Accurate Quantization of Measures via Interacting Particle-based Optimization

Lantian Xu [1] **Anna Korba** [2] **Dejan Slepčev** [1]

Abstract

Approximating a target probability distribution can be cast as an optimization problem where the objective functional measures the dissimilarity to the target. This optimization can be addressed by approximating Wasserstein and related gradient flows. In practice, these are simulated by interacting particle systems, whose stationary states define an empirical measure approximating the target distribution. This approach has been popularized recently to design sampling algorithms, e.g. Stein Variational Gradient Descent, or by minimizing the Maximum Mean or Kernel Stein Discrepancy. However, little is known about quantization properties of these approaches, i.e. how well is the target approximated by a finite number particles. We investigate this question theoretically and numerically. In particular, we prove general upper bounds on the quantization error of MMD and KSD at rates which significantly outperform quantization by i.i.d. samples. We conduct experiments which show that the particle systems at study achieve fast rates in practice, and notably outperform greedy algorithms, such as kernel herding. We compare different gradient flows and highlight their quantization rates. Furthermore we introduce a Normalized Stein Variational Gradient Descent and argue in favor of adaptive kernels, which exhibit faster convergence. Finally we compare the Gaussian and Laplace kernels and argue that the Laplace kernel provides a more robust quantization.

1. Introduction

Approximating a probability distribution π only known up to a normalization constant by a finite set of points, to compute functionals $\int f(x)d\pi(x)$ (e.g. expectations, quantiles) is a central task in machine learning and computational statistics, often referred to as sampling. The error made when approximating the integral of interest by an average over n support points is typically called the integral approximation error. A large number of methods have been developed to tackle this problem. For instance, Markov Chain Monte Carlo (MCMC) methods generate a Markov chain whose law converges to π under mild assumptions (Roberts & Rosenthal, 2004). However, iterates of these chains might bunch up and cover the target distribution in an irregular, thus inefficient way with respect to the number of points. More precisely, the integral approximation error of MCMC methods is of order $\mathcal{O}(n^{-\frac{1}{2}})$ when using n particles (Łatuszyński et al., 2013). Hence, a large body of work has been focused on designing post-processing methods on MCMC algorithms output to reduce the finite-sample error (Riabiz et al., 2020; Hodgkinson et al., 2020; Teymur et al., 2021; Chopin & Ducrocq, 2021). In contrast, Quasi-Monte Carlo methods (Sobol, 1998; Dick & Pillichshammer, 2010) create more regularly-spaced sample sets to achieve faster convergence.

Recently, several algorithms for sampling relying on deterministic particle systems have been proposed in the literature, as alternatives to MCMC algorithms. In a nutshell, one casts the sampling problem as minimizing a discrepancy (between probability distributions) to the target measure π, and discretize Wasserstein, or other, gradient flows of this discrepancy. The best known example is Stein Variational Gradient Descent (SVGD) algorithm (Liu & Wang, 2016), that proposes a deterministic gradient descent of the Kullback-Leibler divergence in Stein geometry that can be seen as a kernel smoothed relative of the Wasserstein metric. In particular the velocities of SVGD are smoother than for the Wasserstein gradient flow of the same energy. The simplicity of the algorithm and its relative success on machine learning tasks, such as sampling e.g. for Bayesian inference (Liu & Wang, 2016; Liu & Zhu, 2018), learning deep probabilistic models (Pu et al., 2017), and more recently Bayesian deep learning (D'Angelo et al., 2021; D'Angelo & Fortuin, 2021) has raised a lot of interest in the sampling literature and popularized the method. More recently, the minimization of other discrepancies have been investigated, such as the Maximum Mean Discrepancy (Arbel et al., 2019) or the Kernel Stein Discrepancy (Korba et al., 2021). While the first one is closely related to optimizing shallow neural

[1]Carnegie Mellon University [2]CREST, ENSAE, IP Paris. Correspondence to: Lantian Xu <lxu2@andrew.cmu.edu>.

Proceedings of the 39^{th} International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022.

networks and requires to know the density of π or at least to have access to samples from it, the second one was introduced as a sampling algorithm when π is only known up to a normalization constant. In all cases, these methods implement interacting particle systems, whose empirical distribution aims at approximating the target distribution π. The corresponding particle systems generally involve attractive force and repulsive force terms, that drive the particles to the target distribution while preventing them from collapsing. The sample particles are correlated and approximate the target distribution as a whole. In this view, these algorithms are closer in spirit to Quasi Monte Carlo sampling. They are typically more computationally expensive than many MCMC methods, however, as we will discuss more in depth later, they can provide a better approximation of the target π for a finite number of samples.

In this paper, we study the approximation properties of particle systems derived from Wasserstein (and related) gradient flows for a finite number of particles n, at stationarity. Our goal is to encourage a fair comparison, for a given computational budget, between MCMC algorithms and these particle algorithms, that both depend on the number of iterations and a number of particles. Our contributions are twofold. We investigate the quantization properties of these methods, assuming the particles have attained a minimizer of their discrepancy objective. Furthermore, as these algorithms might be difficult to tune to guarantee particles convergence, we also discuss practical improvements. The paper is organized as follows. Section 2 provides the necessary background on the discrepancies of interest and interacting particles-based algorithms, including SVGD. Section 3 discusses related work relevant to our study. Section 4 presents a new normalized choice of kernel for SVGD. Section 5 is devoted to our theoretical results on quantization. Our numerical results are to be found in Section 6.

Notations. The space of l continuously differentiable functions on $\mathbb{R}^d$ is $C^l(\mathbb{R}^d)$, and the space of smooth functions with compact support $C_c^\infty(\mathbb{R}^d)$. $\mathcal{P}(\mathbb{R}^d)$ is the set of probability distributions over $\mathbb{R}^d$. For any $\mu \in \mathcal{P}(\mathbb{R}^d)$, $L^p(\mu)$ is the space of μ-measurable functions $f : \mathbb{R}^d \to \mathbb{R}^d$ such that $\int \|f\|^p d\mu < \infty$. We denote by $\|\cdot\|_{L^p(\mu)}$ the norm of the Banach space $L^p(\mu)$. The Sobolev space $W^{d,2}(\mathbb{R}^d)$ is denoted by $H^d = \{u \in L^2(\mathbb{R}^d), \forall \alpha \text{ s. t. } |\alpha| \leq d,\ D^\alpha u \in L^2(\mathbb{R}^d)\}$ where D denotes a partial derivative. The convolution of f and g is denoted $f \star g(x) = \int f(x-y)g(y)dy$. In the following, we assume that π admits a density proportional to $\exp(-U)$ with respect to Lebesgue measure over $\mathbb{R}^d$.

2. Background

2.1. MMD and KSD Gradient Flows

Consider a positive semi-definite kernel $k : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ and its corresponding RKHS $\mathcal{H}_k$ of real-valued functions on $\mathbb{R}^d$. The space $\mathcal{H}_k$ is a Hilbert space with inner product $\langle \cdot, \cdot \rangle_{\mathcal{H}_k}$ and norm $\|\cdot\|_{\mathcal{H}_k}$. Moreover, k satisfies the reproducing property: $\forall\ f \in \mathcal{H}_k,\ x \in \mathbb{R}^d,\ f(x) = \langle f, k(x, \cdot)\rangle_{\mathcal{H}_k}$. We denote by $\mathcal{H}_k^d$ the Cartesian product RKHS consisting of elements $f = (f_1, \ldots, f_d)$ with $f_i \in \mathcal{H}_k$, and with inner product $\langle f, g\rangle_{\mathcal{H}_k^d} = \sum_{i=1}^d \langle f_i, g_i\rangle_{\mathcal{H}_k}$. For a differentiable kernel $k : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$, $\nabla_x k(x, y)$ (resp. $\nabla_y k(x, y)$) is the gradient of the kernel w.r.t. the first (resp. second) variable evaluated at (x, y), and $\boldsymbol{\nabla} \cdot_x \nabla_y k(x, y)$ denotes the divergence of $\nabla_y k(x, y)$ w.r.t. x.

Let $\mu \in \mathcal{P}(\mathbb{R}^d)$. If $\int \sqrt{k(x, x)} d\mu(x) < \infty$, then the kernel mean embedding $m_\mu = \int k(x, .)d\mu(x) \in \mathcal{H}_k$ and $\mathbb{E}_\mu[f(X)] = \langle f, m_\mu\rangle_{\mathcal{H}_k}$ (Smola et al., 2007). If the map $m : \mathcal{P}(\mathbb{R}^d) \to \mathcal{H}_k, \mu \mapsto m_\mu$ is injective, the kernel k is said to be characteristic (Sriperumbudur et al., 2009); this includes common kernels such as the Gaussian $k(x, y) = \exp\big(-\|x - y\|^2/h\big)$ or Laplace kernel $k(x, y) = \exp(-\|x - y\|/h)$ where h denotes some bandwith parameter. For a characteristic kernel, the kernel mean embedding can be used to define a metric for probability distributions, namely the maximum mean discrepancy (MMD) (Gretton et al., 2012) defined for any $\mu \in \mathcal{P}(\mathbb{R}^d)$ as:

$$\begin{aligned}\mathrm{MMD}^2(\mu, \pi) &= \sup_{f \in \mathcal{H}_k, \|f\|_{\mathcal{H}_k} \leq 1} \left| \int f d\mu - \int f d\pi \right|^2 \\ &= \|m_\mu - m_\pi\|^2_{\mathcal{H}_k} = \iint_{\mathbb{R}^d} k(x, y) d\mu(x) d\mu(y) \\ &+ \iint_{\mathbb{R}^d} k(x, y) d\pi(x) d\pi(y) - 2 \iint_{\mathbb{R}^d} k(x, y) d\mu(x) d\pi(y), \quad (1)\end{aligned}$$

where the last equality is obtained by the reproducing property. The MMD (1) writes as a sum of integrals, hence it can be estimated as soon as one has access to samples of μ and π. Also, it enables to bound the integral approximation error for any $f \in \mathcal{H}_k$, since by the reproducing property and Cauchy-Schwartz inequality,

$$\left| \int_{\mathbb{R}^d} f(x) d\pi(x) - \int_{\mathbb{R}^d} f(x) d\mu(x) \right| \leq \|f\|_{\mathcal{H}_k} \mathrm{MMD}(\mu, \pi). \quad (2)$$

If one has only access to the score of π defined by $s(x) = \nabla \log \pi(x)$, as in many applications such as Bayesian inference, it is still possible to compute the Kernel Stein Discrepancy (KSD) (Liu et al., 2016; Chwialkowski et al., 2016; Gorham & Mackey, 2017) defined by

$$\mathrm{KSD}^2(\mu|\pi) = \iint_{\mathbb{R}^d} k_\pi(x, y) d\mu(x) d\mu(y), \quad (3)$$

where $k_\pi : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ is the Stein kernel, defined

through the score s and k as:

$$k_\pi(x,y) = s(x)\cdot s(y)k(x,y) + s(x)\cdot\nabla_y k(x,y) + \nabla_x k(x,y)\cdot s(y) + \boldsymbol{\nabla}\cdot_x\nabla_y k(x,y), \quad (4)$$

where $\cdot$ denotes the scalar product in $\mathbb{R}^d$. The Stein kernel k_π is a reproducing kernel and satisfies a Stein identity ($\int_{\mathbb{R}^d} k_\pi(x,\cdot)d\pi(x) = 0$) under mild boundary conditions on k and π, see Oates et al. (2017, Lemma 1) and Chwialkowski et al. (2016, Lemma 5.1). Hence (3) is obtained by using the Stein kernel (4) in the MMD (1), and KSD can be seen as a particular case of MMD. Several properties of KSD were studied in (Gorham & Mackey, 2017); in particular for a distantly dissipative target π, the KSD was shown to metrize weak convergence for the Inverse MultiQuadratic (IMQ) kernel defined by $\eta(x) = (c + \|x\|)^\beta$, $\beta \in (-1, 0), c > 0$, while this is not the case for kernels with lighter tails such as Gaussian or Matérn kernels. In contrast, MMD metrizes weak convergence for all these kernels Sriperumbudur (2016, Thm.3.2).

