Heriot-Watt University
Research Gateway

Lipschitz-Regularized Gradient Flows and Generative Particle Algorithms for High-Dimensional Scarce Data

Citation for published version:
Gu, H, Birmpa, P, Pantazis, Y, Rey-Bellet, L & Katsoulakis, MA 2024, 'Lipschitz-Regularized Gradient Flows and Generative Particle Algorithms for High-Dimensional Scarce Data', *SIAM Journal on Mathematics of Data Science*, vol. 6, no. 4, pp. 1205-1235. https://doi.org/10.1137/23m1587841

Digital Object Identifier (DOI):
10.1137/23m1587841

Link:
Link to publication record in Heriot-Watt Research Portal

Document Version:
Peer reviewed version

Published In:
SIAM Journal on Mathematics of Data Science

Publisher Rights Statement:
© 2024 Society for Industrial and Applied Mathematics.

General rights
Copyright for the publications made accessible via Heriot-Watt Research Portal is retained by the author(s) and / or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy
Heriot-Watt University has made every reasonable effort to ensure that the content in Heriot-Watt Research Portal complies with UK legislation. If you believe that the public display of this file breaches copyright please contact open.access@hw.ac.uk providing details, and we will remove access to the work immediately and investigate your claim.

Download date: 17. Oct. 2026

arXiv:2210.17230v3 [stat.ML] 24 Jul 2023

LIPSCHITZ-REGULARIZED GRADIENT FLOWS AND GENERATIVE PARTICLE ALGORITHMS FOR HIGH-DIMENSIONAL SCARCE DATA

Hyemin Gu
Department of Mathematics and Statistics
University of Massachusetts
Amherst, MA 01003, USA
hgu@umass.edu

Panagiota Birmpa
Department of Actuarial Mathematics & Statistics
Heriot-Watt University
Edinburgh, EH14 4AS, U.K.
P.Birmpa@hw.ac.uk

Yannis Pantazis
Institute of Applied & Computational Mathematics
Foundation for Research & Technology - Hellas
Heraklion, Greece
pantazis@iacm.forth.gr

Luc Rey-Bellet
Department of Mathematics and Statistics
University of Massachusetts
Amherst, MA 01003, USA
luc@math.umass.edu

Markos A. Katsoulakis
Department of Mathematics and Statistics
University of Massachusetts
Amherst, MA 01003, USA
markos@umass.edu

ABSTRACT

We build a new class of generative algorithms capable of efficiently learning an arbitrary target distribution from possibly scarce, high-dimensional data and subsequently generate new samples. These generative algorithms are particle-based and are constructed as gradient flows of Lipschitz-regularized Kullback-Leibler or other f-divergences, where data from a source distribution can be stably transported as particles, towards the vicinity of the target distribution. As a highlighted result in data integration, we demonstrate that the proposed algorithms correctly transport gene expression data points with dimension exceeding 54K, while the sample size is typically only in the hundreds.

1 INTRODUCTION AND MAIN RESULTS

We construct new algorithms that are capable of efficiently transporting samples from a source distribution to a target data set. The transportation mechanism is built as the gradient flow (in probability space) for Lipschitz-regularized divergences, Dupuis & Mao (2022); Birrell et al. (2022a;c). Samples are viewed as particles and are transported along the gradient of the discriminator of the divergence towards the target data set. Lipschitz regularized f-divergences interpolate between the Wasserstein metric and f-divergences and provide a flexible family of loss functions to compare non-absolutely continuous probability measures. In machine learning one needs to build algorithms to handle target distributions Q which are singular, either by their intrinsic nature such as probability densities concentrated on low dimensional structures and/or because Q is usually only known through N samples. The Lipschitz regularization also provides numerically stable, mesh free, particle algorithms that can act as a generative model for high-dimensional target distributions. The proposed generative approach is validated on a wide variety of datasets and applications ranging from heavy-tailed distributions and image generation to gene expression data integration, including

problems in very high dimensions and with scarce target data. In this introduction we provide an outline of our main results, background material and related prior work.

Generative modeling In generative modeling, which is a form of unsupervised learning, a data set $(X^{(i)})_{i=1}^N$ from an unknown "target" distribution Q is given and the goal is to construct an approximating model in the form of a distribution $P \approx Q$ which is easy to simulate, with the goal to generate additional, inexpensive, approximate samples from the distribution Q. Succinctly, the goal of generative modeling is to learn the target distribution Q from input data $(X^{(i)})_{i=1}^N$. This is partly in contrast to sampling, where typically Q is known up to normalization. In the last 10 years, generative modeling has been revolutionized by new innovative algorithms taking advantage of neural networks (NNs) and more generally deep learning. On one hand NNs provide enormous flexibility to parametrize functions and probabilities and on the other, lead to efficient optimization algorithms in function spaces. Generative adversarial networks (GANs) Goodfellow et al. (2014); Arjovsky et al. (2017), for example, are able to generate complex distributions and are quickly becoming a standard tool in image analysis, medical data, cosmology, computational chemistry, materials science and so on. Many other algorithms have been proposed since, such as normalizing flows Köhler et al. (2020); Chen et al. (2018b), diffusion models Sohl-Dickstein et al. (2015); Ho et al. (2020), score-based generative flows Song & Ermon (2020); Song et al. (2021), variational autoencoders Kingma & Welling (2013) and energy-based methods LeCun et al. (2006).

Information theory, divergences and optimal transport Divergences such as Kullback-Leibler (KL) and f-divergences, and probability metrics such as Wasserstein, provide a notion of 'distance' between probability distributions, thus allowing for comparison of models with one another and with data. Divergences and metrics are used in many theoretical and practical problems in mathematics, engineering, and the natural sciences, ranging from statistical physics, large deviations theory, uncertainty quantification, partial differential equations (PDE) and statistics to information theory, communication theory, and machine learning. In particular, in the context of GANs, the choice of objective functional (in the form of a probability divergence plus a suitable regularization) plays a central role.

A very flexible family of divergences, the (f, Γ)-divergences, were introduced in Birrell et al. (2022a). These new divergences interpolate between f-divergences (e.g KL, α-divergence, Shannon-Jensen) and Γ-Integral Probability Metrics (IPM) like 1-Wasserstein and MMD distances (where Γ is the 1-Lipschitz functions or an RKHS 1-ball respectively). Another way to think of Γ is as a regularization to avoid over-fitting, built directly in the divergence, see for instance structure-preserving GANs Birrell et al. (2022c). In this paper, we focus on one specific family which we view as a Lipschitz regularization of the KL-divergence (or f-divergences) or as an entropic regularization of the 1-Wasserstein metric. In this context, the interpolation is mathematically described by the Infimal Convolution formula

$$D_f^{\Gamma_L}(P\|Q) = \inf_{\gamma \in \mathcal{P}(\mathbb{R}^d)} \left\{ L \cdot W^{\Gamma_1}(P, \gamma) + D_f(\gamma\|Q) \right\} , \tag{1}$$

where $\mathcal{P}(\mathbb{R}^d)$ is the space of all Borel probability measures on $\mathbb{R}^d$ and $\Gamma_L = \{\phi : \mathbb{R}^d \to \mathbb{R} : |\phi(x) - \phi(y)| \leq L|x-y| \text{ for all } x, y\}$ is the space of Lipschitz continuous functions with Lipschitz constant bounded by L (note that $L\Gamma_1 = \Gamma_L$). Furthermore, $W^{\Gamma_1}(P, Q)$ denotes the 1-Wasserstein metric with transport cost $|x - y|$ which is an integral probability metric, and has the dual representation

$$W^{\Gamma_1}(P, Q) = \sup_{\phi \in \Gamma_1} \{E_P[\phi] - E_Q[\phi]\} . \tag{2}$$

Finally, if $f : [0, \infty) \to \mathbb{R}$ is strictly convex and lower-semicontinuous with $f(1) = 0$ the f-divergence of P with respect to Q is defined by $D_f(P\|Q) = E_Q[f(\frac{dP}{dQ})]$ if $P \ll Q$ and set to be $+\infty$ otherwise. The new divergences inherit desirable properties from both objects, for example we have

$$0 \leq D_f^{\Gamma_L}(P\|Q) \leq \min \left\{ D_f(P\|Q), L \cdot W^{\Gamma_1}(P, Q) \right\} . \tag{3}$$

The Lipschitz-regularized f-divergences equation 1 admit a dual variational representation,

$$D_f^{\Gamma_L}(P\|Q) := \sup_{\phi \in \Gamma_L} \left\{ E_P[\phi] - \inf_{\nu \in \mathbb{R}} \{\nu + E_Q[f^*(\phi - \nu)]\} \right\} , \tag{4}$$

where f^* is the Legendre transform of f. Some of the important properties of Lipschitz regularized f-divergences, which summarizes results from Dupuis & Mao (2022); Birrell et al. (2022a) are given in Appendix A. Typical examples of f-divergences include the KL-divergence with $f_{\mathrm{KL}}(x) = x \log x$, and the α-divergences with $f_\alpha(x) = \frac{x^\alpha - 1}{\alpha(\alpha-1)}$. The corresponding Legendre transforms are $f^*_{\mathrm{KL}}(y) = e^{y-1}$ and $f^*_\alpha \propto y^{\frac{\alpha}{(\alpha-1)}}$. In the KL case the infimum over ν can be solved analytically and yields the Lipschitz-regularized Donsker-Varadhan formula with a $\log E_Q[e^\phi]$ term, see Birrell et al. (2022b) for more on variational representations.

Gradient flows in probability space The groundbreaking work of Jordan et al. (1998); Otto (2001) recasted the Fokker-Planck (FP) and the porous media equations as gradient flows in the 2-Wasserstein space of probability measures. More specifically, the Fokker-Planck equation can be thought as the gradient flow of the KL divergence

$$\partial_t p_t = \nabla \cdot \left(p_t \nabla \frac{\delta D_{KL}(p_t \| q)}{\delta \rho} \right) = \nabla \cdot \left(p_t \nabla \log \left(\frac{p_t}{q} \right) \right) \tag{5}$$

where p_t and q are the densities at time t and the stationary density respectively. A similar result relates weighted porous media equation and gradient flows for f divergences Otto (2001). This probabilistic formulation allowed the use of such gradient flows and related perspectives to build new Machine Learning concepts and tools. For instance, the Fokker-Planck equation plays a key role in both generative modeling and in sampling.

In the remaining part of this Introduction we provide an outline of our main results, as well as a discussion of related prior work.

Lipschitz-regularized gradient flows in probability space From a generative model perspective where Q is known only through samples (and may not have density in the first place as Q is concentrated on low dimensional structure), one cannot use gradient flows such as equation 5 without further regularization. For instance, related generative methods such as score matching and diffusion models regularize data by adding noise, Song & Ermon (2020); Song et al. (2021). Here we propose a different and complementary approach by regularizing the divergence directly and without adding noise to the data.

In this paper we propose gradient flows for the Lipschitz-regularized divergences equation 4 of the form

$$\partial_t P_t = \operatorname{div} \left(P_t \nabla \frac{\delta D_f^{\Gamma_L}(P_t \| Q)}{\delta P} \right), \quad P_0 = P \in \mathcal{P}_1(\mathbb{R}^d), \tag{6}$$

for an initial (source) probability measure P and an equilibrium (target) measure Q, for P, Q in the Wasserstein space $\mathcal{P}_1(\mathbb{R}^d) = \left\{ P \in \mathcal{P}(\mathbb{R}^d) : \int |x| dP(x) < \infty \right\}$. We want to emphasize that $\mathcal{P}_1(\mathbb{R}^d)$ includes singular measures such as empirical distributions constructed from data. In Section 2 we prove the first variation formula

$$\frac{\delta D_f^{\Gamma_L}(P \| Q)}{\delta P} = \phi^{L,*} = \operatorname*{argmax}_{\phi \in \Gamma_L} \left\{ E_P[\phi] - \inf_{\nu \in \mathbb{R}} (\nu + E_Q[f^*(\phi - \nu)]) \right\}. \tag{7}$$

The optimal $\phi^{L,*}$ in equation 7 (called the discriminator in the GAN literature) in the variational representation of the divergence equation 4 serves as a potential to transport probability measures, leading to the *transport/variational* PDE reformulation of equation 6:

$$\begin{aligned} &\partial_t P_t + \operatorname{div}(P_t v_t^L) = 0, \quad P_0 = P \in \mathcal{P}_1(\mathbb{R}^d), \\ &v_t^L = -\nabla \phi_t^{L,*}, \quad \phi_t^{L,*} = \operatorname*{argmax}_{\phi \in \Gamma_L} \left\{ E_{P_t}[\phi] - \inf_{\nu \in \mathbb{R}} (\nu + E_Q[f^*(\phi - \nu)]) \right\}, \end{aligned} \tag{8}$$

where we remind that $\Gamma_L = \{\phi : \mathbb{R}^d \to \mathbb{R} : |\phi(x) - \phi(y)| \leq L|x - y| \text{ for all } x, y\}$. This transport/variational PDE should be understood in a weak sense since P_t and Q are not necessarily assumed to have densities. However, the purpose of this paper is not to develop the PDE theory for this new gradient flow but rather to first establish its computational feasibility through associated particle algorithms, explore its usefulness in generative modeling for problems with high-dimensional scarce data, and overall computational efficiency and scalability. Based on these findings we believe

that a rigorous PDE analysis that includes well-posedness, stability, regularity, convergence to equilibrium and so on will be subsequently in order. To this end, the DiPerna-Lions theory for transport equations with rough velocity fields and its more recent powerful variants, as well as functional inequalities for porous medium and Fokker-Planck equations could prove to be useful mathematical methods, but that remains to be explored and demonstrated. For example, given sufficient regularity, along a trajectory of a smooth solution P_t of (8) we have the following dissipation identity:

$$\frac{d}{dt}D_f^{\Gamma_L}(P_t\|Q) = -I_f^{\Gamma_L}(P_t\|Q) \leq 0 \quad \text{where} \quad I_f^{\Gamma_L}(P_t\|Q) = E_{P_t}\left[|\nabla\phi_t^{L,*}|^2\right] \tag{9}$$

and $I_f^{\Gamma_L}(P\|Q)$ is a Lipschitz-regularized version of the Fisher Information. Due to the transport/variational PDE (8) $I_f^{\Gamma_L}(P\|Q)$ can be interpreted as a total kinetic energy, see Section 2, as well as Section 3 for its practical importance in the particle algorithms introduced next.

Lipschitz-regularized Generative Particle Algorithms (GPA) In the context of generative models, the target Q and the generative model P_t in equation 6 are available only through their samples and associated empirical distributions. However, as it can be seen from equation 3 the divergence $D_f^{\Gamma_L}(P\|Q)$ can compare directly singular distributions (e.g. empirical measures) without need for extra regularization such as adding noise to our models. For precisely this reason the proposed gradient flow equation 6 is a natural mathematical object to consider as a generative model.

From a computational perspective, it becomes feasible to solve high-dimensional transport PDE such as equation 6 when considering the Lagrangian formulation of the transport PDE in (8), i.e. the ODE/variational problem

$$\begin{aligned}
&\frac{d}{dt}Y_t = v_t^L(Y_t) = -\nabla\phi_t^{L,*}(Y_t)\,, \quad Y_0 \sim P\,,\\
&\phi_t^{L,*} = \underset{\phi\in\Gamma_L}{\operatorname{argmax}}\left\{E_{P_t}[\phi] - \inf_{\nu\in\mathbb{R}}\{\nu + E_Q[f^*(\phi-\nu)]\}\right\}.
\end{aligned} \tag{10}$$

In order to turn equation 10 into a particle algorithm we need the following ingredients:

- Consider samples $(X^{(i)})_{i=1}^N$ from the target Q and $(Y^{(i)})_{i=1}^M$ samples from an initial (source) distribution $P = P_0$. In this case for the corresponding empirical measures $\widehat{Q}^N$ and $\widehat{P}^M$ we will consider the gradient flow equation 6 for $D_f^{\Gamma_L}(\widehat{P}^M\|\widehat{Q}^N)$. A key observation in our algorithms is that the divergence $D_f^{\Gamma_L}(\widehat{P}^M\|\widehat{Q}^N)$ is always well-defined and finite due to Lipschitz regularization and equation 3.
- Corresponding estimators for the objective functional in the variational representation of the divergence $D_f^{\Gamma_L}(\widehat{P}^M\|\widehat{Q}^N)$, see equation 4 and also equation 10:

$$E_{\widehat{P}^M}[\phi] - \inf_{\nu}(\nu + E_{\widehat{Q}^N}[f^*(\phi-\nu)]) = \frac{\sum_{i=1}^M \phi(Y_n^{(i)})}{M} - \inf_{\nu\in\mathbb{R}}\left\{\nu + \frac{\sum_{i=1}^N f^*(\phi(X^{(i)})-\nu)}{N}\right\}.$$

- The function space Γ_L in equation 10 is approximated by a space of neural network approximations Γ_L^{NN}. The Lipschitz condition can be implemented via neural network spectral normalization as discussed in Section 3.
- The transport ODE in equation 10 is discretized in time using an Euler or a higher order scheme, see Section 3. Furthermore the gradient $\nabla\phi_t^{L,*}$ is evaluated by automatic differentiation of neural networks at the positions of the particles.

