013^{\pm 0.0001}$
5×10^{-2}	$0.0010^{\pm 0.0003}$	$0.0061^{\pm 0.0000}$	$0.0006^{\pm 0.0002}$	$0.0031^{\pm 0.0004}$	$0.0013^{\pm 0.0002}$	$1.0926^{\pm 0.0206}$	$0.0010^{\pm 0.0001}$	$1.1166^{\pm 0.0107}$	$0.0012^{\pm 0.0005}$	$0.0014^{\pm 0.0005}$	$0.0006^{\pm 0.0004}$
3×10^{-2}	$0.0008^{\pm 0.0002}$	$0.0031^{\pm 0.0003}$	$0.0010^{\pm 0.0001}$	$0.0018^{\pm 0.0002}$	$0.0008^{\pm 0.0002}$	$1.1155^{\pm 0.0033}$	$0.0006^{\pm 0.0001}$	$1.1021^{\pm 0.0320}$	$0.0009^{\pm 0.0001}$	$0.0007^{\pm 0.0002}$	$0.0007^{\pm 0.0002}$
1×10^{-2}	$0.0002^{\pm 0.0001}$	$0.1316^{\pm 0.0190}$	$0.0002^{\pm 0.0001}$	$0.0384^{\pm 0.0497}$	$0.0002^{\pm 0.0000}$	$0.0003^{\pm 0.0001}$	$0.0002^{\pm 0.0000}$	$0.0003^{\pm 0.0313}$	$0.0001^{\pm 0.0000}$	$0.0002^{\pm 0.0000}$	$0.0009^{\pm 0.0001}$
8×10^{-3}	$0.0007^{\pm 0.0009}$	$0.1939^{\pm 0.0405}$	$0.0001^{\pm 0.0001}$	$0.1640^{\pm 0.0036}$	$0.0002^{\pm 0.0000}$	$0.0002^{\pm 0.0000}$	$0.0002^{\pm 0.0000}$	$0.0002^{\pm 0.0211}$	$0.1260^{\pm 0.0075}$	$0.0014^{\pm 0.0009}$	$0.1106^{\pm 0.0398}$
5×10^{-3}	$0.1965^{\pm 0.0290}$	$0.3497^{\pm 0.0149}$	$0.2026^{\pm 0.0270}$	$0.3047^{\pm 0.0215}$	$0.2563^{\pm 0.0217}$	$0.1894^{\pm 0.0047}$	$0.2271^{\pm 0.0173}$	$0.2197^{\pm 0.0233}$	$0.3223^{\pm 0.0112}$	$0.1987^{\pm 0.0204}$	$0.2964^{\pm 0.0164}$
3×10^{-3}	$0.4310^{\pm 0.0204}$	$0.4287^{\pm 0.0222}$	$0.4433^{\pm 0.0227}$	$0.4484^{\pm 0.0108}$	$0.4080^{\pm 0.0145}$	$0.4035^{\pm 0.0251}$	$0.4203^{\pm 0.0266}$	$0.4207^{\pm 0.0243}$	$0.4304^{\pm 0.0058}$	$0.4490^{\pm 0.0226}$	$0.4122^{\pm 0.0402}$
1×10^{-3}	$0.8561^{\pm 0.0290}$	$0.8367^{\pm 0.0442}$	$0.7788^{\pm 0.0083}$	$0.8168^{\pm 0.0181}$	$0.7574^{\pm 0.0171}$	$0.7765^{\pm 0.0445}$	$0.8061^{\pm 0.0452}$	$0.8306^{\pm 0.0391}$	$0.7096^{\pm 0.0249}$	$0.8049^{\pm 0.0369}$	$0.7294^{\pm 0.0128}$
8×10^{-4}	$0.8471^{\pm 0.0388}$	$0.8914^{\pm 0.0049}$	$0.8349^{\pm 0.0178}$	$0.8478^{\pm 0.0113}$	$0.8654^{\pm 0.0206}$	$0.8516^{\pm 0.0207}$	$0.8791^{\pm 0.0156}$	$0.8743^{\pm 0.0473}$	$0.8134^{\pm 0.0245}$	$0.8864^{\pm 0.0115}$	$0.7763^{\pm 0.0230}$
5×10^{-4}	$0.9638^{\pm 0.0207}$	$0.9201^{\pm 0.0121}$	$0.9686^{\pm 0.0407}$	$0.9992^{\pm 0.0064}$	$0.9837^{\pm 0.0163}$	$0.9355^{\pm 0.0290}$	$0.9339^{\pm 0.0409}$	$0.9269^{\pm 0.0258}$	$0.9093^{\pm 0.0198}$	$0.9818^{\pm 0.0177}$	$0.9462^{\pm 0.0092}$
3×10^{-4}	$1.0198^{\pm 0.0345}$	$0.9876^{\pm 0.0126}$	$0.9971^{\pm 0.0412}$	$0.9834^{\pm 0.0111}$	$1.0509^{\pm 0.0250}$	$0.9861^{\pm 0.0259}$	$1.0586^{\pm 0.0182}$	$1.0561^{\pm 0.0192}$	$0.9803^{\pm 0.0350}$	$1.0486^{\pm 0.0266}$	$0.9542^{\pm 0.0045}$
1×10^{-4}	$1.0491^{\pm 0.0545}$	$1.0148^{\pm 0.0181}$	$1.0628^{\pm 0.0423}$	$1.0362^{\pm 0.0304}$	$1.0967^{\pm 0.0192}$	$1.0240^{\pm 0.0293}$	$1.0540^{\pm 0.0237}$	$1.0638^{\pm 0.0370}$	$1.0043^{\pm 0.0284}$	$1.0805^{\pm 0.0667}$	$1.0699^{\pm 0.0781}$
8×10^{-5}	$1.1368^{\pm 0.0152}$	$1.0651^{\pm 0.0629}$	$1.0756^{\pm 0.0461}$	$1.0883^{\pm 0.0437}$	$1.0514^{\pm 0.0400}$	$1.0657^{\pm 0.0399}$	$1.1419^{\pm 0.0481}$	$1.0979^{\pm 0.0350}$	$1.1419^{\pm 0.0481}$	$1.1368^{\pm 0.0152}$	$1.0124^{\pm 0.0534}$
5×10^{-5}	$1.1340^{\pm 0.0280}$	$1.0858^{\pm 0.0165}$	$0.9856^{\pm 0.0501}$	$1.1182^{\pm 0.0655}$	$1.0685^{\pm 0.0534}$	$1.0926^{\pm 0.0206}$	$1.0581^{\pm 0.0062}$	$1.1166^{\pm 0.0107}$	$1.0581^{\pm 0.0062}$	$1.1340^{\pm 0.0280}$	$1.1027^{\pm 0.0240}$
3×10^{-5}	$1.1742^{\pm 0.0390}$	$1.0918^{\pm 0.0456}$	$1.1101^{\pm 0.0523}$	$1.1371^{\pm 0.0335}$	$1.1516^{\pm 0.0201}$	$1.1155^{\pm 0.0033}$	$1.0978^{\pm 0.0461}$	$1.1021^{\pm 0.0320}$	$1.0978^{\pm 0.0461}$	$1.1742^{\pm 0.0390}$	$1.1002^{\pm 0.0187}$
1×10^{-5}	$1.1502^{\pm 0.0119}$	$1.1172^{\pm 0.0257}$	$1.1101^{\pm 0.0553}$	$1.0822^{\pm 0.0282}$	$1.1097^{\pm 0.0482}$	$1.1363^{\pm 0.0244}$	$1.1149^{\pm 0.0311}$	$1.0607^{\pm 0.0759}$	$1.1149^{\pm 0.0311}$	$1.1284^{\pm 0.0389}$	$1.1040^{\pm 0.0386}$
8×10^{-6}	$1.1234^{\pm 0.0241}$	$1.1757^{\pm 0.0295}$	$1.1211^{\pm 0.0590}$	$1.1268^{\pm 0.0387}$	$1.1028^{\pm 0.0081}$	$1.1288^{\pm 0.0206}$	$1.1419^{\pm 0.0481}$	$1.1154^{\pm 0.0252}$	$1.1419^{\pm 0.0481}$	$1.1368^{\pm 0.0152}$	$1.1397^{\pm 0.0270}$
5×10^{-6}	$1.0738^{\pm 0.0273}$	$1.0898^{\pm 0.0584}$	$1.1256^{\pm 0.0493}$	$1.1105^{\pm 0.0139}$	$1.1040^{\pm 0.0153}$	$1.1205^{\pm 0.0187}$	$1.0581^{\pm 0.0062}$	$1.1064^{\pm 0.0194}$	$1.0581^{\pm 0.0062}$	$1.1340^{\pm 0.0280}$	$1.1027^{\pm 0.0240}$
3×10^{-6}	$1.1109^{\pm 0.0355}$	$1.1249^{\pm 0.0357}$	$1.0902^{\pm 0.0320}$	$1.1371^{\pm 0.0335}$	$1.1516^{\pm 0.0201}$	$1.1130^{\pm 0.0274}$	$1.0978^{\pm 0.0461}$	$1.0785^{\pm 0.0528}$	$1.0978^{\pm 0.0461}$	$1.1742^{\pm 0.0390}$	$1.1002^{\pm 0.0187}$
1×10^{-6}	$1.1502^{\pm 0.0119}$	$1.1172^{\pm 0.0257}$	$1.1101^{\pm 0.0553}$	$1.0822^{\pm 0.0282}$	$1.1097^{\pm 0.0482}$	$1.1694^{\pm 0.0198}$	$1.1149^{\pm 0.0311}$	$1.0951^{\pm 0.0297}$	$1.1149^{\pm 0.0311}$	$1.1284^{\pm 0.0389}$	$1.1040^{\pm 0.0386}$

Table 7: Numerical results for Gradient Flow with the 8 Gaussians dataset ($d = 2$)

LR	SW	MaxSW	DSW	MaxKSW	iMSW	viMSW	oMSW	rMSW	EBSW	RPSW	EBRPSW
100	$0.7900^{\pm 0.0704}$	$4.4368^{\pm 0.0966}$	$3.0902^{\pm 0.1026}$	$2.9073^{\pm 0.3947}$	$2.2985^{\pm 0.1220}$	$1.2282^{\pm 0.0578}$	$0.8969^{\pm 0.0500}$	$1.2429^{\pm 0.1167}$	$10.5129^{\pm 0.2240}$	$1.2812^{\pm 0.0165}$	$11.0450^{\pm 0.8609}$
80	$0.6047^{\pm 0.0850}$	$3.4799^{\pm 0.0544}$	$2.4792^{\pm 0.2602}$	$2.0292^{\pm 0.2711}$	$1.7091^{\pm 0.0921}$	$0.9662^{\pm 0.0284}$	$0.6897^{\pm 0.0442}$	$0.9003^{\pm 0.0392}$	$8.5668^{\pm 0.3944}$	$1.0411^{\pm 0.0586}$	$7.8665^{\pm 0.8317}$
50	$0.3459^{\pm 0.0234}$	$2.7423^{\pm 0.5278}$	$1.0806^{\pm 0.3335}$	$1.7169^{\pm 0.1028}$	$1.0745^{\pm 0.0742}$	$0.5512^{\pm 0.0653}$	$0.3995^{\pm 0.0698}$	$0.5736^{\pm 0.0181}$	$4.3396^{\pm 0.3767}$	$0.6150^{\pm 0.0368}$	$4.0199^{\pm 0.0448}$
30	$0.2229^{\pm 0.0178}$	$1.8470^{\pm 0.1703}$	$0.8920^{\pm 0.0499}$	$1.5803^{\pm 0.1051}$	$0.7473^{\pm 0.0193}$	$0.3307^{\pm 0.0363}$	$0.2335^{\pm 0.0071}$	$0.3464^{\pm 0.0307}$	$2.3719^{\pm 0.0339}$	$0.3622^{\pm 0.0363}$	$2.7110^{\pm 0.3379}$
10	$0.0819^{\pm 0.0056}$	$1.2519^{\pm 0.0267}$	$0.1853^{\pm 0.0391}$	$0.9043^{\pm 0.0135}$	$0.5119^{\pm 0.0033}$	$0.1162^{\pm 0.0097}$	$0.0825^{\pm 0.0035}$	$0.1102^{\pm 0.0206}$	$1.0896^{\pm 0.0531}$	$0.1307^{\pm 0.0169}$	$1.0587^{\pm 0.0079}$
8	$0.0592^{\pm 0.0014}$	$1.0781^{\pm 0.0115}$	$0.1659^{\pm 0.0566}$	$0.7843^{\pm 0.0256}$	$0.4259^{\pm 0.0210}$	$0.0821^{\pm 0.0011}$	$0.0590^{\pm 0.0082}$	$0.0835^{\pm 0.0041}$	$1.0515^{\pm 0.0428}$	$0.1071^{\pm 0.0068}$	$0.9913^{\pm 0.0529}$
5	$0.0402^{\pm 0.0026}$	$0.7808^{\pm 0.0114}$	$0.0734^{\pm 0.0121}$	$0.5545^{\pm 0.0083}$	$0.3104^{\pm 0.0177}$	$0.0649^{\pm 0.0070}$	$0.0410^{\pm 0.0007}$	$0.0560^{\pm 0.0027}$	$0.0412^{\pm 0.0027}$	$0.0693^{\pm 0.0138}$	$0.0429^{\pm 0.0037}$
3	$0.0234^{\pm 0.0026}$	$0.5466^{\pm 0.0103}$	$0.0374^{\pm 0.0089}$	$0.4019^{\pm 0.0028}$	$0.2041^{\pm 0.0022}$	$0.0386^{\pm 0.0027}$	$0.0265^{\pm 0.0017}$	$0.0348^{\pm 0.0009}$	$0.0258^{\pm 0.0015}$	$0.0430^{\pm 0.0029}$	$0.0217^{\pm 0.0019}$
1	$0.0073^{\pm 0.0002}$	$0.2788^{\pm 0.0092}$	$0.0125^{\pm 0.0025}$	$0.1792^{\pm 0.0007}$	$0.0912^{\pm 0.0016}$	$0.0113^{\pm 0.0013}$	$0.0070^{\pm 0.0009}$	$0.0115^{\pm 0.0005}$	$0.0081^{\pm 0.0008}$	$0.0144^{\pm 0.0009}$	$0.0089^{\pm 0.0009}$
8×10^{-1}	$0.0067^{\pm 0.0007}$	$0.2448^{\pm 0.0066}$	$0.0099^{\pm 0.0007}$	$0.1514^{\pm 0.0017}$	$0.0805^{\pm 0.0016}$	$0.0082^{\pm 0.0013}$	$0.0063^{\pm 0.0003}$	$0.0094^{\pm 0.0010}$	$0.0063^{\pm 0.0006}$	$0.0117^{\pm 0.0003}$	$0.0062^{\pm 0.0004}$
5×10^{-1}	$0.0038^{\pm 0.0004}$	$0.1762^{\pm 0.0045}$	$0.0064^{\pm 0.0006}$	$0.1104^{\pm 0.0031}$	$0.0573^{\pm 0.0011}$	$0.0057^{\pm 0.0003}$	$0.0036^{\pm 0.0003}$	$0.0057^{\pm 0.0003}$	$0.0042^{\pm 0.0002}$	$0.0074^{\pm 0.0006}$	$0.0043^{\pm 0.0003}$
3×10^{-1}	$0.0026^{\pm 0.0002}$	$0.1282^{\pm 0.0012}$	$0.0036^{\pm 0.0001}$	$0.0788^{\pm 0.0018}$	$0.0381^{\pm 0.0004}$	$0.0034^{\pm 0.0001}$	$0.0023^{\pm 0.0001}$	$0.0035^{\pm 0.0001}$	$0.0022^{\pm 0.0003}$	$0.0041^{\pm 0.0003}$	$0.0023^{\pm 0.0001}$
1×10^{-1}	$0.0008^{\pm 0.0000}$	$0.0597^{\pm 0.0009}$	$0.0015^{\pm 0.0000}$	$0.0342^{\pm 0.0005}$	$0.0107^{\pm 0.0002}$	$0.0012^{\pm 0.0001}$	$0.0008^{\pm 0.0000}$	$0.0012^{\pm 0.0001}$	$0.0008^{\pm 0.0001}$	$0.0015^{\pm 0.0001}$	$0.0008^{\pm 0.0000}$
8×10^{-2}	$0.0006^{\pm 0.0000}$	$0.0499^{\pm 0.0003}$	$0.0010^{\pm 0.0000}$	$0.0282^{\pm 0.0004}$	$0.0086^{\pm 0.0000}$	$0.0010^{\pm 0.0000}$	$0.0006^{\pm 0.0000}$	$0.0009^{\pm 0.0000}$	$0.0006^{\pm 0.0000}$	$0.0012^{\pm 0.0001}$	$0.0006^{\pm 0.0000}$
5×10^{-2}	$0.0004^{\pm 0.0000}$	$0.0342^{\pm 0.0003}$	$0.0007^{\pm 0.0000}$	$0.0171^{\pm 0.0003}$	$0.0054^{\pm 0.0000}$	$0.0006^{\pm 0.0000}$	$0.0003^{\pm 0.0000}$	$0.0006^{\pm 0.0000}$	$0.0004^{\pm 0.0000}$	$0.0008^{\pm 0.0000}$	$0.0004^{\pm 0.0000}$
3×10^{-2}	$0.0002^{\pm 0.0000}$	$0.0272^{\pm 0.0072}$	$0.0004^{\pm 0.0000}$	$0.0080^{\pm 0.0000}$	$0.0031^{\pm 0.0002}$	$0.0004^{\pm 0.0000}$	$0.0002^{\pm 0.0000}$	$0.0003^{\pm 0.0000}$	$0.0003^{\pm 0.0000}$	$0.0005^{\pm 0.0001}$	$0.0002^{\pm 0.0000}$
1×10^{-2}	$0.1699^{\pm 0.0262}$	$0.0530^{\pm 0.0060}$	$0.0395^{\pm 0.0020}$	$0.0639^{\pm 0.0039}$	$0.1088^{\pm 0.0063}$	$0.1371^{\pm 0.0141}$	$0.1715^{\pm 0.0089}$	$0.1507^{\pm 0.0100}$	$0.0535^{\pm 0.0008}$	$0.0056^{\pm 0.0077}$	$0.0641^{\pm 0.0027}$
8×10^{-3}	$0.2678^{\pm 0.0292}$	$0.0706^{\pm 0.0028}$	$0.0564^{\pm 0.0037}$	$0.0922^{\pm 0.0046}$	$0.1784^{\pm 0.0108}$	$0.2101^{\pm 0.0076}$	$0.2381^{\pm 0.0090}$	$0.2667^{\pm 0.0288}$	$0.0968^{\pm 0.0027}$	$0.0010^{\pm 0.0012}$	$0.0925^{\pm 0.0011}$
5×10^{-3}	$0.4264^{\pm 0.0227}$	$0.1373^{\pm 0.0006}$	$0.1307^{\pm 0.0060}$	$0.2096^{\pm 0.0176}$	$0.3224^{\pm 0.0325}$	$0.3775^{\pm 0.0155}$	$0.4109^{\pm 0.0132}$	$0.3907^{\pm 0.0216}$	$0.1715^{\pm 0.0034}$	$0.0770^{\pm 0.0092}$	$0.1853^{\pm 0.0080}$
3×10^{-3}	$0.5863^{\pm 0.0435}$	$0.2441^{\pm 0.0004}$	$0.2907^{\pm 0.0178}$	$0.3481^{\pm 0.0071}$	$0.4836^{\pm 0.0280}$	$0.5586^{\pm 0.0159}$	$0.5553^{\pm 0.0304}$	$0.5572^{\pm 0.0043}$	$0.2927^{\pm 0.0148}$	$0.2330^{\pm 0.0223}$	$0.3111^{\pm 0.0292}$
1×10^{-3}	$0.7987^{\pm 0.0118}$	$0.7585^{\pm 0.0052}$	$0.7517^{\pm 0.0287}$	$0.7607^{\pm 0.0420}$	$0.7778^{\pm 0.0164}$	$0.7404^{\pm 0.0106}$	$0.7858^{\pm 0.0109}$	$0.7806^{\pm 0.0178}$	$0.7751^{\pm 0.0066}$	$0.7427^{\pm 0.0242}$	$0.7704^{\pm 0.0552}$
8×10^{-4}	$0.7785^{\pm 0.0360}$	$0.7701^{\pm 0.0215}$	$0.7445^{\pm 0.0253}$	$0.7571^{\pm 0.0562}$	$0.7751^{\pm 0.0262}$	$0.7760^{\pm 0.0339}$	$0.7826^{\pm 0.0458}$	$0.7852^{\pm 0.0142}$	$0.7615^{\pm 0.0246}$	$0.7313^{\pm 0.0053}$	$0.7520^{\pm 0.0261}$
5×10^{-4}	$0.7931^{\pm 0.0415}$	$0.7580^{\pm 0.0470}$	$0.7146^{\pm 0.0403}$	$0.7738^{\pm 0.0153}$	$0.7709^{\pm 0.0115}$	$0.7612^{\pm 0.0216}$	$0.7837^{\pm 0.0457}$	$0.8190^{\pm 0.0229}$	$0.7759^{\pm 0.0414}$	$0.7704^{\pm 0.0147}$	$0.7323^{\pm 0.0210}$
3×10^{-4}	$0.7726^{\pm 0.0087}$	$0.7806^{\pm 0.0640}$	$0.7752^{\pm 0.0142}$	$0.7582^{\pm 0.0184}$	$0.7892^{\pm 0.0316}$	$0.7936^{\pm 0.0294}$	$0.7779^{\pm 0.0213}$	$0.7843^{\pm 0.0390}$	$0.7901^{\pm 0.0189}$	$0.7528^{\pm 0.0385}$	$0.7730^{\pm 0.0263}$
1×10^{-4}	$0.7724^{\pm 0.0251}$	$0.7373^{\pm 0.0580}$	$0.7380^{\pm 0.0290}$	$0.7984^{\pm 0.0495}$	$0.7581^{\pm 0.0325}$	$0.8025^{\pm 0.0279}$	$0.7750^{\pm 0.0102}$	$0.7991^{\pm 0.0153}$	$0.7713^{\pm 0.0014}$	$0.7643^{\pm 0.0285}$	$0.7380^{\pm 0.0197}$
8×10^{-5}	$0.7802^{\pm 0.0147}$	$0.7643^{\pm 0.0223}$	$0.7921^{\pm 0.0231}$	$0.7104^{\pm 0.0534}$	$0.7957^{\pm 0.0327}$	$0.7725^{\pm 0.0192}$	$0.7747^{\pm 0.0062}$	$0.7413^{\pm 0.0228}$	$0.7748^{\pm 0.0297}$	$0.7585^{\pm 0.0187}$	$0.7867^{\pm 0.0122}$
5×10^{-5}	$0.7682^{\pm 0.0403}$	$0.7974^{\pm 0.0291}$	$0.8113^{\pm 0.0685}$	$0.7807^{\pm 0.0192}$	$0.7793^{\pm 0.0381}$	$0.8181^{\pm 0.0332}$	$0.8045^{\pm 0.0346}$	$0.7838^{\pm 0.0321}$	$0.7908^{\pm 0.0161}$	$0.7631^{\pm 0.0420}$	$0.7142^{\pm 0.0102}$
3×10^{-5}	$0.7980^{\pm 0.0513}$	$0.7476^{\pm 0.0426}$	$0.7775^{\pm 0.0282}$	$0.7561^{\pm 0.0425}$	$0.7576^{\pm 0.0305}$	$0.7579^{\pm 0.0212}$	$0.7556^{\pm 0.0193}$	$0.7656^{\pm 0.0462}$	$0.7890^{\pm 0.0293}$	$0.8213^{\pm 0.0123}$	$0.7827^{\pm 0.0255}$
1×10^{-5}	$0.7860^{\pm 0.0132}$	$0.7859^{\pm 0.0271}$	$0.7955^{\pm 0.0397}$	$0.7627^{\pm 0.0285}$	$0.7747^{\pm 0.0025}$	$0.7576^{\pm 0.0395}$	$0.7869^{\pm 0.0353}$	$0.7682^{\pm 0.0255}$	$0.7753^{\pm 0.0191}$	$0.7573^{\pm 0.0311}$	$0.7831^{\pm 0.0199}$
8×10^{-6}	$0.7860^{\pm 0.0132}$	$0.7859^{\pm 0.0271}$	$0.7955^{\pm 0.0397}$	$0.7627^{\pm 0.0285}$	$0.7747^{\pm 0.0025}$	$0.7576^{\pm 0.0395}$	$0.7869^{\pm 0.0353}$	$0.7682^{\pm 0.0255}$	$0.7753^{\pm 0.0191}$	$0.7573^{\pm 0.0311}$	$0.7831^{\pm 0.0199}$
5×10^{-6}	$0.7860^{\pm 0.0132}$	$0.7859^{\pm 0.0271}$	$0.7955^{\pm 0.0397}$	$0.7627^{\pm 0.0285}$	$0.7747^{\pm 0.0025}$	$0.7576^{\pm 0.0395}$	$0.7869^{\pm 0.0353}$	$0.7682^{\pm 0.0255}$	$0.7753^{\pm 0.0191}$	$0.7573^{\pm 0.0311}$	$0.7831^{\pm 0.0199}$
3×10^{-6}	$0.7980^{\pm 0.0513}$	$0.7476^{\pm 0.0426}$	$0.7775^{\pm 0.0282}$	$0.7561^{\pm 0.0425}$	$0.7576^{\pm 0.0305}$	$0.7579^{\pm 0.0212}$	$0.7556^{\pm 0.0193}$	$0.7656^{\pm 0.0462}$	$0.7890^{\pm 0.0293}$	$0.8213^{\pm 0.0123}$	$0.7827^{\pm 0.0255}$
1×10^{-6}	$0.7860^{\pm 0.0132}$	$0.7859^{\pm 0.0271}$	$0.7955^{\pm 0.0397}$	$0.7627^{\pm 0.0285}$	$0.7747^{\pm 0.0025}$	$0.7576^{\pm 0.0395}$	$0.7869^{\pm 0.0353}$	$0.7682^{\pm 0.0255}$	$0.7753^{\pm 0.0191}$	$0.7573^{\pm 0.0311}$	$0.7831^{\pm 0.0199}$