Now, let $\mathcal{F}(\mu) = {}^1\!/_2 D^2(\mu,\pi)$ where D is the MMD or KSD. A Wasserstein gradient flow of $\mathcal{F}$ (Ambrosio et al., 2005) can be described by the following continuity equation:

$$\frac{\partial\mu_t}{\partial t} + \boldsymbol{\nabla}\cdot(\mu_t v_{\mu_t}) = 0, \quad \text{for } v_{\mu_t} = -\nabla\mathcal{F}'(\mu_t), \quad (5)$$

where $\mathcal{F}'$ denotes the first variation [1] of $\mathcal{F}$. For discrete measures $\mu_n = {}^1\!/_n\sum_{i=1}^n \delta_{x^i}$, we can define $F(X_n) := \mathcal{F}(\mu_n)$ where $X_n = (x_1,\dots,x_n)$. For the functionals for which F is well defined (e.g., MMD or KSD), the Wasserstein gradient flow of $\mathcal{F}$ becomes the standard euclidean gradient flow of the particle based function F. Furthermore, the forward Euler discretization of (5) writes as gradient descent on the position of the particles. If D is the MMD, the gradient of F is readily obtained as

$$\nabla_{x^i}F(X_n) = \frac{1}{n}\sum_{j=1}^n \nabla_{x^j}k(x^i,x^j) - \int\nabla_x k(x^i,x)d\pi(x). \quad (6)$$

Notice that the integral with respect to π requires to know the density π, or at least samples to approximate this integral. In contrast, thanks to the property of the Stein kernel, if D is the KSD,

$$\nabla_{x^i}F(X_n) = \frac{1}{n}\sum_{j=1}^n \nabla_{x^j}k_\pi(x^i,x^j). \quad (7)$$

Hence, minimizing the MMD or KSD result in particle systems that interact through the gradient of the objective. We will refer later to these algorithms as MMD Descent and KSD Descent. While the choice of the MMD is not adapted to the task of sampling for Bayesian inference where the density of π or samples are not available, MMD gradient flow can be related to the optimization of shallow neural networks with gradient descent (Arbel et al., 2019).

[1] If it exists, the first variation of $\mathcal{F}$ at ν is the function $\mathcal{F}'(\nu) : \mathbb{R}^d \to \mathbb{R}$ s. t. for $\nu,\mu \in \mathcal{P}(\mathbb{R}^d)$: $\lim_{\epsilon\to 0} {}^1\!/_\epsilon[\mathcal{F}(\nu+\epsilon(\mu-\nu)) - \mathcal{F}(\nu)] = \int\mathcal{F}'(\nu)(x)(d\mu(x) - d\nu(x))$.

2.2. Stein Variational Gradient Descent

Stein Variational Gradient Descent (SVGD) is a sampling algorithm introduced in (Liu & Wang, 2016), that performs gradient descent of the Kullback-Leibler (KL) divergence (relative to π) in the space of probability distributions, where the gradient is smoothed through a kernel integral operator (Korba et al., 2020). Recall that the KL divergence is defined by

$$\mathrm{KL}(\mu|\pi) = \int\log\left(\frac{d\mu}{d\pi}\right)d\mu,$$

where $d\mu/d\pi$ is the Radon-Nikodym derivative, if μ is absolutely continuous with respect to π and $\mathrm{KL}(\mu|\pi) = +\infty$ otherwise. More precisely, SVGD dynamics corresponds to the gradient flow of the KL with respect to a *Stein geometry* (Liu, 2017; Duncan et al., 2019). The Stein geometry shares formal similarities with the Wasserstein metric. The difference is that the length of the paths in Wasserstein geometry is measured by the L^2 norm of the velocity, while in the Stein geometry one measures the velocity in an RKHS. That is the length of the path $\partial_t\mu_t = -\boldsymbol{\nabla}\cdot(\mu_t v_t)$ for $t\in[0,T]$ is $\int_0^T\|v\|_{\mathcal{H}_k}dt$, as compared to $\int_0^T\|v\|_{L^2(\mu_t)}dt$ for Wasserstein geometry. Thus the velocities of paths of finite length need to be more regular in Stein geometry.

Fix a reproducing kernel k. In continuous time, SVGD flow is defined by the continuity equation

$$\frac{\partial\mu_t}{\partial t} + \boldsymbol{\nabla}\cdot(\mu_t v_{\mu_t}) = 0,\ v_{\mu_t} = -k\star(\mu_t\nabla U) - \nabla k\star\mu_t. \quad (8)$$

As was noted (Liu, 2017) the velocity of the flow v_{μ_t} is well defined even if μ_t is a discrete measure. As v_{μ_t} depends on continuously on μ_t with respect to weak convergence of measures, (8) can be approximated by an interacting particle system. In practice, SVGD algorithm implements a particle approximation of a forward discretization of (8). Let $\gamma > 0$ be a fixed step-size. Starting from n i.i.d. samples $x_0^i \sim \mu_0$, SVGD algorithm updates the n particles as follows at each iteration :

$$x_{l+1}^i = x_l^i - \frac{\gamma}{n}\sum_{i=1}^n\left[\nabla U(x_l^i)k(x_l^i,x_l^j) + \nabla_{x_l^j}k(x_l^i,x_l^j)\right],$$

for any $i = 1,\dots,n$. While the discrete particle flow is no longer a gradient flow (as the KL divergence is not well defined) it approximates the continuum gradient flow (8), see (Lu et al., 2019). However, despite the simplicity of the algorithm, choosing the right k and its parameters (e.g.,

bandwith) is critical. Moreover, the convergence may be slow, especially for a bad choice of k; and practitioners rely on different modifications/tricks (e.g., use Adagrad (Duchi et al., 2011)) to improve the convergence of SVGD.

3. Related Work

Quantization problems and Quadrature rules. Finding an optimal discrete distribution $\mu_n = \sum_{i=1}^n w_i \delta_{x_i}$ supported on a finite number n of Dirac masses, in order to approximate a probability density is referred to as the *quantization problem* (Graf & Luschgy, 2000). If the weights are fixed to $w_i = 1/n$ and μ_n is the uniform measure over the points, then the resulting problem of optimally placing the points x_i is known as the *empirical quantization problem* or *uniform quantization*. Closely related to quantization, a quadrature rule aims at finding μ_n such that the integrals of some test function f w.r.t. μ_n and π are close (Briol et al., 2015). The quality of the quantization can be measured through a discrepancy between π and μ_n. Many works have investigated the case where the metric is the Wasserstein distance (Fournier & Guillin, 2015; Merigot et al., 2021), where it is known that the typical quantization error between π and μ_n is of order $\mathcal{O}(n^{-1/d})$ for $d > 2$. However, when the metric is the MMD or KSD, the question of how many points are required to get a good approximation of π remains largely open (Oates, 2021).

Kernel herding (KH) and Stein points (SP). Some approaches attempt to solve MMD or KSD quantization in a greedy manner, i.e. by sequentially constructing μ_n to minimize these discrepancies, adding one new particle at each iteration. The most famous example is kernel herding (KH) (Chen et al., 2012; Bach et al., 2012; Pronzato, 2021; Tsuji & Tanaka, 2021; Khanna et al., 2021), that minimizes greedily the MMD through the Frank-Wolfe algorithm (Frank et al., 1956), assuming one has access to the mean embedding m_π. It is also a first-order optimization scheme for the MMD objective; but it does not rely on the Wasserstein geometry like MMD Descent. This is the reason why KH adds sequentially particles, while MMD descent continuously displaces a given set of particles. (Bach et al., 2012) obtain a linear rate of convergence $\mathcal{O}(e^{-bn})$ for KH if the optimum, i.e. the mean embedding m_π, lies in the relative interior of the marginal polytope with distance b away from the boundary; however for infinite-dimensional kernels $b = 0$ and the rate does not hold, which was pointed out. Similarly to KH, (Chen et al., 2018; 2019) construct Stein Points (SP): a sequence of points that greedily minimize KSD. In (Chen et al., 2018), the authors establish a $\mathcal{O}((\log(n)/n)^{\frac{1}{2}})$ rate, and acknowledge that this rate seem slower than their empirical observations. However, as these greedy methods perform an exhaustive search for the best point at each iteration, they are more computationally expensive than particle methods derived from Wasserstein gradient flows, that perform an explicit descent step (where one iteration is of complexity $\mathcal{O}(n^2)$).

SVGD with a finite number of particles. (Korba et al., 2020) studies non-asymptotic properties of SVGD, i.e., for a finite time $t \leq T$ and number of particles n, how far is the SVGD particles distribution $\hat{\mu}_t^n$ from the target π. The authors obtained propagation of chaos (POC) bounds, that quantify how far the distribution of the interacting particle system $\hat{\mu}_t^n$ is from the one of a particle system composed of i.i.d. particles distributed as the true continuous process (non-interacting but non-implementable), denoted $\bar{\mu}_t^n$ (i.e., $\bar{\mu}_t^n$ is the n-empirical version of μ_t where μ_t is the distribution of SVGD dynamics). More precisely, they obtain a bound of the form $\mathbb{E}[W_2(\hat{\mu}_t^n, \bar{\mu}_t^n)] \leq \frac{C_T}{\sqrt{n}}$. While this bound is interesting since it yields a control on the deviation (or POC) of the particle system, it cannot characterize the quantization properties of SVGD. Indeed, quantization quantifies how far is the particle system (for a finite n), when $T \to \infty$, from the target π. The POC bound of (Korba et al., 2020) becomes vacuous as $T \to \infty$ and $\bar{\mu}_t^n$ converges to an empirical version of π (supported on n points) as $T \to \infty$. Moreover, to the best of our knowledge, POC bounds always yield at best $\mathcal{O}(1/\sqrt{n})$, i.e. as the Monte Carlo rate.

4. Normalized SVGD

We now turn to introducing a modification to the SVGD flow. One of the structural issues with the SVGD flow (8) is that the equation is quadratic in the density μ_t. A particular problem is that the velocity v_{μ_t} is small where μ_t is small. This creates convergence issues, especially if the initial distribution, μ_0, is spread in space. Moreover choosing a wide initial distribution is otherwise a good idea, if the target distribution is multimodal or the location of its mode or geometry in general are unknown a priori. To mitigate the above issue we introduce a normalized SVGD (NSVGD) which scales linearly in μ_t. This is achieved by reweighing the kernel by a kernel-density estimate of μ.

Consider a translation-invariant kernel parameterized by a bandwidth $\tau > 0$: $\eta_\tau(x - y) = \eta(\frac{x-y}{\tau})$ with $\eta \in C^1(\mathbb{R}^d \setminus \{0\})$, and μ a, potentially discrete, distribution. We now introduce a density-dependent kernel:

$$K_\mu(x, y) = \eta_\tau(x - y)\mu_h(x)^{-\frac{1}{2}}\mu_h(y)^{-\frac{1}{2}} \tag{9}$$

where μ_h denotes the smoothed density $\mu \star \eta_h$. We note that K_μ would be the same if we considered $\eta_\tau(x - y) = \frac{1}{\tau^d}\eta(\frac{x-y}{\tau})$. For each μ the kernel is still symmetric and positive definite on $\mathbb{R}^d$. Thus there exists a unique Hilbert space of functions on $\mathbb{R}^d$ for which K_μ is a reproducing kernel; however it is no longer translation-invariant. We note that the kernel now depends on the measure μ, and thus the way we measure the lengths of curves in the space of mea-

sures depends on the location, as is the case for the Wasserstein metric. The length of the path $\partial_t \mu_t = -\nabla \cdot (\mu_t v_t)$ for $t \in [0,T]$ in this new, normalized Stein geometry is $\int_0^T \|v\|_{\mathcal{H}_{K_{\mu_t}}}$. The resulting gradient flow, namely NSVGD flow, has the velocity vector field which can be written explicitly using the kernel (9) in the vector field given in (8). This choice of density-dependent kernel also defines a gradient flow of the Kullback-Leibler divergence in Stein geometry for the new kernel K_μ, see Lemma A.1. We also remark that it was observed by (Duncan et al., 2019) that for Laplace kernel, weighting the kernel as in (9) but with μ_h replaced by π in one spatial dimension enables one to prove that the Hessian at the steady state has a positive spectral gap, indicating exponential convergence to equilibrium.