By incorporating these approximations we derive from equation 10, upon Euler time discretization the *Lipschitz-regularized generative particle algorithm* (GPA):

$$\begin{aligned}
&Y_{n+1}^{(i)} = Y_n^{(i)} - \Delta t\nabla\phi_n^{L,*}(Y_n^{(i)})\,, \quad Y_0^{(i)} = Y^{(i)}\,,\, Y^{(i)} \sim P\,, \quad i = 1,...,M\\
&\phi_n^{L,*} = \underset{\phi\in\Gamma_L^{NN}}{\operatorname{argmax}}\left\{\frac{\sum_{i=1}^M \phi(Y_n^{(i)})}{M} - \inf_{\nu\in\mathbb{R}}\left\{\nu + \frac{\sum_{i=1}^N f^*(\phi(X^{(i)})-\nu)}{N}\right\}\right\},
\end{aligned} \tag{11}$$

Besides the transport aspect of equation 11, it can be also viewed as a new generative algorithm, where the input is samples $(X^{(i)})_{i=1}^N$ from the "target" Q. Initial data, usually referred to as "source" data , $(Y_0^{(i)})_{i=1}^M$ from P are transported via equation 11, after time $T = n_T \Delta t$, where n_T is the total number of steps, to a new set of generated data $(Y_{n_T}^{(i)})_{i=1}^M$ that are approximately $\sim Q$. See for instance the demonstration in Figure 1.

In analogy to equation 10, this Lagrangian point of view has been recently introduced to write the solution of the Fokker-Planck equation equation 5 as the density of particles evolving according to its Lagrangian formulation, Maoutsa et al. (2020),

$$\frac{d}{dt} Y_t = v_t(Y_t) = \nabla \log q(Y_t) - \nabla \log p_t(Y_t)\,, \quad \text{where } Y_t \sim P_t\,. \tag{12}$$

In fact, in Song et al. (2021), the authors proposed the deterministic probability flow equation 12 as an alternative to generative stochastic samplers for score generative models due to advantages related to obtaining better statistical estimators. We note here that the score term $\nabla \log p_t(Y_t)$ in equation 12 is not a priori known and can be estimated by score-based methods Hyvärinen (2005). In practice, these Lagrangian tools are used both for generation, Song et al. (2021) as well as sampling Reich & Weissmann (2021); Boffi & Vanden-Eijnden (2022).

Main contributions As discussed earlier, the purpose of this paper is to introduce the new Lipschitz-regularized gradient flow equation 6 and establish its computational feasibility through associated particle algorithms, its computational efficiency and scalability, and explore its usefulness in generative modeling for problems with high-dimensional scarce data. Towards these goals our main findings can be summarized in the following directions.

1. *Lipschitz-regularization*. We demonstrate that Lipschitz-regularized divergences provide a well-behaved pseudo-metric between models and data or data and data. They remain finite under very broad conditions, making the training of generative particle algorithms equation 11 on data always well-defined and numerically stable. In fact, Lipschitz regularization corresponds to effectively imposing an advection-type Courant – Friedrichs – Lewy (CFL) numerical stability condition on the Fokker-Planck PDE equation 5 through the Lipschitz-regularization parameter L in equation 6. The example in Section 4 demonstrates empirically that the selection of L is important.
2. *Choice of f-divergence in equation 6*. Although KL is often a natural choice, a careful selection of f-divergences, for example the family of α-divergences where $f_\alpha = \frac{x^\alpha - 1}{\alpha(\alpha-1)}$, will allow for training that is numerically stable, including examples with heavy-tailed data, see Section 5.
3. *Generative algorithms for scarce data.* We demonstrate that our proposed GPA can learn distributions from very small data sets, including MNIST and other benchmarks, often supported on low-dimensional structures, see Figure 1. Specifically, we observe that GPA significantly outperforms GANs, including even those trained with the same divergence. A possible explanation is that GPA does not have to train a generator and thus requires less data and/or training time, and as a result is both more stable and more accurate. We refer to the examples in Section 6. For these reasons, GPA can also be used as an effective tool for *data augmentation* in scarce data settings, see the related demonstration example in Section 6.
4. *Latent-space GPA for very high-dimensional problems.* GPA can be effective even for scarce data sets in very high dimensions. We provide a demonstration example where we integrate gene expression data sets exceeding 50,000 dimensions. The goal of data transportation in this context is to mitigate batch effects between studies of different groups of patients, see Section 7. From a practical perspective, to be able to operate in such high-dimensions we need a latent-space representation of the data and subsequently we use GPA to transport particles in the latent space. We provide *performance guarantees* which are applicable to general latent-space generative models using a new Data Processing Inequality (DPI) for Lipschitz-regularized divergences. The DPI serves as an *a posteriori* bound in the sense of numerical analysis, where the approximation in the tractable latent space bounds the error in the much less tractable real space.

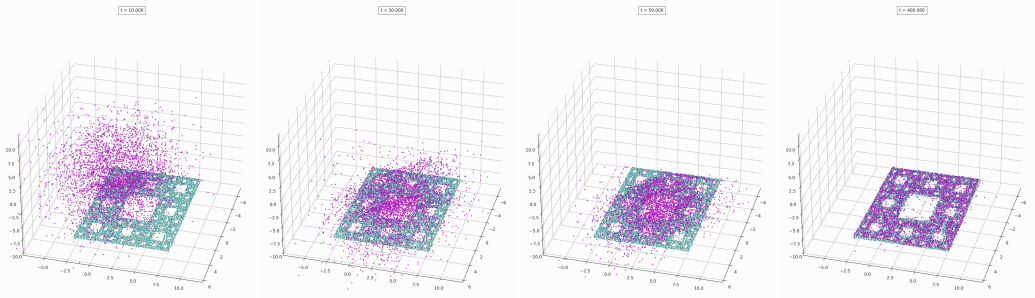


Figure 1: Sierpinski carpet embedded in 3D. Source data (purple particles) are transported via GPA close to the target data (cyan particles). The target particles were sampled from a Sierpinski carpet of level 4 by omitting all finer scales. See Figure 5 for a related 2D demonstration and a comparison to GANs.

Structure of the paper In Section 2, we construct Lipschitz-regularized gradient flows in probability space by computing the first variation of Lipschitz-regularized divergences. In Section 3 we define generative particle algorithms (GPA) for these gradient flows. In Sections 4-7 we provide our primary demonstration examples as discussed earlier. In particular in Section 7 we present the new DPI that provides performance guarantees as we transition from latent to real space in very high-dimensional data sets.

Related work Our approach is inspired by the MMD and KALE gradient flows from Arbel et al. (2019); Glaser et al. (2021) based on an entropic regularization of the MMD metrics, and related work using the Kernelized Sobolev Discrepancy Mroueh et al. (2019). Furthermore, the recent work of Dupuis & Mao (2022); Birrell et al. (2022a) built the mathematical foundations for a large class of new divergences which contains the Lipschitz regularized f-divergences and used them to construct GANs, and in particular symmetry preserving GANs Birrell et al. (2022c). Also related is the Sinkhorn divergence Genevay et al. (2016) which is a different entropic regularization of the 2-Wasserstein metrics. Lipschitz regularizations and the related spectral normalization have been shown to improve the stability of GANs Miyato et al. (2018); Arjovsky et al. (2017); Gulrajani et al. (2017). Our particle algorithms share similarities with GANs Goodfellow et al. (2014); Arjovsky et al. (2017), sharing the same discriminator but having a different generator step. They are also broadly related to continuous-time generative algorithms, such as continuous-time normalizing flows (NF) Chen et al. (2018a); Köhler et al. (2020); Chen et al. (2018b), diffusion models Sohl-Dickstein et al. (2015); Ho et al. (2020) and score-based generative flows Song & Ermon (2020); Song et al. (2021). However, the aforementioned continuous-time models, along with variational autoencoders Kingma & Welling (2013) and energy based methods LeCun et al. (2006), are mostly KL/likelihood-based.

On the other hand, particle gradient flows such as the ones proposed here, can be classified as a separate class within implicit generative models. Within such generative models that include GANs, there is more flexibility in selecting the loss function in terms of a suitable divergence or probability metric, enabling the direct comparison of even mutually singular distributions, e.g. Arjovsky et al. (2017); Gulrajani et al. (2017). Gradient flows in probability spaces related to the Kullback-Leibler (KL) divergence, such as the Fokker-Planck equations and Langevin dynamics Roberts & Tweedie (1996); Durmus & Moulines (2017) or Stein variational gradient descent Liu & Wang (2016); Liu (2017); Lu et al. (2019), form the basis of a variety of sampling algorithms when the target distribution Q has a known density (up to normalization). The weighted porous media equations form another family of gradient flows based on α-divergences Markowich & Villani (2000); Otto (2001); Ambrosio et al. (2005); Dolbeault et al. (2008); Vázquez (2014) which are very useful in the presence of heavy tails. Our gradient flows are Lipschitz-regularizations of such classical PDE's (Fokker-Planck and porous medium equations). Finally, deterministic particle methods and associated probabilistic flows of ODEs such as the ones derived here for Lipschitz-regularized gradient flows, were considered in recent works for classical KL-divergences and associated Fokker-Planck equations as sampling tools Maoutsa et al. (2020); Boffi & Vanden-Eijnden (2022), for Bayesian inference Reich & Weissmann (2021) and as generative models Song et al. (2021). Our latent gen-

erative particles approach is inspired by latent diffusion models using auto-encoders Rombach et al. (2021a) and by autoencoders used for model reduction in coarse-graining for molecular dynamics, Vlachas et al. (2022); Wang & Gómez-Bombarelli (2019); Stieffenhofer et al. (2021).

2 LIPSCHITZ-REGULARIZED GRADIENT FLOWS

In this Section, our primary focus is to introduce the concept of Lipschitz-regularized gradient flows in probability space, including the key computation of the first variation of Lipschitz-regularized divergences. The latter result will allow us to build effective particle-based algorithms in Section 3. Indeed, given a target probability measure Q, we build an evolution equation for probability measures based on the Lipschitz regularized f-divergences $D_f^{\Gamma_L}(P\|Q)$ in equation 4, by considering the PDE

$$\partial_t P_t = \operatorname{div}\left(P_t \nabla \frac{\delta D_f^{\Gamma_L}(P_t\|Q)}{\delta P}\right), \quad P_0 = P \in \mathcal{P}_1(\mathbb{R}^d) \tag{13}$$

where $\frac{\delta D_f^{\Gamma_L}(P\|Q)}{\delta P}$ is the first variation of $D_f^{\Gamma_L}(P\|Q)$, to be discussed below in Theorem 2.1. An advantage of the Lipschitz regularized f-divergences is its ability to compare singular measures and thus equation 13 needs to be understood in a weak sense. For this reason we use the probability measure P_t notation in equation 13, instead of density notation p_t as in the Fokker-Planck (FP) equation equation 5. In the formal asymptotic limit $L \to \infty$ and if $P \ll Q$, equation 13 yields the FP equation equation 5 (for KL divergence) and the weighted porous medium equation (for α-divergences) Otto (2001); Dolbeault et al. (2008), see Remark 2.6. We note that the purpose of this paper is not to develop the PDE theory for equation 13 but rather to first establish its computational feasibility through associated particle algorithms and subsequently demonstrate its usefulness in generative modeling for problems with high-dimensional scarce data.

Theorem 2.1 (first variation of Lipschitz regularized f-divergences)**.** *Assume f is superlinear, strictly convex and $P, Q \in \mathcal{P}_1(\mathbb{R}^d)$. We define*

$$\phi^{L,*} := \operatorname*{argmax}_{\phi\in\Gamma_L}\left\{E_P[\phi] - \inf_{\nu\in\mathbb{R}}\{\nu + E_Q[f^*(\phi-\nu)]\}\right\}. \tag{14}$$

where the optimizer $\phi^{L,*} \in \Gamma_L$ exists, is defined on supp$(P) \cup$ supp(Q), and is unique up to a constant. Subsequently, we extend $\phi^{L,*}$ in all of $\mathbb{R}^d$ using equation 18. Let ρ be a signed measure of total mass 0 and let $\rho = \rho_+ - \rho_-$ where $\rho_\pm \in \mathcal{P}_1(\mathbb{R}^d)$ are mutually singular.

If $P + \epsilon\rho \in \mathcal{P}_1(\mathbb{R}^d)$ for sufficiently small $|\epsilon|$, then $D_f^{\Gamma_L}(P+\epsilon\rho\|Q)$ is differentiable at $\epsilon = 0$ and

$$\lim_{\epsilon\to 0}\frac{1}{\epsilon}\left(D_f^{\Gamma_L}(P+\epsilon\rho\|Q) - D_f^{\Gamma_L}(P\|Q)\right) = \int \phi^{L,*} d\rho\,. \tag{15}$$

Then we write

$$\frac{\delta D_f^{\Gamma_L}(P\|Q)}{\delta P}(P) = \phi^{L,*}\,. \tag{16}$$

Remark 2.2. The first variation of the Lipschitz-regularized KL divergence given in Theorem 2.1, is defined on $\mathcal{P}_1(\mathbb{R}^d)$ which includes singular measures such as empirical distributions. On the other hand, the classical Fokker-Planck equation 5 (where $L = \infty$) can be re-written in a gradient flow formulation

$$\begin{aligned}\partial_t p_t &= \nabla\cdot(\nabla\phi^*(x,t)p_t)\,, \quad \text{where}\\ \phi_t^* &= \log\frac{p_t(x)}{q(x)} = \operatorname{argmax}\left\{E_{P_t}[\phi] - \inf_{\nu\in\mathbb{R}}\left\{\nu + E_Q[e^{\phi-\nu-1}]\right\}\right\}\end{aligned} \tag{17}$$

is built on the first variation of the (un-regularized) KL divergence given by

$$\frac{\delta D_{KL}(P\|Q)}{\delta P} = \log\frac{dP}{dQ} = \phi^* = \operatorname*{argmax}_{\phi\in C_b(\mathbb{R}^d)}\left\{E_P[\phi] - \inf_{\nu\in\mathbb{R}}\left\{\nu + E_Q[e^{\phi-\nu-1}]\right\}\right\}$$

In this case, the first variation is defined on the space of probability measures which are absolutely continuous with respect to Q. This can be generalized for any f-divergence, see also the relevant discussions at the end of this Section.

A version of Theorem 2.1 was proved in Dupuis & Mao (2022) for the special case of KL divergence and here is extended to general f-divergences. This type of generality is necessary in generative modeling based on both past experience in GANs Nowozin et al. (2016); Mao et al. (2017); Birrell et al. (2022a;c), as well as the demonstration examples considered here. The proof of Theorem 2.1 is based on the next lemma.

Lemma 2.3. *Let f be superlinear and strictly convex and $P, Q \in \mathcal{P}_1(\mathbb{R}^d)$. For $y \notin \text{supp}(P) \cup \text{supp}(Q)$, we define*

$$\phi^{L,*}(y) = \sup_{x \in \text{supp}(Q)} \left\{ \phi^{L,*}(x) + L|x - y| \right\} . \tag{18}$$

Then $\phi^{L,*}$ is Lipschitz continuous on $\mathbb{R}^d$ with Lipschitz constant L and $\phi^{L,*} = \sup\{h(x) : h \in \Gamma_L,\ h(y) = \phi^{L,*}(y), \text{for every } y \in \text{supp}(Q)\}$.