Table 8: Numerical results for Gradient Flow with the Knot dataset ($d = 50$)

LR	SW	MaxSW	DSW	MaxKSW	iMSW	viMSW	oMSW	rMSW	EBSW	RPSW	EBRPSW
100	$0.7547^{\pm 0.0159}$	$5.2763^{\pm 0.0581}$	$2.7308^{\pm 0.2356}$	$3.2047^{\pm 0.1553}$	$2.2315^{\pm 0.0152}$	$1.1926^{\pm 0.1009}$	$0.7655^{\pm 0.0725}$	$1.0225^{\pm 0.0259}$	$8.2665^{\pm 0.6763}$	$1.2582^{\pm 0.1625}$	$8.2852^{\pm 0.4012}$
80	$0.7061^{\pm 0.0588}$	$4.4078^{\pm 0.8741}$	$2.1265^{\pm 0.5707}$	$2.6758^{\pm 0.2432}$	$1.5470^{\pm 0.0262}$	$0.9951^{\pm 0.0213}$	$0.6729^{\pm 0.0501}$	$0.8837^{\pm 0.0816}$	$6.7290^{\pm 0.0731}$	$1.1278^{\pm 0.1272}$	$6.6422^{\pm 0.3327}$
50	$0.4126^{\pm 0.0458}$	$3.1516^{\pm 0.4662}$	$1.5598^{\pm 0.1044}$	$2.6188^{\pm 0.1909}$	$1.0823^{\pm 0.0751}$	$0.6173^{\pm 0.0160}$	$0.4052^{\pm 0.0346}$	$0.5515^{\pm 0.0068}$	$4.5307^{\pm 0.1181}$	$0.6737^{\pm 0.0310}$	$4.1734^{\pm 0.1881}$
30	$0.2164^{\pm 0.0158}$	$2.5630^{\pm 0.0389}$	$0.8967^{\pm 0.0765}$	$2.0890^{\pm 0.0750}$	$1.2381^{\pm 0.0289}$	$0.3187^{\pm 0.0404}$	$0.2560^{\pm 0.0072}$	$0.3502^{\pm 0.0163}$	$2.8793^{\pm 0.0917}$	$0.4439^{\pm 0.0344}$	$2.8941^{\pm 0.1512}$
10	$0.0798^{\pm 0.0005}$	$1.4811^{\pm 0.0137}$	$0.1845^{\pm 0.0386}$	$1.1085^{\pm 0.0246}$	$0.6727^{\pm 0.0053}$	$0.1076^{\pm 0.0092}$	$0.0733^{\pm 0.0010}$	$0.1136^{\pm 0.0098}$	$1.6626^{\pm 0.0213}$	$0.1442^{\pm 0.0078}$	$1.5850^{\pm 0.0564}$
8	$0.0701^{\pm 0.0045}$	$1.2975^{\pm 0.0386}$	$0.1219^{\pm 0.0369}$	$0.9595^{\pm 0.0109}$	$0.5404^{\pm 0.0106}$	$0.0885^{\pm 0.0099}$	$0.0584^{\pm 0.0057}$	$0.0884^{\pm 0.0097}$	$1.3687^{\pm 0.0730}$	$0.1110^{\pm 0.0185}$	$1.4399^{\pm 0.0461}$
5	$0.0387^{\pm 0.0020}$	$1.0414^{\pm 0.0267}$	$0.0892^{\pm 0.0231}$	$0.7089^{\pm 0.0243}$	$0.3520^{\pm 0.0050}$	$0.0595^{\pm 0.0041}$	$0.0373^{\pm 0.0030}$	$0.0508^{\pm 0.0028}$	$0.0395^{\pm 0.0016}$	$0.0723^{\pm 0.0041}$	$0.0426^{\pm 0.0054}$
3	$0.0253^{\pm 0.0009}$	$0.7394^{\pm 0.0176}$	$0.0374^{\pm 0.0112}$	$0.4655^{\pm 0.0047}$	$0.1601^{\pm 0.0228}$	$0.0301^{\pm 0.0002}$	$0.0235^{\pm 0.0011}$	$0.0314^{\pm 0.0018}$	$0.0226^{\pm 0.0022}$	$0.0438^{\pm 0.0020}$	$0.0239^{\pm 0.0020}$
1	$0.0079^{\pm 0.0003}$	$0.2967^{\pm 0.0070}$	$0.0146^{\pm 0.0022}$	$0.1618^{\pm 0.0059}$	$0.0558^{\pm 0.0016}$	$0.0112^{\pm 0.0014}$	$0.0076^{\pm 0.0004}$	$0.0120^{\pm 0.0013}$	$0.0078^{\pm 0.0009}$	$0.0150^{\pm 0.0005}$	$0.0089^{\pm 0.0003}$
8×10^{-1}	$0.0065^{\pm 0.0005}$	$0.2347^{\pm 0.0065}$	$0.0106^{\pm 0.0018}$	$0.0932^{\pm 0.0034}$	$0.0469^{\pm 0.0011}$	$0.0089^{\pm 0.0002}$	$0.0059^{\pm 0.0003}$	$0.0091^{\pm 0.0008}$	$0.0068^{\pm 0.0004}$	$0.0126^{\pm 0.0013}$	$0.0059^{\pm 0.0003}$
5×10^{-1}	$0.0039^{\pm 0.0002}$	$0.1408^{\pm 0.0040}$	$0.0072^{\pm 0.0003}$	$0.0627^{\pm 0.0017}$	$0.0350^{\pm 0.0005}$	$0.0053^{\pm 0.0004}$	$0.0043^{\pm 0.0003}$	$0.0056^{\pm 0.0006}$	$0.0038^{\pm 0.0002}$	$0.0081^{\pm 0.0007}$	$0.0039^{\pm 0.0003}$
3×10^{-1}	$0.0025^{\pm 0.0001}$	$0.0740^{\pm 0.0014}$	$0.0038^{\pm 0.0004}$	$0.0448^{\pm 0.0006}$	$0.0256^{\pm 0.0004}$	$0.0033^{\pm 0.0003}$	$0.0024^{\pm 0.0003}$	$0.0035^{\pm 0.0002}$	$0.0024^{\pm 0.0004}$	$0.0052^{\pm 0.0002}$	$0.0024^{\pm 0.0002}$
1×10^{-1}	$0.0009^{\pm 0.0001}$	$0.0338^{\pm 0.0004}$	$0.0014^{\pm 0.0001}$	$0.0250^{\pm 0.0009}$	$0.0123^{\pm 0.0001}$	$0.0013^{\pm 0.0001}$	$0.0008^{\pm 0.0001}$	$0.0011^{\pm 0.0002}$	$0.0008^{\pm 0.0000}$	$0.0017^{\pm 0.0001}$	$0.0009^{\pm 0.0000}$
8×10^{-2}	$0.0006^{\pm 0.0001}$	$0.0303^{\pm 0.0009}$	$0.0012^{\pm 0.0002}$	$0.0216^{\pm 0.0007}$	$0.0102^{\pm 0.0004}$	$0.0008^{\pm 0.0001}$	$0.0006^{\pm 0.0000}$	$0.0010^{\pm 0.0000}$	$0.0007^{\pm 0.0000}$	$0.0013^{\pm 0.0001}$	$0.0006^{\pm 0.0000}$
5×10^{-2}	$0.0004^{\pm 0.0000}$	$0.0246^{\pm 0.0009}$	$0.0007^{\pm 0.0001}$	$0.0154^{\pm 0.0001}$	$0.0065^{\pm 0.0001}$	$0.0006^{\pm 0.0001}$	$0.0004^{\pm 0.0000}$	$0.0006^{\pm 0.0000}$	$0.0004^{\pm 0.0000}$	$0.0009^{\pm 0.0001}$	$0.0004^{\pm 0.0000}$
3×10^{-2}	$0.1207^{\pm 0.0565}$	$0.0171^{\pm 0.0004}$	$0.0004^{\pm 0.0001}$	$0.1128^{\pm 0.0265}$	$0.1479^{\pm 0.0204}$	$0.0808^{\pm 0.0513}$	$0.0953^{\pm 0.0630}$	$0.1041^{\pm 0.0235}$	$0.0002^{\pm 0.0000}$	$0.0005^{\pm 0.0000}$	$0.0002^{\pm 0.0000}$
1×10^{-2}	$0.5471^{\pm 0.0314}$	$0.3074^{\pm 0.0063}$	$0.2104^{\pm 0.0210}$	$0.3963^{\pm 0.0099}$	$0.4644^{\pm 0.0191}$	$0.5370^{\pm 0.0153}$	$0.5546^{\pm 0.0361}$	$0.5783^{\pm 0.0261}$	$0.3110^{\pm 0.0059}$	$0.1862^{\pm 0.0244}$	$0.3056^{\pm 0.0195}$
8×10^{-3}	$0.6607^{\pm 0.0252}$	$0.3471^{\pm 0.0367}$	$0.2931^{\pm 0.0161}$	$0.4406^{\pm 0.0058}$	$0.5575^{\pm 0.0223}$	$0.6040^{\pm 0.0050}$	$0.6462^{\pm 0.0206}$	$0.6438^{\pm 0.0218}$	$0.3705^{\pm 0.0090}$	$0.2789^{\pm 0.0171}$	$0.3714^{\pm 0.0098}$
5×10^{-3}	$0.8309^{\pm 0.0754}$	$0.4564^{\pm 0.0277}$	$0.4910^{\pm 0.0170}$	$0.5249^{\pm 0.0144}$	$0.6980^{\pm 0.0091}$	$0.7606^{\pm 0.0329}$	$0.7976^{\pm 0.0299}$	$0.8461^{\pm 0.0325}$	$0.5150^{\pm 0.0186}$	$0.4749^{\pm 0.0196}$	$0.4967^{\pm 0.0190}$
3×10^{-3}	$0.9346^{\pm 0.0160}$	$0.5658^{\pm 0.0233}$	$0.6897^{\pm 0.0284}$	$0.6755^{\pm 0.0063}$	$0.7964^{\pm 0.0128}$	$0.8806^{\pm 0.0164}$	$0.9133^{\pm 0.0362}$	$0.9116^{\pm 0.0411}$	$0.6939^{\pm 0.0332}$	$0.6552^{\pm 0.0181}$	$0.7147^{\pm 0.0048}$
1×10^{-3}	$1.0732^{\pm 0.0251}$	$0.8076^{\pm 0.0065}$	$0.9769^{\pm 0.0156}$	$0.9318^{\pm 0.0178}$	$1.0055^{\pm 0.0251}$	$1.0421^{\pm 0.0599}$	$1.0351^{\pm 0.0384}$	$1.0009^{\pm 0.0010}$	$0.9582^{\pm 0.0105}$	$0.9311^{\pm 0.0464}$	$0.9782^{\pm 0.0298}$
8×10^{-4}	$1.0660^{\pm 0.0099}$	$0.8543^{\pm 0.0055}$	$0.9574^{\pm 0.0476}$	$0.9526^{\pm 0.0065}$	$1.0176^{\pm 0.0510}$	$1.0811^{\pm 0.0094}$	$1.0274^{\pm 0.0524}$	$1.0162^{\pm 0.0435}$	$1.0022^{\pm 0.0106}$	$1.0049^{\pm 0.0174}$	$0.9958^{\pm 0.0305}$
5×10^{-4}	$1.0721^{\pm 0.0250}$	$0.9709^{\pm 0.0153}$	$0.9846^{\pm 0.0195}$	$0.9930^{\pm 0.0107}$	$1.0416^{\pm 0.0409}$	$1.0991^{\pm 0.0175}$	$1.1032^{\pm 0.0322}$	$1.0534^{\pm 0.0326}$	$1.0223^{\pm 0.0399}$	$1.0613^{\pm 0.0227}$	$1.0550^{\pm 0.0291}$
3×10^{-4}	$1.0848^{\pm 0.0382}$	$1.0400^{\pm 0.0263}$	$1.0311^{\pm 0.0734}$	$1.0448^{\pm 0.0222}$	$1.0676^{\pm 0.0230}$	$1.0752^{\pm 0.0319}$	$1.0726^{\pm 0.0208}$	$1.1065^{\pm 0.0231}$	$1.0811^{\pm 0.0291}$	$1.0522^{\pm 0.0462}$	$1.0969^{\pm 0.0233}$
1×10^{-4}	$1.1030^{\pm 0.0147}$	$1.0385^{\pm 0.0311}$	$1.1083^{\pm 0.0502}$	$1.1365^{\pm 0.0700}$	$1.1079^{\pm 0.0175}$	$1.1241^{\pm 0.0141}$	$1.0896^{\pm 0.0660}$	$1.1286^{\pm 0.0108}$	$1.1013^{\pm 0.0297}$	$1.0749^{\pm 0.0316}$	$1.1096^{\pm 0.0245}$
8×10^{-5}	$1.0942^{\pm 0.0655}$	$1.0906^{\pm 0.0433}$	$1.1044^{\pm 0.0174}$	$1.1250^{\pm 0.0374}$	$1.1226^{\pm 0.0615}$	$1.0875^{\pm 0.0187}$	$1.0936^{\pm 0.0374}$	$1.0882^{\pm 0.0041}$	$1.0679^{\pm 0.0441}$	$1.1070^{\pm 0.0730}$	$1.0884^{\pm 0.0650}$
5×10^{-5}	$1.1349^{\pm 0.0215}$	$1.0513^{\pm 0.0381}$	$1.0550^{\pm 0.0217}$	$1.1064^{\pm 0.0454}$	$1.1607^{\pm 0.0151}$	$1.1231^{\pm 0.0358}$	$1.0709^{\pm 0.0248}$	$1.1488^{\pm 0.0098}$	$1.0924^{\pm 0.0221}$	$1.0870^{\pm 0.0166}$	$1.0661^{\pm 0.0188}$
3×10^{-5}	$1.0990^{\pm 0.0313}$	$1.0985^{\pm 0.0104}$	$1.0675^{\pm 0.0382}$	$1.0855^{\pm 0.0425}$	$1.1012^{\pm 0.0343}$	$1.1434^{\pm 0.0657}$	$1.1014^{\pm 0.0307}$	$1.1131^{\pm 0.0082}$	$1.1001^{\pm 0.0724}$	$1.0806^{\pm 0.0281}$	$1.1312^{\pm 0.0125}$
1×10^{-5}	$1.0485^{\pm 0.0162}$	$1.1895^{\pm 0.0512}$	$1.0654^{\pm 0.0337}$	$1.1409^{\pm 0.0482}$	$1.1335^{\pm 0.0215}$	$1.1170^{\pm 0.0207}$	$1.1062^{\pm 0.0395}$	$1.1249^{\pm 0.0194}$	$1.1434^{\pm 0.0072}$	$1.1078^{\pm 0.0490}$	$1.1283^{\pm 0.0508}$
8×10^{-6}	$1.1477^{\pm 0.0352}$	$1.1061^{\pm 0.0364}$	$1.0993^{\pm 0.0411}$	$1.1129^{\pm 0.0117}$	$1.1289^{\pm 0.0148}$	$1.0819^{\pm 0.0535}$	$1.0914^{\pm 0.0531}$	$1.0946^{\pm 0.0192}$	$1.1224^{\pm 0.0029}$	$1.0909^{\pm 0.0091}$	$1.0903^{\pm 0.0345}$
5×10^{-6}	$1.1039^{\pm 0.0245}$	$1.0986^{\pm 0.0137}$	$1.1386^{\pm 0.0528}$	$1.1225^{\pm 0.0556}$	$1.0472^{\pm 0.0103}$	$1.0801^{\pm 0.0381}$	$1.0740^{\pm 0.0291}$	$1.1074^{\pm 0.0248}$	$1.0940^{\pm 0.0571}$	$1.1170^{\pm 0.0424}$	$1.1369^{\pm 0.0183}$
3×10^{-6}	$1.0944^{\pm 0.0144}$	$1.1067^{\pm 0.0313}$	$1.0853^{\pm 0.0478}$	$1.1657^{\pm 0.0502}$	$1.0876^{\pm 0.0485}$	$1.0898^{\pm 0.0251}$	$1.1064^{\pm 0.0428}$	$1.1198^{\pm 0.0351}$	$1.1420^{\pm 0.0470}$	$1.1234^{\pm 0.0185}$	$1.1284^{\pm 0.0406}$
1×10^{-6}	$1.0992^{\pm 0.0299}$	$1.0725^{\pm 0.0256}$	$1.1476^{\pm 0.0510}$	$1.1265^{\pm 0.0194}$	$1.1256^{\pm 0.0139}$	$1.1005^{\pm 0.0335}$	$1.0683^{\pm 0.0558}$	$1.0980^{\pm 0.0285}$	$1.1159^{\pm 0.0161}$	$1.0996^{\pm 0.0567}$	$1.0926^{\pm 0.0170}$