In the discrete setting where $\mu = {}^1\!/_n \sum_{i=1}^n \delta_{x_i}$, we can write the NSVGD vector field ruling the particle system as

$$v_\mu(x) = -\frac{1}{n}\sum_{j=1}^n \left(\mu_h(x)\mu_h(x^j)\right)^{-\frac{1}{2}} w^j(x), \tag{10}$$

where $\mu_h(x) = \frac{1}{n}\sum_{i=1}^n \eta_h(x - x_i)$, and

$$w^j(x) = \nabla\eta_\tau(x-x^j) + \eta_\tau(x-x^j)\nabla U(x^j) \tag{11}$$

$$+ \frac{\eta_\tau(x-x^j)}{2\mu_h(x^j)}\frac{1}{n}\sum_{m=1}^n \nabla\eta_h(x^j - x^m). \tag{12}$$

In v_μ, the term $\mu_h(.)\mu_h(x^j)$ acts as a preconditioner, (11) is the vector field of the original SVGD algorithm, while (12) can be understood as a weighted repulsive term inherited from its neighbors. We will refer to this algorithm as Normalised SVGD (NSVGD), whose pseudocode is given in Algorithm 1. The preconditioner accelerates or slows down the dynamic depending on the density regions and makes NSVGD less sensitive to the choice of the step-size than regular SVGD. Furthermore, we consider a self-tuning kernel bandwidth $h = ({}^1\!/_{n^2}\sum_{i,j}\|x_i - x_j\|^2)^{1/2} n^{-1/d+4}$ inspired from Scott's rule (Scott, 1979), so we are not introducing more hyperparameters. In practice, $\tau = h$ might be a good choice for kernel bandwidth, as discussed in the experiments section.

Algorithm 1 Normalized SVGD (NSVGD)

Input: initial distribution μ_0, number of particles n, number of iterations L, step-size γ, bandwidth τ for η
Initialize $x_0^1, \dots, x_0^n \sim \mu_0$.
for $l = 1$ **to** L **do**
  Compute $\mu_h(.) = \frac{1}{n}\sum_{i=1}^n \eta_h(. - x_l^i)$
  For $i = 1, \dots, n$,
    $x_{l+1}^i = x_l^i - \gamma v_\mu(x_l^i)$, where v_μ is defined as (10).
end for
Return: $\mu_n = \frac{1}{n}\sum_{i=1}^n \delta_{x_L^i}$

5. MMD and KSD Quantization

In this section, we focus on establishing how well can a measure be approximated by an empirical measure with respect to MMD or KSD. We are interested in establishing bounds on the quantization error

$$Q_n = \inf_{x_1,\dots,x_n} \mathrm{D}(\pi, \mu_n), \quad \text{for } \mu_n = \frac{1}{n}\sum_{i=1}^n \delta_{x_i},$$

where D is the MMD or KSD. It is well known (Gretton et al., 2006; Lu & Lu, 2020) that if $x_1, x_2, \dots, x_n$ are i.i.d. samples of μ then the error is independent of the dimension d and $\mathrm{D}(\pi, \mu_n) = \mathcal{O}(n^{-\frac{1}{2}})$, which matches the minimax lower bound for empirically estimating the mean embedding in L^2 norm, (Tolstikhin et al., 2017). Here we show that if instead of random, we consider appropriately chosen points, then one can obtain a much lower approximation error. We first consider the following assumption on the kernel k.

Assumption 1. Assume that the kernel is d-times continuously differentiable. Assume also that any mixed partial derivative of the kernel of order smaller than d has a RKHS norm bounded by a constant $C_{k,d} \geq 0$.

Assumption 2. Let $k(x,y) = \eta(x-y)$ a translation invariant kernel on $\mathbb{R}^d$. Assume that $\eta \in C(\mathbb{R}^d) \cap L^1(\mathbb{R}^d)$, and that its Fourier transform verifies : $\exists C_{k,d} \geq 0$ such that $(1+|\xi|^2)^d \leq C_{k,d}|\hat{\eta}(\xi)|^{-1}$ for any $\xi \in \mathbb{R}^d$.

We will prove that the Gaussian kernel satisfies Assumption 1. Moreover, for the Gaussian kernel $\eta(z) = \exp\left(-\|z\|^2/2h^2\right)$, for any $\xi \in \mathbb{R}^d$, $|\hat{\eta}(\xi)|^{-1} = h^{-d}\exp\left(h^2\|\xi\|^2/2\right)$ where $\hat{\eta} = (2\pi)^{-d/2}\int \eta(z)e^{-iz\xi}dz$. Hence, it satisfies Assumption 2. Moreover, Assumption 2 includes kernels which are not smooth, such as Matern kernels that can be defined through their Fourier transform $\hat{\eta}(\xi) \propto \frac{1}{(1+\|\xi\|^2)^j}$, $j \geq d$ whose RKHS correspond to Sobolev spaces of order j, and which are not smooth at $z = 0$.

Theorem 5.1. *Suppose Assumption 1 or Assumption 2 holds. Assume that (i) π is the Lebesgue measure or (ii) a general probability measure on $[0,1]^d$. Then, there exists a constant C_d depending on d, such that for all $n \geq 2$,*

- *if (i): there exist points $x_1, \dots, x_n$ such that*

$$\mathrm{MMD}(\pi, \mu_n) \leq C_d \frac{(\log n)^{d-1}}{n}.$$

- *if (ii): there exist points $x_1, \dots, x_n$ such that*

$$\mathrm{MMD}(\pi, \mu_n) \leq C_d \frac{(\log n)^{\frac{3d+1}{2}}}{n}.$$

Proof. Below we present the proof under Assumption 1 which uses RKHS reproducing properties. Our proof relies on bounding the star discrepancy of the point set

$X_n = \{x_1, \dots, x_n\}$ and the Hardy-Krause variation of a function belonging to the unit RKHS ball when k satisfies Assumption 1. The proof under Assumption 2 relies on Sobolev space techniques and can be found in Appendix B.

The star discrepancy of the point set X_n is defined as

$$\mathcal{D}(X_n, \pi) = \sup_{I=\Pi_{i=1}^n [a_i, b_i]} |\pi(I) - \mu_n(I)|. \quad (13)$$

For a probability measure π on $[0,1]^d$, the latter quantity is upper bounded by $C_{\pi,d}(\log n)^{\frac{3d+1}{2}}/n$ where $C_{\pi,d}$ is of order $\sqrt{d}$, see Aistleitner & Dick (2013, Theorem 1).

Now, for a subset $\alpha \subseteq \{1, \dots, d\}$, and $x \in \mathbb{R}^d$, denote by $x^\alpha \in \mathbb{R}^{|\alpha|}$ the components of x indexed by α, and denote $f(y) := f(x_\alpha, 1)$ where $y \in \mathbb{R}^d$ is the vector whose i-th coordinate is equal to x_i if $i \in \alpha$, and is 1 otherwise. We denote by $\frac{\partial^{|\alpha|} f(x_\alpha, 1)}{\partial x_\alpha}$ the mixed partial derivative of f with respects to the components of x^α. The variation of a function $f : [0,1]^d \to \mathbb{R}$ with continuous mixed partial derivatives is defined as

$$V(f) = \sum_{\alpha \subseteq \{1,\dots,d\}} \int_{[0,1]^{|\alpha|}} \left| \frac{\partial^{|\alpha|} f(x_\alpha, 1)}{\partial x_\alpha} \right| dx_\alpha. \quad (14)$$

Remarkably, the latter variation coincides with the Hardy-Krause variation, a more general notion that be applied to functions that do not have continuous partial derivatives, see Dick & Pillichshammer (2010, Remark 2.19).

Then, for any $x \in \mathbb{R}^d$, $\alpha \subseteq \{1, \dots, d\}$ and f such that $\|f\|_{\mathcal{H}_k} \leq 1$, applying the reproducing property on partial derivatives, Cauchy-Schwarz inequality Steinwart & Christmann (2008, Corollary 4.36), and Assumption 1, gives

$$\left| \frac{\partial^{|\alpha|} f(x_\alpha, 1)}{\partial x_\alpha} \right| \leq \left\| \frac{\partial^{|\alpha|} k((x_\alpha, 1), \cdot)}{\partial^{|\alpha|} x_\alpha} \right\|_{\mathcal{H}_k} \|f\|_{\mathcal{H}_k} \leq C_{k,d}.$$

Thus, the variation can be bounded as

$$V(f) \leq C_{k,d} \sum_{\alpha \subseteq \{1,\dots,d\}} \int_{[0,1]^{|\alpha|}} dx_\alpha = C_{k,d} 2^d. \quad (15)$$

We now recall a generalized form of the well-known Koksma-Hlawka inequality:

$$\left| \int_{[0,1]^d} f(x) d\pi(x) - \frac{1}{n} \sum_{i=1}^n f(x_i) \right| \leq \mathcal{D}(X_n, \pi) V(f),$$

that holds for a probability measure on $[0,1]^d$ Aistleitner & Dick (2015, Theorem 1), where f is function with bounded variation. Using the previous inequality and the above computations, we can conclude by taking $C_d = C_{k,d} C_{\pi,d}$. □

Remark 5.2. Assumption 1 is satisfied by the Gaussian kernel with $C_{k,d} = (2d)!$.

Proof. By the reproducing property, we have

$$\left\| \frac{\partial^{|\alpha|} k((x_\alpha, 1), \cdot)}{\partial^{|\alpha|} x_\alpha} \right\|_{\mathcal{H}_k} = \left(\frac{\partial^{|\alpha|, |\alpha|} k((x_\alpha, 1), (x_\alpha, 1))}{\partial^{|\alpha|} x_\alpha \partial^{|\alpha|} y_\alpha} \right)^{\frac{1}{2}}.$$

Consider the Gaussian kernel, i.e. for $x, y \in \mathbb{R}^d$, $k(x,y) = e^{-\|x-y\|^2/h}$. Hence, for any $x, y \in \mathbb{R}^d$, the $|\alpha|$-th partial derivative of the kernel in both variables is equal to

$$\frac{\partial^{|\alpha|, |\alpha|} k(x,y)}{\partial^{|\alpha|} x_\alpha \partial^{|\alpha|} y_\alpha} = (-1)^{|\alpha|} \frac{\partial^{2|\alpha|} e^{-t^2}}{\partial^{2|\alpha|} t} = (-1)^{|\alpha|} e^{-t^2} h_{2|\alpha|}(t)$$

where h_u, $u \geq 0$ denotes the u-th Hermite polynomial, see Steinwart & Christmann (2008, Section A.1). In particular for $x = y$, i.e. $t = 0$, evaluations of Hermite polynomials at zero correspond to the well-known Hermite numbers $(-1)^{|\alpha|} 2^{|\alpha|} (2|\alpha| - 1)!!$ with $(2|\alpha| - 1)!! = 1 \times 3 \times \cdots \times (2|\alpha| - 1)$. We conclude using $|\alpha| \leq d$. □

The next Proposition, proved in Appendix C, extends the result to non compactly supported distributions.