Proof. The fact that $\phi^{L,*}$ is Lipschitz continuous on $\mathbb{R}^d$ is straightforward by using the triangle inequality. Moreover, since $h \in \Gamma_L$, we have that $h(x) \leq h(y) + L\|x - y\|$. This implies that for $y \in \text{supp}(Q)$ and $x \notin \text{supp}(Q)$, $h(x) \leq \inf_{y \in \text{supp}(Q)} \{h(y) + L\|x - y\|\} = \inf_{y \in \text{supp}(Q)} \{\phi^{L,*}(y) + L\|x - y\|\} = \phi^{L,*}(x)$. Since $\phi^{L,*}(y) \in \Gamma_L$, this concludes the proof. □

Proof of Theorem 2.1. We use the variational formula equation 4 for $D_f^{\Gamma_L}(P + \epsilon\rho\|Q)$ where we suppose that $P + \epsilon\rho \in \mathcal{P}_1(\mathbb{R}^d)$.

$$\begin{aligned} D_f^{\Gamma_L}(P + \epsilon\rho\|Q) &= \sup_{\phi \in \Gamma_L} \left\{ E_{P+\epsilon\rho}[\phi] - \inf_{\nu \in \mathbb{R}} \{\nu + E_Q[f^*(\phi - \nu)]\} \right\} \\ &\geq \int \phi^{L,*}\, d(P + \epsilon\rho) - \inf_{\nu \in \mathbb{R}} \left\{ \nu + \int f^*(\phi^{L,*} - \nu) dQ \right\} \\ &= \epsilon \int \phi^{L,*} d\rho + D_f^{\Gamma_L}(P\|Q) \end{aligned} \tag{19}$$

Thus

$$\liminf_{\epsilon \to 0^+} \frac{1}{\epsilon} \left(D_f^{\Gamma_L}(P + \epsilon\rho\|Q) - D_f^{\Gamma_L}(P\|Q) \right) \geq \int \phi^{L,*} d\rho$$

For the other direction let us define $F(\epsilon) = D_f^{\Gamma_L}(P + \epsilon\rho\|Q)$. By Theorem A.1 $F(\epsilon)$ is convex, lower semicontinuous and finite on $[0, \epsilon_0]$. Due to the convexity of F, it is differentiable on $(0, \epsilon_0)$ except for a countable number of points. Let $\epsilon \in (0, \epsilon_0)$ such that F is differentiable and $\delta > 0$ small. Also, let $\phi_\epsilon^{L,*}$ be the optimizer of $D_f^{\Gamma_L}(P + \epsilon\rho\|Q)$ satisfying $\phi_\epsilon^{L,*}(0) = 0$ so that

$$D_f^{\Gamma_L}(P + \epsilon\rho\|Q) = \int \phi_\epsilon^{L,*}\, d(P + \epsilon\rho) - \inf_{\nu \in \mathbb{R}} \left\{ \nu + \int f^*(\phi_\epsilon^{L,*} - \nu) dQ \right\} .$$

By using the same argument as before in the proof, we have that

$$D_f^{\Gamma_L}(P + (\epsilon + \delta)\rho\|Q) - D_f^{\Gamma_L}(P + \epsilon\rho\|Q) \geq \delta \int \phi_\epsilon^{L,*} d\rho \tag{20}$$

and

$$D_f^{\Gamma_L}(P + (\epsilon - \delta)\rho\|Q) - D_f^{\Gamma_L}(P + \epsilon\rho\|Q) \geq -\delta \int \phi_\epsilon^{L,*} d\rho \tag{21}$$

which gives us that

$$\begin{aligned} \int \phi_\epsilon^{L,*} d\rho &\leq \lim_{\delta \to 0} \frac{1}{\delta} \left(D_f^{\Gam_L}(P + (\epsilon + \delta)\rho\|Q) - D_f^{\Gamma_L}(P + \epsilon\rho\|Q) \right) \\ &= F'(\epsilon) \\ &= \lim_{\delta \to 0} \frac{1}{\delta} \left(D_f^{\Gamma_L}(P + \epsilon\rho\|Q) - D_f^{\Gamma_L}(P + (\epsilon - \delta)\rho\|Q) \right) \\ &\leq \int \phi_\epsilon^{L,*} d\rho\,. \end{aligned} \tag{22}$$

Consequently,

$$F'(\epsilon) = \int \phi_\epsilon^{L,*} d\rho\,. \tag{23}$$

Let $F'_+(0)$ be the right derivative at $\epsilon = 0$, i.e. $F'_+(0) = \lim_{\epsilon\to 0^+} \frac{1}{\epsilon}\left(F(\epsilon) - F(0)\right)$. By convexity, for any sequence $\{\epsilon_n\}_{n\in\mathbb{N}}$ such that F is differentiable at ϵ_n and $\epsilon_0 > \epsilon_n \downarrow 0$, we have

$$F'_+(0) = \lim_{n\to\infty} F'(\epsilon_n) = \lim_{n\to\infty} \int \phi_{\epsilon_n}^{L,*} d\rho\,.$$

As $\mathbb{R}^d$ can be written as $\mathbb{R}^d = \cup_{m\in\mathbb{N}} K_m$ with $K_m \subset \mathbb{R}^d$ being compact set and $K_m \subset K_{m+1}$, we have that for any $m \in \mathbb{N}$, $\phi_{\epsilon_n}^{L,*}$ is bounded and equicontinuous on K_m. By the Arzelà-Ascoli theorem, there exists a subsequence of $\phi_{\epsilon_n}^{L,*}$ that converges uniformly in K_m. Using diagonal argument, by taking subsequences sequentially along $\{K_m\}_{m\in\mathbb{N}}$ where the next subsequence is a subsequence of the former one, and take one element from each sequence, we conclude there exists a subsequence such that $\phi_{\epsilon_{n_k}}^{L,*}$ converges uniformly in any K_m. Since $\mathbb{R}^d = \cup_{m\in\mathbb{N}} K_m$, we conclude that $\phi_{\epsilon_{n_k}}^{L,*}$ converges pointwise in $\mathbb{R}^d$. Let $\phi_0^{L,*} \in \mathrm{Lip}^L(\mathbb{R}^d)$ be the limit. For simplicity, from now on we denote by $\phi_{\epsilon_n}^{L,*}$ the convergent subsequence, by a slight abuse of notation. At this point, we recall that for any $\epsilon \in (0, \epsilon_0)$, $\phi_\epsilon^{L,*}(0) = 0$. For any x, $|\phi_\epsilon^{L,*}(x) - \phi_\epsilon^{L,*}(0)| \le L|x|$ which implies that

$$|\phi_\epsilon^{L,*}(x)| \le L|x|$$

The function $L|x|$ is integrable with respect to ρ because $\rho = \rho_+ - \rho_-$ and $\rho_\pm \in \mathcal{P}_1(\mathbb{R}^d)$. Thus by the dominated convergence theorem

$$F'_+(0) = \lim_{n\to\infty} \int \phi_{\epsilon_n}^{L,*} d\rho = \int \phi_0^* d\rho$$

By the lower semicontinuity of $D_f^{\Gamma_L}(\cdot\|Q)$, see Theorem A.1, we have

$$\begin{aligned}
D_f^{\Gamma_L}(P\|Q) &\le \liminf_{n\to\infty} D_f^{\Gamma_L}(P + \epsilon_n\rho\|Q) \\
&= \liminf_{n\to\infty} \left\{ E_{P+\epsilon_n\rho}[\phi_{\epsilon_n}^{L,*}] - \inf_{\nu\in\mathbb{R}} \left\{\nu + E_Q[f^*(\phi_{\epsilon_n}^{L,*} - \nu)]\right\} \right\} \\
&= \liminf_{n\to\infty} E_{P+\epsilon_n\rho}[\phi_{\epsilon_n}^{L,*}] - \limsup_{n\to\infty} \inf_{\nu\in\mathbb{R}} \left\{\nu + E_Q[f^*(\phi_{\epsilon_n}^{L,*} - \nu)]\right\} \\
&\le E_P[\phi_0^{L,*}] - \inf_{\nu\in\mathbb{R}} \left\{\nu + E_Q[f^*(\phi_0^{L,*} - \nu)]\right\} \\
&\le D_f^{\Gamma_L}(P\|Q)
\end{aligned} \tag{24}$$

where for the second inequality we use the dominated convergence theorem, equation 23 and that by Fatou's lemma

$$\limsup_{n\to\infty} \int f^*(\phi_{\epsilon_n}^{L,*}) dQ \ge \liminf_{n\to\infty} \int f^*(\phi_{\epsilon_n}^{L,*}) dQ \ge \int f^*(\phi_0^{L,*}) dQ$$

Since both sides of the inequality coincide, $\phi_0^{L,*}$ must be the optimizer. By Lemma 2.3 and Theorem A.1, we have that $\phi_0^{L,*}(x) \le \phi^{L,*}$ for all x. Thus

$$F'_+(0) = \int \phi_0^* d\rho \le \int \phi^* d\rho.$$

which concludes the proof. For the uniqueness of the optimizer $\phi^{L,*}$ we refer to Theorem A.1. □

Using Theorem 2.1 we can now rewrite equation 13 as a *transport/variational* PDE,

$$\begin{aligned}
&\partial_t P_t + \mathrm{div}(P_t v_t^L) = 0\,, \quad P_0 = P \in \mathcal{P}_1(\mathbb{R}^d)\,, \\
&v_t^L = -\nabla\phi_t^{L,*}\,, \quad \phi_t^{L,*} = \operatorname*{argmax}_{\phi\in\Gamma_L} \left\{ E_{P_t}[\phi] - \inf_{\nu\in\mathbb{R}} (\nu + E_Q[f^*(\phi - \nu)]) \right\}.
\end{aligned} \tag{25}$$

Remark 2.4. **(a)** The transport/variational reformulation equation 25 is the starting point for developing our generative particle algorithms in Section 3 based on data, when P and Q are replaced by their empirical measures $\hat{P}^M$, $\hat{Q}^N$ based on M and N i.i.d. samples respectively. **(b)** Furthermore, equation 25 provides a numerical stability perspective on the Lipschitz regularization equation 13 In particular, the Lipschitz condition on $\phi \in \Gamma_L$ enforces a finite speed of propagation of at most L in the transport equation in equation 25. This is in sharp contrast with the FP equation equation 5, which is a diffusion equation and has infinite speed of propagation. We refer to Section 4 for connections to the Courant, Friedrichs, and Lewy (CFL) stability condition and practical implications.

The gradient flow structure of equation 13 is reflected in dissipation estimates, namely an equation for the rate of change (dissipation) of the divergence along smooth solutions P_t of equation 13.

Theorem 2.5 (Lipschitz-regularized dissipation)**.** *Along a trajectory of a smooth solution $\{P_t\}_{t\geq 0}$ of equation 25 with source probability distribution $P_0 = P$ we have the following rate of decay identity:*

$$\frac{d}{dt} D_f^{\Gamma_L}(P_t \| Q) = -I_f^{\Gamma_L}(P_t \| Q) \leq 0 \tag{26}$$

where we define the Lipschitz-regularized Fisher Information as

$$I_f^{\Gamma_L}(P_t \| Q) = E_{P_t}\left[|\nabla \phi^{L,*}|^2\right] . \tag{27}$$

Consequently, for any $T \geq 0$, we have $D_f^{\Gamma_L}(P_T \| Q) = D_f^{\Gamma_L}(P \| Q) - \int_0^T I_f^{\Gamma_L}(P_s \| Q) ds$.

For the generative particle algorithms of Section 3 the Lipschitz-regularized Fisher Information will be interpreted as the total kinetic energy of the particles equation 34.

Proof of Theorem 2.5. We obtain equation 26 by the next calculation, assuming sufficient smoothness. We use the divergence theorem together with vanishing boundary conditions, as well as equation 7 and equation 8.

$$\begin{aligned}
\frac{d}{dt} D_f^{\Gamma_L}(P_t \| Q) &= \left\langle \frac{\delta D_f^{\Gamma_L}(P \| Q)}{\delta P}(P), \frac{\partial P_t}{\partial t} \right\rangle \\
&= \left\langle \phi_t^{L,*}, \operatorname{div}\left(P_t \nabla \phi_t^{L,*}\right) \right\rangle \\
&= -\int |\nabla \phi_t^{L,*}|^2 dP_t \\
&= -E_{P_t}\left[|\nabla \phi_t^{L,*}|^2\right] .
\end{aligned} \tag{28}$$

□

Remark 2.6 (Formal asymptotics of Lipschitz-regularized gradient flows). The rigorous asymptotic results presented in Dupuis & Mao (2022); Birrell et al. (2022a) regarding the limit of the Lipschitz-regularized f-divergences to (un-regularized) f-divergences as Γ_L gets larger (see also Theorem A.1), motivates a discussion on the formal asymptotics of the corresponding gradient flows. In particular, the Lipschitz-regularization $L \to \infty$ asymptotics towards the (unregularized) gradient flows,

$$\partial_t P_t = \operatorname{div}\left(P_t \nabla \phi_t^*\right) \text{ with } \phi_t^* = f'\left(\frac{dP_t(x)}{dQ(x)}\right) = \operatorname{argmax}\left\{E_{P_t}[\phi] - \inf_{\nu \in \mathbb{R}} \{\nu + E_Q[f^*(\phi)]\}\right\} ,$$

that can be formally obtained as the limit of the transport/variational PDEs equation 25, i.e.,

$$\underbrace{\partial_t P_t = \operatorname{div}\left(P_t \nabla \phi_t^{L,*}\right)}_{\text{Lip. regularized } f\text{-divergence flow}} \quad \underset{L\to\infty}{\longrightarrow} \quad \underbrace{\partial_t P_t = \operatorname{div}\left(P_t \nabla \phi_t^*\right)}_{f\text{-divergence flow}}, \text{ where } \phi_t^* = f'\left(\frac{dP_t}{dQ}\right) \tag{29}$$

When p_t and q are the probability densities of P_t and Q respectively, and $f(x) = f_{\mathrm{KL}}(x) = x\log(x)$ and $f_\alpha(x) = \frac{x^\alpha - 1}{\alpha(\alpha-1)}$, the PDEs on the right hand side in equation 29 become the classical Fokker-Planck and the Weighted Porous Medium equations respectively, and equation 29 takes the form

$$\partial_t p_t = \operatorname{div}\left(p_t \nabla \phi_t^{L,*}\right) \quad \underset{L\to\infty}{\longrightarrow} \quad \partial_t p_t = \operatorname{div}\left(p_t \nabla \log\left(\frac{p_t}{q}\right)\right) , \tag{30}$$

where $f'_{\mathrm{KL}}(x) = \log x - 1$, and

$$\partial_t p_t = \operatorname{div}\left(p_t \nabla \phi_t^{L,*}\right) \quad \underset{L\to\infty}{\longrightarrow} \quad \partial_t p_t = \frac{1}{\alpha-1}\operatorname{div}\left(p_t \nabla \left(\frac{p_t}{q}\right)^{\alpha-1}\right), \tag{31}$$

where $f'_\alpha(x) = \frac{x^{\alpha-1}}{\alpha-1}$.

Similar formal asymptotics can be considered for the Lipschitz-regularized Fisher Information $I_f^{\Gamma_L}(P\|Q)$ and its convergence to the (classical) Fisher Information as $L \to \infty$, for instance, $I_{f_{\mathrm{KL}}}^{\Gamma_L}(P\|Q) \underset{L\to\infty}{\longrightarrow} I(P\|Q) = E_P\left[|\nabla \log\left(\frac{dP}{dQ}\right)|^2\right]$. Finally, we note that functional inequalities involving the usual Fisher Information $I(P\|Q)$ are, in general, useful tools to prove convergence of a PDE to its equilibrium, e.g. exponential or polynomial convergence. Classical examples of such inequalities are the Poincaré and the Logarithmic Sobolev-type inequalities, and generalizations thereof for Fokker-Planck and porous medium equations, see Markowich & Villani (2000); Otto & Villani (2000); Toscani & Villani (2000); Dolbeault et al. (2008); Wang (2005). However, proving convergence of the new class of PDE gradient flows equation 13 to their equilibrium states, will require new functional inequalities entailing the Lipschitz-regularized Fisher Information.

3 Generative Particle Algorithms

In this section we build a numerical algorithm to solve the transport/discriminator gradient flow equation 25 when N i.i.d. samples from the target distribution Q are given. We first discretize the system in time using a forward-Euler scheme as follows,

$$\begin{aligned} P_{n+1} &= \left(I - \Delta t \nabla \phi_n^{L,*}\right)_{\#} P_n, \quad \text{where } P_0 = P \\ \phi_n^{L,*} &= \underset{\phi\in\Gamma_L}{\arg\max}\left\{E_{P_n}[\phi] - \inf_{\nu\in\mathbb{R}}\left\{\nu + E_Q[f^*(\phi-\nu)]\right\}\right\}. \end{aligned} \tag{32}$$

Here, the pushforward measure for a map $T : \mathbb{R}^d \to \mathbb{R}^d$ and $P \in \mathcal{P}(\mathbb{R}^d)$ is denoted by $T_{\#}P$ (i.e. $T_{\#}P(A) = P(T^{-1}(A))$.

Next, given N i.i.d. samples $\{X^{(i)}\}_{i=1}^N$ from the target distribution Q, we consider the empirical measure $\hat{Q}^N = N^{-1}\sum_{i=1}^N \delta_{X^{(i)}}$. Likewise, given M i.i.d. samples $\{Y_0^{(i)}\}_{i=1}^M$ from a known initial (source) probability measure P and consider the empirical measure $\hat{P}^M = M^{-1}\sum_{i=1}^M \delta_{Y_0^{(i)}}$. By replacing the measures P and Q in equation 32 by their empirical measures $\hat{P}^M$ and $\hat{Q}^N$ we obtain the following particle system.

$$\begin{aligned} Y_{n+1}^{(i)} &= Y_n^{(i)} - \Delta t \nabla \phi_n^{L,*}(Y_n^{(i)}), \quad Y_0^{(i)} = Y^{(i)}, \, Y^{(i)} \sim P, \quad i = 1, ..., M \\ \phi_n^{L,*} &= \underset{\phi\in\Gamma_L^{NN}}{\operatorname{argmax}}\left\{\frac{\sum_{i=1}^M \phi(Y_n^{(i)})}{M} - \inf_{\nu\in\mathbb{R}}\left\{\nu + \frac{\sum_{i=1}^N f^*(\phi(X^{(i)})-\nu)}{N}\right\}\right\}, \end{aligned} \tag{33}$$

where the function space Γ_L in equation 32 is approximated by a space of neural network approximations Γ_L^{NN}. We will refer to this particle algorithm as (f, Γ_L)-GPA or simply GPA. The transport mechanism given by equation 33 corresponds to a linear transport PDE in equation 25. However, between particles nonlinear interactions are introduced via the discriminator $\phi_n^{L,*}$ which in turn depends on all particles in equation 33 at step n of the algorithm, namely the generated particles $(Y_n^{(i)})_{i=1}^M$, as well as the "target" particles $(X^{(i)})_{i=1}^N$. Notice that $\phi_n^{L,*}$ discriminates the generated samples at time n from the target data using the second equation of equation 33, and is not directly using the generated data of the previous step $n-1$. The Lipschitz condition for Γ_L^{NN} can be implemented via neural network spectral normalization for neural network architectures, Miyato et al. (2018). Moreover the gradient of the discriminator is computed only at the positions of the particles. The procedure is summarized in Algorithm 1.