Table 9: Numerical results for Gradient Flow with the 8 Gaussians dataset ($d = 50$)

LR	SW	MaxSW	DSW	MaxKSW	iMSW	viMSW	oMSW	rMSW	EBSW	RPSW	EBRPSW
100	$0.7762^{\pm 0.1023}$	$4.1470^{\pm 0.0352}$	$2.9805^{\pm 0.0642}$	$2.5629^{\pm 0.4443}$	$2.0662^{\pm 0.1062}$	$1.2985^{\pm 0.1430}$	$0.7226^{\pm 0.0545}$	$1.0492^{\pm 0.0837}$	$8.9485^{\pm 0.8584}$	$1.2748^{\pm 0.0332}$	$8.9753^{\pm 0.7660}$
80	$0.6647^{\pm 0.0359}$	$3.2381^{\pm 0.0367}$	$2.2192^{\pm 0.2220}$	$2.1054^{\pm 0.0841}$	$1.6892^{\pm 0.1181}$	$0.9497^{\pm 0.1181}$	$0.5743^{\pm 0.0987}$	$0.9389^{\pm 0.0556}$	$7.5420^{\pm 0.9074}$	$1.1116^{\pm 0.0758}$	$7.5470^{\pm 0.5091}$
50	$0.3936^{\pm 0.0427}$	$2.3955^{\pm 0.5774}$	$1.4745^{\pm 0.1405}$	$1.5428^{\pm 0.0838}$	$1.1088^{\pm 0.0616}$	$0.5304^{\pm 0.0832}$	$0.3697^{\pm 0.0686}$	$0.6062^{\pm 0.0722}$	$4.2090^{\pm 0.0792}$	$0.6496^{\pm 0.0087}$	$4.4983^{\pm 0.4854}$
30	$0.2592^{\pm 0.0305}$	$1.7158^{\pm 0.0576}$	$0.9065^{\pm 0.0385}$	$1.3463^{\pm 0.0646}$	$0.7396^{\pm 0.0385}$	$0.3412^{\pm 0.0455}$	$0.2242^{\pm 0.0245}$	$0.3475^{\pm 0.0257}$	$2.6767^{\pm 0.0900}$	$0.3805^{\pm 0.0476}$	$2.5791^{\pm 0.0469}$
10	$0.0843^{\pm 0.0069}$	$1.2179^{\pm 0.0184}$	$0.2198^{\pm 0.0271}$	$0.8623^{\pm 0.0304}$	$0.5016^{\pm 0.0059}$	$0.1124^{\pm 0.0121}$	$0.0718^{\pm 0.0044}$	$0.1114^{\pm 0.0134}$	$1.0026^{\pm 0.0296}$	$0.1323^{\pm 0.0029}$	$0.9923^{\pm 0.0272}$
8	$0.0563^{\pm 0.0017}$	$1.0353^{\pm 0.0213}$	$0.2052^{\pm 0.0283}$	$0.7273^{\pm 0.0190}$	$0.4534^{\pm 0.0015}$	$0.0943^{\pm 0.0109}$	$0.0583^{\pm 0.0041}$	$0.0887^{\pm 0.0075}$	$0.9543^{\pm 0.0348}$	$0.1118^{\pm 0.0083}$	$0.9426^{\pm 0.0210}$
5	$0.0399^{\pm 0.0046}$	$0.7122^{\pm 0.0174}$	$0.0919^{\pm 0.0103}$	$0.5057^{\pm 0.0135}$	$0.3198^{\pm 0.0128}$	$0.0546^{\pm 0.0033}$	$0.0387^{\pm 0.0038}$	$0.0542^{\pm 0.0050}$	$0.0414^{\pm 0.0033}$	$0.0714^{\pm 0.0053}$	$0.0407^{\pm 0.0013}$
3	$0.0298^{\pm 0.0048}$	$0.5158^{\pm 0.0225}$	$0.0368^{\pm 0.0029}$	$0.3866^{\pm 0.0041}$	$0.2153^{\pm 0.0097}$	$0.0334^{\pm 0.0026}$	$0.0256^{\pm 0.0007}$	$0.0322^{\pm 0.0026}$	$0.0271^{\pm 0.0061}$	$0.0448^{\pm 0.0043}$	$0.0280^{\pm 0.0066}$
1	$0.0084^{\pm 0.0009}$	$0.2754^{\pm 0.0034}$	$0.0120^{\pm 0.0024}$	$0.1864^{\pm 0.0049}$	$0.1010^{\pm 0.0016}$	$0.0105^{\pm 0.0007}$	$0.0072^{\pm 0.0003}$	$0.0113^{\pm 0.0020}$	$0.0082^{\pm 0.0001}$	$0.0150^{\pm 0.0013}$	$0.0076^{\pm 0.0006}$
8×10^{-1}	$0.0064^{\pm 0.0006}$	$0.2443^{\pm 0.0031}$	$0.0164^{\pm 0.0065}$	$0.1644^{\pm 0.0012}$	$0.0856^{\pm 0.0006}$	$0.0103^{\pm 0.0005}$	$0.0059^{\pm 0.0009}$	$0.0092^{\pm 0.0001}$	$0.0066^{\pm 0.0007}$	$0.0128^{\pm 0.0010}$	$0.0122^{\pm 0.0084}$
5×10^{-1}	$0.0039^{\pm 0.0003}$	$0.1865^{\pm 0.0033}$	$0.0061^{\pm 0.0005}$	$0.1229^{\pm 0.0014}$	$0.0555^{\pm 0.0006}$	$0.0058^{\pm 0.0005}$	$0.0044^{\pm 0.0003}$	$0.0056^{\pm 0.0004}$	$0.0042^{\pm 0.0003}$	$0.0075^{\pm 0.0009}$	$0.0040^{\pm 0.0002}$
3×10^{-1}	$0.0023^{\pm 0.0000}$	$0.1347^{\pm 0.0006}$	$0.0035^{\pm 0.0003}$	$0.0823^{\pm 0.0012}$	$0.0367^{\pm 0.0031}$	$0.0038^{\pm 0.0001}$	$0.0022^{\pm 0.0002}$	$0.0032^{\pm 0.0002}$	$0.0022^{\pm 0.0003}$	$0.0041^{\pm 0.0004}$	$0.0024^{\pm 0.0002}$
1×10^{-1}	$0.0008^{\pm 0.0001}$	$0.0598^{\pm 0.0005}$	$0.0013^{\pm 0.0000}$	$0.0301^{\pm 0.0001}$	$0.0080^{\pm 0.0001}$	$0.0011^{\pm 0.0002}$	$0.0073^{\pm 0.0092}$	$0.0011^{\pm 0.0000}$	$0.0009^{\pm 0.0001}$	$0.0015^{\pm 0.0000}$	$0.0008^{\pm 0.0000}$
8×10^{-2}	$0.0006^{\pm 0.0001}$	$0.0479^{\pm 0.0024}$	$0.0011^{\pm 0.0001}$	$0.0232^{\pm 0.0001}$	$0.0059^{\pm 0.0001}$	$0.0010^{\pm 0.0000}$	$0.0006^{\pm 0.0000}$	$0.0009^{\pm 0.0001}$	$0.0007^{\pm 0.0001}$	$0.0011^{\pm 0.0001}$	$0.0007^{\pm 0.0000}$
5×10^{-2}	$0.0004^{\pm 0.0000}$	$0.0358^{\pm 0.0047}$	$0.0007^{\pm 0.0000}$	$0.0165^{\pm 0.0048}$	$0.0034^{\pm 0.0001}$	$0.0005^{\pm 0.0000}$	$0.0004^{\pm 0.0000}$	$0.0006^{\pm 0.0000}$	$0.0004^{\pm 0.0000}$	$0.0008^{\pm 0.0000}$	$0.0004^{\pm 0.0000}$
3×10^{-2}	$0.0030^{\pm 0.0040}$	$0.0210^{\pm 0.0051}$	$0.0004^{\pm 0.0000}$	$0.0124^{\pm 0.0082}$	$0.0021^{\pm 0.0001}$	$0.0003^{\pm 0.0000}$	$0.0002^{\pm 0.0000}$	$0.0003^{\pm 0.0000}$	$0.0002^{\pm 0.0000}$	$0.0004^{\pm 0.0000}$	$0.0073^{\pm 0.0100}$
1×10^{-2}	$0.1316^{\pm 0.0033}$	$0.0501^{\pm 0.0031}$	$0.0302^{\pm 0.0003}$	$0.0598^{\pm 0.0009}$	$0.0906^{\pm 0.0095}$	$0.1090^{\pm 0.0164}$	$0.1320^{\pm 0.0154}$	$0.1244^{\pm 0.0033}$	$0.0472^{\pm 0.0033}$	$0.0001^{\pm 0.0000}$	$0.0546^{\pm 0.0011}$
8×10^{-3}	$0.2177^{\pm 0.0424}$	$0.0575^{\pm 0.0036}$	$0.0526^{\pm 0.0021}$	$0.0824^{\pm 0.0072}$	$0.1306^{\pm 0.0105}$	$0.1858^{\pm 0.0147}$	$0.2047^{\pm 0.0427}$	$0.2182^{\pm 0.0078}$	$0.0760^{\pm 0.0004}$	$0.0111^{\pm 0.0155}$	$0.0743^{\pm 0.0014}$
5×10^{-3}	$0.3952^{\pm 0.0247}$	$0.1087^{\pm 0.0024}$	$0.1025^{\pm 0.0047}$	$0.1622^{\pm 0.0081}$	$0.3173^{\pm 0.0275}$	$0.3845^{\pm 0.0282}$	$0.4071^{\pm 0.0187}$	$0.4151^{\pm 0.0329}$	$0.1511^{\pm 0.0098}$	$0.0622^{\pm 0.0067}$	$0.1411^{\pm 0.0066}$
3×10^{-3}	$0.5586^{\pm 0.0151}$	$0.2257^{\pm 0.0155}$	$0.2471^{\pm 0.0098}$	$0.3464^{\pm 0.0133}$	$0.4916^{\pm 0.0096}$	$0.5395^{\pm 0.0327}$	$0.5485^{\pm 0.0271}$	$0.5645^{\pm 0.0297}$	$0.3053^{\pm 0.0138}$	$0.2302^{\pm 0.0240}$	$0.2800^{\pm 0.0173}$
1×10^{-3}	$0.7368^{\pm 0.0071}$	$0.5209^{\pm 0.0295}$	$0.6224^{\pm 0.0271}$	$0.6223^{\pm 0.0124}$	$0.6734^{\pm 0.0601}$	$0.6885^{\pm 0.0230}$	$0.7442^{\pm 0.0126}$	$0.7277^{\pm 0.0133}$	$0.6449^{\pm 0.0427}$	$0.6091^{\pm 0.0403}$	$0.6653^{\pm 0.0580}$
8×10^{-4}	$0.7768^{\pm 0.0225}$	$0.6110^{\pm 0.0293}$	$0.6699^{\pm 0.0478}$	$0.6629^{\pm 0.0105}$	$0.7314^{\pm 0.0226}$	$0.7445^{\pm 0.0266}$	$0.7542^{\pm 0.0457}$	$0.7287^{\pm 0.0117}$	$0.6765^{\pm 0.0522}$	$0.6724^{\pm 0.0188}$	$0.6419^{\pm 0.0300}$
5×10^{-4}	$0.7490^{\pm 0.0225}$	$0.6869^{\pm 0.0197}$	$0.6954^{\pm 0.0774}$	$0.7295^{\pm 0.0491}$	$0.7891^{\pm 0.0284}$	$0.7690^{\pm 0.0318}$	$0.7695^{\pm 0.0567}$	$0.7668^{\pm 0.0369}$	$0.7451^{\pm 0.0210}$	$0.6965^{\pm 0.0405}$	$0.7216^{\pm 0.0103}$
3×10^{-4}	$0.7827^{\pm 0.0395}$	$0.6642^{\pm 0.0428}$	$0.7586^{\pm 0.0062}$	$0.7633^{\pm 0.0165}$	$0.7398^{\pm 0.0400}$	$0.7723^{\pm 0.0548}$	$0.7431^{\pm 0.0404}$	$0.7707^{\pm 0.0251}$	$0.7329^{\pm 0.0414}$	$0.7311^{\pm 0.0240}$	$0.7943^{\pm 0.0136}$
1×10^{-4}	$0.8140^{\pm 0.0313}$	$0.7916^{\pm 0.0414}$	$0.7861^{\pm 0.0208}$	$0.7831^{\pm 0.0390}$	$0.8007^{\pm 0.0424}$	$0.8189^{\pm 0.0443}$	$0.7720^{\pm 0.0456}$	$0.8255^{\pm 0.0249}$	$0.7897^{\pm 0.0402}$	$0.7853^{\pm 0.0086}$	$0.7956^{\pm 0.0433}$
8×10^{-5}	$0.8251^{\pm 0.0557}$	$0.7742^{\pm 0.0468}$	$0.7564^{\pm 0.0264}$	$0.7817^{\pm 0.0112}$	$0.7730^{\pm 0.0625}$	$0.7962^{\pm 0.0255}$	$0.7882^{\pm 0.0464}$	$0.7486^{\pm 0.0271}$	$0.8021^{\pm 0.0180}$	$0.7926^{\pm 0.0361}$	$0.7754^{\pm 0.0085}$
5×10^{-5}	$0.8447^{\pm 0.0485}$	$0.8429^{\pm 0.0513}$	$0.7968^{\pm 0.0078}$	$0.7805^{\pm 0.0351}$	$0.7909^{\pm 0.0394}$	$0.7495^{\pm 0.0347}$	$0.7897^{\pm 0.0257}$	$0.8281^{\pm 0.0520}$	$0.8203^{\pm 0.0164}$	$0.7809^{\pm 0.0271}$	$0.8060^{\pm 0.0349}$
3×10^{-5}	$0.8487^{\pm 0.0216}$	$0.8205^{\pm 0.0185}$	$0.8784^{\pm 0.0763}$	$0.8266^{\pm 0.0327}$	$0.7722^{\pm 0.0272}$	$0.7971^{\pm 0.0129}$	$0.8065^{\pm 0.0415}$	$0.7762^{\pm 0.0201}$	$0.7999^{\pm 0.0260}$	$0.7997^{\pm 0.0335}$	$0.8139^{\pm 0.0303}$
1×10^{-5}	$0.8185^{\pm 0.0507}$	$0.8229^{\pm 0.0235}$	$0.8475^{\pm 0.0495}$	$0.8096^{\pm 0.0189}$	$0.8023^{\pm 0.0096}$	$0.7732^{\pm 0.0190}$	$0.7889^{\pm 0.0274}$	$0.8114^{\pm 0.0154}$	$0.8616^{\pm 0.0231}$	$0.8170^{\pm 0.0166}$	$0.8138^{\pm 0.0658}$
8×10^{-6}	$0.7791^{\pm 0.0343}$	$0.8024^{\pm 0.0081}$	$0.8053^{\pm 0.0170}$	$0.7963^{\pm 0.0172}$	$0.8017^{\pm 0.0222}$	$0.7826^{\pm 0.0209}$	$0.8119^{\pm 0.0277}$	$0.8194^{\pm 0.0162}$	$0.8093^{\pm 0.0391}$	$0.7836^{\pm 0.0407}$	$0.8051^{\pm 0.0071}$
5×10^{-6}	$0.8004^{\pm 0.0436}$	$0.8115^{\pm 0.0065}$	$0.7933^{\pm 0.0083}$	$0.8196^{\pm 0.0535}$	$0.8087^{\pm 0.0554}$	$0.8442^{\pm 0.0464}$	$0.8336^{\pm 0.0498}$	$0.7760^{\pm 0.0264}$	$0.7825^{\pm 0.0188}$	$0.8363^{\pm 0.0375}$	$0.8199^{\pm 0.0169}$
3×10^{-6}	$0.8095^{\pm 0.0577}$	$0.7820^{\pm 0.0421}$	$0.8062^{\pm 0.0289}$	$0.7960^{\pm 0.0257}$	$0.8325^{\pm 0.0368}$	$0.8312^{\pm 0.0423}$	$0.8435^{\pm 0.0051}$	$0.8024^{\pm 0.0159}$	$0.7826^{\pm 0.0256}$	$0.8178^{\pm 0.0533}$	$0.7688^{\pm 0.0116}$
1×10^{-6}	$0.8090^{\pm 0.0194}$	$0.8014^{\pm 0.0267}$	$0.7747^{\pm 0.0278}$	$0.7973^{\pm 0.0361}$	$0.8236^{\pm 0.0444}$	$0.8056^{\pm 0.0109}$	$0.8253^{\pm 0.0041}$	$0.8288^{\pm 0.0236}$	$0.8208^{\pm 0.0290}$	$0.8208^{\pm 0.0422}$	$0.8211^{\pm 0.0454}$