Proposition 5.3. *Suppose Assumption 1 or Assumption 2 holds and that k is bounded. Assume π is a light-tailed distribution on $\mathbb{R}^d$ (i.e. which has a thinner tail than an exponential distribution). Then, for $n \geq 2$ there exist points $x_1, ..., x_n$ such that*

$$\mathrm{MMD}(\pi, \mu_n) \leq C_d \frac{(\log n)^{\frac{5d+1}{2}}}{n}.$$

Sketch of proof. The proof relies on decomposing $\mathrm{MMD}(\pi, \mu_n) \leq \mathrm{MMD}(\pi, \mu) + \mathrm{MMD}(\mu, \mu_n)$ and choosing μ compactly supported on $A_n = [-\log n, \log n]^d$. As π is light-tailed, μ is close to π in total variation distance, and we first get $\mathrm{MMD}(\pi, \mu) \leq C/n$. Then, we can take μ_n supported on A_n and bound $\mathrm{MMD}(\mu, \mu_n)$ using similar arguments as Theorem 5.1. □

Now we turn to the quantization upper bound in KSD. The proof of the theorem below can be found in Appendix D.

Theorem 5.4. *Assume that k is a Gaussian kernel and that $\pi \propto \exp(-U)$ with $U \in C^\infty(\mathbb{R}^d)$. Assume furthermore that $U(x) > c_1 \|x\|$ for large enough x, and that there exists a real-valued polynomial V of degree $m \geq 0$, such that for any multi-index[2] β, $\left| \frac{\partial^\beta U(x)}{\partial^{\beta_1} x_1 ... \partial^{\beta_j} x_j} \right| \leq V(x)$ for all $1 \leq |\beta| \leq d+1$. Then there exist points $x_1, ..., x_n$ such that*

$$\mathrm{KSD}(\mu_n | \pi) \leq C_d \frac{(\log n)^{\frac{6d+2m+1}{2}}}{n}.$$

[2] A multi-index β of order $|\beta| := j$ is a d-tuple of nonnegative integers whose sum is equal to j. In contrast to the multi-indices α used in Theorem 5.1, β can have repeated entries.

We note that for Gaussian mixtures π, U satisfies the conditions of the theorem.

Sketch of proof. The proof relies on bounding the first and last term of the $\mathrm{KSD}(\mu_n, \pi)$ (3), as the cross terms can be upper bounded by the former ones, see Lemma A.2. Then, the two remaining terms in the $\mathrm{KSD}(\mu_n, \pi)$ are treated independently as two $\mathrm{MMD}(\mu_n, \pi)$, with $k_1(x, y) = s(x)^T s(y) k(x, y)$ and $k_2(x, y) = \boldsymbol{\nabla} \cdot_x \nabla_y k(x, y)$. □

The results of this section indicate that the quantization error w.r.t. MMD or KSD decreases faster than the Monte Carlo error rate of $\mathcal{O}(n^{-\frac{1}{2}})$ for i.i.d. samples Assuming MMD or KSD Descent algorithms find global minimizers (or good approximates), we can thus attain very fast rates of quantization. A consequence of our rates is that the integration error (2) is controlled for functions in the RKHS associated to the kernel of the MMD. For some k, the RKHS is well-characterized - for instance, for the Gaussian kernel. In contrast, the RKHS of the Stein kernel k_π will depend on both a kernel k and the target π, resulting in a specific class of functions. Identifying the Stein kernel RKHS requires an extended study, which does not exist in the literature to the best of our knowledge.

Characterizing quantization properties of SVGD stationary states remain an open problem. In contrast to MMD and KSD descent, SVGD does not decrease a functional (the KL divergence is meaningless for discrete measures). Moreover, there is no clear relation between approximately minimizing the KL divergence and the integral approximation error. Hence, it is unclear how to show SVGD quantization properties. We think that it is a wide open and very challenging problem, that we also leave for future work. However, our experimental results in the next Section provide guidance on what the conjectured quantization rate would be.

6. Experiments

In this section, we investigate numerically the quantization properties of the sampling algorithms discussed above, namely SVGD (Liu & Wang, 2016), Normalized SVGD presented in Section 4, MMD descent (Arbel et al., 2019), and KSD Descent (Korba et al., 2021), as well as greedy algorithms such as Kernel Herding (KH) (Chen et al., 2012) and Stein points (SP) (Chen et al., 2018).

6.1. Practical Behavior of the Algorithms

We first discuss the optimization properties of the algorithms at study, as they can be difficult to tune to reach convergence.

Optimization. KSD (3) and MMD (1), as well as their gradients (6)-(7), can be evaluated exactly for discrete measures. Note that for a Gaussian target π and Gaussian kernel, the mean embedding m_π can be written in closed-form, see Tolstikhin et al. (2017, Eq. 25). Thus for Gaussian mixture targets, one can compute MMD descent exactly. For general measures one can only approximate the MMD descent if a sample of the target measure is available, which limits the practical applicability the algorithm. For KSD and MMD minimization, we use L-BFGS (Liu & Nocedal, 1989), a fast and hyperparameter-free optimization algorithm that does not require tuning the step-size. Since L-BFGS requires access to the objective loss and gradient in closed form, it can be used for KSD and MMD descent. However it cannot be used for SVGD, that optimizes KL, since the latter cannot be evaluated for discrete measures. Still, SVGD is often optimized with the AdaGrad optimizer (Duchi et al., 2011), which can be seen as an improved gradient descent algorithm where the learning rate is scaled for each particle. In our experiments, we did not see a significant advantage of using Adagrad versus the standard gradient descent for SVGD. Regarding greedy algorithms such as KH and Stein points, they require at each iteration the solution of a global optimisation problem over $\mathbb{R}^d$, which is, in practice, infeasible. Hence, a feasible numerical optimisation subroutine is used at each iteration to propose a reasonable new particle.

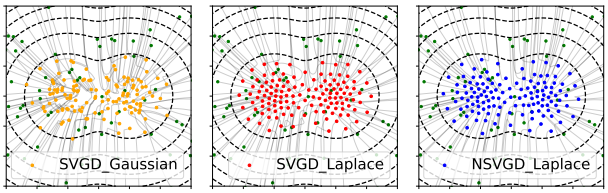


Figure 1. Example of a 2D Gaussian mixture. The configuration of 128 particles are plotted in green at initialization, and in different colors after convergence. The light grey curves correspond to their trajectories. From left to right: SVGD with Gaussian and Laplace kernel, γ=0.5, after 1000 iterations; NSVGD with Laplace kernel and γ=0.1, after 30 iterations.

Reaching quantization states. The convergence behavior of the algorithms considered differ greatly and they exhibit different levels of sensitivity to the properties of the target, choice of bandwidth and initialization. For example, while KSD descent converges reasonably to log-concave targets (e.g., a Gaussian), it can fail to converge when it is multimodal (Korba et al., 2021) (e.g., a mixture of Gaussians). MMD descent behavior can even be worse; it can fail to converge to a simple Gaussian target, hence the tuning of this algorithm is particularly tricky. This can be explained by the fact that the MMD is non convex with respect to the Wasserstein geometry (Arbel et al., 2019).

In contrast, SVGD particles always converged reasonably to the (unimodal and bimodal) target distribution in our experiments, for an appropriate choice of step-size, bandwith of the kernel and initialization of the particles. Further-

more, we found a significant improvement in the speed of convergence and slightly better results when using Normalized SVGD (NSVGD) with a Laplace kernel. We illustrate this with a simple experiment where the target distribution is a 2-dimensional mixture of Gaussians, using particles drawn from a uniform distribution in a ball of radius 5. Figure 1 displays the trajectories and final states of NSVGD and SVGD runs, with different optimization strategies and choices of kernel, with a constant bandwidth 1. NSVGD with a Laplace kernel leads to the best configuration after 30 iterations. Moreover, NSVGD with a fixed-time step γ benefits from a faster convergence than SVGD, as illustrated in Figure 2. This first experiment motivates the use of NSVGD over SVGD. We also evaluate these two algorithms on a Bayesian ICA (Independent Component Analysis) task, where the target distribution is highly non log-concave, and Bayesian logistic regression on 13 benchmark datasets, as in (Korba et al., 2021). Our results, provided in Appendix E, advocate for NSVGD versus its SVGD: it has either the same, or slightly better performance than SVGD, while converging much faster, in particular when the initial distribution has low-density regions.

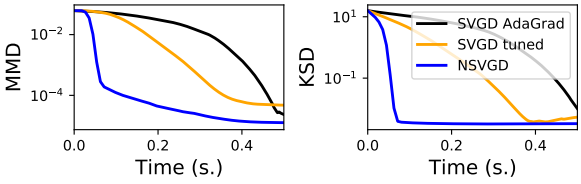


Figure 2. Convergence speed of SVGD (tuned time-step or AdaGrad) and Normalized SVGD (fixed time-step) on a 2D mixture of Gaussians, with 128 particles.

Practical considerations. The context in which these approaches can be used vary. MMD Descent can be used only in the case where the density or samples of π are available. KSD Descent only requires the score of π, but particles may be stuck in low density regions when the target is not unimodal (which we noticed in our experiments with a mixture of Gaussians target, see Appendix E) thus limiting the practical interest of this algorithm. Finally, SVGD and NSVGD also only requires the score of π, and are reasonably robust with respect to the choice of target distribution.

6.2. Quantization Rates

We start by visually comparing the sample sets obtained by different algorithms, Figure 3. In particular we have observed that for algorithms that use smooth kernels, like the Gaussian, the final states develop internal structures like the rings in figure (c). This is more often encountered for MMD and KSD minimization, but we observed it in SVGD as well. In contrast, the algorithms with kernels which are pointy at the origin, like the Laplace kernel, have stronger

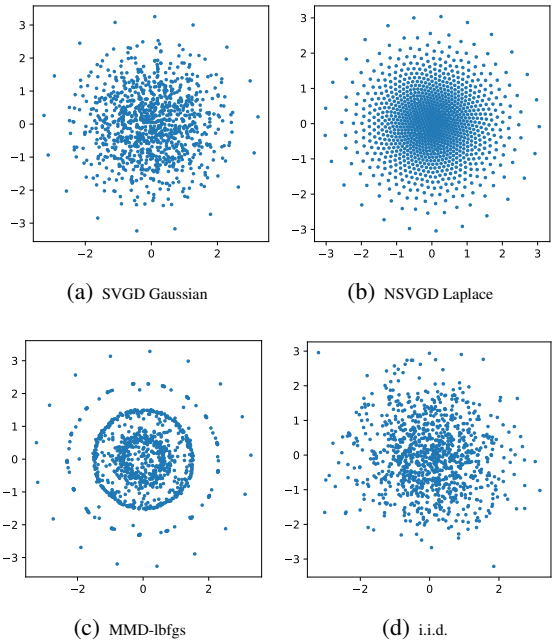


Figure 3. (a)-(c) Final states of the algorithms for 1024 particles, after 1e4 iterations. The kernel bandwidth for all algorithm is set to 1. Target measure is Gaussian $\mathcal{N}(0, I_2)$, whose i.i.d. samples are given for comparison, (d).

d	Eval.	SVGD	NSVGD	MMD-lbfgs	KSD-lbfgs	KH	SP
2	**KSD**	-0.98	-0.94	-1.48	-1.46	-0.84	-0.77
	MMD	-1.04	-1.00	-1.60	-1.54	-0.93	-0.77
3	**KSD**	-0.91	-0.81	-1.38	-1.44	-0.84	-0.78
	MMD	-0.96	-0.91	-1.51	-1.49	-0.92	-0.75
4	**KSD**	-0.91	-0.81	-1.35	-1.39	-0.89	–
	MMD	-0.94	-0.89	-1.46	-1.40	-0.95	–
8	**KSD**	-0.84	-0.80	-1.14	-1.16	–	–
	MMD	-0.77	-0.90	-1.25	-1.13	–	–

Table 1. Slopes for the quantization measured in KSD/MMD, for the different algorithms at study and several dimensions d. R^2 coefficient for the linear regression is always between $0.95 \sim 1$. The slopes remain much steeper than the Monte Carlo rate, even when the dimension increases.

local repulsion which results in a more regular distribution of points, see figure (b). Also, this choice of kernel is less sensitive to the bandwith as one can see Figure 5, and we focus on the latter in the following experiments.