Overall, equation 33 is an approximation scheme of the Lagrangian formulation equation 10 of the Lipschitz-regularized gradient flow equation 6, where we have (a) discretized time, (b) approximated the function space Γ_L in terms of neural networks, and (c) used empirical distributions/particles to

build approximations of the target Q, (d) used gradient-based optimization methods to approximate the discriminator $\phi_n^{L,*}$ such as stochastic gradient descent or the Adam optimizer. All these elements are combined in Algorithm 1.

Remark 3.1 (Generator for GPA). The transport/discriminator formulation equation 32 is the mechanism for generating samples in GPA, directly transporting particles to the target by solving equation 33 iteratively in time, without having to learn a generator as GANs do, Goodfellow et al. (2014); Arjovsky et al. (2017); Birrell et al. (2022a). We also refer to the examples in Section 6. Furthermore, a comparison of the generative mechanisms of GPA and GANs is included in Section D.

Remark 3.2 (Lipschitz regularization & numerical stability). The Lipschitz bound L on the discriminator space implies a pointwise bound $|\nabla\phi_n^{L,*}(Y_n^{(i)})| \leq L$ and thus the particle speed is bounded by L. Hence the Lipschitz regularization imposes a speed limit L on the particles, ensuring the stability of the algorithm for suitable choices of L, see also Section 4. This speed constraint on the particles induces an implicit spatial discretization grid Δx where particles are transported for each Δt by at most $\Delta x = L\Delta t$. These heuristics are also related to the Courant, Friedrichs, and Lewy (CFL) condition for the stability of discrete schemes for transport PDEs, LeVeque (2007), we also refer to the discussion in Section 4.

Remark 3.3 (GPA kinetic energy and Lipschitz-regularized Fisher Information). Theorem 2.5 suggests the empirical Lipschitz-regularized Fisher Information,

$$I_f^{\Gamma_L}(\hat{P}_n^M \| \hat{Q}^N) = \int |\nabla\phi_n^{L,*}|^2 \hat{P}_n^M(dx) = \frac{1}{M}\sum_{i=1}^{M} |\nabla\phi_n^{L,*}(Y_n^{(i)})|^2 \,, \tag{34}$$

as a quantity of interest to monitor the convergence of GPA equation 33. Here $\hat{P}_n^M$ denotes the empirical distribution of the generative particles $(Y_n^{(i)})_{i=1}^M$. Indeed, $I_f^{\Gamma_L}(\hat{P}_n^M \| \hat{Q}^N)$ is equal to the total kinetic energy of the generative particles since $\nabla\phi_n^{L,*}(Y_n^{(i)})$ is the velocity of the i^{th} particle at time step n. The algorithm will stop when the total kinetic energy $I_f^{\Gamma_L}(\hat{P}_n^M \| \hat{Q}^N)$ is zero.

Algorithm 1 [(f, Γ_L)-GPA] Lipschitz regularized generative particles algorithm

Require: f for the choice of f-divergence and its Legendre conjugate f^*, L: Lipschitz constant, $n_{\max}$: number of updates for the particles, Δt: time step size, M: number of initial particles, N: number of target particles

Require: $W = \{W^l\}_{l=1}^D$: parameters for the NN $\phi : \mathbb{R}^d \to \mathbb{R}$, D: depth of the NN, δ: learning rate of the NN, $m_{\max}$: number of updates for the NN.

Result: $\{Y_{n_{\max}}^{(i)}\}_{i=1}^M$

1: Sample $\{Y_0^{(i)}\}_{i=1}^M \sim P_0 = P$, a batch of prior samples
2: Sample $\{X^{(j)}\}_{j=1}^N \sim Q$, a batch from the real data
3: Initialize $\nu \leftarrow 0$
4: Initialize W randomly and $W^l \leftarrow L^{1/D} * W^l / \|W^l\|_2, l = 1, \cdots, D$ ▷ $\phi_0^L(\cdot; W) \in \Gamma_L$
5: **for** $n = 0$ **to** $(n_{\max} - 1)$ **do**
6: **for** $m = 0$ **to** $m_{\max} - 1$ **do**
7: $grad_{W,\nu} \leftarrow \nabla_{W,\nu}\left[M^{-1}\sum_{i=1}^M \phi_n^L(Y_n^{(i)}; W) - N^{-1}\sum_{j=1}^N f^*(\phi_n^L(X^{(j)}; W) - \nu) + \nu\right]$
8: $(\nu, W) \leftarrow (\nu, W) + \delta \cdot optimizer(grad_\nu, grad_W)$
9: $W^l \leftarrow L^{1/D} * W^l / \|W^l\|_2 \,, l = 1, \cdots, D$
10: **end for** ▷ $\phi_n^{L,*}(\cdot; W) \in \Gamma_L$
11: $Y_{n+1}^{(i)} \leftarrow Y_n^{(i)} - \Delta t \nabla\phi_n^{L,*}(Y_n^{(i)}; W), \quad i = 1, \cdots, M$ ▷ forward Euler
12: **end for**

L-Lipschitz continuity is imposed by $W^l \leftarrow L^{1/D} * W^l / \|W^l\|_2, l = 1, \cdots, D$.

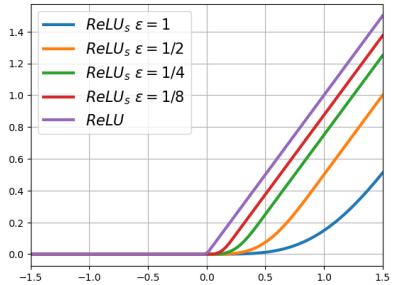
Remark 3.4 (Lipschitz regularization for NN). We can enforce the Lipschitz regularization in equation 33 and Algorithm 1 in two ways. **(a)** Hard constraints: Spectral Normalization Miyato et al. (2018) in Algorithm 1, **(b)** Soft constraints: Gradient Penalty (GP) added to the loss Birrell et al. (2022a).

Remark 3.5 (Higher-order discretization schemes). Replacing the forward Euler in equation 33 or Line 10 in Algorithm 1 with Heun's predictor/corrector method is found to improve significantly

the accuracy of the GPA for several examples, see for instance Figure 10. We refer to Section B for additional details as well as other higher-order schemes for time-discretization. For instance, Heun's method involves two calculations of the discriminator and higher-order methods can involve even more.

Remark 3.6 (Smooth activation functions). Smoother discriminators $\phi_n^{L,*}$ allow us to take smaller time step sizes Δt in equation 33 so that the algorithm can slow down and eventually stop, avoiding oscillations around the target. We build smoother discriminators by replacing the standard $ReLU$ activation function in NNs by a smoother one, namely $ReLU_s^\epsilon \in C^3$ with $0 \leq (ReLU_s^\epsilon)'(x) \leq 1$ given by equation 35. This activation function is compatible with spectral normalization technique for imposing Lipschitz continuity to a NN and is given by

$$ReLU_s^\epsilon(x), \quad \epsilon = 2^{-n}$$
$$= \begin{cases} 0, & x \leq 0 \\ \frac{x^2}{4\epsilon} + \frac{\epsilon}{2\pi^2}(\cos(\frac{\pi}{\epsilon}x) - 1), & 0 < x < 2\epsilon \\ x - \epsilon, & x \geq 2\epsilon. \end{cases} \tag{35}$$



Remark 3.7. [Estimators] Algorithm 1 estimates two natural quantities of interest: Lipschitz regularized f-divergence $M^{-1}\sum_{i=1}^M \phi_n^{L,*}(Y_n^{(i)};W) - N^{-1}\sum_{j=1}^N f^*(\phi_n^{L,*}(X^{(j)};W) - \nu^*) + \nu^*$ and Lipschitz regularized Fisher information equation 34 These quantities will be calculated in all our experiments.

4 Lipschitz regularization and numerical stability

In this section, we discuss the numerical stability of GPA induced by Lipschitz regularization.

First, we illustrate how Lipschitz regularization works in GPA Algorithm 1 by an example of a mixture of 2D Gaussians. As stated in Remark 3.2 (b), Lipschitz regularization provides a finite propagation speed bound on particles. We explore the influence of the Lipschitz regularization constant L by monitoring the Lipschitz regularized Fisher information equation 34 (i.e. kinetic energy of particles). In Figure 2 we track this quantity in time. We empirically observe that a proper choice of L enables the particles slow down and eventually stop near the target particles as discussed in Remark 3.3. Time trajectories of particles are displayed in Figure 3.

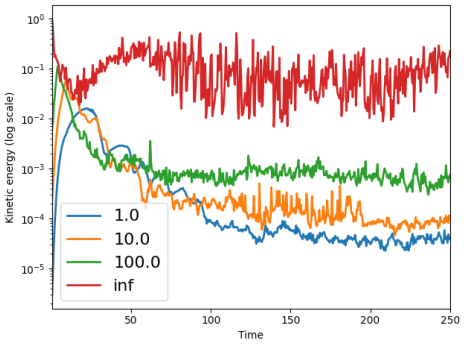


Figure 2: **(2D Mixture of Gaussians)** Kinetic energy of particles equation 34 for $(f_{\mathrm{KL}}, \Gamma_L)$-GPA with different L's. Theorem 2.5 suggests that particles need to slow down and practically stop when they reach the "vicinity" of the target particles. Time trajectories of particles are displayed in Figure 3.

Individual curves in Figure 2 result from the L-Lipschitz regularized $(f_{\mathrm{KL}}, \Gamma_L)$-GPA with $L = 1, 10, 100, \infty$. We fix all other parameters including the time step size Δt to figure out the influence of the Lipschitz constant L. For $L = 1, 10$, the kinetic energy decreases and particles eventually stop. However, without Lipschitz regularization, the particles keep (realtively) high speeds of propagation. Trajectories of particles in Figure 3 verify that in this case ($L = \infty$) the algorithm fails to converge.

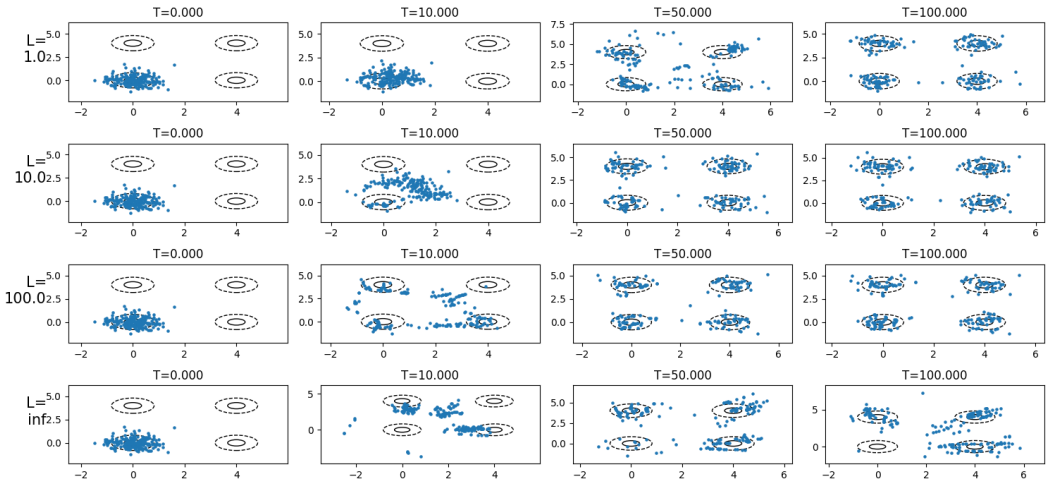


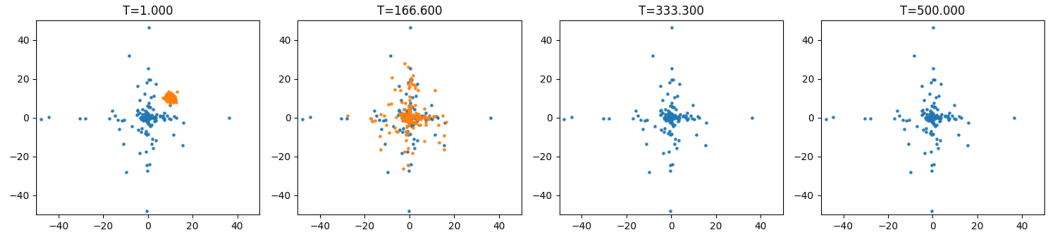
Figure 3: **(2D Mixture of Gaussians)** We empirically observe that Lipschitz constant L controls the propagation speed of $(f_{\mathrm{KL}}, \Gamma_L)$-GPA with different L's. For $L < \infty$, the particles are propagated to the 4 wells. As L gets larger, the algorithm becomes more unstable. For $L = \infty$ (unregularized KL) GPA fails to capture the target.

Numerical stability of GPA Based on our empirical findings, we observe a close relationship between a finite propagation speed and numerical stability of the algorithm. Indeed, from a numerical analysis point of view, equation 32 is a particle-based explicit scheme for the PDE equation 25. In this context, the Courant, Friedrichs, and Lewy (CFL) condition for stability of discrete schemes for transport PDEs such as the first equation in equation 25 becomes $\sup_x |\nabla \phi_t^{L,*}(x)| \frac{\Delta t}{\Delta x} \leq 1$, LeVeque (2007). Clearly, the Lipschitz regularization $|\nabla \phi_t^{L,*}(x)| \leq L$ enforces a CFL type condition with a learning rate Δt proportional to the inverse of L. It remains an open question how to rigorously extend these CFL-based heuristics to particle-based algorithms where the spatial discretization grid $\Delta x = L \Delta t$ is known only implicitly as noted in Remark 3.2(b), see also related questions in Carrillo et al. (2017). Intuitively though, this implicit spatio-temporal discretization suggests that $\sup_x |\nabla \phi_t^{L,*}(x)| \frac{\Delta t}{\Delta x} = \frac{\sup_x |\nabla \phi_t^{L,*}(x)|}{L} \leq 1$. Hence equation 33 or Algorithm 1 are expected to satisfy the same CFL condition for the transport PDE in equation 25. Based on these CFL heuristics for particles, here we keep the inversely proportional relation between L and Δt as a criterion for tuning the learning rate Δt.

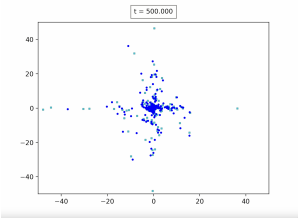
5 Generative particle algorithms for heavy-tailed data

Lipschitz regularized gradient flows in Section 2 and GPA in Section 3 are built on a family of f's which defines f-divergences as discussed in Section 1. In this Section, we study the role of the choice of f_{KL} and f_α on GPA for data samples from distributions with various tails, e.g. gaussian, stretched exponential, or polynomial.

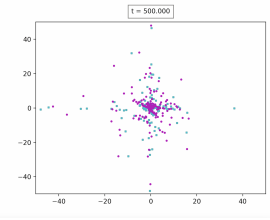
In our first experiment we observe the following. The choice of f_{KL} for heavy-tailed data renders the function optimization step in equation 33 numerically unstable and eventually leads to the collapse of the algorithm. On the other hand, the choice of f_α with $\alpha > 1$ makes the algorithm stable. However, it still takes a long time to transport particles deep into the heavy tails due to the speed restriction of the Lipshitz regularization. The different behaviors of f_{KL} and f_α on heavy-tailed data is illustrated in Figure 4.



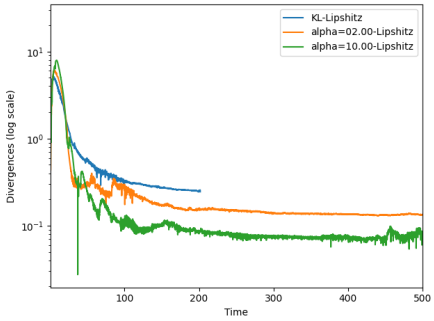
(a) f_{KL}, $L = 1$ collapses when the particle distribution become heavy-tailed



(b) f_α with $\alpha = 2$, $L = 1$



(c) f_α with $\alpha = 10$, $L = 1$



(d) (f, Γ_1)-divergences

Figure 4: **(Gaussian to Student-t with $\nu = 0.5$ in 2D)** We consider 200 initial samples from $N((10, 10), 0.5^2 I)$ which are transported towards 200 target samples from $Student - t(\nu)$ with $\nu = 0.5$ using (f, Γ_1)-GPA's. **(a)** $(f_{\text{KL}}, \Gamma_1)$-GPA collapses at around $t = 202$ as the function optimization step with f_{KL} is numerically unstable on heavy-tailed data. **(b - c)** (f_α, Γ_1)-GPA with $\alpha = 2, 10$ propagate particles in a stable manner during the entire simulation window. However, GPA still appears to take a long time to transport particles deep into the heavy tails due to the speed restriction of the Lipshitz regularization. This stability in performance that lacks in accuracy is manifested in (d) in the relatively large size of the α divergence **(d)** (f, Γ_1)-divergences computed by the corresponding estimator in equation 33 (see also Remark 3.7) are tracked in time.