Table 10: Numerical results for Gradient Flow with the Swiss dataset ($d = 50$)

LR	SW	MaxSW	DSW	MaxKSW	iMSW	viMSW	oMSW	rMSW	EBSW	RPSW	EBRPSW
100	$0.8643^{\pm 0.0480}$	$2.4103^{\pm 0.0243}$	$3.0249^{\pm 0.0992}$	$2.8531^{\pm 0.3598}$	$1.9101^{\pm 0.0904}$	$0.9712^{\pm 0.0227}$	$0.8463^{\pm 0.0478}$	$0.9859^{\pm 0.0645}$	$16.7324^{\pm 0.3058}$	$0.8955^{\pm 0.0380}$	$16.3655^{\pm 0.8700}$
80	$0.6125^{\pm 0.0264}$	$1.7029^{\pm 0.0444}$	$2.2926^{\pm 0.1929}$	$2.2264^{\pm 0.1102}$	$1.5574^{\pm 0.1757}$	$0.7954^{\pm 0.0465}$	$0.6772^{\pm 0.0312}$	$0.7505^{\pm 0.0809}$	$10.8613^{\pm 0.1851}$	$0.7372^{\pm 0.0058}$	$12.4947^{\pm 0.8208}$
50	$0.4262^{\pm 0.0147}$	$2.0519^{\pm 0.0738}$	$1.5044^{\pm 0.0269}$	$1.7652^{\pm 0.2612}$	$1.1316^{\pm 0.0373}$	$0.4388^{\pm 0.0366}$	$0.4079^{\pm 0.0182}$	$0.4803^{\pm 0.0080}$	$7.0184^{\pm 0.8989}$	$0.4605^{\pm 0.0160}$	$7.5545^{\pm 0.5874}$
30	$0.2685^{\pm 0.0022}$	$2.0192^{\pm 0.1526}$	$0.9237^{\pm 0.0327}$	$1.3850^{\pm 0.1726}$	$0.7873^{\pm 0.0145}$	$0.2954^{\pm 0.0180}$	$0.2619^{\pm 0.0018}$	$0.3041^{\pm 0.0170}$	$3.7698^{\pm 0.2624}$	$0.2759^{\pm 0.0187}$	$4.3376^{\pm 0.2378}$
10	$0.0972^{\pm 0.0068}$	$1.3748^{\pm 0.0235}$	$0.2432^{\pm 0.0166}$	$1.1019^{\pm 0.0105}$	$0.6308^{\pm 0.0093}$	$0.1131^{\pm 0.0041}$	$0.0959^{\pm 0.0036}$	$0.1160^{\pm 0.0059}$	$1.3600^{\pm 0.0370}$	$0.1129^{\pm 0.0066}$	$1.3044^{\pm 0.0268}$
8	$0.0767^{\pm 0.0032}$	$1.2367^{\pm 0.0192}$	$0.1845^{\pm 0.0193}$	$0.9386^{\pm 0.0170}$	$0.5502^{\pm 0.0083}$	$0.0966^{\pm 0.0035}$	$0.0768^{\pm 0.0016}$	$0.0957^{\pm 0.0045}$	$1.2489^{\pm 0.0797}$	$0.0904^{\pm 0.0078}$	$1.2369^{\pm 0.0945}$
5	$0.0558^{\pm 0.0024}$	$0.9070^{\pm 0.0095}$	$0.0955^{\pm 0.0141}$	$0.6769^{\pm 0.0079}$	$0.3837^{\pm 0.0111}$	$0.0724^{\pm 0.0017}$	$0.0563^{\pm 0.0020}$	$0.0718^{\pm 0.0008}$	$0.0564^{\pm 0.0016}$	$0.0581^{\pm 0.0027}$	$0.0559^{\pm 0.0017}$
3	$0.0430^{\pm 0.0009}$	$0.6628^{\pm 0.0086}$	$0.0400^{\pm 0.0069}$	$0.4668^{\pm 0.0040}$	$0.2629^{\pm 0.0053}$	$0.0583^{\pm 0.0009}$	$0.0437^{\pm 0.0001}$	$0.0611^{\pm 0.0002}$	$0.0441^{\pm 0.0005}$	$0.0349^{\pm 0.0015}$	$0.0432^{\pm 0.0005}$
1	$0.0224^{\pm 0.0001}$	$0.3285^{\pm 0.0080}$	$0.0095^{\pm 0.0010}$	$0.2235^{\pm 0.0048}$	$0.1153^{\pm 0.0003}$	$0.0296^{\pm 0.0004}$	$0.0224^{\pm 0.0002}$	$0.0296^{\pm 0.0004}$	$0.0225^{\pm 0.0003}$	$0.0109^{\pm 0.0006}$	$0.0225^{\pm 0.0003}$
8×10^{-1}	$0.0165^{\pm 0.0001}$	$0.2840^{\pm 0.0054}$	$0.0088^{\pm 0.0004}$	$0.1902^{\pm 0.0019}$	$0.1007^{\pm 0.0003}$	$0.0164^{\pm 0.0001}$	$0.0166^{\pm 0.0002}$	$0.0163^{\pm 0.0002}$	$0.0165^{\pm 0.0001}$	$0.0104^{\pm 0.0004}$	$0.0167^{\pm 0.0002}$
5×10^{-1}	$0.0080^{\pm 0.0000}$	$0.2212^{\pm 0.0012}$	$0.0057^{\pm 0.0008}$	$0.1405^{\pm 0.0036}$	$0.0740^{\pm 0.0003}$	$0.0046^{\pm 0.0003}$	$0.0080^{\pm 0.0000}$	$0.0047^{\pm 0.0001}$	$0.0080^{\pm 0.0001}$	$0.0067^{\pm 0.0009}$	$0.0080^{\pm 0.0001}$
3×10^{-1}	$0.0046^{\pm 0.0001}$	$0.1589^{\pm 0.0011}$	$0.0035^{\pm 0.0006}$	$0.1022^{\pm 0.0014}$	$0.0498^{\pm 0.0002}$	$0.0029^{\pm 0.0001}$	$0.0046^{\pm 0.0001}$	$0.0030^{\pm 0.0000}$	$0.0045^{\pm 0.0000}$	$0.0038^{\pm 0.0002}$	$0.0045^{\pm 0.0001}$
1×10^{-1}	$0.0013^{\pm 0.0002}$	$0.0787^{\pm 0.0024}$	$0.0012^{\pm 0.0001}$	$0.0449^{\pm 0.0002}$	$0.0149^{\pm 0.0001}$	$0.0010^{\pm 0.0000}$	$0.0015^{\pm 0.0000}$	$0.0010^{\pm 0.0000}$	$0.0013^{\pm 0.0002}$	$0.0012^{\pm 0.0000}$	$0.0013^{\pm 0.0002}$
8×10^{-2}	$0.0011^{\pm 0.0001}$	$0.0650^{\pm 0.0015}$	$0.0010^{\pm 0.0001}$	$0.0410^{\pm 0.0020}$	$0.0117^{\pm 0.0001}$	$0.0008^{\pm 0.0001}$	$0.0012^{\pm 0.0001}$	$0.0008^{\pm 0.0000}$	$0.0012^{\pm 0.0000}$	$0.0010^{\pm 0.0000}$	$0.0012^{\pm 0.0000}$
5×10^{-2}	$0.0004^{\pm 0.0000}$	$0.0485^{\pm 0.0026}$	$0.0007^{\pm 0.0001}$	$0.0257^{\pm 0.0003}$	$0.0075^{\pm 0.0001}$	$0.0005^{\pm 0.0000}$	$0.0004^{\pm 0.0000}$	$0.0005^{\pm 0.0000}$	$0.0004^{\pm 0.0000}$	$0.0006^{\pm 0.0001}$	$0.0004^{\pm 0.0000}$
3×10^{-2}	$0.0045^{\pm 0.0060}$	$0.0390^{\pm 0.0057}$	$0.0004^{\pm 0.0000}$	$0.0242^{\pm 0.0062}$	$0.0115^{\pm 0.0104}$	$0.0257^{\pm 0.0048}$	$0.0105^{\pm 0.0119}$	$0.0343^{\pm 0.0043}$	$0.0003^{\pm 0.0000}$	$0.0004^{\pm 0.0000}$	$0.0003^{\pm 0.0000}$
1×10^{-2}	$0.2804^{\pm 0.0226}$	$0.0720^{\pm 0.0046}$	$0.0502^{\pm 0.0035}$	$0.1019^{\pm 0.0020}$	$0.1841^{\pm 0.0085}$	$0.2933^{\pm 0.0198}$	$0.2789^{\pm 0.0208}$	$0.3458^{\pm 0.0357}$	$0.1314^{\pm 0.0043}$	$0.0346^{\pm 0.0075}$	$0.1413^{\pm 0.0088}$
8×10^{-3}	$0.3572^{\pm 0.0416}$	$0.0894^{\pm 0.0071}$	$0.0633^{\pm 0.0024}$	$0.1397^{\pm 0.0037}$	$0.2467^{\pm 0.0085}$	$0.4134^{\pm 0.0194}$	$0.3828^{\pm 0.0295}$	$0.4188^{\pm 0.0111}$	$0.1794^{\pm 0.0026}$	$0.0640^{\pm 0.0079}$	$0.1718^{\pm 0.0033}$
5×10^{-3}	$0.7284^{\pm 0.0214}$	$0.1556^{\pm 0.0114}$	$0.1280^{\pm 0.0096}$	$0.2343^{\pm 0.0145}$	$0.4257^{\pm 0.0203}$	$0.5644^{\pm 0.0408}$	$0.5199^{\pm 0.0729}$	$0.5625^{\pm 0.0292}$	$0.2970^{\pm 0.0092}$	$0.1951^{\pm 0.0367}$	$0.3066^{\pm 0.0145}$
3×10^{-3}	$0.5855^{\pm 0.0378}$	$0.2609^{\pm 0.0091}$	$0.2891^{\pm 0.0310}$	$0.3839^{\pm 0.0267}$	$0.5528^{\pm 0.0394}$	$0.6505^{\pm 0.0218}$	$0.6237^{\pm 0.0231}$	$0.6863^{\pm 0.0198}$	$0.4115^{\pm 0.0076}$	$0.4189^{\pm 0.0281}$	$0.4130^{\pm 0.0205}$
1×10^{-3}	$0.7769^{\pm 0.0223}$	$0.5345^{\pm 0.0245}$	$0.6896^{\pm 0.0310}$	$0.6344^{\pm 0.0232}$	$0.6984^{\pm 0.0159}$	$0.7895^{\pm 0.0378}$	$0.7531^{\pm 0.0458}$	$0.7658^{\pm 0.0333}$	$0.6480^{\pm 0.0414}$	$0.5981^{\pm 0.0180}$	$0.6073^{\pm 0.0028}$
8×10^{-4}	$0.7223^{\pm 0.0554}$	$0.5782^{\pm 0.0436}$	$0.6573^{\pm 0.0227}$	$0.6719^{\pm 0.0370}$	$0.6832^{\pm 0.0032}$	$0.8203^{\pm 0.0277}$	$0.7193^{\pm 0.0335}$	$0.7771^{\pm 0.0393}$	$0.6816^{\pm 0.0188}$	$0.6958^{\pm 0.0269}$	$0.6699^{\pm 0.0186}$
5×10^{-4}	$0.5284^{\pm 0.0214}$	$0.6497^{\pm 0.0282}$	$0.7132^{\pm 0.0115}$	$0.7223^{\pm 0.0115}$	$0.7746^{\pm 0.0050}$	$0.7589^{\pm 0.0108}$	$0.7435^{\pm 0.0341}$	$0.8034^{\pm 0.0074}$	$0.7169^{\pm 0.0187}$	$0.6618^{\pm 0.0051}$	$0.6806^{\pm 0.0069}$
3×10^{-4}	$0.7738^{\pm 0.0239}$	$0.6827^{\pm 0.0282}$	$0.7920^{\pm 0.0172}$	$0.7138^{\pm 0.0237}$	$0.7059^{\pm 0.0415}$	$0.7849^{\pm 0.0082}$	$0.7744^{\pm 0.0022}$	$0.8368^{\pm 0.0076}$	$0.7549^{\pm 0.0262}$	$0.7491^{\pm 0.0323}$	$0.7581^{\pm 0.0283}$
1×10^{-4}	$0.7769^{\pm 0.0223}$	$0.7339^{\pm 0.0433}$	$0.8135^{\pm 0.0696}$	$0.7113^{\pm 0.0250}$	$0.7601^{\pm 0.0404}$	$0.8799^{\pm 0.0589}$	$0.7841^{\pm 0.0240}$	$0.8516^{\pm 0.0175}$	$0.7906^{\pm 0.0125}$	$0.7495^{\pm 0.0050}$	$0.7495^{\pm 0.0843}$
8×10^{-5}	$0.7693^{\pm 0.0421}$	$0.7755^{\pm 0.0336}$	$0.8598^{\pm 0.0077}$	$0.7652^{\pm 0.0212}$	$0.7680^{\pm 0.0261}$	$0.8133^{\pm 0.0309}$	$0.8206^{\pm 0.0287}$	$0.8416^{\pm 0.0336}$	$0.7758^{\pm 0.0432}$	$0.7305^{\pm 0.0133}$	$0.7520^{\pm 0.0274}$
5×10^{-5}	$0.7831^{\pm 0.0237}$	$0.7964^{\pm 0.0122}$	$0.7991^{\pm 0.0563}$	$0.7612^{\pm 0.0231}$	$0.7013^{\pm 0.0435}$	$0.8473^{\pm 0.0135}$	$0.8103^{\pm 0.0092}$	$0.8216^{\pm 0.0423}$	$0.7741^{\pm 0.0211}$	$0.7610^{\pm 0.0130}$	$0.8096^{\pm 0.0157}$
3×10^{-5}	$0.7928^{\pm 0.0350}$	$0.7748^{\pm 0.0484}$	$0.8389^{\pm 0.0115}$	$0.7534^{\pm 0.0318}$	$0.7539^{\pm 0.0087}$	$0.8650^{\pm 0.0221}$	$0.7570^{\pm 0.0352}$	$0.8469^{\pm 0.0187}$	$0.7786^{\pm 0.0070}$	$0.7785^{\pm 0.0239}$	$0.7547^{\pm 0.0152}$
1×10^{-5}	$0.7588^{\pm 0.0398}$	$0.7469^{\pm 0.0374}$	$0.8472^{\pm 0.0336}$	$0.7930^{\pm 0.0286}$	$0.7672^{\pm 0.0217}$	$0.8318^{\pm 0.0185}$	$0.7799^{\pm 0.0125}$	$0.8329^{\pm 0.0328}$	$0.7835^{\pm 0.0145}$	$0.7804^{\pm 0.0246}$	$0.7885^{\pm 0.0138}$
8×10^{-6}	$0.7720^{\pm 0.0243}$	$0.7906^{\pm 0.0141}$	$0.8344^{\pm 0.0208}$	$0.7622^{\pm 0.0255}$	$0.7477^{\pm 0.0283}$	$0.8835^{\pm 0.0241}$	$0.7494^{\pm 0.0229}$	$0.8103^{\pm 0.0076}$	$0.7818^{\pm 0.0347}$	$0.7729^{\pm 0.0175}$	$0.7692^{\pm 0.0128}$
5×10^{-6}	$0.7414^{\pm 0.0346}$	$0.7499^{\pm 0.0495}$	$0.8574^{\pm 0.0395}$	$0.8278^{\pm 0.0333}$	$0.7953^{\pm 0.0533}$	$0.8492^{\pm 0.0526}$	$0.7898^{\pm 0.0170}$	$0.8617^{\pm 0.0261}$	$0.7642^{\pm 0.0455}$	$0.7480^{\pm 0.0184}$	$0.7467^{\pm 0.0161}$
3×10^{-6}	$0.8039^{\pm 0.0412}$	$0.7761^{\pm 0.0199}$	$0.8475^{\pm 0.0209}$	$0.8175^{\pm 0.0604}$	$0.7411^{\pm 0.0401}$	$0.8466^{\pm 0.0131}$	$0.7385^{\pm 0.0290}$	$0.8266^{\pm 0.0428}$	$0.7709^{\pm 0.0121}$	$0.8049^{\pm 0.0306}$	$0.7930^{\pm 0.0426}$
1×10^{-6}	$0.7711^{\pm 0.0045}$	$0.7751^{\pm 0.0052}$	$0.8456^{\pm 0.0207}$	$0.8004^{\pm 0.0423}$	$0.7304^{\pm 0.0497}$	$0.8543^{\pm 0.0218}$	$0.7518^{\pm 0.0316}$	$0.8319^{\pm 0.0460}$	$0.7475^{\pm 0.0265}$	$0.7541^{\pm 0.0503}$	$0.7424^{\pm 0.0243}$