The quality of sample sets. We now evaluate the quantization properties of the algorithms at study, for different target distributions and in different dimensions. We compute the MMD and KSD between the target π and its discrete approximation $\mu_n = {}^1\!/_n \sum_{i=1}^n \delta_{x_i}$ where the particles $(x_i)_{i=1}^n$ are either i.i.d. samples, or the output of the algorithms we mentioned earlier, for different values of n. We chose a Gaussian kernel for the MMD, since MMD between a dis-

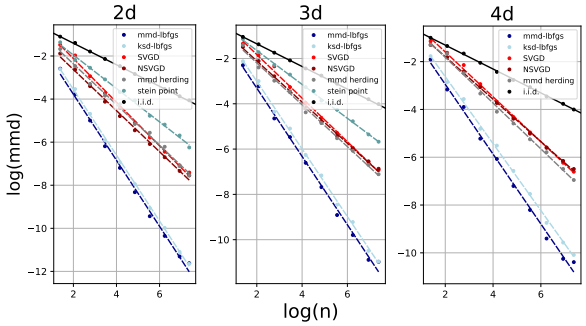


Figure 4. Quantization rates of the algorithms at study when $\pi = \mathcal{N}(0, {}^1\!/_d I_d)$. MMD/KSD Descent use bandwidth 1; SVGD use Laplace kernel with median trick; NSVGD use Laplace kernel with adaptive choice of bandwidth. Stein points use gridsize = 200 points in 2d, 50 in 3d; in 4d grid search was too slow.

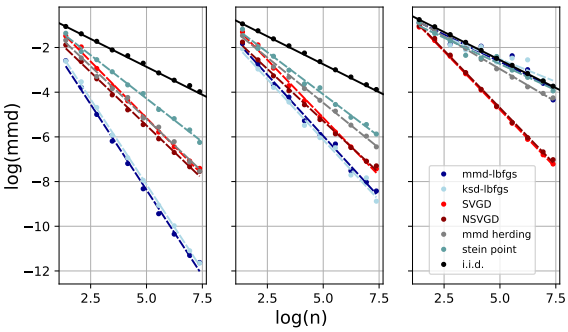


Figure 5. Importance of the choice of the bandwidth in the MMD evaluation metric when evaluating the final states, in 2d. From Left to Right: (evaluation) MMD bandwidth = 1, 0.7, 0.3.

crete distribution and a Gaussian (continuous) target can be computed in closed form; similarly for the KSD. The results are reported Figure 4 and illustrate the rate of convergence of the different methods with respect of the number of particles n, when the target is a Gaussian distribution, for $d = 2, 3, 4$. Each point is the result of averaging 10 runs of each algorithm run for 1e4 iterations, where the initial particles are i.i.d. samples of π (this does not apply to greedy algorithms). The slopes are reported in Table 1. They highlight that the algorithms at study show much faster rates than i.i.d. samples, which are faster than what we proved in Section 5. In the Appendix, we provide additional details and illustrations on the experiments of this Section, as well as additional experiments (e.g., with a Gaussian mixture target distribution). We observe the following. The quantization error of i.i.d. samples is of order $\mathcal{O}(n^{-\frac{1}{2}})$ and is always higher than the ones of the other methods. Greedy algorithms such as KH and SP enjoy appreciable quantization rates, but are computationally expensive due to the optimization subroutine when adding a new particle. As the global optimization cannot be performed exactly, the quantization errors of KH and SP may be overestimated, but they correspond to what is achieved in practice. Particles systems we considered, namely MMD Descent, KSD descent, SVGD and Normalised SVGD, showed the best rates of convergence. This is particularly the case for MMD and KSD descent, which are designed to minimize the MMD and KSD respectively, in contrast to SVGD. In fact we observe that the MMD and KSD quantization error of the associated flows have steeper slope (about -1.5 in low dimensions) than our theoretical guarantees of Section 5.

Robustness to evaluation discrepancy. In Figure 5 we compare the quality of the samples obtained by using MMD with different kernel bandwidths to evaluate the difference between the set of samples and the target distribution. The sample sets have been obtained by using the bandwidth 1 in all methods. We see that when the discrepancy is measured using the bandwidth which was used in the algorithms, MMD minimization provides the best quantization, as expected. However if we measure the discrepancy using a kernel with smaller bandwidth, MMD and KSD results deteriorate significantly and SVGD and NSVGD with Laplacian kernel perform the best. The likely reason is that the distribution of samples obtained by SVGD is more regular, while for MMD and KSD with Gaussian kernel the optimization can create internal structures which can affect the discrepancy at lower bandwidths. Since MMD measures the maximal integration error over the RKHS ball, the above figure suggests that the samples obtained using SVGD and Laplace kernel will perform better at integration tasks for wider families, which include less regular functions.

7. Conclusion

In this work, we studied quantization properties of interacting particle systems derived from Wasserstein (and related) gradient flows, such as SVGD, MMD and KSD descent. We highlighted both theoretically and numerically that they can create "super-samples", i.e. that they approximate the target distribution with a very fast rate compared to Monte Carlo samples, as measured by the MMD or KSD. Furthermore, we proposed a normalized version of SVGD which accelerates the dynamics and observed that Laplace kernels produce more regular sample point distributions. A number of open questions remain about the particle systems at study. In particular, proving quantization for SVGD is challenging, as it does not minimize a functional for discrete measures. Furthermore the fact that some of the observed quantization rates for KSD and MMD were faster than the guarantees we proved, points to possible theoretical improvements. Finally, it would be of great interest to establish non-asymptotic, unified bounds for these particle systems, i.e. that quantify the quantization properties these finite particles systems for a finite number of iterations of algorithm.

References

Adams, R. A. and Fournier, J. J. *Sobolev spaces*. Elsevier, 2003.

Aistleitner, C. and Dick, J. Low-discrepancy point sets for non-uniform measures. *Acta Arithmetica*, 163, 08 2013. doi: 10.4064/aa163-4-4.

Aistleitner, C. and Dick, J. Functions of bounded variation, signed measures, and a general Koksma-Hlawka inequality. *Acta Arith.*, 167(2):143–171, 2015. ISSN 0065-1036. doi: 10.4064/aa167-2-4. URL https://doi.org/10.4064/aa167-2-4.

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

Arbel, M., Korba, A., Salim, A., and Gretton, A. Maximum mean discrepancy gradient flow. *Advances in Neural Information Processing Systems*, 32, 2019.

Bach, F., Lacoste-Julien, S., and Obozinski, G. On the equivalence between herding and conditional gradient algorithms. In *ICML 2012 International Conference on Machine Learning*, 2012.

Briol, F.-X., Oates, C., Girolami, M., and Osborne, M. A. Frank-Wolfe bayesian quadrature: Probabilistic integration with theoretical guarantees. In *Advances in Neural Information Processing Systems*, volume 28, 2015. URL https://proceedings.neurips.cc/paper/2015/file/ba3866600c3540f67c1e9575e213be0a-Paper.pdf.

Chen, W. Y., Mackey, L., Gorham, J., Briol, F.-X., and Oates, C. J. Stein points. *International Conference on Machine Learning (ICML)*, 2018.

Chen, W. Y., Barp, A., Briol, F.-X., Gorham, J., Girolami, M., Mackey, L., Oates, C., et al. Stein point Markov Chain Monte Carlo. *International Conference on Machine Learning (ICML)*, 2019.

Chen, Y., Welling, M., and Smola, A. Super-samples from kernel herding. *arXiv preprint arXiv:1203.3472*, 2012.

Chopin, N. and Ducrocq, G. Fast compression of mcmc output. *Entropy*, 23(8):1017, 2021.

Chwialkowski, K., Strathmann, H., and Gretton, A. A kernel test of goodness of fit. In *International Conference on Machine Learning (ICML)*, pp. 2606–2615, 2016.

D'Angelo, F. and Fortuin, V. Annealed stein variational gradient descent. *arXiv preprint arXiv:2101.09815*, 2021.

D'Angelo, F., Fortuin, V., and Wenzel, F. On stein variational neural network ensembles. *arXiv preprint arXiv:2106.10760*, 2021.

Dick, J. and Pillichshammer, F. *Digital nets and sequences: discrepancy theory and quasi–Monte Carlo integration*. Cambridge University Press, 2010.

Duchi, J., Hazan, E., and Singer, Y. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research*, 12(7), 2011.

Duncan, A., Nüsken, N., and Szpruch, L. On the geometry of Stein variational gradient descent. *arXiv preprint arXiv:1912.00894*, 2019.

Fournier, N. and Guillin, A. On the rate of convergence in Wasserstein distance of the empirical measure. *Probab. Theory Related Fields*, 162(3-4): 707–738, 2015. ISSN 0178-8051. doi: 10.1007/s00440-014-0583-7. URL https://doi.org/10.1007/s00440-014-0583-7.

Frank, M., Wolfe, P., et al. An algorithm for quadratic programming. *Naval research logistics quarterly*, 3(1-2): 95–110, 1956.

Gorham, J. and Mackey, L. Measuring sample quality with kernels. In *International Conference on Machine Learning (ICML)*, volume 70, pp. 1292–1301. JMLR. org, 2017.

Graf, S. and Luschgy, H. *Foundations of quantization for probability distributions*, volume 1730 of *Lecture Notes in Mathematics*. Springer-Verlag, Berlin, 2000. ISBN 3-540-67394-6. doi: 10.1007/BFb0103945. URL https://doi.org/10.1007/BFb0103945.

Gretton, A., Borgwardt, K., Rasch, M., Schölkopf, B., and Smola, A. A kernel method for the two-sample-problem. *Advances in neural information processing systems*, 19: 513–520, 2006.

Gretton, A., Borgwardt, K. M., Rasch, M. J., Schölkopf, B., and Smola, A. A kernel two-sample test. *Journal of Machine Learning Research*, 13:723–773, 2012.

Hauser, N. and Mikhailov, S. Applied computational methods. 2022. URL http://www.ltcc.ac.uk/courses/applied-computational-methods/.

Hodgkinson, L., Salomone, R., and Roosta, F. The reproducing stein kernel approach for post-hoc corrected sampling. *arXiv preprint arXiv:2001.09266*, 2020.

Kanagawa, M., Hennig, P., Sejdinovic, D., and Sriperumbudur, B. K. Gaussian processes and kernel methods: A

review on connections and equivalences. *arXiv preprint arXiv:1807.02582*, 2018.

Khanna, R., Hodgkinson, L., and Mahoney, M. W. Geometric rates of convergence for kernel-based sampling algorithms. In *Uncertainty in Artificial Intelligence*, pp. 2156–2164. PMLR, 2021.

Korba, A., Salim, A., Arbel, M., Luise, G., and Gretton, A. A non-asymptotic analysis for stein variational gradient descent. *Advances in Neural Information Processing Systems*, 33, 2020.

Korba, A., Aubin-Frankowski, P.-C., Majewski, S., and Ablin, P. Kernel Stein discrepancy descent. *arXiv preprint arXiv:2105.09994*, 2021.

Łatuszyński, K., Miasojedow, B., and Niemiro, W. Nonasymptotic bounds on the estimation error of mcmc algorithms. *Bernoulli*, 19(5A):2033–2066, 2013.

Liu, C. and Zhu, J. Riemannian stein variational gradient descent for bayesian inference. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

Liu, D. C. and Nocedal, J. On the limited memory BFGS method for large scale optimization. *Mathematical programming*, 45(1-3):503–528, 1989.

Liu, Q. Stein variational gradient descent as gradient flow. In *Advances in Neural Information Processing Systems*, pp. 3115–3123, 2017.

Liu, Q. and Wang, D. Stein variational gradient descent: A general purpose Bayesian inference algorithm. In *Advances in Neural Information Processing Systems*, pp. 2378–2386, 2016.

Liu, Q., Lee, J., and Jordan, M. A kernelized Stein discrepancy for goodness-of-fit tests. In *International Conference on Machine Learning (ICML)*, pp. 276–284, 2016.