Next, we explore the performance of GPA for several distributions with varying degrees of heavy-tailed structure. Initial distributions P_0 are chosen as heavy-tailed distributions in cases 1-4 in Table 1, whereas target distribution Q are chosen as heavy-tailed distributions in cases 5-8. We chose Generalized Gaussian distribution (Stretched exponential distribution, $GMM(\beta) \propto \exp(-|x|^\beta)$) with $\beta = 0.5$ as a heavy-tailed distribution because it fails to be subexponential. But it has finite moments of all orders. On the other hand, Student-t distributions with degree of freedom ν ($Student - t(\nu)$) have polynomial tails. Among them, $Student-t(3.0)$ has a finite second moment, $Student-t(1.5)$ has an infinite second moment but has a finite first moment, and $Student - t(0.5)$ has an infinite second moment but its first moment is undefined. In all cases in Table 1 we use the Gaussian distribution $N((10, 10), I)$ as either source or target. Table 1 displays the summary of the transportation of particles for different cases.

Overall, with the exception of especially heavy-tailed distributions in cases 3 & 4 (both with infinite second moments), KL and/or α-divergences work reasonably well. We also note that α-divergences in GANs for images can provide superior performance to KL and related divergences, even in the abscence of heavy tails Nowozin et al. (2016); Mao et al. (2017); Birrell et al. (2022a;c).

6 Learning from scarce data

In this section, we empirically demonstrate that GPA can be an effective generative model when only scarce target data is available. We analyze three examples: GPA for approximating a low-dimensional multi-scale distribution represented by scarce data, GPA for generating images in a high-dimensional space given scarce target data, and GPA for data augmentation. The mechanism of generating samples from GPA is by directly transporting source particles to the target by iteratively

Case	GPA source P_0	GPA target Q	$D_{\mathrm{KL}}^{\Gamma_1}$	$D_{\alpha}^{\Gamma_1}$ with $\alpha = 2$
1	$GGM(0.5)$	$\mathcal{N}((10,10), 0.5^2 I)$	$O(10^{-6})$	$O(10^{-6})$
2	$Student - t(3)$	$\mathcal{N}((10,10), 0.5^2 I)$	$O(10^{-4})$	$O(10^{-4})$
3	$Student - t(1.5)$	$\mathcal{N}((10,10), 0.5^2 I)$	**diverged at** $t = 0$	$O(10^{0})$
4	$Student - t(0.5)$	$\mathcal{N}((10,10), 0.5^2 I)$	**diverged at** $t = 0$	$O(10^{7})$
5	$\mathcal{N}((10,10), 0.5^2 I)$	$GGM(0.5)$	$O(10^{-6})$	$O(10^{-3})$
6	$\mathcal{N}((10,10), 0.5^2 I)$	$Student - t(3)$	$O(10^{-6})$	$O(10^{-4})$
7	$\mathcal{N}((10,10), 0.5^2 I)$	$Student - t(1.5)$	$O(10^{-3})$	$O(10^{-3})$
8	$\mathcal{N}((10,10), 0.5^2 I)$	$Student - t(0.5)$	**diverged at** $t = 202$	$O(10^{-1})$

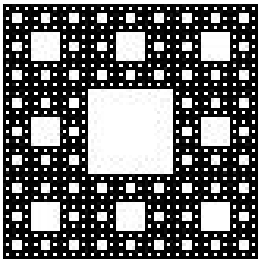
Table 1: Transportation of heavy-tails to Gaussian (cases 1-4) and Gaussian to heavy-tails (cases 5-8) by (f, Γ_1)-GPA with f_{KL} and f_α with $\alpha = 2$. When the algorithm collapses, the corresponding time is reported. In other cases, the converged $D_f^{\Gamma_1}(P_T, Q)$'s are reported.

solving equation 33, without having to learn a generator as GANs do, Goodfellow et al. (2014); Arjovsky et al. (2017); Birrell et al. (2022a), see also Section D. Here we demonstrate that GPA vastly outperforms GANs in the context of scarce data.

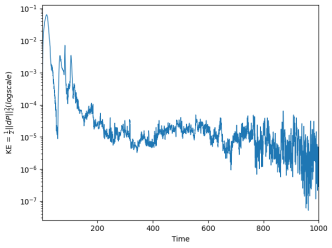
GPA for low-dimensional multi-scale distribution We consider a target distribution with a multi-scale (fractal) structure such as a Sierpinski carpet of level 4. This uniform distribution is constructed from a fractal set by keeping the 4 largest scales and truncating all finer scales. We refer to Figure 5a where we consider 4096 target particles in $[0, 10] \times [0, 10]$. Each target particle is random-sampled only once in each dark pixel with size of $[0, 10/3^4] \times [0, 10/3^4]$. We transport 4096 initial samples from $N(0, 3^2 I)$ using $(f_{\mathrm{KL}}, \Gamma_1)$-GPA. Figure 5b and 5c indicate that $(f_{\mathrm{KL}}, \Gamma_1)$-GPA approximates the target distribution and stops in a reasonable time $T = 1000$ with time steps $n = 5000$. We also refer to the related 3D result in Figure 1, where particles in 3D find a multi-scale structure in the 2D plane. Comparatively in Wasserstein GANs and $(f_{\mathrm{KL}}, \Gamma_1)$-GAN Arjovsky et al. (2017); Birrell et al. (2022a), training the generator for a multi-scale distribution with this amount of data turns out to be difficult, as we can readily see in Figure 5d to 5f.

GPA for image generation given scarce target data Here we consider the example of MNIST image generation using GPA, given a target data set that is relatively sparse compared to the corresponding spatial dimensionality. Recall the entire MNIST data set has $60,000$ images. We demonstrate an example of generating images for MNIST in $\mathbb{R}^{784}$ from 200 target samples in Figure 6. On the other hand, we observe that GANs cannot learn with the same amount of data. We argue that the failure of GANs is due to their additional generator learning step that requires more data, similarly to the previous example, see also the discussion in Section D. We demonstrate these points by comparing the results of $(f_{\mathrm{KL}}, \Gamma_1)$-GPA, $(f_{\mathrm{KL}}, \Gamma_1)$-GAN and Wasserstein-GAN side by side in Figure 6. Additional experiments that validate the diversity of the generated samples with $(f_{\mathrm{KL}}, \Gamma_1)$-GPA can be found in Section E and Figure 11. In these experiments the sample diversity of $(f_{\mathrm{KL}}, \Gamma_1)$-GPA is addressed by selecting the number of source samples to be larger than the target samples, because $(f_{\mathrm{KL}}, \Gamma_1)$-GPA is so efficient and accurate as a transport mechanism that, if $M = N$ in Algorithm 1, it would typically transport the N source particles almost directly on the N target particles, see Figure 12 in Section E. The same strategy regarding sample diversity is validated again, as well as used for data augmentation, in the swiss roll example discussed next.

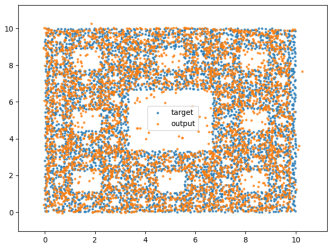
GPA for data augmentation GPA can provide a data augmentation tool for GANs and thus alleviate GANs' problematic behavior when a limited number of target samples is available as in Figure 6. Hence further generation of samples can be done with GANs in one-shot, once their training data is augmented by GPA. Augmented data obtained by GPA will be close to the empirical target distribution in (f, Γ_L)-divergence as suggested by Theorem 2.5. Other data augmentation techniques such as small noise injection or transformation do not a priori guarantee proximity to the target distribution. We demonstrate GPA-based data augmentation with a simple Swiss roll example in 3D in Figure 7. The number of target samples are limited to 200 which turn out to be not enough for generating the shape. Specifically, when $(f_{\mathrm{KL}}, \Gamma_1)$-GAN is trained on these samples, the loss decreases with large oscillations and the resulting generated samples are inaccurate. On the other hand,



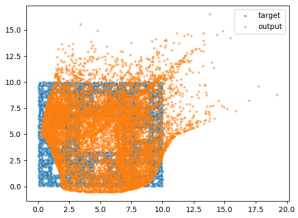
(a) Target distribution



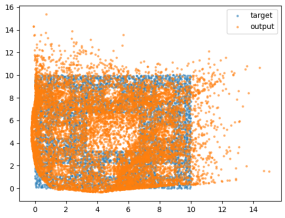
(b) Kinetic energy of particles equation 34 for $(f_{\mathrm{KL}}, \Gamma_1)$-GPA



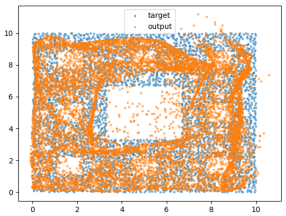
(c) Output of $(f_{\mathrm{KL}}, \Gamma_1)$-GPA



(d) Output of Wasserstein GAN (2-sided penalty)

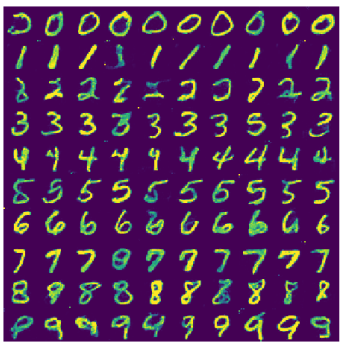


(e) Output of Wasserstein GAN (1-sided penalty)

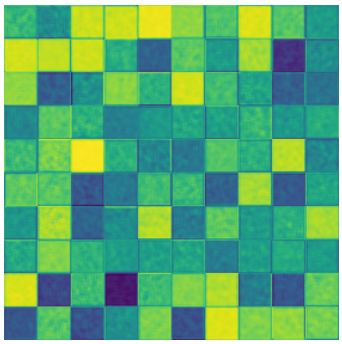


(f) Output of $(f_{\mathrm{KL}}, \Gamma_1)$-GAN

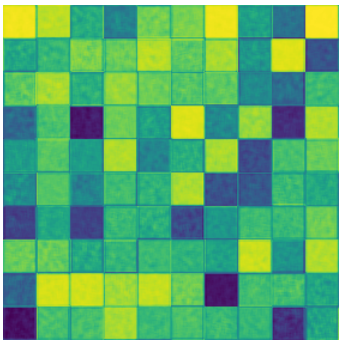
Figure 5: **Sierpinski carpet of level 4 - GPA for multi-scale distributions.** **(a)** We approximate a target distribution which consists of four different scales. **(b - c)** $(f_{\mathrm{KL}}, \Gamma_1)$-GPA can transport Gaussian samples to capture the three largest scales of the target and eventually terminates. **(d - f)** GANs show vastly inferior behavior compared to GPA for the same amount of data. **(f)** Note that the $(f_{\mathrm{KL}}, \Gamma_1)$-GAN is trained with the same loss function and the same NN structure for the discriminator, see Section D.



(a) $(f_{\mathrm{KL}}, \Gamma_1)$-GPA



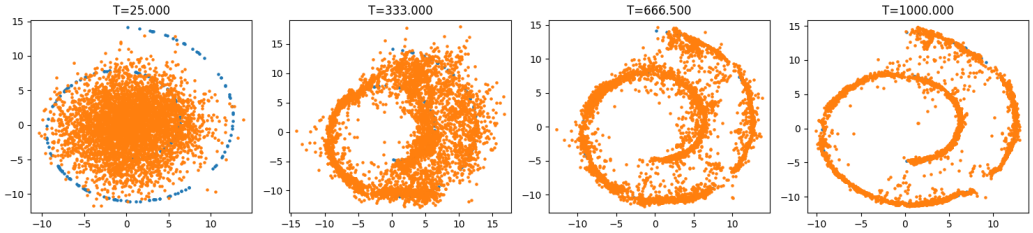
(b) $(f_{\mathrm{KL}}, \Gamma_1)$-GAN



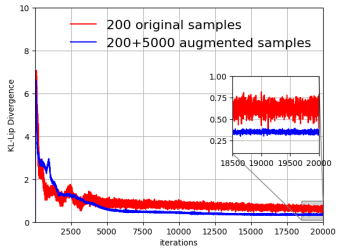
(c) Wasserstein-GAN

Figure 6: **(MNIST) Learning digits by different generative models with 200 training target samples.** The $(f_{\mathrm{KL}}, \Gamma_1)$-GPA was able to learn digits from this small data set, while the other methods failed. Additional experiments that validate the diversity of the generated samples with $(f_{\mathrm{KL}}, \Gamma_1)$-GPA can be found in Section E and Figure 11. On the other hand, GANs still failed to generate any digit-like samples with 600 training samples, but were barely successful with 2000 training samples. In our comparison we maintained the same discriminator architecture for all methods. In particular, the $(f_{\mathrm{KL}}, \Gamma_1)$-GAN is trained with both the same divergence and the same NN structure for the discriminator as the $(f_{\mathrm{KL}}, \Gamma_1)$-GPA but was still not able to generate any meaningful samples.

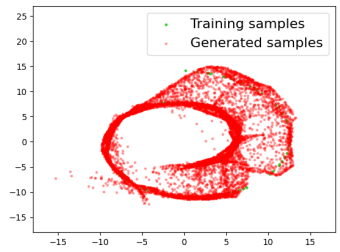
from these 200 particles we generate 5000 additional samples using $(f_{\mathrm{KL}}, \Gamma_1)$-GPA. Then we select the newly generated particles at $T = 1000$ with $D_{f_{\mathrm{KL}}}^{\Gamma_1}(P_T \| Q) \leq 1.506e - 3$ as our augmented data. These particles contain some noise but have enough diversity to be fairly densely distributed



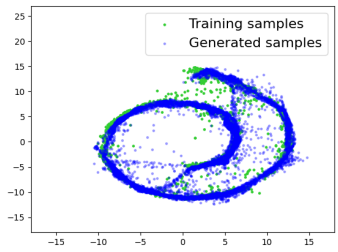
(a) Trajectory of $(f_{\mathrm{KL}}, \Gamma_1)$-GPA. **(blue)** 200 target samples, **(orange)** 5000 transported particles



(b) $(f_{\mathrm{KL}}, \Gamma_1)$-divergences



(c) without data augmentation



(d) with data augmentation

Figure 7: **(Swiss roll) Data augmentation using GPA. (a)** Given 200 samples from the Swiss roll uniform distribution Q, 5000 additional samples are generated by transporting initial samples from $P = \mathcal{N}(0, 3^2 I)$ using $(f_{\mathrm{KL}}, \Gamma_1)$-GPA. Particles at time $T = 1000$ with $D_{f_{\mathrm{KL}}}^{\Gamma_1}(P_T \| Q) = 1.506e - 3$ are used as the augmented data. **(b - d)** When $(f_{\mathrm{KL}}, \Gamma_1)$-GAN is trained from 200 original samples **(red)**, the loss (divergence) oscillates, see inset in (b). Furthermore the generated particles are not sufficiently accurate, reconstructing an incorrect manifold in (c). To improve the GAN, we train it with 200 original samples + 5000 augmented samples **(blue)**. By GPA-data augmentation, the GAN loss decreases stably, see inset in (b), and the generated samples provide an improved representation of the Swiss roll manifold in (d).

to cover the manifold. Indeed, similarly to the previous MNIST example, we obtain sufficient sample diversity in particles generated with $(f_{\mathrm{KL}}, \Gamma_1)$-GPA, by selecting the $5,000$ source samples to be significantly larger than the 200 target samples. Subsequently the $(f_{\mathrm{KL}}, \Gamma_1)$-GAN is trained by including these 5000 augmented data, showing a stable decrease of loss as well as improved sample quality.

7 Latent-space GPA for high-dimensional dataset integration

The integration of two or more datasets that essentially contain the same information yet their statistical properties are different due to e.g., distributional shifts is crucial for the successful training and deployment of statistical and machine learning models Kirk et al. (2012); Hendler (2014); Samuelsen et al. (2019). Taking bioinformatics as an example, the available datasets, even when they study the same disease, have been created from different labs around the globe resulting in statistical differences which are also known as batch effects Tran et al. (2020). Furthermore, those datasets typically have low sample size due to budget constraints or limited availability of patients (e.g., rare diseases). GPA offers an elegant solution for dataset integration since it does not impose any assumption regarding the source and target distributions and it can produce stable and accurate results even for very small sample size as it has been already shown in Section 6. However, GPA cannot be straightforwardly applied when the dimensionality of the data is in the order of tens of thousands. Therefore, before applying GPA, we significantly reduce the dimensionality of the data. Then apply GPA in the latent space and, if required, reconstruct the transported particles back to the original space. This three-step approach makes GPA to correctly transport the particles from the latent source to the latent target distribution because the transportation is performed in a low dimensional space. However, it may impose additional errors which could eventually result in poor dataset integration. Therefore, we further prove (see Theorem 7.1 below) that when the target dataset is supported on a lower dimensional manifold, it is theoretically guaranteed via a new data processing

inequality that the error in the original space can be bounded by the error occurred in the latent space.