Table 11: Numerical results for Gradient Flow with the Knot dataset ($d = 100$)

LR	SW	MaxSW	DSW	MaxKSW	iMSW	viMSW	oMSW	rMSW	EBSW	RPSW	EBRPSW
100	$0.8999^{\pm 0.0210}$	$2.7462^{\pm 0.0473}$	$2.8735^{\pm 0.1213}$	$3.2882^{\pm 0.2012}$	$2.0599^{\pm 0.0665}$	$0.9238^{\pm 0.0107}$	$0.8228^{\pm 0.0550}$	$1.0054^{\pm 0.0460}$	$11.1502^{\pm 0.5452}$	$0.9537^{\pm 0.0065}$	$11.5940^{\pm 0.2757}$
80	$0.6750^{\pm 0.0370}$	$2.2929^{\pm 0.0425}$	$2.3658^{\pm 0.1707}$	$2.8335^{\pm 0.0114}$	$1.6259^{\pm 0.0782}$	$0.7803^{\pm 0.0317}$	$0.6715^{\pm 0.0205}$	$0.8045^{\pm 0.0388}$	$9.5332^{\pm 0.1474}$	$0.7804^{\pm 0.0251}$	$9.4759^{\pm 0.1591}$
50	$0.3877^{\pm 0.0209}$	$3.0593^{\pm 0.5077}$	$1.5513^{\pm 0.0461}$	$2.5637^{\pm 0.1616}$	$1.1430^{\pm 0.0018}$	$0.4871^{\pm 0.0066}$	$0.4036^{\pm 0.0349}$	$0.4982^{\pm 0.0160}$	$5.9085^{\pm 0.0718}$	$0.5273^{\pm 0.0293}$	$6.1440^{\pm 0.0923}$
30	$0.2629^{\pm 0.0154}$	$2.4795^{\pm 0.3492}$	$0.9380^{\pm 0.0379}$	$2.3604^{\pm 0.1002}$	$1.3570^{\pm 0.0652}$	$0.2752^{\pm 0.0008}$	$0.2549^{\pm 0.0194}$	$0.2956^{\pm 0.0055}$	$3.7218^{\pm 0.1138}$	$0.3304^{\pm 0.0195}$	$3.6760^{\pm 0.0457}$
10	$0.0815^{\pm 0.0022}$	$1.6234^{\pm 0.0147}$	$0.2669^{\pm 0.0078}$	$1.2923^{\pm 0.0119}$	$0.8413^{\pm 0.0116}$	$0.0996^{\pm 0.0031}$	$0.0862^{\pm 0.0043}$	$0.0859^{\pm 0.0026}$	$1.8692^{\pm 0.0718}$	$0.1037^{\pm 0.0045}$	$1.9169^{\pm 0.0229}$
8	$0.0676^{\pm 0.0019}$	$1.5109^{\pm 0.0179}$	$0.1902^{\pm 0.0150}$	$1.1465^{\pm 0.0356}$	$0.6969^{\pm 0.0184}$	$0.0778^{\pm 0.0033}$	$0.0684^{\pm 0.0045}$	$0.0781^{\pm 0.0031}$	$1.8447^{\pm 0.0816}$	$0.0903^{\pm 0.0086}$	$1.8146^{\pm 0.1152}$
5	$0.0446^{\pm 0.0021}$	$1.1718^{\pm 0.0410}$	$0.0936^{\pm 0.0274}$	$0.8477^{\pm 0.0221}$	$0.4585^{\pm 0.0021}$	$0.0440^{\pm 0.0019}$	$0.0412^{\pm 0.0005}$	$0.0473^{\pm 0.0021}$	$0.0454^{\pm 0.0038}$	$0.0558^{\pm 0.0023}$	$0.0435^{\pm 0.0026}$
3	$0.0279^{\pm 0.0011}$	$0.9264^{\pm 0.0262}$	$0.0375^{\pm 0.0020}$	$0.6073^{\pm 0.0108}$	$0.2748^{\pm 0.0066}$	$0.0288^{\pm 0.0008}$	$0.0288^{\pm 0.0013}$	$0.0292^{\pm 0.0011}$	$0.0280^{\pm 0.0014}$	$0.0360^{\pm 0.0014}$	$0.0282^{\pm 0.0009}$
1	$0.0133^{\pm 0.0003}$	$0.4048^{\pm 0.0096}$	$0.0106^{\pm 0.0007}$	$0.2178^{\pm 0.0017}$	$0.0716^{\pm 0.0009}$	$0.0124^{\pm 0.0006}$	$0.0140^{\pm 0.0002}$	$0.0119^{\pm 0.0005}$	$0.0137^{\pm 0.0010}$	$0.0134^{\pm 0.0002}$	$0.0133^{\pm 0.0004}$
8×10^{-1}	$0.0120^{\pm 0.0003}$	$0.3242^{\pm 0.0024}$	$0.0091^{\pm 0.0015}$	$0.1738^{\pm 0.0022}$	$0.0590^{\pm 0.0012}$	$0.0104^{\pm 0.0004}$	$0.0125^{\pm 0.0005}$	$0.0105^{\pm 0.0001}$	$0.0124^{\pm 0.0002}$	$0.0100^{\pm 0.0004}$	$0.0128^{\pm 0.0007}$
5×10^{-1}	$0.0098^{\pm 0.0001}$	$0.1982^{\pm 0.0021}$	$0.0055^{\pm 0.0004}$	$0.0843^{\pm 0.0015}$	$0.0424^{\pm 0.0009}$	$0.0077^{\pm 0.0004}$	$0.0096^{\pm 0.0001}$	$0.0078^{\pm 0.0002}$	$0.0097^{\pm 0.0002}$	$0.0065^{\pm 0.0005}$	$0.0098^{\pm 0.0001}$
3×10^{-1}	$0.0057^{\pm 0.0000}$	$0.0981^{\pm 0.0066}$	$0.0035^{\pm 0.0001}$	$0.0578^{\pm 0.0012}$	$0.0312^{\pm 0.0008}$	$0.0058^{\pm 0.0001}$	$0.0059^{\pm 0.0000}$	$0.0058^{\pm 0.0000}$	$0.0057^{\pm 0.0001}$	$0.0038^{\pm 0.0001}$	$0.0058^{\pm 0.0001}$
1×10^{-1}	$0.0016^{\pm 0.0001}$	$0.0439^{\pm 0.0013}$	$0.0013^{\pm 0.0001}$	$0.0290^{\pm 0.0003}$	$0.0159^{\pm 0.0003}$	$0.0010^{\pm 0.0000}$	$0.0015^{\pm 0.0000}$	$0.0009^{\pm 0.0001}$	$0.0015^{\pm 0.0001}$	$0.0013^{\pm 0.0001}$	$0.0016^{\pm 0.0001}$
8×10^{-2}	$0.0010^{\pm 0.0000}$	$0.0385^{\pm 0.0006}$	$0.0010^{\pm 0.0001}$	$0.0261^{\pm 0.0002}$	$0.0135^{\pm 0.0002}$	$0.0008^{\pm 0.0000}$	$0.0011^{\pm 0.0002}$	$0.0008^{\pm 0.0000}$	$0.0011^{\pm 0.0001}$	$0.0010^{\pm 0.0000}$	$0.0011^{\pm 0.0001}$
5×10^{-2}	$0.0421^{\pm 0.0445}$	$0.0297^{\pm 0.0008}$	$0.0006^{\pm 0.0001}$	$0.0200^{\pm 0.0001}$	$0.0462^{\pm 0.0330}$	$0.0350^{\pm 0.0487}$	$0.0308^{\pm 0.0428}$	$0.0279^{\pm 0.0351}$	$0.0005^{\pm 0.0000}$	$0.0007^{\pm 0.0000}$	$0.0004^{\pm 0.0000}$
3×10^{-2}	$0.3033^{\pm 0.0398}$	$0.0219^{\pm 0.0003}$	$0.0004^{\pm 0.0000}$	$0.1169^{\pm 0.0074}$	$0.2579^{\pm 0.0305}$	$0.2547^{\pm 0.0021}$	$0.2924^{\pm 0.0156}$	$0.2703^{\pm 0.0201}$	$0.0600^{\pm 0.0394}$	$0.0004^{\pm 0.0000}$	$0.0803^{\pm 0.0208}$
1×10^{-2}	$0.6815^{\pm 0.0581}$	$0.3337^{\pm 0.0280}$	$0.2210^{\pm 0.0083}$	$0.4191^{\pm 0.0139}$	$0.5300^{\pm 0.0373}$	$0.6595^{\pm 0.0196}$	$0.6801^{\pm 0.0112}$	$0.6874^{\pm 0.0216}$	$0.4444^{\pm 0.0156}$	$0.3403^{\pm 0.0255}$	$0.4169^{\pm 0.0118}$
8×10^{-3}	$0.7851^{\pm 0.0479}$	$0.3731^{\pm 0.0236}$	$0.3234^{\pm 0.0314}$	$0.4651^{\pm 0.0083}$	$0.6044^{\pm 0.0072}$	$0.6893^{\pm 0.0299}$	$0.7789^{\pm 0.0269}$	$0.7410^{\pm 0.0204}$	$0.4848^{\pm 0.0111}$	$0.4244^{\pm 0.0203}$	$0.5039^{\pm 0.0169}$
5×10^{-3}	$0.8508^{\pm 0.0152}$	$0.4733^{\pm 0.0185}$	$0.4979^{\pm 0.0280}$	$0.5620^{\pm 0.0089}$	$0.7247^{\pm 0.0213}$	$0.8281^{\pm 0.0402}$	$0.8774^{\pm 0.0232}$	$0.9089^{\pm 0.0383}$	$0.6009^{\pm 0.0188}$	$0.6378^{\pm 0.0079}$	$0.6152^{\pm 0.0155}$
3×10^{-3}	$1.0324^{\pm 0.0447}$	$0.5853^{\pm 0.0110}$	$0.6446^{\pm 0.0219}$	$0.6916^{\pm 0.0141}$	$0.8709^{\pm 0.0107}$	$0.9276^{\pm 0.0026}$	$0.9828^{\pm 0.0431}$	$0.9791^{\pm 0.0088}$	$0.7829^{\pm 0.0191}$	$0.8020^{\pm 0.0098}$	$0.7405^{\pm 0.0263}$
1×10^{-3}	$1.0512^{\pm 0.0197}$	$0.8431^{\pm 0.0326}$	$0.8869^{\pm 0.0205}$	$0.9131^{\pm 0.0176}$	$1.0226^{\pm 0.0193}$	$1.0683^{\pm 0.0350}$	$1.0648^{\pm 0.0060}$	$1.0571^{\pm 0.0341}$	$1.0056^{\pm 0.0210}$	$1.0151^{\pm 0.0527}$	$1.0046^{\pm 0.0041}$
8×10^{-4}	$1.0779^{\pm 0.0182}$	$0.8908^{\pm 0.0561}$	$0.9720^{\pm 0.0380}$	$0.9676^{\pm 0.0211}$	$1.0130^{\pm 0.0372}$	$1.0758^{\pm 0.0137}$	$1.0793^{\pm 0.0151}$	$1.0885^{\pm 0.0170}$	$1.0231^{\pm 0.0193}$	$1.0303^{\pm 0.0372}$	$1.0333^{\pm 0.0203}$
5×10^{-4}	$1.0988^{\pm 0.0184}$	$0.9505^{\pm 0.0212}$	$1.0193^{\pm 0.0270}$	$0.9798^{\pm 0.0157}$	$1.0378^{\pm 0.0249}$	$1.1476^{\pm 0.0273}$	$1.1071^{\pm 0.0083}$	$1.0677^{\pm 0.0578}$	$1.0552^{\pm 0.0253}$	$1.0367^{\pm 0.0548}$	$1.0364^{\pm 0.0551}$
3×10^{-4}	$1.1348^{\pm 0.0345}$	$1.0047^{\pm 0.0080}$	$1.0176^{\pm 0.0270}$	$1.0377^{\pm 0.0328}$	$1.0758^{\pm 0.0090}$	$1.1305^{\pm 0.0379}$	$1.1396^{\pm 0.0675}$	$1.1136^{\pm 0.0423}$	$1.0734^{\pm 0.0079}$	$1.0659^{\pm 0.0355}$	$1.0757^{\pm 0.0183}$
1×10^{-4}	$1.0925^{\pm 0.0405}$	$1.0953^{\pm 0.0485}$	$1.1185^{\pm 0.0088}$	$1.0751^{\pm 0.0584}$	$1.0750^{\pm 0.0136}$	$1.0832^{\pm 0.0300}$	$1.0511^{\pm 0.0267}$	$1.1315^{\pm 0.0190}$	$1.0838^{\pm 0.0455}$	$1.1096^{\pm 0.0402}$	$1.1157^{\pm 0.0268}$
8×10^{-5}	$1.1123^{\pm 0.0250}$	$1.0700^{\pm 0.0271}$	$1.0796^{\pm 0.0224}$	$1.0788^{\pm 0.0533}$	$1.1064^{\pm 0.0140}$	$1.1237^{\pm 0.0328}$	$1.0980^{\pm 0.0204}$	$1.1517^{\pm 0.0253}$	$1.0981^{\pm 0.0303}$	$1.0822^{\pm 0.0578}$	$1.0918^{\pm 0.0191}$
5×10^{-5}	$1.1325^{\pm 0.0183}$	$1.0869^{\pm 0.0071}$	$1.1188^{\pm 0.0054}$	$1.1100^{\pm 0.0342}$	$1.1137^{\pm 0.0182}$	$1.0774^{\pm 0.0188}$	$1.1139^{\pm 0.0363}$	$1.1536^{\pm 0.0548}$	$1.1520^{\pm 0.0330}$	$1.0898^{\pm 0.0257}$	$1.0977^{\pm 0.0389}$
3×10^{-5}	$1.0957^{\pm 0.0242}$	$1.1209^{\pm 0.0221}$	$1.0884^{\pm 0.0234}$	$1.0532^{\pm 0.0352}$	$1.0918^{\pm 0.0186}$	$1.1116^{\pm 0.0255}$	$1.1636^{\pm 0.0557}$	$1.0753^{\pm 0.0131}$	$1.1040^{\pm 0.0511}$	$1.0965^{\pm 0.0552}$	$1.0595^{\pm 0.0265}$
1×10^{-5}	$1.1133^{\pm 0.0149}$	$1.1525^{\pm 0.0283}$	$1.0880^{\pm 0.0347}$	$1.1194^{\pm 0.0511}$	$1.1360^{\pm 0.0050}$	$1.0627^{\pm 0.0312}$	$1.1375^{\pm 0.0438}$	$1.0889^{\pm 0.0484}$	$1.1512^{\pm 0.1043}$	$1.1299^{\pm 0.0701}$	$1.0808^{\pm 0.0393}$
8×10^{-6}	$1.1463^{\pm 0.0507}$	$1.1160^{\pm 0.0364}$	$1.1042^{\pm 0.0119}$	$1.1087^{\pm 0.0458}$	$1.1419^{\pm 0.0305}$	$1.0656^{\pm 0.0384}$	$1.1161^{\pm 0.0382}$	$1.1148^{\pm 0.0397}$	$1.1235^{\pm 0.0733}$	$1.1152^{\pm 0.0240}$	$1.1092^{\pm 0.0640}$
5×10^{-6}	$1.0819^{\pm 0.0338}$	$1.0975^{\pm 0.0435}$	$1.0985^{\pm 0.0218}$	$1.0971^{\pm 0.0492}$	$1.1398^{\pm 0.0347}$	$1.1297^{\pm 0.0235}$	$1.1038^{\pm 0.0064}$	$1.1181^{\pm 0.0381}$	$1.1171^{\pm 0.0500}$	$1.0907^{\pm 0.0028}$	$1.1375^{\pm 0.0591}$
3×10^{-6}	$1.0974^{\pm 0.0341}$	$1.1184^{\pm 0.0258}$	$1.0649^{\pm 0.0178}$	$1.1366^{\pm 0.0448}$	$1.0637^{\pm 0.0139}$	$1.1125^{\pm 0.0327}$	$1.0915^{\pm 0.0324}$	$1.1307^{\pm 0.0267}$	$1.1339^{\pm 0.0241}$	$1.1261^{\pm 0.0402}$	$1.1100^{\pm 0.0351}$
1×10^{-6}	$1.1113^{\pm 0.0431}$	$1.0936^{\pm 0.0302}$	$1.1580^{\pm 0.0174}$	$1.0748^{\pm 0.0342}$	$1.1037^{\pm 0.0867}$	$1.0442^{\pm 0.0314}$	$1.0524^{\pm 0.0236}$	$1.0905^{\pm 0.0209}$	$1.1355^{\pm 0.0364}$	$1.1215^{\pm 0.0309}$	$1.0954^{\pm 0.0074}$

Table 12: Numerical results for Gradient Flow with the 8 Gaussians dataset ($d = 100$)