Lu, J., Lu, Y., and Nolen, J. Scaling limit of the Stein variational gradient descent: The mean field regime. *SIAM Journal on Mathematical Analysis*, 51(2):648–671, 2019.

Lu, Y. and Lu, J. A universal approximation theorem of deep neural networks for expressing probability distributions. *Advances in Neural Information Processing Systems*, 33, 2020. URL https://proceedings.neurips.cc/paper/2020/file/2000f6325dfc4fc3201fc45ed01c7a5d-Paper.pdf.

Manole, T., Balakrishnan, S., and Wasserman, L. Minimax confidence intervals for the sliced wasserstein distance. *Electronic Journal of Statistics*, 16:2252–2345, 2022.

Merigot, Q., Santambrogio, F., and Sarrazin, C. Non-asymptotic convergence bounds for Wasserstein approximation using point clouds. *arXiv preprint arXiv:2106.07911*, 2021.

Oates, C. J. Minimum discrepancy methods in uncertainty quantification. *arXiv preprint arXiv:2109.06075*, 2021.

Oates, C. J., Girolami, M., and Chopin, N. Control functionals for Monte Carlo integration. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 79 (3):695–718, 2017.

Pronzato, L. Performance analysis of greedy algorithms for minimising a Maximum Mean Discrepancy. *arXiv preprint arXiv:2101.07564*, 2021.

Pu, Y., Gan, Z., Henao, R., Li, C., Han, S., and Carin, L. Vae learning via stein variational gradient descent. *arXiv preprint arXiv:1704.05155*, 2017.

Riabiz, M., Chen, W., Cockayne, J., Swietach, P., Niederer, S. A., Mackey, L., Oates, C., et al. Optimal thinning of mcmc output. *arXiv preprint arXiv:2005.03952*, 2020.

Roberts, G. O. and Rosenthal, J. S. General state space Markov chains and mcmc algorithms. *Probability surveys*, 1:20–71, 2004.

Scott, D. W. On optimal and data-based histograms. *Biometrika*, 66(3):605–610, 1979.

Smola, A., Gretton, A., Song, L., and Schölkopf, B. A hilbert space embedding for distributions. In *International Conference on Algorithmic Learning Theory*, pp. 13–31. Springer, 2007.

Sobol, I. M. On quasi-Monte Carlo integrations. *Mathematics and computers in simulation*, 47(2-5):103–112, 1998.

Sriperumbudur, B. On the optimal estimation of probability measures in weak and strong topologies. *Bernoulli*, 22 (3):1839–1893, 2016.

Sriperumbudur, B. K., Fukumizu, K., Gretton, A., Lanckriet, G. R., and Schölkopf, B. Kernel choice and classifiability for rkhs embeddings of probability distributions. In *NIPS*, volume 22, pp. 1750–1758, 2009.

Steinwart, I. and Christmann, A. *Support Vector Machines*. Information Science and Statistics. Springer New York, 2008. ISBN 9780387772424. URL https://books.google.com/books?id=HUnqnrpYt4IC.

Teymur, O., Gorham, J., Riabiz, M., and Oates, C. Optimal quantisation of probability measures using maximum mean discrepancy. In *International Conference on Artificial Intelligence and Statistics*, pp. 1027–1035. PMLR, 2021.

Tolstikhin, I., Sriperumbudur, B. K., and Muandet, K. Minimax estimation of kernel mean embeddings. *The Journal of Machine Learning Research*, 18(1):3002–3048, 2017.

Tsuji, K. and Tanaka, K. Acceleration of the kernel herding algorithm by improved gradient approximation. *arXiv preprint arXiv:2105.07900*, 2021.

Wenliang, L. K. and Kanagawa, H. Blindness of score-based methods to isolated components and mixing proportions. *arXiv preprint arXiv:2008.10087*, 2020.

A. Lemmas

We first derive the equations for the NSVGD flow.

Lemma A.1. *The gradient flow of the* KL *divergence with respect to the Stein geometry corresponding to the normalized kernel K_μ defined in (9) is given by (17).*

We observe that when this flow is applied to particles it takes the form (10) thus confirming the derivation of the equations for NSVGD.

Proof. We denote $\int K_\mu(x,y)u(y)dy = K_\mu * u(x)$, then for $v = K_\mu \star u \in \mathcal{H}^d_{K_\mu} = \mathcal{H}_{K_\mu} \times \cdots \times \mathcal{H}_{K_\mu}$, we can define the new metric tensor to be

$$g_\mu(v,v) = \iint K_\mu(x,y)u(x)\cdot u(y)dxdy.$$

The corresponding Rayleigh functional is

$$\mathcal{R}(u) = \frac{1}{2}\iint K_\mu(x,y)u(x)\cdot u(y)dxdy + \int (\nabla\mu + \mu\nabla U)(x)\cdot K_\mu * u(x)dx. \tag{16}$$

Taking the first variation of $\mathcal{R}(u)$, we have

$$\frac{\delta\mathcal{R}}{\delta u}(w) = \int K_\mu * (u + \nabla\mu + \mu\nabla U)\cdot w dx = 0.$$

We then conclude that the gradient flow of relative entropy with respect to this new metric is

$$\frac{\partial\rho_t}{\partial t} = \nabla\cdot(\rho_t(K_\mu * \nabla\rho_t + K_\mu * (\rho_t\nabla U))). \tag{17}$$

Therefore, the resulting gradient flow, i.e. NSVGD gradient flow, has the velocity vector field which can be written explicitly using the kernel (9) as

$$\begin{aligned} v_\mu &= -K_\mu \star \nabla\mu - K_\mu \star (\mu\nabla U) \\ &= \int (\nabla_y K_\mu(x,y) - K_\mu(x,y)\nabla U(y))\, d\mu(y) \\ &= -\int \left(\frac{1}{\mu_h(x)^{1/2}}\left(\frac{1}{\rho_h(y)^{1/2}}\nabla\eta(x-y) + \frac{\eta(x-y)}{2\rho_h(y)^{3/2}}\mu * \nabla\eta_h(y)\right)\right) + K_\mu(x,y)\nabla U(y)\Big)d\mu(y). \end{aligned}$$

where the second equality uses an integration by part that holds as $\lim_{\|x\|\to\infty} K_\mu(x,.)\mu(x) = 0$ with a vanishing μ. □

Lemma A.2. *Assume that the kernel k is bounded and twice differentiable with bounded derivatives. Assume furthermore that the distribution π is light-tailed with potential $U \in C^1(\mathbb{R}^d)$ and $U(x) = O(\|x\|^m)$ with some $m \in \mathbb{N}$, the following inequality holds:*

$$\mathrm{KSD}^2(\mu_n|\pi) \le \iint_{A_n} 2\left(s(x)\cdot s(y)k(x,y) + \nabla_{x,y}k(x,y)\right) d(\mu_n-\mu)(x)d(\mu_n-\mu)(y) + \frac{C_0}{n^2}$$

where $s(x) = \nabla U(x)$.

Proof. First $\int k_\pi(x,.)d\pi(x) = 0$ holds from Stein's identity, see Oates et al. (2017, Lemma 1). We have

$$\begin{aligned} \mathrm{KSD}^2(\mu_n|\pi) &= \iint_{\mathbb{R}^d} k_\pi(x,y)d(\mu_n-\pi)(x)d(\mu_n-\pi)(y) \\ &= \iint_{\mathbb{R}^d} \left(s(x)\cdot s(y)k(x,y) + \nabla_{x,y}k(x,y) + 2s(x)\cdot\nabla_y k(x,y)\right) d(\mu_n-\pi)(x)d(\mu_n-\pi)(y) \end{aligned}$$

By assumptions, the measure $d\rho = \|s(x)\|e^{-U(x)}dx$ is light-tailed and $\|\nabla_y k(x,y)\|$ is uniformly bounded, we can take $\{x_i\}_{i=1}^n \subset A_n = [-\log n, \log n]^d$ and construct μ compactly supported on A_n as in Proposition 5.3 such that

$$\mathrm{KSD}^2(\mu_n|\pi) = \iint_{A_n} \left(s(x)\cdot s(y)k(x,y) + \nabla_{x,y}k(x,y) + 2s(x)\cdot\nabla_y k(x,y)\right) d(\mu_n-\mu)(x)d(\mu_n-\mu)(y) + \frac{C_0}{n^2}$$

with some constant C_0. Note that $\nabla_y k(x,y) \in \mathcal{H}_k^d$ Steinwart & Christmann (2008, Lemma 4.34), so $f_1(y) = \int \nabla_y k(x,y) d(\mu_n - \mu)(x) \in \mathcal{H}_k^d$ from (Smola et al., 2007). Also, the entries of $s(x)d(\mu_n - \mu)(x)$ are finite signed measures, its kernel embedding $f_2(x) = \int k(x,y)s(y)d(\mu_n - \mu)(y) \in \mathcal{H}_k^d$. Now, by reproducing property and Steinwart & Christmann (2008, Lemma 4.34),

$$\iint 2s(x) \cdot \nabla_y k(x,y) d(\mu_n - \pi)(x) d(\mu_n - \pi)(y) = 2\langle f_1, f_2 \rangle_{\mathcal{H}_k^d} \leq \|f_1\|_{\mathcal{H}_k^d}^2 + \|f_2\|_{\mathcal{H}_k^d}^2$$
$$= \iint_{\mathbb{R}^d} (s(x) \cdot s(y) k(x,y) + \nabla_{x,y} k(x,y)) d(\mu_n - \pi)(x) d(\mu_n - \pi)(y),$$

which concludes the proof. □

Lemma A.3. *Let k be a bounded kernel,then for finite signed Borel measure μ on Borel set $M \subset \mathbb{R}^d$ and any vector-valued function v with $\|v\| \in L^2(M; |\mu|)$,*

$$\sup_{f \in \mathcal{H}_k^d, ||f||_{\mathcal{H}_k^d} \leqslant 1} \left| \int_M v \cdot f d\mu \right|^2 = \iint_M k(x,y) v(x) \cdot v(y) d\mu(x) d\mu(y)$$

Proof. For any vector-valued measure $dP = v d\mu$, let $T_p : \mathcal{H}_k^d \mapsto \mathbb{R}$ be the linear functional defined as

$$T_P(f) = \int_M f(x) \cdot dP(x)$$

with $\|T_P\| := \sup_{f \in \mathcal{H}_k^d, f \neq 0} \frac{|T_P(f)|}{\|f\|_{\mathcal{H}_k^d}}$. Note that this functional is bounded: let $B = \sup_{x \in M} |k(x,x)|$, then for any $f \in \mathcal{H}_k^d$, by Hölder's inequality and $f(x) = \langle f, k(\cdot, x) \rangle_{\mathcal{H}_k}$, there exists constant C such that

$$|T_P(f)| = \left| \int_M f \cdot v \, d\mu \right| = \left| \int_M \sum_{i=1}^d v_i \langle f_i, k(\cdot, x) \rangle_{\mathcal{H}_k} d\mu \right| \leqslant \left(\int_M B \|f\|_{\mathcal{H}_k^d}^2 d|\mu| \cdot \int_M \|v\|^2 d|\mu| \right)^{1/2} < C \|f\|_{\mathcal{H}_k^d}.$$