Gene expression data and dataset integration problem We consider the integration of two gene expression datasets which are publicly available at `https://www.ncbi.nlm.nih.gov/geo/` with accession codes GSE76275 and GSE26639. These dataset have been measured using the GLP570 platform which creates samples with $d = 54,675$ dimensions. Furthermore, each dataset consists of a low number of data while each individual sample corresponds to gene expression levels of a patient and it is labeled by a clinical indicator which informs if the patient was positive or negative to ER (estrogen receptor). The number of positive and negative samples are provided in

	Positive	Negative	Total
GSE26639 (source)	138	88	226
GSE76275 (target)	49	216	265

Table 2: Sample sizes of the gene expression studied datasets.

Table 2. Despite measuring the same quantities, a direct concatenation of the two datasets will result in erroneous statistics as it is evident from Figure 9a where a 2D visualization reveals that the two datasets are completely separated. To tackle this issue, we propose transporting one of the datasets via GPA in a latent space as an approach to integrate high-dimensional gene expression datasets. Next, we present how to construct a low dimensional latent space from the data.

Dimensionality reduction using PCA Applying GPA and essentially any machine learning algorithm in the original high dimensional space is neither efficient nor generalizable. Hence, we apply dimensionality reduction and construct a latent space and perform GPA in the latent space. There are linear dimensionality reduction algorithms such as PCA (Jolliffe & Cadima, 2016; Hastie et al., 2001, Ch. 14.5) as well nonlinear dimensionality reduction algorithms such as neural network autoencoders Pantazis et al. (2020). In the general setting, an autoencoder is a pair of two functions with the first being the encoder, denoted by $\mathcal{E}(\cdot)$, which compresses the information from a high-dimensional space to a lower dimensional latent space and the second being the decoder, denoted by $\mathcal{D}(\cdot)$, which decompresses the latent features back to the original space. Since a neural-based autoencoder requires tens of thousands of samples for its training, we utilize the PCA algorithm on the concatenated dataset for the reduction of the dimensionality of the data.

We first standardize the entire dataset to have mean 0 and standard deviation 1. Mathematically, this standardization is written as $s(x_i) = (x_i - \mu_i)/\sigma_i$ for $i = 1, \cdots, d$ where μ_i and σ_i are the mean and standard deviation of the i-th element. PCA only requires centered mean however, we observe that GPA leads to faster convergence when standard deviation is also normalized therefore we apply both. PCA computes an orthonormal basis via the singular value decomposition on the data covariance matrix. Each eigenvalue λ_i is interpreted as the variance from the corresponding eigenvector $\mathbf{v_i} \in \mathbb{R}^d$. The latent features are obtained by projecting the data into the d' eigenvectors with the highest eigenvalue values. We determine the dimension of the latent space, denoted by d', according to the *explained variance ratio* which is defined as $\sum_{i=1}^{d'} \lambda_i / \sum_{i=1}^{d} \lambda_i$. Figure 8 shows the explained variance ratio as a function of d'. We choose to keep 90% of the explained variance which resulted in $d' = 50$.

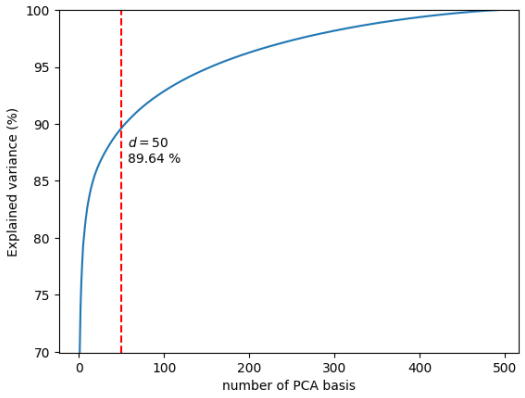
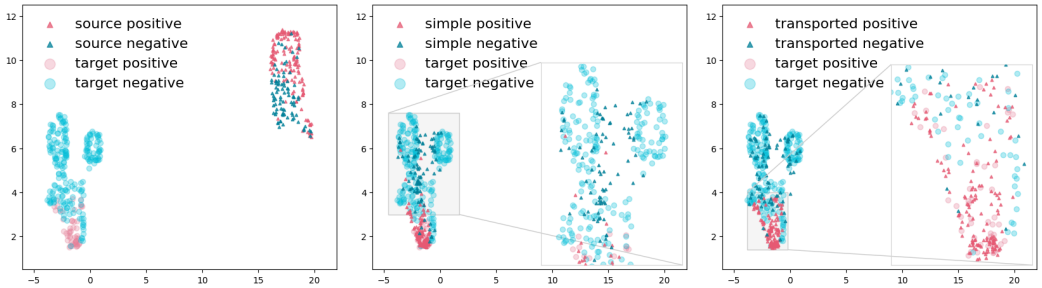


Figure 8: 89.64% of the variance is kept when the latent dimension is set to $d' = 50$.

Results We integrate gene expression datasets by applying the latent-space GPA. Positive-labeled and negative-labeled gene expression datasets are integrated separately so that the labels are preserved after integration. The transportation maps $\mathcal{T}^{n,+}$ and $\mathcal{T}^{n,-}$ are defined as compositions of

$(f_{\mathrm{KL}}, \Gamma_1)$-GPA transport maps defined in equation 32 for $n = 5000$ time steps (and $\Delta t = 0.2$). The visualization of the data set integration in Figure 9c shows that both positive and negative distributions have been efficiently transported via latent-space GPA. As a comparison, we present a baseline data transformation for each class, denoted by $\mathcal{F}^+$ and $\mathcal{F}^-$, respectively, which performs mean and standard deviation adjustment. As it is evident in Figure 9b, the baseline dataset integration fails to relocate the samples from the transformed distribution to the target distribution, which is especially pronounced in the negative samples (see inset in Figure 9b). For simplicity, we drop the symbols $+$ & $-$ from the notation.

We additionally calculate the difference between the transported and target distributions and report the 2-Wasserstein distance in Table 3. We opt to quantify the differences via the 2-Wasserstein distance because it is a metric not used in the latent GPA and it can be computed very efficiently using the Sinkhorn algorithm. Evidently, the error between datasets in the original space ($d = 54,675$) is reduced by two orders of magnitude, while GPA is twice as effective compared to the mean and std adjustment transformation (rows 8 & 9 in Table 3). Finally, we remark that the error as measured by the distance in the latent space is higher for the negative class (row 2 in Table 3). This can be partially attributed to the smaller sample size of the source compared to the target. Interestingly, the relative ordering in the distance is also observed in the original space (row 6 in Table 3). The following paragraph makes the observations in Table 3 rigorous through a new data processing inequality for general auto-encoders.



(a) Two gene expression data sets without any transformation.

(b) Dataset integration using mean and std adjustment.

(c) Dataset integration using latent $(f_{\mathrm{KL}}, \Gamma_1)$-GPA with $d' = 50$.

Figure 9: **Gene expression dataset integration by GPA.** We integrate two high-dimensional gene expression datasets via GPA transportation. **(a)** A direct concatenation of the two datasets results in incorrect integration as visualized in the 2D plane using UMAP McInnes et al. (2018). **(b)** The baseline approach consists of a mean and std adjustment of each feature in the original space. In the inset, we notice that transformed negative samples do not evenly cover the support of the negative target samples. **(c)** The proposed latent GPA data transportation results in transported distributions close to the target ones.

Data Processing Inequality and latent space GPA Motivated by the findings in Table 3, we present in Theorem 7.1 a theoretical result guaranteeing that the error induced by the transportation of a high-dimensional data distribution via combined encoding/decoding and particle transportation in a lower dimensional latent space is controlled by the error only in the (much more tractable) latent space. The corresponding errors are measured by Lipschitz-regularized divergences in the real and the latent space respectively. However, we believe the results of the theorem should also hold true for the Sinkhorn divergence deployed in Table 3, based in part on the analysis for the Sinkhorn divergence in Birrell et al. (2022c) and elements from the proof of Theorem A.2. More precisely, consider the following elements: i) a probability $Q = Q^{\mathcal{Y}}$, defined on the original, high dimensional space $\mathcal{Y}$, typically supported on some low dimensional set $S \subset \mathcal{Y} = \mathbb{R}^d$; ii) an encoder map $\mathcal{E} : \mathcal{Y} \to \mathcal{Z}$ where $\mathcal{Z} \subset \mathbb{R}^{d'}$, $d' < d$ and a decoder map $\mathcal{D} : \mathcal{Z} \to \mathcal{Y}$ which are invertible in S, i.e. $\mathcal{D} \circ \mathcal{E}(S) = S$. Let $\mathcal{E}_{\#}Q^{\mathcal{Y}}$ denote the image of the measure $Q^{\mathcal{Y}}$ by the map $\mathcal{E}$, i.e. for $A \subset \mathcal{Z}$, $\mathcal{E}_{\#}Q^{\mathcal{Y}}(A) := Q^{\mathcal{Y}}(\mathcal{E}^{-1}(A))$ and similarly we define $\mathcal{D}_{\#}P^{\mathcal{Z}}$. For instance, the PCA based on Figure 8 provides a linear autoencoder with a linear encoder $\mathcal{E}$ and a linear decoder $\mathcal{D}$.

Then, Theorem 7.1 expresses how information remains controlled under encoding/decoding and guarantees the performance of the numerical approximation $\mathcal{D}_{\#}P^{\mathcal{Z}}$ in the real space. We recall

Metric	Explanation	Positive	Negative
$W_2(\mathcal{E}_\# P_0, \mathcal{E}_\# Q)$	Distance between source and target in latent space	8.4320e+4	8.3507e+4
$W_2(\mathcal{T}^n_\# \mathcal{E}_\# P_0, \mathcal{E}_\# Q)$	Distance between transported and target in latent space	3.9939e+2	2.0675e+3
$W_2(\mathcal{D}_\# \mathcal{E}_\# Q, Q)$	Reconstruction error of the target distribution (R)	2.9768e+3	3.3805e+3
$W_2(P_0, Q)$	Distance between source and target in original space (D)	2.3883e+5	2.3845e+5
$W_2(\mathcal{F}_\# P_0, Q)$	Distance between mean & std adjusted source and target in original space (S)	9.4402e+3	1.1617e+4
$W_2(\mathcal{D}_\# \mathcal{T}^n_\# \mathcal{E}_\# P_0, Q)$	Distance between transported and target in original space (G)	3.5171e+3	6.2247e+3
(R)/(D)	The least relative distance that can be attained by latent GPA	1.2464e-2	1.4176e-2
(S)/(D)	Relative distance attained by mean & std adjusted source	3.9526e-2	4.8718e-2
(G)/(D)	Relative distance attained by latent GPA	**1.4726e-2**	**2.6104e-2**

Table 3: We compute the 2-Wasserstein distance between datasets in both the latent space ($d' = 50$) and in the original space ($d = 54,675$). 2-Wasserstein distance is approximated by Sinkhorn divergence Genevay et al. (2017); Feydy (2020). For each class, original source P_0 and target Q are determined by finite number of samples in Table 2. The reconstruction error due to PCA dimensionality reduction is below 1.5% as quantified by the ratio (R)/(D). $\mathcal{T}^n$ is the composition of $(f_{\text{KL}}, \Gamma_1)$-GPA transport maps defined in equation 32 for n time steps while $\mathcal{F}$ denotes the baseline transformation which adjusts the mean & std of the source to the mean & std of the target distribution. Dataset integration via latent GPA is approximately twice as effective compared to dataset integration via the baseline data transformation, as can be readily observed in their respective ratios (G)/(D) and (S)/(D).

that $\mathcal{D}_\# P^{\mathcal{Z}}$ is constructed via combined encoding/decoding and particle transportation in a lower dimensional latent space, i.e.

$$P^{\mathcal{Z}} := \mathcal{T}^n_\# \mathcal{E}_\# P_0$$

in the notation of Table 3. The fidelity of the approximation $Q^{\mathcal{Y}} \approx \mathcal{D}_\# P^{\mathcal{Z}}$ of the target measure $Q^{\mathcal{Y}}$ in the original, real space $\mathcal{Y}$ will be then guaranteed by an *a posteriori* estimate equation 36, interpreted in the sense of numerical analysis, where the approximation in the tractable latent space $\mathcal{Z}$ will bound the error in the real space $\mathcal{Y}$. Its proof is a consequence of a new, tighter data processing inequality derived in Birrell et al. (2022a), see also Theorem A.2, that involves both transformations of probabilities and discriminator spaces Γ. Indeed, we have the following.

Theorem 7.1 (Autoencoder performance guarantees)**.** *For $Q^{\mathcal{Y}} \in \mathcal{P}(\mathcal{Y})$, suppose that there is a exact encoder/decoder with encoder $\mathcal{E} : \mathbb{R}^d \to \mathbb{R}^{d'}$ and decoder $\mathcal{D} : \mathbb{R}^{d'} \to \mathbb{R}^d$, where exact means that $\mathcal{D}_\# \mathcal{E}_\# Q^{\mathcal{Y}} = Q^{\mathcal{Y}}$. Furthermore, assume the decoder is Lipschitz continuous with Lipschitz constant $a_{\mathcal{D}}$. Then, for any $P^{\mathcal{Z}} \in \mathcal{P}_1(\mathcal{Z})$ we have*

$$D_f^{\Gamma_L}(\mathcal{D}_\# P^{\mathcal{Z}} \| Q^{\mathcal{Y}}) \leq D_f^{a_{\mathcal{D}}\Gamma_L}(P^{\mathcal{Z}} \| \mathcal{E}_\# Q^{\mathcal{Y}}). \tag{36}$$

Proof. From data processing inequality Theorem A.2 and using that the composition of Lipschitz functions with Lipschitz constants L_1, L_2 is $L_1 L_2$-Lipschitz, we have:

$$D_f^{\Gamma_L}(\mathcal{D}_\# P^{\mathcal{Z}} \| \mathcal{D}_\# \mathcal{E}_\# Q^{\mathcal{Y}}) \leq D_f^{a_{\mathcal{D}}\Gamma_L}(P^{\mathcal{Z}} \| \mathcal{E}_\# Q^{\mathcal{Y}}). \tag{37}$$

Since the encoder $\mathcal{E}$ and the decoder $\mathcal{D}$ perfectly reconstruct $Q^{\mathcal{Y}}$, namely $\mathcal{D}_\# \mathcal{E}_\# Q^{\mathcal{Y}} = Q^{\mathcal{Y}}$, we obtain that

$$D_f^{\Gamma_L}(\mathcal{D}_\# P^{\mathcal{Z}} \| Q^{\mathcal{Y}}) \leq D_f^{a_{\mathcal{D}}\Gamma_L}(P^{\mathcal{Z}} \| \mathcal{E}_\# Q^{\mathcal{Y}}). \tag{38}$$

Note also that, if $a_{\mathcal{D}} \leq 1$,

$$D_f^{\Gamma_L}(\mathcal{D}_\# P^{\mathcal{Z}} \| \mathcal{D}_\# \mathcal{E}_\# Q^{\mathcal{Y}}) \leq D_f^{\Gamma_L}(P^{\mathcal{Z}} \| \mathcal{E}_\# Q^{\mathcal{Y}}). \tag{39}$$

□

Remark 7.2 (Practicalities of Theorem 7.1). The assumption of an "exact" encoder/decoder in Theorem 7.1 is a mathematical idealization. Practically, in our PCA example in Figure 8 we selected the number of basis elements $d' = 50$ in order to reduce the error between the reconstructed target data set and the original target data set and obtain a reasonable approximation of an exact encoder/decoder. See also the relative reconstruction error (R)/(D) in Table 3. Furthermore, the decoder $\mathcal{D}$ obtained by PCA is Lipschitz continuous with the Lipschitz constant $a_{\mathcal{D}} = 1$, in the notation of Theorem 7.1, since $\|\mathcal{D}(\mathbf{z}) - \mathcal{D}(\mathbf{z}')\|^2 = \|\sum_{i=1}^{d'}(z_i - z_i')\mathbf{v}_i\|^2 = \sum_{i=1}^{d'}|z_i - z_i'|^2\|\mathbf{v}_i\|^2 = \|\mathbf{z} - \mathbf{z}'\|^2$, using that the $\mathbf{v}_i$'s are orthonormal. In the case where the PCA decoder is composed with the inverse of the element-wise standardization $s^{-1}(x_i) = \sigma_i x_i + \mu_i$, the Lipschitz constant $a_{\mathcal{D}} = \max_{i=1,\cdots,d}\sigma_i$.

Remark 7.3 (Autoencoder guarantees in generative modeling). It is clear that Theorem 7.1 is a result about autoencoders and it is independent of the choice of any specific transport/generation algorithm in the latent space. In this sense our conclusions from Theorem 7.1 are generally applicable to any other latent space methods for generative modeling, for instance stable diffusion Rombach et al. (2021b).