LR	SW	MaxSW	DSW	MaxKSW	iMSW	viMSW	oMSW	rMSW	EBSW	RPSW	EBRPSW
100	$0.8453^{\pm 0.0023}$	$1.7094^{\pm 0.0385}$	$2.8643^{\pm 0.2356}$	$2.6594^{\pm 0.1227}$	$2.1708^{\pm 0.0707}$	$0.8819^{\pm 0.0498}$	$0.8580^{\pm 0.0126}$	$1.0229^{\pm 0.0815}$	$14.9925^{\pm 0.9149}$	$0.9446^{\pm 0.0524}$	$14.6610^{\pm 0.9302}$
80	$0.6462^{\pm 0.0144}$	$2.3625^{\pm 1.6301}$	$2.5657^{\pm 0.0937}$	$2.1892^{\pm 0.1915}$	$1.6570^{\pm 0.0787}$	$0.7463^{\pm 0.0104}$	$0.6837^{\pm 0.0626}$	$0.7806^{\pm 0.0436}$	$10.8735^{\pm 1.6920}$	$0.7065^{\pm 0.0500}$	$11.0594^{\pm 0.9880}$
50	$0.4404^{\pm 0.0069}$	$1.8941^{\pm 0.0347}$	$1.4256^{\pm 0.0602}$	$1.7448^{\pm 0.0856}$	$1.1720^{\pm 0.0426}$	$0.5025^{\pm 0.0142}$	$0.4153^{\pm 0.0104}$	$0.5028^{\pm 0.0436}$	$6.8204^{\pm 0.2040}$	$0.4404^{\pm 0.0366}$	$6.4021^{\pm 0.4725}$
30	$0.2634^{\pm 0.0069}$	$1.9391^{\pm 0.0814}$	$0.8874^{\pm 0.0252}$	$1.3388^{\pm 0.0205}$	$0.8062^{\pm 0.0750}$	$0.2830^{\pm 0.0113}$	$0.2585^{\pm 0.0043}$	$0.2728^{\pm 0.0145}$	$4.2188^{\pm 0.4268}$	$0.2838^{\pm 0.0237}$	$3.8089^{\pm 0.3879}$
10	$0.0925^{\pm 0.0017}$	$1.1779^{\pm 0.1149}$	$0.2637^{\pm 0.0336}$	$1.0617^{\pm 0.0060}$	$0.6473^{\pm 0.0297}$	$0.1026^{\pm 0.0080}$	$0.0863^{\pm 0.0036}$	$0.1088^{\pm 0.0061}$	$1.2991^{\pm 0.0956}$	$0.1026^{\pm 0.0062}$	$1.4357^{\pm 0.1469}$
8	$0.0805^{\pm 0.0066}$	$1.1334^{\pm 0.0718}$	$0.1843^{\pm 0.0303}$	$0.8777^{\pm 0.0231}$	$0.5424^{\pm 0.0050}$	$0.0784^{\pm 0.0049}$	$0.0720^{\pm 0.0038}$	$0.0819^{\pm 0.0019}$	$1.1654^{\pm 0.0456}$	$0.0922^{\pm 0.0077}$	$1.1620^{\pm 0.0320}$
5	$0.0511^{\pm 0.0016}$	$0.8703^{\pm 0.0221}$	$0.0881^{\pm 0.0173}$	$0.6358^{\pm 0.0074}$	$0.3933^{\pm 0.0074}$	$0.0533^{\pm 0.0052}$	$0.0512^{\pm 0.0027}$	$0.0461^{\pm 0.0020}$	$0.0530^{\pm 0.0013}$	$0.0613^{\pm 0.0094}$	$0.0513^{\pm 0.0030}$
3	$0.0380^{\pm 0.0004}$	$0.6049^{\pm 0.0037}$	$0.0383^{\pm 0.0082}$	$0.4436^{\pm 0.0070}$	$0.2733^{\pm 0.0050}$	$0.0265^{\pm 0.0031}$	$0.0390^{\pm 0.0005}$	$0.0279^{\pm 0.0015}$	$0.0366^{\pm 0.0013}$	$0.0359^{\pm 0.0020}$	$0.0411^{\pm 0.0050}$
1	$0.0177^{\pm 0.0003}$	$0.3048^{\pm 0.0009}$	$0.0111^{\pm 0.0010}$	$0.2203^{\pm 0.0029}$	$0.1282^{\pm 0.0017}$	$0.0100^{\pm 0.0004}$	$0.0210^{\pm 0.0050}$	$0.0098^{\pm 0.0004}$	$0.0175^{\pm 0.0003}$	$0.0131^{\pm 0.0003}$	$0.0178^{\pm 0.0002}$
8×10^{-1}	$0.0132^{\pm 0.0002}$	$0.2803^{\pm 0.0073}$	$0.0144^{\pm 0.0094}$	$0.2008^{\pm 0.0039}$	$0.1082^{\pm 0.0018}$	$0.0080^{\pm 0.0005}$	$0.0132^{\pm 0.0001}$	$0.0079^{\pm 0.0002}$	$0.0130^{\pm 0.0003}$	$0.0104^{\pm 0.0008}$	$0.0129^{\pm 0.0001}$
5×10^{-1}	$0.0070^{\pm 0.0002}$	$0.2261^{\pm 0.0033}$	$0.0052^{\pm 0.0006}$	$0.1506^{\pm 0.0017}$	$0.0731^{\pm 0.0010}$	$0.0046^{\pm 0.0001}$	$0.0071^{\pm 0.0001}$	$0.0049^{\pm 0.0002}$	$0.0071^{\pm 0.0002}$	$0.0062^{\pm 0.0006}$	$0.0070^{\pm 0.0001}$
3×10^{-1}	$0.0035^{\pm 0.0001}$	$0.1716^{\pm 0.0022}$	$0.0036^{\pm 0.0003}$	$0.1090^{\pm 0.0017}$	$0.0521^{\pm 0.0018}$	$0.0097^{\pm 0.0098}$	$0.0035^{\pm 0.0000}$	$0.0029^{\pm 0.0003}$	$0.0037^{\pm 0.0000}$	$0.0037^{\pm 0.0003}$	$0.0035^{\pm 0.0001}$
1×10^{-1}	$0.0009^{\pm 0.0000}$	$0.0774^{\pm 0.0003}$	$0.0012^{\pm 0.0001}$	$0.0426^{\pm 0.0005}$	$0.0124^{\pm 0.0001}$	$0.0010^{\pm 0.0000}$	$0.0010^{\pm 0.0000}$	$0.0081^{\pm 0.0101}$	$0.0010^{\pm 0.0000}$	$0.0013^{\pm 0.0001}$	$0.0009^{\pm 0.0000}$
8×10^{-2}	$0.0007^{\pm 0.0000}$	$0.0675^{\pm 0.0002}$	$0.0009^{\pm 0.0000}$	$0.0357^{\pm 0.0027}$	$0.0093^{\pm 0.0003}$	$0.0008^{\pm 0.0000}$	$0.0007^{\pm 0.0000}$	$0.0008^{\pm 0.0000}$	$0.0007^{\pm 0.0000}$	$0.0010^{\pm 0.0000}$	$0.0007^{\pm 0.0000}$
5×10^{-2}	$0.0004^{\pm 0.0000}$	$0.0507^{\pm 0.0039}$	$0.0007^{\pm 0.0001}$	$0.0209^{\pm 0.0006}$	$0.0050^{\pm 0.0001}$	$0.0005^{\pm 0.0000}$	$0.0005^{\pm 0.0000}$	$0.0005^{\pm 0.0000}$	$0.0005^{\pm 0.0000}$	$0.0007^{\pm 0.0000}$	$0.0004^{\pm 0.0000}$
3×10^{-2}	$0.0189^{\pm 0.0099}$	$0.0342^{\pm 0.0022}$	$0.0004^{\pm 0.0000}$	$0.0122^{\pm 0.0010}$	$0.0129^{\pm 0.0081}$	$0.0116^{\pm 0.0092}$	$0.0075^{\pm 0.0102}$	$0.0195^{\pm 0.0063}$	$0.0081^{\pm 0.0111}$	$0.0004^{\pm 0.0000}$	$0.0003^{\pm 0.0000}$
1×10^{-2}	$0.2747^{\pm 0.0110}$	$0.0596^{\pm 0.0018}$	$0.0432^{\pm 0.0029}$	$0.0831^{\pm 0.0027}$	$0.1521^{\pm 0.0144}$	$0.2559^{\pm 0.0187}$	$0.2458^{\pm 0.0317}$	$0.2713^{\pm 0.0137}$	$0.1024^{\pm 0.0050}$	$0.0238^{\pm 0.0042}$	$0.1028^{\pm 0.0059}$
8×10^{-3}	$0.3540^{\pm 0.0134}$	$0.0722^{\pm 0.0044}$	$0.0560^{\pm 0.0033}$	$0.1025^{\pm 0.0034}$	$0.2334^{\pm 0.0147}$	$0.3502^{\pm 0.0444}$	$0.3782^{\pm 0.0345}$	$0.3498^{\pm 0.0251}$	$0.1400^{\pm 0.0049}$	$0.0539^{\pm 0.0031}$	$0.1535^{\pm 0.0130}$
5×10^{-3}	$0.5383^{\pm 0.0144}$	$0.1306^{\pm 0.0024}$	$0.1028^{\pm 0.0079}$	$0.1954^{\pm 0.0088}$	$0.4190^{\pm 0.0172}$	$0.4794^{\pm 0.0464}$	$0.5292^{\pm 0.0509}$	$0.5352^{\pm 0.0263}$	$0.2629^{\pm 0.0107}$	$0.1642^{\pm 0.0141}$	$0.2691^{\pm 0.0017}$
3×10^{-3}	$0.6450^{\pm 0.0303}$	$0.2361^{\pm 0.0044}$	$0.2410^{\pm 0.0100}$	$0.3577^{\pm 0.0077}$	$0.5810^{\pm 0.0530}$	$0.6048^{\pm 0.0730}$	$0.6123^{\pm 0.0442}$	$0.6584^{\pm 0.0437}$	$0.3916^{\pm 0.0181}$	$0.4047^{\pm 0.0529}$	$0.3907^{\pm 0.0190}$
1×10^{-3}	$0.7594^{\pm 0.0267}$	$0.5412^{\pm 0.0211}$	$0.6013^{\pm 0.0194}$	$0.6430^{\pm 0.0117}$	$0.7360^{\pm 0.0238}$	$0.7566^{\pm 0.0276}$	$0.7211^{\pm 0.0482}$	$0.7498^{\pm 0.0312}$	$0.6015^{\pm 0.0460}$	$0.6453^{\pm 0.0495}$	$0.7185^{\pm 0.0149}$
8×10^{-4}	$0.7160^{\pm 0.0087}$	$0.5678^{\pm 0.0112}$	$0.6477^{\pm 0.0109}$	$0.6684^{\pm 0.0222}$	$0.7186^{\pm 0.0174}$	$0.7828^{\pm 0.0445}$	$0.7589^{\pm 0.0165}$	$0.7703^{\pm 0.0604}$	$0.6718^{\pm 0.0382}$	$0.6836^{\pm 0.0211}$	$0.7163^{\pm 0.0128}$
5×10^{-4}	$0.7653^{\pm 0.0262}$	$0.6971^{\pm 0.0526}$	$0.7226^{\pm 0.0039}$	$0.6803^{\pm 0.0267}$	$0.7617^{\pm 0.0291}$	$0.7912^{\pm 0.0393}$	$0.7891^{\pm 0.0067}$	$0.7953^{\pm 0.0266}$	$0.7830^{\pm 0.0425}$	$0.7378^{\pm 0.0145}$	$0.6751^{\pm 0.0220}$
3×10^{-4}	$0.8457^{\pm 0.0352}$	$0.7321^{\pm 0.0326}$	$0.7639^{\pm 0.0215}$	$0.7465^{\pm 0.0305}$	$0.7948^{\pm 0.0215}$	$0.7908^{\pm 0.0204}$	$0.7646^{\pm 0.0033}$	$0.7864^{\pm 0.0466}$	$0.7100^{\pm 0.0429}$	$0.7491^{\pm 0.0203}$	$0.7938^{\pm 0.0186}$
1×10^{-4}	$0.8093^{\pm 0.0227}$	$0.7885^{\pm 0.0195}$	$0.7749^{\pm 0.0105}$	$0.7928^{\pm 0.0380}$	$0.8168^{\pm 0.0158}$	$0.8058^{\pm 0.0158}$	$0.8151^{\pm 0.0481}$	$0.8039^{\pm 0.0251}$	$0.7846^{\pm 0.0221}$	$0.7626^{\pm 0.0589}$	$0.8299^{\pm 0.0233}$
8×10^{-5}	$0.7991^{\pm 0.0525}$	$0.7841^{\pm 0.0170}$	$0.8303^{\pm 0.0346}$	$0.8011^{\pm 0.0606}$	$0.8028^{\pm 0.0015}$	$0.7600^{\pm 0.0358}$	$0.7743^{\pm 0.0226}$	$0.8303^{\pm 0.0111}$	$0.7847^{\pm 0.0166}$	$0.7977^{\pm 0.0512}$	$0.7738^{\pm 0.0072}$
5×10^{-5}	$0.8101^{\pm 0.0105}$	$0.7783^{\pm 0.0233}$	$0.7970^{\pm 0.0311}$	$0.8237^{\pm 0.0238}$	$0.8331^{\pm 0.0377}$	$0.7835^{\pm 0.0162}$	$0.7558^{\pm 0.0366}$	$0.8068^{\pm 0.0126}$	$0.7932^{\pm 0.0526}$	$0.7831^{\pm 0.0308}$	$0.7982^{\pm 0.0385}$
3×10^{-5}	$0.8117^{\pm 0.0263}$	$0.7992^{\pm 0.0453}$	$0.8197^{\pm 0.0457}$	$0.7834^{\pm 0.0271}$	$0.7962^{\pm 0.0431}$	$0.8351^{\pm 0.0116}$	$0.7937^{\pm 0.0126}$	$0.7483^{\pm 0.0244}$	$0.7942^{\pm 0.0143}$	$0.8096^{\pm 0.0368}$	$0.8043^{\pm 0.0252}$
1×10^{-5}	$0.7595^{\pm 0.0237}$	$0.8169^{\pm 0.0375}$	$0.7732^{\pm 0.0479}$	$0.8115^{\pm 0.0069}$	$0.8231^{\pm 0.0475}$	$0.8176^{\pm 0.0199}$	$0.8142^{\pm 0.0101}$	$0.8302^{\pm 0.0278}$	$0.8018^{\pm 0.0126}$	$0.8152^{\pm 0.0308}$	$0.8371^{\pm 0.0359}$
8×10^{-6}	$0.7845^{\pm 0.0314}$	$0.8265^{\pm 0.0131}$	$0.8054^{\pm 0.0373}$	$0.8196^{\pm 0.0185}$	$0.8357^{\pm 0.0154}$	$0.8271^{\pm 0.0202}$	$0.8130^{\pm 0.0471}$	$0.8007^{\pm 0.0579}$	$0.8058^{\pm 0.0192}$	$0.8395^{\pm 0.0284}$	$0.8022^{\pm 0.0566}$
5×10^{-6}	$0.8083^{\pm 0.0300}$	$0.8336^{\pm 0.0266}$	$0.8146^{\pm 0.0384}$	$0.8508^{\pm 0.0368}$	$0.7954^{\pm 0.0439}$	$0.8371^{\pm 0.0248}$	$0.8227^{\pm 0.0695}$	$0.8602^{\pm 0.0203}$	$0.8230^{\pm 0.0152}$	$0.8052^{\pm 0.0382}$	$0.8197^{\pm 0.0247}$
3×10^{-6}	$0.7842^{\pm 0.0003}$	$0.7975^{\pm 0.0232}$	$0.8045^{\pm 0.0418}$	$0.7585^{\pm 0.0186}$	$0.8474^{\pm 0.0376}$	$0.8087^{\pm 0.0283}$	$0.8302^{\pm 0.0095}$	$0.8099^{\pm 0.0333}$	$0.7705^{\pm 0.0159}$	$0.7857^{\pm 0.0354}$	$0.8152^{\pm 0.0362}$
1×10^{-6}	$0.8476^{\pm 0.0189}$	$0.8257^{\pm 0.0266}$	$0.7709^{\pm 0.0330}$	$0.8412^{\pm 0.0244}$	$0.7549^{\pm 0.0221}$	$0.7657^{\pm 0.0264}$	$0.8248^{\pm 0.0183}$	$0.8425^{\pm 0.0392}$	$0.8022^{\pm 0.0302}$	$0.8207^{\pm 0.0338}$	$0.7887^{\pm 0.0366}$

Table 13: Numerical results for Gradient Flow with the Swiss dataset ($d = 100$)

LR	SW	MaxSW	DSW	MaxKSW	iMSW	viMSW	oMSW	rMSW	EBSW	RPSW	EBRPSW
100	294.00 ± 0.38	287.00 ± 4.65	227.00 ± 4.40	275.00 ± 7.53	286.00 ± 7.55	286.00 ± 7.36	294.00 ± 0.38	201.00 ± 3.34	291.00 ± 0.71	291.00 ± 0.79	291.00 ± 0.52
80	294.00 ± 0.38	286.00 ± 4.00	227.00 ± 4.49	275.00 ± 7.71	286.00 ± 7.55	286.00 ± 7.36	294.00 ± 0.38	168.00 ± 4.27	289.00 ± 0.56	289.00 ± 0.54	290.00 ± 0.47
50	294.00 ± 0.38	280.00 ± 4.74	228.00 ± 5.45	276.00 ± 0.85	286.00 ± 7.55	286.00 ± 7.36	294.00 ± 0.38	147.00 ± 0.45	282.00 ± 0.86	282.00 ± 1.65	284.00 ± 0.81
30	294.00 ± 0.38	263.00 ± 6.38	232.00 ± 11.00	258.00 ± 1.30	286.00 ± 7.50	286.00 ± 7.36	294.00 ± 0.38	162.00 ± 7.25	232.00 ± 35.60	272.00 ± 8.80	268.00 ± 0.79
10	294.00 ± 0.38	187.00 ± 15.70	179.00 ± 1.86	173.00 ± 1.69	286.00 ± 7.50	286.00 ± 7.36	145.00 ± 1.13	95.90 ± 0.61	161.00 ± 3.78	191.00 ± 1.47	163.00 ± 1.10
8	294.00 ± 0.38	166.00 ± 17.30	168.00 ± 3.83	151.00 ± 3.05	286.00 ± 7.50	286.00 ± 7.36	122.00 ± 0.20	89.70 ± 0.32	152.00 ± 0.88	169.00 ± 4.85	181.00 ± 1.50
5	165.00 ± 5.93	123.00 ± 17.90	134.00 ± 10.20	101.00 ± 4.33	286.00 ± 7.50	286.00 ± 7.36	53.40 ± 0.29	50.90 ± 0.31	143.00 ± 5.02	123.00 ± 4.24	133.00 ± 5.45
3	48.40 ± 0.01	85.60 ± 15.60	88.40 ± 8.86	71.40 ± 1.34	293.90 ± 0.40	293.90 ± 0.40	29.60 ± 0.01	34.60 ± 1.36	85.50 ± 3.18	43.30 ± 2.77	80.20 ± 6.35
1	0.03 ± 0.02	28.60 ± 4.04	27.70 ± 3.07	22.40 ± 2.20	293.90 ± 0.40	293.90 ± 0.40	11.60 ± 0.01	10.10 ± 1.78	29.20 ± 1.93	23.90 ± 8.57	30.00 ± 2.60
8×10^{-1}	0.04 ± 0.04	23.70 ± 7.05	25.50 ± 6.84	16.60 ± 0.06	293.90 ± 0.40	293.90 ± 0.40	8.59 ± 0.43	7.33 ± 0.25	27.50 ± 0.30	23.70 ± 4.18	25.90 ± 0.43
5×10^{-1}	0.04 ± 0.04	13.90 ± 3.30	11.50 ± 4.65	11.30 ± 0.70	293.90 ± 0.40	293.90 ± 0.40	4.99 ± 0.01	3.86 ± 0.16	14.00 ± 0.19	7.90 ± 2.08	15.40 ± 0.36
3×10^{-1}	0.09 ± 0.07	10.30 ± 3.68	9.24 ± 0.24	6.73 ± 0.14	293.90 ± 0.40	293.90 ± 0.40	2.97 ± 0.00	2.50 ± 0.55	10.20 ± 0.38	5.19 ± 2.25	11.20 ± 0.88
1×10^{-1}	0.03 ± 0.01	5.55 ± 1.22	2.98 ± 0.39	4.33 ± 0.01	293.90 ± 0.40	293.90 ± 0.40	0.97 ± 0.00	0.89 ± 0.01	4.99 ± 0.22	1.69 ± 0.49	5.55 ± 0.00
8×10^{-2}	0.04 ± 0.00	5.30 ± 1.26	2.90 ± 0.09	3.86 ± 0.18	293.90 ± 0.40	293.90 ± 0.40	0.78 ± 0.00	0.60 ± 0.02	4.17 ± 0.00	1.32 ± 0.41	4.89 ± 0.01
5×10^{-2}	0.03 ± 0.01	4.85 ± 1.70	1.91 ± 0.04	3.13 ± 0.01	210.00 ± 7.20	184.00 ± 11.00	0.49 ± 0.00	0.55 ± 0.17	3.81 ± 0.03	0.87 ± 0.28	3.85 ± 0.03
3×10^{-2}	0.06 ± 0.04	3.68 ± 1.01	1.18 ± 0.06	2.74 ± 0.08	5.59 ± 0.00	5.07 ± 0.03	0.29 ± 0.00	0.22 ± 0.00	3.30 ± 0.00	0.56 ± 0.15	3.47 ± 0.06
1×10^{-2}	1.18 ± 0.08	2.48 ± 0.48	0.25 ± 0.07	2.00 ± 0.01	0.35 ± 0.00	0.31 ± 0.01	0.10 ± 0.00	0.07 ± 0.00	2.45 ± 0.04	0.21 ± 0.04	2.55 ± 0.01
8×10^{-3}	1.36 ± 0.10	2.26 ± 0.47	0.17 ± 0.04	1.81 ± 0.02	0.25 ± 0.00	0.23 ± 0.00	0.07 ± 0.00	0.06 ± 0.00	2.34 ± 0.02	0.19 ± 0.03	2.41 ± 0.02
5×10^{-3}	1.73 ± 0.08	1.70 ± 0.42	0.17 ± 0.00	1.30 ± 0.01	0.13 ± 0.03	0.13 ± 0.02	0.05 ± 0.00	0.04 ± 0.00	2.00 ± 0.03	0.13 ± 0.02	2.03 ± 0.02
3×10^{-3}	2.12 ± 0.05	1.14 ± 0.40	0.11 ± 0.01	0.76 ± 0.01	0.09 ± 0.00	0.06 ± 0.00	0.04 ± 0.00	0.03 ± 0.00	1.48 ± 0.02	0.10 ± 0.00	1.50 ± 0.01
1×10^{-3}	3.05 ± 0.03	0.15 ± 0.10	0.03 ± 0.00	0.09 ± 0.05	0.03 ± 0.00	0.03 ± 0.00	0.03 ± 0.00	0.03 ± 0.00	0.03 ± 0.00	0.13 ± 0.02	0.10 ± 0.07
8×10^{-4}	3.34 ± 0.02	0.18 ± 0.14	0.03 ± 0.00	0.04 ± 0.00	0.03 ± 0.00	0.03 ± 0.00	0.06 ± 0.03	0.03 ± 0.00	0.03 ± 0.00	0.13 ± 0.00	0.04 ± 0.01
5×10^{-4}	4.25 ± 0.00	0.09 ± 0.06	0.03 ± 0.00	0.03 ± 0.00	0.03 ± 0.00	0.03 ± 0.00	0.05 ± 0.03	0.13 ± 0.10	0.07 ± 0.01	0.14 ± 0.05	0.03 ± 0.01
3×10^{-4}	6.02 ± 0.04	0.03 ± 0.00	0.05 ± 0.02	0.03 ± 0.00	0.03 ± 0.00	0.03 ± 0.00	0.06 ± 0.03	0.03 ± 0.00	0.05 ± 0.03	0.16 ± 0.01	0.08 ± 0.03
1×10^{-4}	13.91 ± 0.32	6.76 ± 4.76	0.84 ± 0.07	1.19 ± 0.20	0.03 ± 0.00	0.03 ± 0.00	1.38 ± 0.11	1.42 ± 0.08	1.32 ± 0.11	5.20 ± 0.05	1.57 ± 0.02