By Riesz representation theorem, there exists a unique $\lambda_P = [\lambda_{P_i}]_{i=1}^d \in \mathcal{H}_k^d$ s.t. $T_P(f) = \langle f, \lambda_P \rangle_{\mathcal{H}_k^d}$. Thus for any $u = [u_i]_{i=1}^d \in U$, define $k_u(\cdot) = [k(\cdot, u_1), ...k(\cdot, u_d)]^T$, we have

$$T_P([k_u(\cdot)]) = \int_M \sum_{i=1}^d k(x, u_i) v_i(x) d\mu(x) = \langle k_u(\cdot), \lambda_P \rangle_{\mathcal{H}_k^d} = \sum_{i=1}^d \langle k(\cdot, u_i), \lambda_{P_i} \rangle_{\mathcal{H}_k} = \sum_{i=1}^d \lambda_{P_i}(u_i),$$

which implies $\lambda_{P_i}(x) = \int_M k(y,x) v_i(y) d\mu(y)$, $i = 1, ..., d$. We conclude

$$\sup_{f \in \mathcal{H}_k^d, ||f||_{\mathcal{H}_k^d}} \left| \int_M v \cdot f d\mu \right| = \sup_{f \in \mathcal{H}_k^d, ||f||_{\mathcal{H}_k^d}} \langle f, \lambda_P \rangle_{\mathcal{H}_k^d} = \|\lambda_P\|_{\mathcal{H}_k^d} = \sqrt{\sum_{i=1}^d \iint_M k(x,y) v_i(x) v_i(y) d\mu(x) d\mu(y)}. \quad \square$$

B. Proof of Theorem 5.1 under Assumption 2

Denote by $\mathcal{H}_k$ the RKHS of k. Recall that, see Kanagawa et al. (2018, Theorem 2.4):

$$\mathcal{H}_k = \left\{ f \in C(\mathbb{R}^d) \cap L^2(\mathbb{R}^d) \, : \, \|f\|_{\mathcal{H}_k}^2 := \frac{1}{(2\pi)^{d/2}} \int_{\mathbb{R}^d} |\hat{\eta}(\xi)|^{-1} |\hat{f}(\xi)|^2 d\xi < \infty \right\}$$

where $\hat{f}$, $\hat{\eta}$ are the Fourier transform of f and η respectively, where the Fourier transform

$$\hat{f}(\xi) = \frac{1}{(2\pi)^{d/2}} \int_{\mathbb{R}^d} f(x) e^{-i\xi x} dx.$$

Recall also that for $j \geq 0$, the H^j-Sobolev norm of f is spectrally defined as

$$\|f\|^2_{H^j(\mathbb{R}^j)} = \int (1 + \|\xi\|^2)^j |\hat{f}(\xi)|^2 d\xi.$$

We note that for any d, by Assumption 1 there exists a constant $C_{k,d}$ such that for all ξ, $(1+\|\xi\|^2)^d \leq C_{k,d}(2\pi)^{d/2}|\hat{\eta}(\xi)|^{-1}$. Hence, for any $f \in \mathcal{H}_k$, $\|f\|_{H^d(\mathbb{R}^d)} \leq \|f\|_{\mathcal{H}_k}$.

To control the integrals on the faces, we recall the Sobolev trace theorem from $\mathbb{R}^{j+1}$ to a hyperplane $\mathbb{R}^j$. From Hauser & Mikhailov (2022, Theorem 4.1), it follows that

$$\|f\|_{H^j(\mathbb{R}^j)} \leq \sqrt{2(j+1)}\|f\|_{H^{j+1}(\mathbb{R}^{j+1})}.$$

Hence, recursively we obtain

$$\|f\|_{H^j(\mathbb{R}^j)} \leq 2^{\frac{d-j}{2}} \sqrt{\frac{d!}{j!}}\, \|f\|_{H^d(\mathbb{R}^d)}.$$

For $\alpha \subseteq \{1, \ldots, d\}$ we let $|\alpha| = j$. Let $[0,1]^j$ be the face of the cube corresponding to the coordinates $\alpha_1, \ldots, \alpha_j$. Let $j = |\alpha|$. Then by Parseval's identity,

$$\|\partial^j f\|_{L^1([0,1]^j)} \leq \|\partial^j f\|_{L^2(\mathbb{R}^j)} \leq \left(\int_{\mathbb{R}^j} \|\xi\|^{2j} |\hat{f}(\xi)|^2 d\xi\right)^{\frac{1}{2}} \leq \|f\|_{H^j(\mathbb{R}^j)} \leq 2^{\frac{d-j}{2}} \sqrt{\frac{d!}{j!}}\, \|f\|_{H^d(\mathbb{R}^d)}.$$

Summing up over all of the faces we obtain an upper bound on variation (14):

$$V(f) = \sum_{j=1}^{d} \sum_{\alpha \subseteq \{1,\ldots,d\}, |\alpha|=j} \|\partial^j f\|_{L^1([0,1]^j)} \leq 3^d \sqrt{d!}\, \|f\|_{H^d(\mathbb{R}^d)}.$$

The result now follows, as before, by the Koksma-Hlawka inequality (15).

C. Proof of Proposition 5.3

Proof. From triangle inequality, for any $\mu \in \mathcal{P}(\mathbb{R}^d)$,

$$\mathrm{MMD}(\pi, \mu_n) \leq \mathrm{MMD}(\pi, \mu) + \mathrm{MMD}(\mu, \mu_n).$$

Denote $A_n = [-\log n, \log n]^d$. Recall that we assumed that π is light-tailed, i.e. $\pi(\|x\| \geq t) \leq c\exp(-\lambda t)$ for some $\lambda, c \geq 0$. Hence, for some constant $C \geq 0$, $\pi(\mathbb{R}^d \setminus A_{n/2}) \leq \frac{C}{n}$ and without loss of generality we assume $\pi(A_n) > \pi(A_{n/2})$. We can define a probability measure $\mu = \pi\big|_{A_{n/2}} + \frac{1-\pi(A_{n/2})}{\pi(A_n)-\pi(A_{n/2})}\pi\big|_{A_n \setminus A_{n/2}}$ such that μ is compactly supported in $A_n = [-\log n, \log n]^d$ and

$$\|\pi - \mu\|_{TV(\mathbb{R}^d)} \leq 2(1 - \pi(A_{n/2})) \leq \frac{C'_{\pi,d}}{n}$$

where $\|\cdot\|_{TV}$ is the total variation distance and $C'_{\pi,d}$ is a positive constant. Then, since there exists $B > 0$ such that $k(x,y) \leq B$ for any $x, y \in \mathbb{R}^d$,

$$\mathrm{MMD}^2(\pi, \mu) = \iint_{\mathbb{R}^d} k(x,y) d(\pi - \mu)(x) d(\pi - \mu)(y) \leq B^2 \|\pi - \mu\|^2_{TV(\mathbb{R}^d)} \leq \frac{C''_{\pi,d}}{n^2},$$

for some constant $C''_{\pi,d}$. We now turn to bounding $\mathrm{MMD}(\mu, \mu_n)$. Define $T : \mathbb{R}^d \to \mathbb{R}^d$, $T(x) = \log n \cdot (x - [\frac{1}{2}, ..., \frac{1}{2}])$. For n large enough, consider μ_n supported on A_n. Let $\nu = T^{-1}_{\#}\mu$, $\nu_n = T^{-1}_{\#}\mu_n$ and $X_n = (x_1, \ldots, x_n)$. For any $f \in \mathcal{H}_k$,

$$\left|\int_{A_n} f d\mu - \frac{1}{n}\sum_{i=1}^{n} f(x_i)\right| = \left|\int_{A_n} f(x) d(\mu - \mu_n)(x)\right| = \left|\int_{[0,1]^d} f \circ T d(\nu - \nu_n)\right| \leq \mathcal{D}(T^{-1}(X_n), \nu) V(f \circ T),$$

which implies that

$$\mathrm{MMD}(\mu,\mu_n) \leq \sup_{f\in\mathcal{H}_k,\|f\|_{\mathcal{H}_k}\leq 1} D(T^{-1}(X_n),\nu)V(f\circ T).$$

We will choose X_n so that $D(T^{-1}(X_n),\nu)$ is bounded as in the proof of Theorem 5.1. To bound the Hardy-Krause variation, observe that under Assumption 1,

$$V(f\circ T) = \sum_{\alpha\subseteq\{1,\dots,d\}} \int_{[0,1]^{|\alpha|}} \left|\frac{\partial^{|\alpha|} f\circ T(x_\alpha,1)}{\partial x_\alpha}\right| dx_\alpha \leq \sum_{\alpha\subseteq\{1,\dots,d\}} (\log n)^d \left\|\frac{\partial^{|\alpha|}k((x_\alpha,1),\cdot)}{\partial^{|\alpha|}x_\alpha}\right\|_{\mathcal{H}_k} \|f\|_{\mathcal{H}_k} \leq C_{k,d}2^d(\log n)^d,$$

where $c_{1,d} = C_{k,d}2^d$. Under Assumption 2, we obtain:

$$V(f\circ T) \leq 3^d\sqrt{d!}\,\|f\circ T\|_{H^d(\mathbb{R}^d)} \leq 3^d\sqrt{C_{k,d}d!}(2\pi)^{d/4}(\log n)^d\|f\|_{\mathcal{H}_k} \leq c_{2,d}(\log n)^d,$$

where $c_{2,d} = 3^d\sqrt{C_{k,d}d!}(2\pi)^{d/4}$. In both cases, multiplying $c_{1,d}$ or $c_{2,d}$ by $C_{\pi,d}$, there exists a constant c_d depending on the dimension such that $\mathrm{MMD}(\mu,\mu_n) \leq \frac{c_d(\log n)^{\frac{5d+1}{2}}}{n}$. Hence there exists C_d such that $\mathrm{MMD}(\pi,\mu_n) \leq C_d\frac{(\log n)^{\frac{5d+1}{2}}}{n}$. □

D. Proof of Theorem 5.4

Proof. Assume $k(x,y) = e^{-\frac{\|x-y\|^2}{2h^2}}$ and denote $s(x) = -\nabla U(x)$. By Lemma A.2, there exists a measure μ and a constant C_0 such that

$$\mathrm{KSD}^2(\mu_n|\pi) \leq \frac{C_0}{n^2} + \iint_{A_n} 2\left(k_1(x,y)+k_2(x,y)\right)d(\mu_n-\mu)(x)d(\mu_n-\mu)(y)$$

where $k_1(x,y) = s(x)\cdot s(y)k(x,y)$ and $k_2(x,y) = \nabla_{x,y}k(x,y)$. Since k_2 is a bounded kernel which satisfies Assumption 1, $\mathrm{MMD}_{k_2}(\mu,\mu_n)$ can be upper bounded using Proposition 5.3. Hence, we now turn to derive an upper bound for $\mathrm{MMD}_{k_1}(\mu,\mu_n)$. Note that since partial derivatives of U are bounded by a polynomial V of degree $m\geq 0$, $\|s(x)\| \in L^2(A_n;|\mu-\mu_n|)$, and Lemma A.3 yields:

$$\iint_{A_n} s(x)\cdot s(y)k(x,y)d(\mu_n-\mu)(x)d(\mu_n-\mu)(y) = \sup_{\|f\|_{\mathcal{H}_k^d}\leq 1}\left|\int_{A_n} f(x)\cdot s(x)d(\mu_n-\mu)(x)\right|.$$

Let $\varphi\in C^\infty(\mathbb{R},[0,1])$ be a cut-off function, i.e. such that $\varphi(x)=1$ on $x\in[-1,1]$ and $\varphi(x)=0$ whenever $x\geq 2$. For any $a>0$, let $\varphi_a(x)=\varphi(x/a)$. Note that for $i=1,\dots,d$, the i-th coordinate of the score $s_i\in H^d(2\sqrt{d}[-\log n,\log n]^d)$ and that $\|s_i\|_{H^d(2\sqrt{d}[-\log n,\log n]^d)} \leq C'_d(\log n)^{m+d/2}$. Indeed,

$$\begin{aligned}\|s_i\|^2_{H^d(2\sqrt{d}[-\log n,\log n]^d)} &= \sum_{\beta=(\beta_1,\dots,\beta_k),|\beta|\leq d}\int_{2\sqrt{d}[-\log n,\log n]^d}\left|\frac{\partial^\beta s_i(x)}{\partial^{\beta_1}x_1\dots\partial^{\beta_j}x_j}\right|^2 dx\\ &\leq 2^d\int_{2\sqrt{d}[-\log n,\log n]^d}|V(x)|^2dx \leq C'^2 2^d(2\sqrt{d}\log n)^{2m}\int_{2\sqrt{d}[-\log n,\log n]^d}dx \leq C'_d(\log n)^{2m+d},\end{aligned}$$

using that $|V(x)|\leq C'\|x\|^m$ for some constant $C'>0$ and C'_d depending on the dimension.