REFERENCES

Luigi Ambrosio, Nicola Gigli, and Giuseppe Savaré. *Gradient flows: in metric spaces and in the space of probability measures*. Springer Science & Business Media, 2005.

Michael Arbel, Anna Korba, Adil Salim, and Arthur Gretton. Maximum mean discrepancy gradient flow. *Advances in Neural Information Processing Systems*, 32, 2019.

Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In Doina Precup and Yee Whye Teh (eds.), *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 214–223. PMLR, 06–11 Aug 2017. URL https://proceedings.mlr.press/v70/arjovsky17a.html.

Jeremiah Birrell, Paul Dupuis, Markos A Katsoulakis, Yannis Pantazis, and Luc Rey-Bellet. (f-γ)-divergences: Interpolating between f-divergences and integral probability metrics. *Journal of Machine Learning Research*, 23(39):1–70, 2022a.

Jeremiah Birrell, Markos A. Katsoulakis, and Yannis Pantazis. Optimizing variational representations of divergences and accelerating their statistical estimation. *IEEE Transactions on Information Theory*, pp. 1–1, 2022b. doi: 10.1109/TIT.2022.3160659.

Jeremiah Birrell, Markos A. Katsoulakis, Luc Rey-Bellet, and Wei Zhu. Structure-preserving gans. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvári, Gang Niu, and Sivan Sabato (eds.), *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pp. 1982–2020. PMLR, 2022c. URL https://proceedings.mlr.press/v162/birrell22a.html.

Nicholas M. Boffi and Eric Vanden-Eijnden. Probability flow solution of the Fokker-Planck equation. *arXiv e-prints*, art. arXiv:2206.04642, June 2022.

José Antonio Carrillo, Yanghong Huang, Francesco Saverio Patacchini, and Gershon Wolansky. Numerical study of a particle method for gradient flows. *Kinetic and Related Models*, 10(3): 613–641, 2017.

Changyou Chen, Chunyuan Li, Liqun Chen, Wenlin Wang, Yunchen Pu, and Lawrence Carin Duke. Continuous-time flows for efficient inference and density estimation. In Jennifer Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 824–833. PMLR, 10–15 Jul 2018a. URL https://proceedings.mlr.press/v80/chen18d.html.

Ricky T. Q. Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural ordinary differential equations. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018b. URL https://proceedings.neurips.cc/paper/2018/file/69386f6bb1dfed68692a24c8686939b9-Paper.pdf.

Jean Dolbeault, Ivan Gentil, Arnaud Guillin, and Feng-Yu Wang. L^q-functional inequalities and weighted porous media equations. *Potential Anal.*, 28(1):35–59, 2008. ISSN 0926-2601. doi: 10.1007/s11118-007-9066-0.

Paul Dupuis and Yixiang Mao. Formulation and properties of a divergence used to compare probability measures without absolute continuity. *ESAIM: Control, Optimisation and Calculus of Variations*, 28:10, 2022.

Alain Durmus and Eric Moulines. Nonasymptotic convergence analysis for the unadjusted Langevin algorithm. *The Annals of Applied Probability*, 27(3):1551 – 1587, 2017. doi: 10.1214/16-AAP1238.

Jean Feydy. Geometric data analysis, beyond convolutions, 2020. URL https://www.jeanfeydy.com/geometric_data_analysis.pdf.

Aude Genevay, Marco Cuturi, Gabriel Peyré, and Francis Bach. Stochastic optimization for large-scale optimal transport. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016. URL https://proceedings.neurips.cc/paper/2016/file/2a27b8144ac02f67687f76782a3b5d8f-Paper.pdf.

Aude Genevay, Gabriel Peyré, and Marco Cuturi. Learning generative models with sinkhorn divergences, 2017.

Pierre Glaser, Michael Arbel, and Arthur Gretton. Kale flow: A relaxed kl gradient flow for probabilities with disjoint support. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 8018–8031. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper/2021/file/433a6ea5429d6d75f0be9bf9da26e24c-Paper.pdf.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein gans. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper/2017/file/892c3b1c6dccd52936e27cbd0ff683d6-Paper.pdf.

Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning*. Springer Series in Statistics. Springer New York Inc., New York, NY, USA, 2001.

James Hendler. Data integration for heterogenous datasets. *Big Data*, 2(4):205–215, December 2014.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models, 2020.

Aapo Hyvärinen. Estimation of non-normalized statistical models by score matching. *J. Mach. Learn. Res.*, 6:695–709, December 2005. ISSN 1532-4435. URL http://dl.acm.org/citation.cfm?id=1046920.1088696.

Ian Jolliffe and Jorge Cadima. Principal component analysis: A review and recent developments. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 374:20150202, 04 2016. doi: 10.1098/rsta.2015.0202.

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

Diederik P Kingma and Max Welling. Auto-Encoding Variational Bayes. *arXiv e-prints*, art. arXiv:1312.6114, December 2013.

Paul Kirk, Jim E Griffin, Richard S Savage, Zoubin Ghahramani, and David L Wild. Bayesian correlated clustering to integrate multiple datasets. *Bioinformatics*, 28(24):3290–3297, October 2012.

Jonas Köhler, Leon Klein, and Frank Noe. Equivariant flows: Exact likelihood generative learning for symmetric densities. In Hal Daumé III and Aarti Singh (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 5361–5370. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/kohler20a.html.

Yann LeCun, Sumit Chopra, Raia Hadsell, Marc'Aurelio Ranzato, and Fu-Jie Huang. A tutorial on energy-based learning. In G. Bakir, T. Hofman, B. Schölkopf, A. Smola, and B. Taskar (eds.), *Predicting Structured Data*. MIT Press, 2006.

Randall LeVeque. *Finite Difference Methods for Ordinary and Partial Differential Equations: Steady-State and Time-Dependent Problems (Classics in Applied Mathematics Classics in Applied Mathemat)*. Society for Industrial and Applied Mathematics, USA, 2007. ISBN 0898716292.

Qiang Liu. Stein variational gradient descent as gradient flow. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper/2017/file/17ed8abedc255908be746d245e50263a-Paper.pdf.

Qiang Liu and Dilin Wang. Stein variational gradient descent: A general purpose bayesian inference algorithm. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016. URL https://proceedings.neurips.cc/paper/2016/file/b3ba8f1bee1238a2f37603d90b58898d-Paper.pdf.

Jianfeng Lu, Yulong Lu, and James Nolen. Scaling limit of the stein variational gradient descent: The mean field regime. *SIAM Journal on Mathematical Analysis*, 51(2):648–671, 2019. doi: 10.1137/18M1187611. URL https://doi.org/10.1137/18M1187611.

X. Mao, Q. Li, H. Xie, R. K. Lau, Z. Wang, and S. Smolley. Least squares generative adversarial networks. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 2813–2821, Los Alamitos, CA, USA, oct 2017. IEEE Computer Society. doi: 10.1109/ICCV.2017.304. URL https://doi.ieeecomputersociety.org/10.1109/ICCV.2017.304.

Dimitra Maoutsa, Sebastian Reich, and Manfred Opper. Interacting particle solutions of fokker–planck equations through gradient–log–density estimation. *Entropy*, 22(8), 2020. ISSN 1099-4300. doi: 10.3390/e22080802. URL https://www.mdpi.com/1099-4300/22/8/802.

Peter A Markowich and Cédric Villani. On the trend to equilibrium for the fokker-planck equation: an interplay between physics and functional analysis. *Mat. Contemp*, 19:1–29, 2000.

Leland McInnes, John Healy, Nathaniel Saul, and Lukas Grossberger. Umap: Uniform manifold approximation and projection. *Journal of Open Source Software*, 3(29):861, 2018. doi: 10.21105/joss.00861. URL https://doi.org/10.21105/joss.00861.

Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. 02 2018.

Youssef Mroueh, Tom Sercu, and Anant Raj. Sobolev descent. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 2976–2985. PMLR, 2019.

Sebastian Nowozin, Botond Cseke, and Ryota Tomioka. F-GAN: Training Generative Neural Samplers Using Variational Divergence Minimization. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, pp. 271–279, Red Hook, NY, USA, 2016. Curran Associates Inc. ISBN 9781510838819.

F. Otto and C. Villani. Generalization of an inequality by talagrand and links with the logarithmic sobolev inequality. *Journal of Functional Analysis*, 173(2):361–400, 2000. ISSN 0022-1236. doi: https://doi.org/10.1006/jfan.1999.3557. URL https://www.sciencedirect.com/science/article/pii/S0022123699935577.

Felix Otto. The geometry of dissipative evolution equations: the porous medium equation. *Comm. Partial Differential Equations*, 26(1-2):101–174, 2001. ISSN 0360-5302. doi: 10.1081/PDE-100002243.

Yannis Pantazis, Christos Tselas, Kleanthi Lakiotaki, Vincenzo Lagani, and Ioannis Tsamardinos. Latent feature representations for human gene expression data improve phenotypic predictions. In *2020 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pp. 2505–2512, 2020. doi: 10.1109/BIBM49941.2020.9313286.

Sebastian Reich and Simon Weissmann. Fokker–planck particle systems for bayesian inference: Computational approaches. *SIAM/ASA Journal on Uncertainty Quantification*, 9(2):446–482, 2021. doi: 10.1137/19M1303162.

Gareth O. Roberts and Richard L. Tweedie. Exponential convergence of Langevin distributions and their discrete approximations. *Bernoulli*, 2(4):341 – 363, 1996. doi: bj/1178291835. URL https://doi.org/.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. *CoRR*, abs/2112.10752, 2021a. URL https://arxiv.org/abs/2112.10752.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. *CoRR*, abs/2112.10752, 2021b. URL https://arxiv.org/abs/2112.10752.

Jeanette Samuelsen, Weiqin Chen, and Barbara Wasson. Integrating multiple data sources for learning analytics—review of literature. *Research and Practice in Technology Enhanced Learning*, 14 (1):11, Aug 2019. ISSN 1793-7078. doi: 10.1186/s41039-019-0105-4.

Jascha Sohl-Dickstein, Eric A. Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics, 2015.

Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution, 2020.

Yang Song, Jascha Narain Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *ArXiv*, abs/2011.13456, 2021.

Marc Stieffenhofer, Tristan Bereau, and Michael Wand. Adversarial reverse mapping of condensed-phase molecular structures: Chemical transferability. *APL Materials*, 9:031107, 03 2021. doi: 10.1063/5.0039102.

Guiseppe Toscani and Cédric Villani. On the trend to equilibrium for some dissipative systems with slowly increasing a priori bounds. *Journal of Statistical Physics*, 98(5):1279–1309, 2000.

Hoa Thi Nhu Tran, Kok Siong Ang, Marion Chevrier, Xiaomeng Zhang, Nicole Yee Shin Lee, Michelle Goh, and Jinmiao Chen. A benchmark of batch-effect correction methods for single-cell rna sequencing data. *Genome Biology*, 21(1):12–12, 2020. ISSN 1474-760X.

Juan Luis Vázquez. Barenblatt solutions and asymptotic behaviour for a nonlinear fractional heat equation of porous medium type. *Journal of the European Mathematical Society*, 16(4):769–803, 2014.

Pantelis Vlachas, Georgios Arampatzis, Caroline Uhler, and Petros Koumoutsakos. Multiscale simulations of complex systems by learning their effective dynamics. *Nature Machine Intelligence*, 03 2022. doi: 10.48550/arXiv.2006.13431.

Feng-Yu Wang. A generalization of Poincaré and log-Sobolev inequalities. *Potential Anal.*, 22(1): 1–15, 2005. ISSN 0926-2601. doi: 10.1007/s11118-004-4006-8.

Wujie Wang and Rafael Gómez-Bombarelli. Coarse-graining auto-encoders for molecular dynamics. *npj Computational Materials*, 5:125, 12 2019. doi: 10.1038/s41524-019-0261-5.

SUPPLEMENTARY MATERIALS

A BACKGROUND ON LIPSCHITZ-REGULARIZED DIVERGENCES

In the paper Dupuis & Mao (2022), continuing with Birrell et al. (2022a) a new general class of divergences has been constructed which interpolate between f-divergences and integral probability metrics and inherit desirable properties from both. We focus here one specific family which we view as a Lipschitz regularization of the KL-divergence (or f-divergences) or as an entropic regularization of the 1-Wasserstein metric. We denote by $\mathcal{P}(\mathbb{R}^d)$ the space of all Borel probability measures on $\mathbb{R}^d$ by $\mathcal{P}_1(\mathbb{R}^d) = \left\{P \in \mathcal{P}(\mathbb{R}^d) : \int |x| dP(x) < \infty\right\}$. We denote by $C_b(\mathbb{R}^d)$ the bounded continuous function and by $\Gamma_L = \{f : \mathbb{R}^d \to \mathbb{R} : |f(x) - f(y)| \leq L|x-y| \text{ for all } x, y\}$ the Lipschitz continous functions with Lipschitz constant bounded by L (note that $a\Gamma_L = \Gamma_{aL}$).

f-divergences If $f : [0, \infty) \to \mathbb{R}$ is strictly convex and lower-semicontinuous with $f(1) = 0$ the f-divergence of P with respect to Q is defined by $D_f(P\|Q) = E_Q[f(\frac{dP}{dQ})]$ if $P \ll Q$ and set to be $+\infty$ otherwise. We have the variational representation (see e.g. Birrell et al. (2022a) for a proof)

$$D_f(P\|Q) = \sup_{\phi \in C_b(\mathbb{R}^d)} \left\{ E_P[\phi] - \inf_{\nu \in \mathbb{R}} \{\nu + E_Q[f^*(\phi - \nu)]\} \right\} \tag{40}$$

where $f^*(s) = \sup_{t \in \mathbb{R}} \{st - f(t)\}$ is the Legendre-Fenchel transform of f. We will use the KL-divergence with $f_{\mathrm{KL}}(x) = x \log x$ and the α-divergence: $f_\alpha = \frac{x^\alpha - 1}{\alpha(\alpha-1)}$ with Legendre transforms $f^*_{\mathrm{KL}}(y) = e^{y-1}$ and $f^*_\alpha \propto y^{\frac{\alpha}{(\alpha-1)}}$. For KL the infimum over ν can be solved analytically and yields the Donsker-Varadhan with a $\log E_Q[e^\phi]$ term (see Birrell et al. (2022b) for more on variational representations).

Wasserstein metrics The 1-Wasserstein metrics $W^{\Gamma_1}(P, Q)$ with transport cost $|x - y|$ is an integral probability metrics, see Arjovsky et al. (2017). By keeping the Lipschitz constant as a regularization parameter we set

$$W^{\Gamma_L}(P, Q) = \sup_{\phi \in \Gamma_L} \{E_P[\phi] - E_Q[\phi]\} \tag{41}$$

and note that we have $W^{\Gamma_L}(P, Q) = LW^{\Gamma_1}(P, Q)$.

Lipschitz-regularized f-divergences The Lipschitz regularized f-divergences are defined directly in terms their variational representations, by replacing the optimization over bounded continuous functions in equation 40 by Lipschitz continuous functions in Γ_L.

$$D_f^{\Gamma_L}(P\|Q) := \sup_{\phi \in \Gamma_L} \left\{ E_P[\phi] - \inf_{\nu \in \mathbb{R}} \{\nu + E_Q[f^*(\phi - \nu)]\} \right\}. \tag{42}$$

Some of the important properties of Lipschitz regularized f-divergences, which summarizes results from Dupuis & Mao (2022); Birrell et al. (2022a) are given in Theorem A.1. It is assumed there that f is super-linear (called admissible in Birrell et al. (2022a)), that is $\lim_{s\to\infty} f(s)/s = +\infty$. The case of α-divergences for $\alpha < 1$ is discussed in detail in Birrell et al. (2022a).

Theorem A.1. *Assume that f is superlinear and strictly convex. Then for $P, Q \in \mathcal{P}_1(\mathbb{R}^d)$ we have*

1. **Divergence:** *$D_f^{\Gamma_L}(P\|Q)$ is a divergence, i.e. $D_f^{\Gamma_L}(P\|Q) \geq 0$ and $D_f^{\Gamma_L}(P\|Q) = 0$ if and only if $P = Q$. Moreover the map $(P, Q) \to D_f^{\Gamma_L}(P\|Q)$ is convex and lower-semicontinuous.*

2. **Infimal Convolution Formula:** *We have*

$$D_f^{\Gamma_L}(P\|Q) = \inf_{\gamma \in \mathcal{P}(\Omega)} \left\{ W^{\Gamma_L}(P, \gamma) + D_f(\gamma\|Q) \right\}. \tag{43}$$

In particular we have

$$0 \leq D_f^{\Gamma_L}(P\|Q) \leq \min\left\{ D_f(P\|Q), W^{\Gamma_L}(P, Q) \right\}. \tag{44}$$

3. **Interpolation and limiting behavior of $D_f^{\Gamma_L}(P\|Q)$:**

$$\lim_{L\to\infty} D_f^{\Gamma_L}(P\|Q) = D_f(P\|Q) \quad \text{and} \quad \lim_{L\to 0} \frac{1}{L} D_f^{\Gamma_L}(P\|Q) = W^{\Gamma_1}(P,Q)\,. \tag{45}$$

4. **Optimizers:** *There exists an optimizer $\phi^{L,*} \in \Gamma_L$, unique up to a constant in supp$(P) \cup$ supp(Q). The optimizer $\gamma^{L,*}$ in the infimal convolution formula exists and is unique and we have $d\gamma^{L,*} \propto (f^*)'(\phi^{L,*})dQ$ (see Birrell et al. (2022a) for details). For example for KL we get $d\gamma^{L,*} \propto e^{\phi^{L,*}} dQ$.*

For connections with Sinkhorn regularizations Genevay et al. (2016) we refer to Birrell et al. (2022c). Another useful result established in Birrell et al. (2022a) is a new type of data processing inequality. For probability kernel $K(x, dy)$ we denote $K_{\#}P(dy) = \int K(x, dy)P(dx)$ and $Kf(x) = \int f(y)K(x, dy)$. We have

Theorem A.2 (Data Processing Inequality)**.** *For probability kernel $K(x, dy)$ we have*

$$D_f^{\Gamma}(K_{\#}P\|K_{\#}Q) \le D_f^{K(\Gamma)}(P\|Q) \tag{46}$$

Note that this is a stronger form than the usual data processing inequality since $K(\Gamma)$ maybe (much smaller) than Γ. This inequality will be used to construct and assess GPA in latent space in Section 7, see Theorem 7.1.