Table 14: Results for Color Transfer (Set 1).

LR	SW	MaxSW	DSW	MaxKSW	iMSW	viMSW	oMSW	rMSW	EBSW	RPSW	EBRPSW
100	271.54 ± 0.22	255.40 ± 2.41	268.04 ± 0.71	266.08 ± 0.17	250.17 ± 2.50	250.21 ± 2.46	269.68 ± 2.09	184.92 ± 0.00	269.68 ± 2.09	267.88 ± 0.27	267.89 ± 0.27
80	271.54 ± 0.22	267.88 ± 0.27	268.93 ± 0.93	265.72 ± 0.30	250.17 ± 2.50	250.21 ± 2.46	269.68 ± 2.09	183.33 ± 0.08	271.54 ± 0.22	267.84 ± 0.03	267.86 ± 0.03
50	271.54 ± 0.22	267.67 ± 0.25	270.75 ± 0.57	264.57 ± 0.25	250.17 ± 2.50	250.21 ± 2.46	269.68 ± 2.09	181.88 ± 0.37	271.54 ± 0.22	267.48 ± 0.22	267.66 ± 0.25
30	206.09 ± 0.32	266.45 ± 0.25	271.54 ± 0.22	263.23 ± 0.32	250.21 ± 2.46	250.21 ± 2.46	269.33 ± 1.63	150.88 ± 0.68	266.07 ± 0.27	265.47 ± 0.10	266.50 ± 0.30
10	267.89 ± 0.27	231.34 ± 0.18	237.76 ± 8.30	207.42 ± 14.88	250.45 ± 2.28	250.45 ± 2.28	145.92 ± 0.14	110.15 ± 0.05	258.87 ± 0.32	209.55 ± 8.08	188.90 ± 1.52
8	267.89 ± 0.27	202.37 ± 0.25	208.99 ± 0.19	170.61 ± 3.62	267.89 ± 0.27	267.89 ± 0.27	122.14 ± 0.03	70.60 ± 0.03	241.47 ± 0.33	197.62 ± 2.41	189.23 ± 7.49
5	267.89 ± 0.27	155.92 ± 7.23	160.85 ± 2.04	104.98 ± 7.47	250.96 ± 2.44	250.78 ± 2.66	63.83 ± 0.05	52.42 ± 0.06	145.10 ± 23.12	109.62 ± 6.34	138.06 ± 4.07
3	67.92 ± 3.09	81.71 ± 3.84	82.30 ± 16.46	62.80 ± 2.47	249.98 ± 2.76	249.98 ± 2.76	29.07 ± 0.02	26.25 ± 0.45	93.20 ± 1.46	66.69 ± 3.35	89.42 ± 2.56
1	4.82 ± 0.05	39.43 ± 1.40	17.67 ± 9.12	21.70 ± 0.29	259.34 ± 0.43	259.94 ± 0.03	9.43 ± 0.19	10.18 ± 1.32	30.18 ± 1.27	22.54 ± 3.17	34.46 ± 2.11
8×10^{-1}	0.01 ± 0.00	31.62 ± 2.06	17.77 ± 4.39	18.39 ± 0.09	267.62 ± 0.01	267.62 ± 0.01	9.74 ± 0.08	7.77 ± 1.12	26.50 ± 1.50	18.69 ± 0.09	25.70 ± 1.05
5×10^{-1}	0.01 ± 0.00	19.61 ± 0.76	11.12 ± 0.13	15.37 ± 1.67	267.89 ± 0.27	267.89 ± 0.27	5.38 ± 0.33	4.35 ± 0.32	17.35 ± 0.83	12.48 ± 0.86	17.65 ± 0.39
3×10^{-1}	0.03 ± 0.01	12.00 ± 0.85	1.31 ± 0.01	8.26 ± 0.09	267.89 ± 0.27	267.89 ± 0.27	3.31 ± 0.00	2.58 ± 0.29	10.92 ± 0.30	7.50 ± 0.91	11.34 ± 0.26
1×10^{-1}	0.03 ± 0.00	6.39 ± 0.52	3.40 ± 0.21	5.72 ± 0.00	267.89 ± 0.27	267.89 ± 0.27	1.06 ± 0.01	0.81 ± 0.04	6.36 ± 0.04	2.54 ± 0.29	7.46 ± 0.09
8×10^{-2}	0.03 ± 0.00	6.62 ± 0.65	2.65 ± 0.13	3.82 ± 0.02	267.89 ± 0.27	267.89 ± 0.27	0.95 ± 0.01	0.70 ± 0.01	5.10 ± 0.47	1.97 ± 0.23	6.87 ± 0.09
5×10^{-2}	0.33 ± 0.27	5.60 ± 0.38	2.12 ± 0.10	3.28 ± 0.08	250.80 ± 2.77	251.21 ± 2.88	0.59 ± 0.05	0.35 ± 0.00	5.81 ± 0.00	1.28 ± 0.15	6.03 ± 0.05
3×10^{-2}	0.63 ± 0.57	4.60 ± 0.46	1.18 ± 0.00	2.74 ± 0.00	3.36 ± 1.76	3.19 ± 1.57	0.30 ± 0.00	0.22 ± 0.01	5.41 ± 1.05	0.76 ± 0.10	5.51 ± 1.02
1×10^{-2}	2.20 ± 0.51	2.71 ± 0.12	0.25 ± 0.00	2.00 ± 0.00	0.34 ± 0.01	0.33 ± 0.01	0.10 ± 0.00	0.08 ± 0.01	4.22 ± 0.04	0.22 ± 0.02	4.25 ± 0.02
8×10^{-3}	2.57 ± 0.61	2.31 ± 0.02	0.15 ± 0.01	1.83 ± 0.01	0.24 ± 0.00	0.24 ± 0.00	0.08 ± 0.00	0.07 ± 0.01	3.96 ± 0.01	0.21 ± 0.02	4.00 ± 0.02
5×10^{-3}	3.00 ± 0.63	1.57 ± 0.06	0.10 ± 0.03	1.32 ± 0.01	0.13 ± 0.00	0.13 ± 0.00	0.05 ± 0.00	0.04 ± 0.00	3.28 ± 0.64	0.17 ± 0.02	3.30 ± 0.64
3×10^{-3}	4.07 ± 0.98	0.92 ± 0.11	0.11 ± 0.00	0.76 ± 0.00	0.09 ± 0.00	0.07 ± 0.01	0.04 ± 0.00	0.03 ± 0.00	1.49 ± 0.01	0.10 ± 0.00	1.50 ± 0.00
1×10^{-3}	8.48 ± 2.72	0.15 ± 0.00	0.04 ± 0.01	0.10 ± 0.06	0.03 ± 0.00	0.03 ± 0.00	0.03 ± 0.00	0.03 ± 0.00	0.03 ± 0.00	0.13 ± 0.00	0.07 ± 0.02
8×10^{-4}	9.10 ± 2.88	0.12 ± 0.03	0.04 ± 0.00	0.04 ± 0.00	0.03 ± 0.00	0.03 ± 0.00	0.06 ± 0.00	0.03 ± 0.00	0.03 ± 0.00	0.16 ± 0.02	0.04 ± 0.00
5×10^{-4}	11.10 ± 3.42	0.06 ± 0.01	0.04 ± 0.01	0.03 ± 0.00	0.03 ± 0.00	0.03 ± 0.00	0.05 ± 0.00	0.07 ± 0.03	0.04 ± 0.02	0.15 ± 0.00	0.03 ± 0.00
3×10^{-4}	13.97 ± 3.97	0.03 ± 0.00	0.04 ± 0.00	0.03 ± 0.00	0.03 ± 0.00	0.03 ± 0.00	0.05 ± 0.01	0.03 ± 0.00	0.04 ± 0.01	0.15 ± 0.00	0.06 ± 0.01
1×10^{-4}	11.85 ± 0.63	4.04 ± 1.36	0.86 ± 0.01	1.25 ± 0.03	0.03 ± 0.00	0.03 ± 0.00	1.15 ± 0.12	1.16 ± 0.13	1.32 ± 0.00	4.59 ± 0.31	1.49 ± 0.04

Table 15: Results for Color Transfer (Set 2).

LR	SW	MaxSW	DSW	MaxKSW	iMSW	viMSW	oMSW	rMSW	EBSW	RPSW	EBRPSW
100	320.55 ± 0.09	320.55 ± 0.09	279.16 ± 0.05	308.63 ± 0.18	320.55 ± 0.09	320.55 ± 0.09	320.55 ± 0.09	157.93 ± 0.04	320.55 ± 0.09	320.55 ± 0.09	320.55 ± 0.09
80	320.55 ± 0.09	320.55 ± 0.09	279.16 ± 0.05	307.70 ± 0.20	320.55 ± 0.09	320.55 ± 0.09	320.55 ± 0.09	168.12 ± 1.25	320.55 ± 0.09	320.55 ± 0.09	320.55 ± 0.09
50	320.55 ± 0.09	320.55 ± 0.09	279.16 ± 0.05	304.23 ± 0.21	320.55 ± 0.09	320.55 ± 0.09	320.55 ± 0.09	124.61 ± 0.79	320.55 ± 0.09	320.55 ± 0.09	320.55 ± 0.09
30	320.55 ± 0.09	320.55 ± 0.09	279.16 ± 0.05	298.63 ± 0.38	320.55 ± 0.09	320.55 ± 0.09	320.45 ± 0.11	123.85 ± 10.63	149.82 ± 0.07	320.52 ± 0.06	320.57 ± 0.05
10	320.55 ± 0.09	286.84 ± 0.29	279.14 ± 0.03	202.30 ± 1.04	320.17 ± 0.03	320.16 ± 0.03	183.41 ± 0.22	62.75 ± 0.48	157.47 ± 28.22	266.66 ± 2.21	205.07 ± 78.24
8	320.52 ± 0.07	251.89 ± 0.26	210.92 ± 6.75	165.78 ± 0.11	311.04 ± 0.20	311.02 ± 0.19	143.62 ± 0.14	91.55 ± 0.01	171.06 ± 18.71	222.87 ± 15.73	165.19 ± 37.58
5	254.50 ± 0.29	175.13 ± 4.67	161.67 ± 0.60	95.66 ± 15.39	219.26 ± 19.68	232.89 ± 0.29	72.48 ± 0.04	30.00 ± 1.59	123.75 ± 58.87	111.06 ± 3.36	124.90 ± 7.31
3	59.33 ± 0.07	108.52 ± 1.52	107.15 ± 8.57	57.36 ± 3.19	320.55 ± 0.09	320.55 ± 0.09	34.35 ± 0.22	29.95 ± 0.29	87.52 ± 10.23	66.06 ± 1.43	107.12 ± 1.10
1	0.00 ± 0.00	35.58 ± 1.42	39.61 ± 2.14	21.60 ± 0.55	320.55 ± 0.09	320.55 ± 0.09	9.70 ± 0.01	10.02 ± 0.49	34.00 ± 0.45	19.71 ± 3.98	32.68 ± 4.24
8×10^{-1}	0.00 ± 0.00	28.51 ± 1.06	17.42 ± 0.97	17.33 ± 0.41	320.55 ± 0.09	320.55 ± 0.09	7.68 ± 0.01	7.67 ± 0.39	27.35 ± 1.00	18.13 ± 2.59	27.86 ± 1.00
5×10^{-1}	0.00 ± 0.00	19.69 ± 1.77	12.75 ± 4.32	11.51 ± 0.11	320.55 ± 0.09	320.55 ± 0.09	4.80 ± 0.01	5.08 ± 0.89	16.78 ± 0.14	10.86 ± 0.59	18.69 ± 1.52
3×10^{-1}	0.20 ± 0.25	12.93 ± 1.49	7.88 ± 1.00	8.03 ± 0.05	320.55 ± 0.09	320.55 ± 0.09	3.38 ± 0.01	2.75 ± 0.22	11.10 ± 0.32	6.10 ± 0.27	11.51 ± 0.58
1×10^{-1}	0.03 ± 0.00	6.24 ± 0.71	3.97 ± 0.02	5.10 ± 0.24	320.15 ± 0.03	320.16 ± 0.03	0.95 ± 0.00	0.79 ± 0.01	6.17 ± 0.16	1.94 ± 0.23	6.96 ± 0.52
8×10^{-2}	0.20 ± 0.24	5.93 ± 0.44	2.75 ± 0.40	4.72 ± 0.06	311.04 ± 0.20	311.02 ± 0.19	0.77 ± 0.00	0.73 ± 0.01	5.51 ± 0.21	1.59 ± 0.29	6.35 ± 0.41
5×10^{-2}	0.17 ± 0.24	4.33 ± 0.74	2.02 ± 0.07	4.47 ± 0.08	219.26 ± 19.68	232.89 ± 0.29	0.48 ± 0.00	0.49 ± 0.06	5.03 ± 0.16	0.97 ± 0.14	5.30 ± 0.10
3×10^{-2}	0.95 ± 0.01	3.68 ± 0.00	1.17 ± 0.01	4.06 ± 0.08	5.99 ± 0.56	5.49 ± 0.59	0.29 ± 0.00	0.24 ± 0.02	4.56 ± 0.06	0.58 ± 0.05	4.78 ± 0.08
1×10^{-2}	2.16 ± 0.04	2.31 ± 0.12	0.28 ± 0.05	2.93 ± 0.01	0.34 ± 0.01	0.32 ± 0.01	0.10 ± 0.00	0.07 ± 0.00	3.51 ± 0.05	0.28 ± 0.08	3.59 ± 0.01
8×10^{-3}	2.37 ± 0.07	2.09 $± 0.12	0.22 ± 0.07	2.64 ± 0.01	0.24 ± 0.01	0.23 ± 0.00	0.09 ± 0.02	0.07 ± 0.01	3.32 ± 0.02	0.23 ± 0.06	3.42 ± 0.01
5×10^{-3}	2.99 ± 0.02	1.61 ± 0.06	0.18 ± 0.01	1.51 ± 0.01	0.13 ± 0.00	0.13 ± 0.00	0.05 ± 0.00	0.04 ± 0.00	2.41 ± 0.02	0.16 ± 0.05	2.44 ± 0.01
3×10^{-3}	3.24 ± 0.03	1.15 ± 0.01	0.12 ± 0.01	0.93 ± 0.12	0.09 ± 0.00	0.06 ± 0.00	0.04 ± 0.00	0.04 ± 0.01	1.76 ± 0.01	0.10 ± 0.00	1.79 ± 0.01
1×10^{-3}	3.48 ± 0.17	0.15 ± 0.00	0.05 ± 0.01	0.66 ± 0.37	0.03 ± 0.00	0.10 ± 0.09	0.03 ± 0.00	0.03 ± 0.00	0.03 ± 0.00	0.15 ± 0.01	0.11 ± 0.11
8×10^{-4}	3.84 ± 0.35	0.12 ± 0.05	0.03 ± 0.00	0.04 ± 0.00	0.08 ± 0.08	0.03 ± 0.00	0.03 ± 0.00	0.03 ± 0.00	0.05 ± 0.02	0.09 ± 0.05	0.05 ± 0.03
5×10^{-4}	4.93 ± 0.48	0.06 ± 0.02	0.03 ± 0.00	0.03 ± 0.00	0.03 ± 0.00	0.03 ± 0.00	0.05 ± 0.00	0.08 ± 0.07	0.03 ± 0.00	0.11 ± 0.02	0.03 ± 0.00
3×10^{-4}	6.37 ± 0.25	0.03 ± 0.00	0.08 ± 0.02	0.03 ± 0.00	0.03 ± 0.00	0.03 ± 0.00	0.05 ± 0.01	0.03 ± 0.00	0.07 ± 0.02	0.15 ± 0.01	0.05 ± 0.02
1×10^{-4}	13.61 ± 0.21	5.55 ± 0.86	0.89 ± 0.03	6.88 ± 0.15	0.70 ± 0.09	0.74 ± 0.14	11.42 ± 0.11	12.17 ± 0.14	3.64 ± 0.15	7.50 ± 0.16	3.68 ± 0.01

Table 16: Results for Color Transfer (Set 3).