B HIGHER ORDER EXPLICIT ODE SOLVERS

Besides the forward Euler scheme considered in equation 32 and equation 33, we can also take advantage of higher order schemes for differential equations such as Heun's method

$$\begin{aligned} \tilde{y}_{t+1} &= y_t - \Delta t \nabla \phi_t(y_t) \\ y_{t+1} &= y_t - \frac{\Delta t}{2}(\phi_t(y_t) + \phi_{t+1}(\tilde{y}_{t+1})) \end{aligned} \tag{47}$$

and RK4. As we demonstrate in an example in Figure 10, they can substantially improve the accuracy of solution. In this example GPA learns a 2D Mixture of Gaussians embedded in 12D. We consider 600 particles from the 12D Gaussian ball $P_0 = N(8 * \mathbf{1}_{12}, 0.5^2 I_{12})$, which are transported via GPA to the target distribution. In this example, forward Euler produces an oscillatory pattern in the orthogonal 10D subspace while equation 47 produces a convergent approximation.

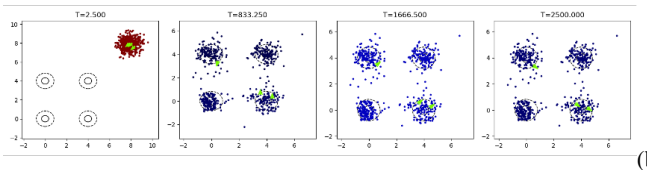
C COMPUTATIONAL SETTING

Neural network architectures Discriminators $\phi : \mathbb{R}^d \to \mathbb{R}$'s are implemented using neural networks. We implemented FNN discriminators for general $\mathbb{R}^d$ problems and CNN discriminator especially for 2D image generation problems. For both networks, we impose the Lipschitz constraint on ϕ by spectral normalization (SN), where the weight matrix in each layer of the D layers in total has spectral norm $\|W^l\|_2 = L^{1/D}$. See Table 4 for details. Exact numbers of parameters differ in each example and can be found in the code repository `https://github.com/HyeminGu/Lipschitz_regularized_generative_particles_algorithm` v0.2.0.

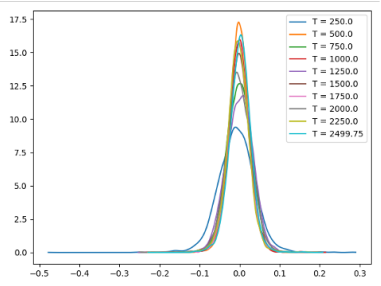
Computational resources General examples including gene expression dataset integration in a latent space are computed in the tensorflow-CPU environment: *tensorflow-gpu=2.8.0 with CPU model Intel(R) Core(TM) i5-10210U CPU @ 1.60GHz ∼ 2.11 GHz*. The high-dimensional MNIST image generation example is computed in the tensorflow-GPU environment: *tensorflow-gpu=2.7.0 with GPU model Tesla K80 in Google cloud platform*.

D GPA VS. GAN

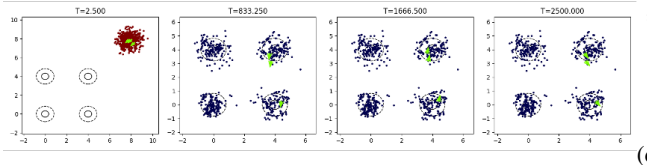
GPA generates particles by iteratively solving equation 33. The velocities of the particles are computed by the evaluation of the gradient of the discriminator $\phi_n^{L,*}$, and updated at each time step n.



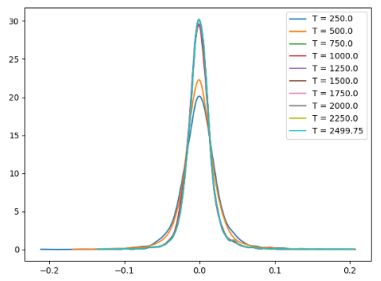
(a) Forward Euler, $\Delta t = 0.25$, Trajectories of $f_{\mathrm{KL}}^{\Gamma_1}$-GPA



(b) Forward Euler, $\Delta t = 0.25$, Evaluation at orthogonal axes



(c) Heun, $\Delta t = 0.25$, Trajectories of $f_{\mathrm{KL}}^{\Gamma_1}$-GPA



(d) Heun, $\Delta t = 0.25$, Evaluation at orthogonal axes

Figure 10: **(2D Mixture of Gaussians embedded in 12D) Forward Euler and Heun.** Both Forward Euler and Heun were able to capture the 4 wells in the 2D subspace but forward Euler shows oscillatory behavior in the orthogonal subspace while Heun shows convergent in the orthogonal subspace.

FNN Discriminator
$W^1 \in \mathbb{R}^{d \times \ell_1}$ with SN, $b^1 \in \mathbb{R}^{\ell_1}$
ReLU
$W^2 \in \mathbb{R}^{\ell_1 \times \ell_2}$ with SN, $b^2 \in \mathbb{R}^{\ell_2}$
ReLU
$W^3 \in \mathbb{R}^{\ell_2 \times \ell_3}$ with SN, $b^3 \in \mathbb{R}^{\ell_3}$
ReLU
$W^4 \in \mathbb{R}^{\ell_3 \times 1}$ with SN, $b^4 \in \mathbb{R}$
Linear

(a) General problems with dimension d

CNN Discriminator
5×5 Conv SN, 2×2 stride $(1 \to ch_1)$
leaky ReLU
Dropout, rate 0.3
5×5 Conv SN, 2×2 stride $(ch_1 \to ch_2)$
leaky ReLU
Dropout, rate 0.3
5×5 Conv SN, 2×2 stride $(ch_2 \to ch_3)$
leaky ReLU
Dropout, rate 0.3
Flatten with dimension ℓ_3
$W^4 \in \mathbb{R}^{\ell_3 \times d}$ with SN, $b^4 \in \mathbb{R}^d$
ReLU
$W^5 \in \mathbb{R}^{d \times 1}$ with SN, $b^5 \in \mathbb{R}$
Linear

(b) 2D image data (MNIST)

Table 4: Neural network architectures of the discriminator $\phi : \mathbb{R}^d \to \mathbb{R}$

This discriminator evaluation feature is shared with GANs Goodfellow et al. (2014); Arjovsky et al. (2017); Birrell et al. (2022a). However in GANs the particle generation step is different and involves also learning a generator $g_\theta : \mathbb{R}^{d'} \to \mathbb{R}^d$ parametrized in turn by a second NN with its own parameters θ. For each time step n, GANs solve two optimization problems on θ and ϕ. For instance, an

(f, Γ_L)-based GAN, Birrell et al. (2022a), is the minmax problem

$$\inf_{\theta} \sup_{\phi} H_f[\phi; g_\theta(Z), X], \quad \text{where the objective function is}$$

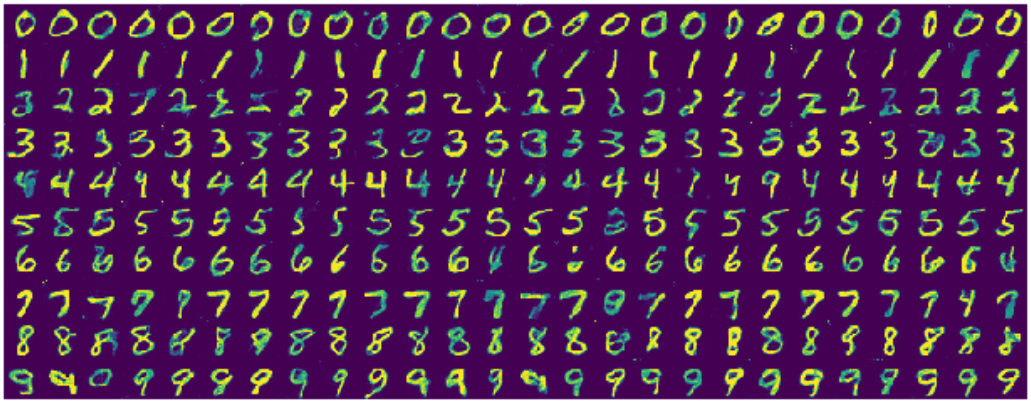
$$H_f[\phi; g_\theta(Z), X] = \frac{\sum_{i=1}^{M} \phi(g_\theta(Z^{(i)}))}{M} - \inf_{\nu \in \mathbb{R}} \left\{ \nu + \frac{\sum_{i=1}^{N} f^*(\phi(X^{(i)}) - \nu)}{N} \right\}. \tag{48}$$

Here $Z^{(i)}$ denote random data usually from the standard Gaussian in $\mathbb{R}^{d'}$ and $X^{(i)}$ correspond to the given training data set. Different GANs Goodfellow et al. (2014); Arjovsky et al. (2017); Nowozin et al. (2016); Gulrajani et al. (2017); Mao et al. (2017) have their own objective functionals $H_f[\phi; g_\theta(Z), X]$, however (f, Γ_L)-based GANs provide a common, mathematically unifying framework Birrell et al. (2022a). Once a GAN is trained, new samples can be reproduced instantly by evaluating the generator $g^*_{\theta_{\text{final}}}$ on random Gaussian samples Z. GANs are discriminator-generator models, while GPA are a discriminator-transport model where the generator is replaced by the transport mechanism, and does not need to be learned, see Algorithm 1. Since GPA does not learn a generator, instant generation is not allowed as in GAN. But GPA can excel in some tasks that GANs fail, see for example Section 6.

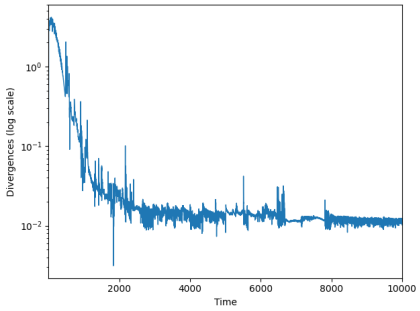
E Additional MNIST results and benchmarks for sample diversity

In Figure 11 we present an additional set of experiments to validate the diversity of the generated samples with $(f_{\text{KL}}, \Gamma_1)$-GPA, as also discussed in Section 6. Here the sample diversity of $(f_{\text{KL}}, \Gamma_1)$-GPA is addressed by selecting the number of source samples (set at $M = 600$) to be three times as large as the $N = 200$ target samples. We note that $(f_{\text{KL}}, \Gamma_1)$-GPA is such an efficient and accurate transportation method that when $M = N$ in Algorithm 1 (source and target particles have the same number), it would typically transport the source particles almost exactly on the target particles, see Figure 12. The same strategy regarding sample diversity was validated again, as well as used for data augmentation, in the swiss roll example in Section 6.

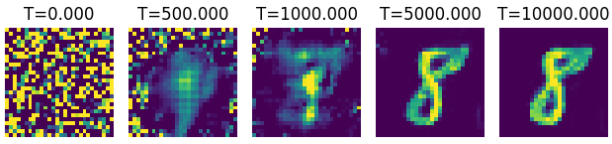
(a) 200 target samples from the true MNIST data set



(b) 260 samples from total 600 generated samples

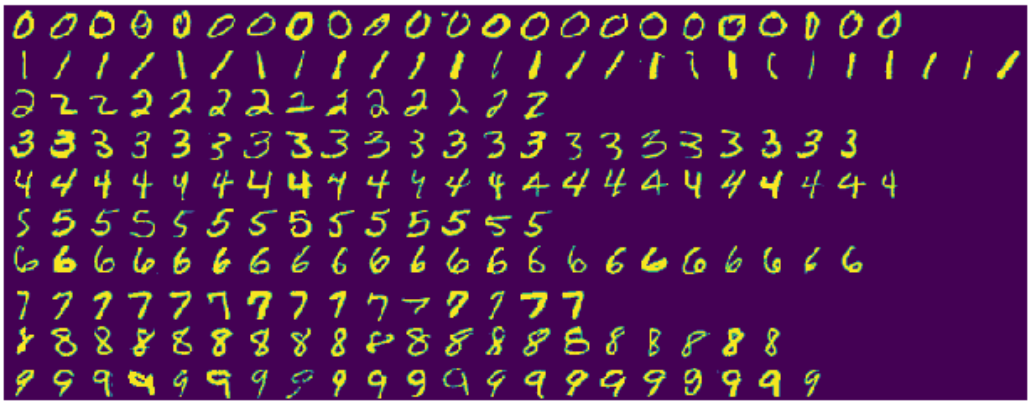


(c) $(f_{\mathrm{KL}}, \Gamma_1)$-Divergence

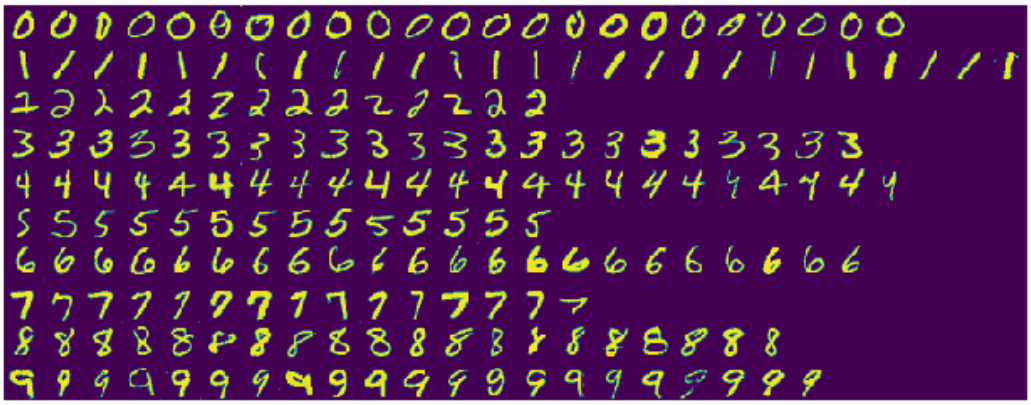


(d) A sample trajectory

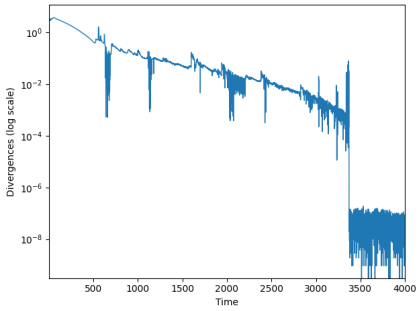
Figure 11: **(Sample diversity in learning MNIST with GPA)** We generate 600 MNIST samples from the initial logistic distribution by learning 200 true MNIST target samples via $(f_{\mathrm{KL}}, \Gamma_1)$-GPA. Target samples and generated samples can be directly compared showing that GPA-generated samples have sufficient diversity. In addition, note in (c) that the divergence converges progressively and begins to level off after the value of 1e-2. We also refer to corresponding samples, generated after that time point in (d).



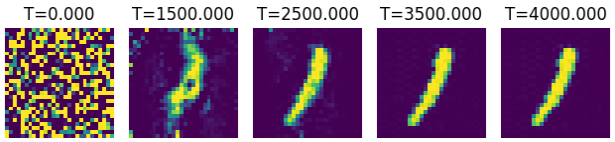
(a) 200 target samples from the true MNIST data set



(b) 200 generated samples



(c) $(f_{\mathrm{KL}}, \Gamma_1)$-Divergence



(d) A sample trajectory

Figure 12: **(Learning MNIST with GPA using same number of source and target particles)** We generate $M = 200$ MNIST samples from the initial logistic distribution by learning $N = 200$ true MNIST target samples via $(f_{\mathrm{KL}}, \Gamma_1)$-GPA. Generated and target samples can be directly compared showing that GPA-generated samples are almost exactly transported to target samples and there is no diversity in sample shapes. In addition, note in (c) that the divergence drastically drops to the value of 1e-7 when the generated particles become exactly matched to the target, and in stark contrast to Figure 12c. We also refer to a trajectory of a sample in (d), where generated samples are modified only in minor details after the drastic drop of the divergence in (c).