LR	SW	MaxSW	DSW	MaxKSW	iMSW	viMSW	oMSW	rMSW	EBSW	RPSW	EBRPSW
1	$14.17_{\pm 0.02}$	-	-	-	-	-	-	-	-	-	-
8×10^{-1}	$14.16_{\pm 0.01}$	-	-	-	-	-	-	-	-	-	-
5×10^{-1}	$14.15_{\pm 0.02}$	-	-	-	-	-	-	-	-	-	-
3×10^{-1}	$14.16_{\pm 0.00}$	-	-	-	-	-	-	-	-	$17.78_{\pm 0.18}$	-
1×10^{-1}	$14.22_{\pm 0.01}$	-	-	$14.50_{\pm 0.03}$	-	-	$14.13_{\pm 0.02}$	$14.17_{\pm 0.02}$	-	$14.14_{\pm 0.00}$	-
8×10^{-2}	$14.26_{\pm 0.01}$	-	$14.78_{\pm 0.02}$	$14.29_{\pm 0.01}$	-	-	$14.12_{\pm 0.01}$	$14.16_{\pm 0.01}$	-	$14.15_{\pm 0.02}$	-
5×10^{-2}	$14.33_{\pm 0.01}$	$14.54_{\pm 0.02}$	$14.46_{\pm 0.03}$	$14.20_{\pm 0.01}$	-	-	$14.19_{\pm 0.00}$	$14.18_{\pm 0.02}$	-	$14.18_{\pm 0.02}$	-
3×10^{-2}	$14.37_{\pm 0.01}$	$14.26_{\pm 0.02}$	$14.25_{\pm 0.02}$	$14.27_{\pm 0.01}$	-	-	$14.20_{\pm 0.01}$	$14.19_{\pm 0.02}$	-	$14.18_{\pm 0.01}$	-
1×10^{-2}	$14.50_{\pm 0.01}$	$14.06_{\pm 0.02}$	$14.12_{\pm 0.02}$	$14.29_{\pm 0.02}$	-	-	$14.17_{\pm 0.01}$	$14.16_{\pm 0.01}$	$14.71_{\pm 0.02}$	$14.15_{\pm 0.02}$	$14.69_{\pm 0.02}$
8×10^{-3}	$14.54_{\pm 0.01}$	$14.02_{\pm 0.02}$	$14.11_{\pm 0.02}$	$14.27_{\pm 0.01}$	-	-	$14.16_{\pm 0.02}$	$14.16_{\pm 0.01}$	$14.72_{\pm 0.04}$	$14.16_{\pm 0.02}$	$14.71_{\pm 0.02}$
5×10^{-3}	$14.67_{\pm 0.00}$	$14.01_{\pm 0.02}$	$14.11_{\pm 0.02}$	$14.26_{\pm 0.03}$	-	-	$14.16_{\pm 0.01}$	$14.18_{\pm 0.01}$	$14.72_{\pm 0.02}$	$14.17_{\pm 0.02}$	$14.70_{\pm 0.04}$
3×10^{-3}	$14.84_{\pm 0.01}$	$14.01_{\pm 0.02}$	$14.16_{\pm 0.02}$	$14.27_{\pm 0.02}$	$14.06_{\pm 0.01}$	$14.09_{\pm 0.01}$	$14.19_{\pm 0.01}$	$14.19_{\pm 0.01}$	$14.77_{\pm 0.03}$	$14.20_{\pm 0.01}$	$14.77_{\pm 0.05}$
1×10^{-3}	$15.36_{\pm 0.01}$	$14.04_{\pm 0.02}$	$14.17_{\pm 0.02}$	$14.42_{\pm 0.01}$	$14.12_{\pm 0.01}$	$14.26_{\pm 0.00}$	$14.34_{\pm 0.01}$	$14.37_{\pm 0.01}$	$14.72_{\pm 0.02}$	$14.36_{\pm 0.00}$	$14.73_{\pm 0.05}$
8×10^{-4}	$15.47_{\pm 0.02}$	$14.08_{\pm 0.01}$	$14.17_{\pm 0.02}$	$14.47_{\pm 0.01}$	$14.14_{\pm 0.02}$	$14.31_{\pm 0.01}$	$14.35_{\pm 0.02}$	$14.38_{\pm 0.01}$	$14.74_{\pm 0.02}$	$14.38_{\pm 0.01}$	$14.76_{\pm 0.02}$
5×10^{-4}	$15.73_{\pm 0.02}$	$14.12_{\pm 0.01}$	$14.22_{\pm 0.02}$	$14.48_{\pm 0.01}$	$14.26_{\pm 0.01}$	$14.34_{\pm 0.01}$	$14.39_{\pm 0.01}$	$14.40_{\pm 0.01}$	$14.85_{\pm 0.03}$	$14.41_{\pm 0.01}$	$14.87_{\pm 0.03}$
3×10^{-4}	$16.59_{\pm 0.04}$	$14.75_{\pm 0.02}$	$14.91_{\pm 0.04}$	$14.94_{\pm 0.02}$	$14.81_{\pm 0.02}$	$14.81_{\pm 0.02}$	$14.88_{\pm 0.03}$	$14.88_{\pm 0.01}$	$14.97_{\pm 0.02}$	$14.48_{\pm 0.01}$	$14.99_{\pm 0.04}$
1×10^{-4}	$17.36_{\pm 0.04}$	$14.41_{\pm 0.02}$	$14.52_{\pm 0.01}$	$14.70_{\pm 0.01}$	$14.40_{\pm 0.01}$	$14.50_{\pm 0.01}$	$14.70_{\pm 0.01}$	$14.69_{\pm 0.00}$	$15.24_{\pm 0.01}$	$14.73_{\pm 0.02}$	$15.23_{\pm 0.02}$
8×10^{-5}	$17.71_{\pm 0.11}$	$14.41_{\pm 0.01}$	$14.53_{\pm 0.01}$	$14.79_{\pm 0.01}$	$14.42_{\pm 0.01}$	$14.53_{\pm 0.01}$	$14.77_{\pm 0.01}$	$14.77_{\pm 0.01}$	$15.15_{\pm 0.02}$	$14.81_{\pm 0.01}$	$15.15_{\pm 0.02}$
5×10^{-5}	$18.11_{\pm 0.04}$	$14.45_{\pm 0.01}$	$14.55_{\pm 0.01}$	$14.97_{\pm 0.02}$	$14.47_{\pm 0.01}$	$14.61_{\pm 0.01}$	$14.97_{\pm 0.01}$	$14.96_{\pm 0.01}$	$15.00_{\pm 0.01}$	$14.99_{\pm 0.01}$	$15.01_{\pm 0.01}$
3×10^{-5}	$18.49_{\pm 0.07}$	$14.50_{\pm 0.01}$	$14.61_{\pm 0.01}$	$15.22_{\pm 0.01}$	$14.55_{\pm 0.01}$	$14.72_{\pm 0.01}$	$15.20_{\pm 0.00}$	$15.23_{\pm 0.01}$	$14.96_{\pm 0.01}$	$15.57_{\pm 0.01}$	$14.95_{\pm 0.02}$
1×10^{-5}	$18.84_{\pm 0.06}$	$14.66_{\pm 0.01}$	$14.75_{\pm 0.01}$	$15.80_{\pm 0.00}$	$14.77_{\pm 0.01}$	$15.09_{\pm 0.01}$	$15.81_{\pm 0.02}$	$15.81_{\pm 0.03}$	$15.39_{\pm 0.01}$	$15.81_{\pm 0.02}$	$15.39_{\pm 0.02}$
8×10^{-6}	$18.89_{\pm 0.04}$	$14.70_{\pm 0.01}$	$14.80_{\pm 0.01}$	$15.94_{\pm 0.02}$	$14.84_{\pm 0.01}$	$15.20_{\pm 0.01}$	$15.95_{\pm 0.02}$	$15.95_{\pm 0.02}$	$15.50_{\pm 0.02}$	$15.93_{\pm 0.02}$	$15.50_{\pm 0.01}$
5×10^{-6}	$19.08_{\pm 0.05}$	$14.84_{\pm 0.01}$	$14.95_{\pm 0.02}$	$16.34_{\pm 0.02}$	$14.99_{\pm 0.01}$	$15.44_{\pm 0.01}$	$16.35_{\pm 0.01}$	$16.34_{\pm 0.04}$	$15.81_{\pm 0.02}$	$16.30_{\pm 0.02}$	$15.79_{\pm 0.01}$
3×10^{-6}	$19.10_{\pm 0.08}$	$15.03_{\pm 0.01}$	$15.20_{\pm 0.02}$	$16.93_{\pm 0.05}$	$15.26_{\pm 0.02}$	$15.77_{\pm 0.01}$	$16.95_{\pm 0.03}$	$16.94_{\pm 0.04}$	$16.25_{\pm 0.02}$	$16.81_{\pm 0.03}$	$16.23_{\pm 0.05}$
1×10^{-6}	$19.05_{\pm 0.06}$	$15.76_{\pm 0.01}$	$16.03_{\pm 0.01}$	$18.18_{\pm 0.07}$	$15.96_{\pm 0.04}$	$16.49_{\pm 0.03}$	$18.18_{\pm 0.07}$	$18.15_{\pm 0.04}$	$17.30_{\pm 0.03}$	$18.10_{\pm 0.06}$	$17.30_{\pm 0.05}$

Table 17: Numerical results for the M2F task.

LR	SW	MaxSW	DSW	MaxKSW	iMSW	viMSW	oMSW	rMSW	EBSW	RPSW	EBRPSW
1	$14.58_{\pm 0.03}$	-	-	-	-	-	-	-	-	-	-
8×10^{-1}	$14.61_{\pm 0.03}$	-	-	-	-	-	-	-	-	-	-
5×10^{-1}	$14.62_{\pm 0.03}$	-	-	-	-	-	-	-	-	-	-
3×10^{-1}	$14.62_{\pm 0.02}$	-	-	-	-	-	-	-	-	$14.68_{\pm 0.13}$	-
1×10^{-1}	$14.70_{\pm 0.02}$	-	-	$14.98_{\pm 0.02}$	-	-	$14.62_{\pm 0.02}$	$14.63_{\pm 0.01}$	-	$14.65_{\pm 0.03}$	-
8×10^{-2}	$14.73_{\pm 0.02}$	$16.48_{\pm 0.00}$	$15.35_{\pm 0.07}$	$14.81_{\pm 0.02}$	-	-	$14.62_{\pm 0.02}$	$14.63_{\pm 0.01}$	-	$14.64_{\pm 0.01}$	-
5×10^{-2}	$14.80_{\pm 0.02}$	$15.17_{\pm 0.08}$	$14.95_{\pm 0.02}$	$14.65_{\pm 0.02}$	-	-	$14.63_{\pm 0.01}$	$14.67_{\pm 0.03}$	-	$14.65_{\pm 0.03}$	-
3×10^{-2}	$14.80_{\pm 0.03}$	$14.63_{\pm 0.04}$	$15.44_{\pm 0.00}$	$15.71_{\pm 0.00}$	-	-	$14.65_{\pm 0.02}$	$14.62_{\pm 0.02}$	-	$14.62_{\pm 0.02}$	-
1×10^{-2}	$14.95_{\pm 0.02}$	$14.60_{\pm 0.02}$	$14.63_{\pm 0.04}$	$14.69_{\pm 0.03}$	-	-	$14.58_{\pm 0.03}$	$14.62_{\pm 0.02}$	$15.18_{\pm 0.04}$	$14.60_{\pm 0.01}$	$15.16_{\pm 0.03}$
8×10^{-3}	$15.01_{\pm 0.03}$	$14.61_{\pm 0.02}$	$14.62_{\pm 0.03}$	$14.70_{\pm 0.02}$	-	-	$14.61_{\pm 0.01}$	$14.60_{\pm 0.02}$	$15.15_{\pm 0.03}$	$14.60_{\pm 0.02}$	$15.15_{\pm 0.05}$
5×10^{-3}	$15.09_{\pm 0.03}$	$14.53_{\pm 0.03}$	$14.60_{\pm 0.04}$	$14.69_{\pm 0.05}$	-	-	$14.61_{\pm 0.03}$	$14.63_{\pm 0.02}$	$15.11_{\pm 0.04}$	$14.64_{\pm 0.03}$	$15.16_{\pm 0.04}$
3×10^{-3}	$15.28_{\pm 0.01}$	$14.52_{\pm 0.02}$	$14.61_{\pm 0.02}$	$14.70_{\pm 0.03}$	$14.62_{\pm 0.04}$	$14.57_{\pm 0.01}$	$14.65_{\pm 0.04}$	$14.65_{\pm 0.04}$	$15.16_{\pm 0.01}$	$14.67_{\pm 0.02}$	$15.15_{\pm 0.05}$
1×10^{-3}	$15.78_{\pm 0.01}$	$14.57_{\pm 0.03}$	$14.65_{\pm 0.01}$	$14.89_{\pm 0.01}$	$14.59_{\pm 0.01}$	$14.72_{\pm 0.03}$	$14.80_{\pm 0.02}$	$14.83_{\pm 0.03}$	$15.13_{\pm 0.01}$	$14.78_{\pm 0.02}$	$15.10_{\pm 0.05}$
8×10^{-4}	$15.88_{\pm 0.03}$	$14.59_{\pm 0.03}$	$14.68_{\pm 0.02}$	$14.92_{\pm 0.02}$	$14.64_{\pm 0.02}$	$14.76_{\pm 0.01}$	$14.78_{\pm 0.02}$	$14.81_{\pm 0.02}$	$15.09_{\pm 0.04}$	$14.81_{\pm 0.02}$	$15.12_{\pm 0.02}$
5×10^{-4}	$16.17_{\pm 0.03}$	$14.62_{\pm 0.02}$	$14.74_{\pm 0.03}$	$14.92_{\pm 0.02}$	$14.75_{\pm 0.01}$	$14.77_{\pm 0.03}$	$14.82_{\pm 0.02}$	$14.85_{\pm 0.03}$	$15.26_{\pm 0.06}$	$14.85_{\pm 0.02}$	$15.25_{\pm 0.02}$
3×10^{-4}	$16.59_{\pm 0.04}$	$14.75_{\pm 0.02}$	$14.91_{\pm 0.04}$	$14.94_{\pm 0.02}$	$14.81_{\pm 0.02}$	$14.81_{\pm 0.02}$	$14.88_{\pm 0.03}$	$14.88_{\pm 0.01}$	$15.50_{\pm 0.04}$	$14.94_{\pm 0.03}$	$15.49_{\pm 0.03}$
1×10^{-4}	$18.05_{\pm 0.05}$	$14.89_{\pm 0.04}$	$14.96_{\pm 0.02}$	$15.16_{\pm 0.02}$	$14.81_{\pm 0.02}$	$14.97_{\pm 0.02}$	$15.14_{\pm 0.04}$	$15.13_{\pm 0.02}$	$15.63_{\pm 0.04}$	$15.17_{\pm 0.02}$	$15.66_{\pm 0.02}$
8×10^{-5}	$18.26_{\pm 0.09}$	$14.88_{\pm 0.03}$	$14.96_{\pm 0.03}$	$15.20_{\pm 0.03}$	$14.87_{\pm 0.01}$	$14.96_{\pm 0.03}$	$15.19_{\pm 0.03}$	$15.19_{\pm 0.02}$	$15.56_{\pm 0.04}$	$15.24_{\pm 0.01}$	$15.54_{\pm 0.03}$
5×10^{-5}	$18.92_{\pm 0.06}$	$14.92_{\pm 0.04}$	$15.03_{\pm 0.04}$	$15.38_{\pm 0.03}$	$14.94_{\pm 0.02}$	$15.05_{\pm 0.04}$	$15.37_{\pm 0.03}$	$15.38_{\pm 0.04}$	$15.43_{\pm 0.01}$	$15.42_{\pm 0.02}$	$15.41_{\pm 0.05}$
3×10^{-5}	$19.21_{\pm 0.05}$	$14.95_{\pm 0.02}$	$15.07_{\pm 0.01}$	$15.61_{\pm 0.03}$	$15.00_{\pm 0.02}$	$15.14_{\pm 0.01}$	$15.60_{\pm 0.01}$	$15.59_{\pm 0.03}$	$15.35_{\pm 0.01}$	$15.64_{\pm 0.01}$	$15.39_{\pm 0.02}$
1×10^{-5}	$19.64_{\pm 0.04}$	$15.12_{\pm 0.01}$	$15.23_{\pm 0.02}$	$16.21_{\pm 0.03}$	$15.22_{\pm 0.04}$	$15.55_{\pm 0.03}$	$16.23_{\pm 0.03}$	$16.22_{\pm 0.03}$	$15.72_{\pm 0.04}$	$16.22_{\pm 0.02}$	$15.70_{\pm 0.04}$
8×10^{-6}	$19.62_{\pm 0.05}$	$15.19_{\pm 0.02}$	$15.27_{\pm 0.02}$	$16.43_{\pm 0.02}$	$15.28_{\pm 0.01}$	$15.61_{\pm 0.01}$	$16.42_{\pm 0.03}$	$16.39_{\pm 0.03}$	$15.91_{\pm 0.05}$	$15.94_{\pm 0.02}$	$15.91_{\pm 0.05}$
5×10^{-6}	$19.79_{\pm 0.06}$	$15.32_{\pm 0.01}$	$15.38_{\pm 0.02}$	$16.86_{\pm 0.02}$	$15.47_{\pm 0.03}$	$15.87_{\pm 0.03}$	$16.85_{\pm 0.02}$	$16.85_{\pm 0.05}$	$16.25_{\pm 0.03}$	$16.30_{\pm 0.02}$	$16.26_{\pm 0.04}$
3×10^{-6}	$19.85_{\pm 0.13}$	$15.56_{\pm 0.02}$	$15.65_{\pm 0.02}$	$17.57_{\pm 0.06}$	$15.70_{\pm 0.02}$	$16.15_{\pm 0.02}$	$17.60_{\pm 0.09}$	$17.53_{\pm 0.07}$	$16.72_{\pm 0.03}$	$16.81_{\pm 0.02}$	$16.74_{\pm 0.04}$
1×10^{-6}	$19.84_{\pm 0.10}$	$16.23_{\pm 0.02}$	$16.46_{\pm 0.02}$	$18.95_{\pm 0.06}$	$16.39_{\pm 0.05}$	$16.93_{\pm 0.05}$	$18.93_{\pm 0.08}$	$18.88_{\pm 0.06}$	$17.92_{\pm 0.07}$	$18.05_{\pm 0.09}$	$18.02_{\pm 0.04}$

Table 18: Numerical results for the A2C translation task.