Let g_i be the real-valued function defined by $g_i(x) = s_i(x)\varphi_{\sqrt{d}\log n}(\|x\|)$ for $i=1,\dots,d$. The vector-valued function $g(x)=[g_1(x),...,g_d(x)]$ satisfies

$$g(x) = \begin{cases} s(x), & x\in A_n;\\ 0, & \|x\|\geq 2\sqrt{d}\log n.\end{cases}$$

Since $\log n\geq 1/2$ for $n\geq 2$ there exists a constant C''_d depending on d

$$\|g_i\|_{H^d(\mathbb{R}^d)} \leq C''_d\|s_i\|_{H^d(2\sqrt{d}[-\log n,\log n]^d)}.$$

Thus there exists a constant C'''_d depending on d such that $\|g_i\|_{H^d(\mathbb{R}^d)} \leq C'''_d(\log n)^{m+d/2}$, for any $i=1,...d$.

Note that $\mathcal{H}_k$ continuously embeds into $H^d(\mathbb{R}^d)$ and $H^d(\mathbb{R}^d)$ is an algebra, see Adams & Fournier (2003, Theorem 4.39). Thus we have for any $f \in \mathcal{H}_k^d$ and $l \in H^d(\mathbb{R}^d; \mathbb{R}^d)$, there exists a constant c_d such that

$$\|f \cdot l\|_{H^d(\mathbb{R}^d)} \leq c_d \|f\|_{\mathcal{H}_k^d} \|l\|_{H^d(\mathbb{R}^d;\mathbb{R}^d)},$$

where $f \cdot l(x) = \sum_{i=1}^d f_i(x) \cdot l_i(x)$. This implies that for the for the function g defined above,

$$\begin{aligned}\sup_{\|f\|_{\mathcal{H}_k^d} \leq 1} \left| \int_{A_n} f(x) \cdot s(x) d(\mu_n - \mu)(x) \right| &= \sup_{\|f\|_{\mathcal{H}_k^d} \leq 1} \left| \int_{A_n} f(x) \cdot g(x) d(\mu_n - \mu)(x) \right| \\ &\leq \sup_{\|h\|_{H^d(\mathbb{R}^d)} \leq c_d C_d''' (\log n)^{m+d/2}} \left| \int_{A_n} h(x) d(\mu_n - \mu)(x) \right|.\end{aligned}$$

We then apply exactly the same approach in Proposition 5.3:

$$\sup_{\|h\|_{H^d(\mathbb{R}^d)} \leq C_3 (\log n)^{m+d/2}} \left| \int_{A_n} h(x) d(\mu_n - \mu)(x) \right| \leq \sup_{\|h\|_{H^d(\mathbb{R}^d)} \leq c_d C_d''' (\log n)^{m+d/2}} \mathcal{D}(T^{-1}(X_n), \nu) V(h \circ T),$$

since there exists a constant $C_d^{(4)}$ depends on d such that

$$V(h \circ T) \leq 3^d \sqrt{d!} \, \|h \circ T\|_{H^d(\mathbb{R}^d)} \leq C_d^{(4)} (\log n)^{m+d/2},$$

and use the bound on the star discrepancy in the proof of Theorem 5.1, there exists $C_d^{(5)}$ such that

$$\mathrm{MMD}_{k_1}(\mu, \mu_n) \leq C_d^{(5)} \frac{(\log n)^{\frac{6d+2m+1}{2}}}{n}.$$

It follows that

$$\mathrm{KSD}^2(\mu_n | \pi) \leq \frac{C_0}{n^2} + \mathrm{MMD}_{k_1}^2(\mu, \mu_n) + \mathrm{MMD}_{k_2}^2(\mu, \mu_n) \leq C_d^2 \frac{(\log n)^{6d+2m+1}}{n^2}$$

holds with a dimension-dependent constant C_d. □

E. Additional Experiments

The code to reproduce the experiments are available at https://github.com/xulant/accurate-quantization-and-nsvgd.

E.1. Comparison of NSVGD and SVGD on Bayesian ICA and Bayesian Logistic regression

In this section we compare the performance of NSVGD and SVGD on real-world problems, i.e. on a synthetic Bayesian ICA task in $R^{p \times p}$, and on logistic regression on 13 benchmark datasets. Both settings are described in (Korba et al., 2021). For Bayesian ICA, initial particles are sampled from the uniform distribution; SVGD is run with AdaGrad for 3000 iterations with well-tuned bandwidth, while NSVGD is run for 1000 iterations. Results are averaged over 50 independent experiments, with 50 $p \times p$ matrices (i.e., the number of particles in this experiment). Figure 6 and Figure 7 illustrate our results on the synthetic Bayesian ICA task. We can notice that NSVGD converges to a configuration that is always better (i.e., that corresponds to lower Amari distance) than SVGD, and faster. Table 2 gives our results on logistic regression on the 13 benchmark datasets. Generally, NSVGD converges much faster to either similar or slightly better solutions than SVGD.

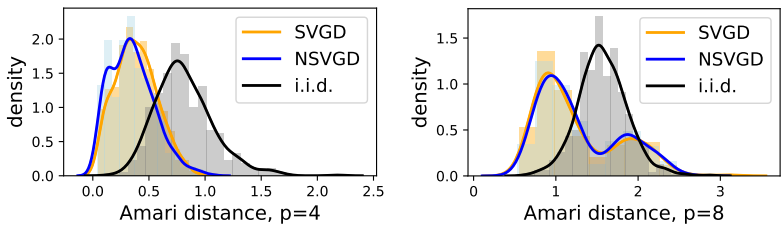
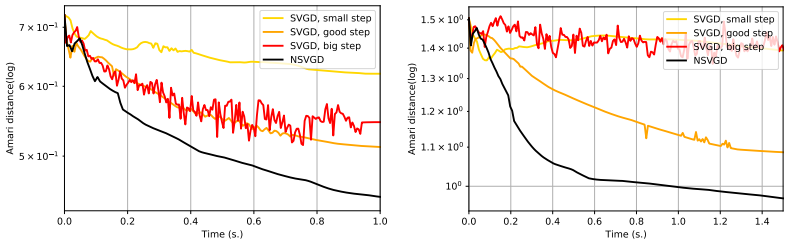


Figure 6. Distribution of Amari distances in the Bayesian ICA tasks for $p = 4$ and $p = 8$.



(a) p=4 (b) p=8

Figure 7. Convergence of Amari distances in the Bayesian ICA tasks. Initial particles are sampled from the uniform distribution on the centered ball with radius 10. Results averaged over 50 independent experiments, with 50 $p \times p$ matrices for each round. We can see that it is hard to find a reasonable step size for SVGD in this setting, while NSVGD works well with a fixed step size.

Dataset	banana	cancer	diabetes	solar	german	heart	image	ringnorm	splice	thyroid	titanic	twonorm	waveform
SVGD	0.60	0.75	0.76	0.66	0.80	0.80	0.82	0.75	0.85	0.85	0.78	0.98	0.87
NSVGD	0.60	0.75	0.76	0.66	0.80	0.81	0.82	0.75	0.85	0.85	0.78	0.98	0.87

Table 2. Bayesian logistic regression accuracies, with a fine-tuned bandwidth, averaged on 10 experiments

E.2. Additional Results and Illustrations on Previous Quantization Experiments

Figure 8 illustrates the quantization rates of the same experiments reported in Figure 4, but where the final states are evaluated in KSD instead of the MMD, for $d = 2 - 4$. **??** reports the results of the experiments when $d = 8$.

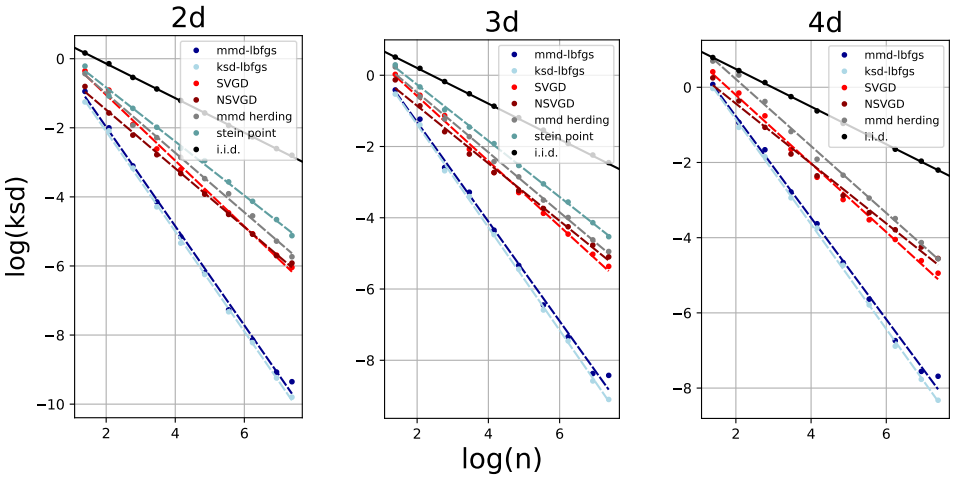


Figure 8. Quantization rates measured in KSD of the algorithms at $\pi = \mathcal{N}(0, {}^1/_d I_d)$.

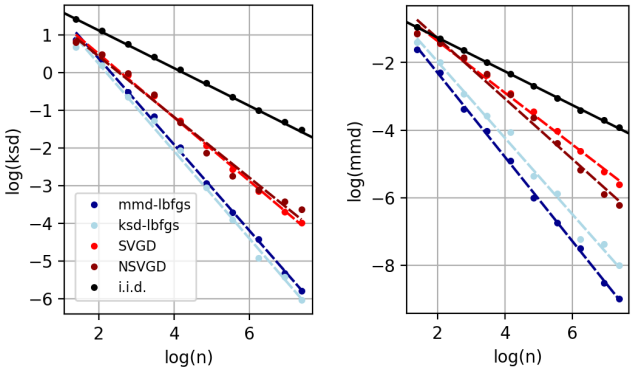


Figure 9. Quantization rates of the algorithms at study for $\pi = \mathcal{N}(0, 1/8 I_8)$, under the same setting as the 2-4d experiments on Figure 4.

Algorithm	Eval.	$n = 4$	$n = 16$	$n = 64$	$n = 256$	$n = 1024$
MMD-lbfgs	**KSD**	3.41± 0.59 e-01	4.65± 0.26 e-02	5.61± 0.60 e-03	6.95± 0.39 e-04	1.12± 0.09 e-04
	MMD	6.43± 1.39 e-02	7.08± 0.49 e-03	7.30± 0.58 e-04	7.73± 0.33 e-05	1.18± 0.08 e-05
KSD-lbfgs	**KSD**	2.86± 0.01 e-01	4.25± 0.02 e-02	4.94± 0.49 e-03	6.43± 0.33 e-04	9.44± 0.36 e-05
	MMD	7.37± 0.02 e-02	9.34± 0.70 e-03	9.72± 0.86 e-04	1.07± 0.06 e-04	1.47± 0.03 e-05
SVGD	**KSD**	7.02± 0.01 e-01	1.50± 0.02 e-01	3.99± 0.03 e-02	1.16± 0.02 e-02	3.39± 0.04 e-03
	MMD	2.26± 0.01 e-01	5.31± 0.03 e-02	1.23± 0.02 e-02	3.10± 0.04 e-03	8.43± 0.03 e-04
NSVGD	**KSD**	4.47± 0.01 e-01	1.12± 0.03 e-01	3.51± 0.04 e-02	1.11± 0.03 e-02	3.45± 0.05 e-03
	MMD	1.51± 0.01 e-01	3.34± 0.03 e-03	8.22± 0.04 e-03	2.29± 0.03 e-03	6.90± 0.05 e-04

Table 3. This table reports the 99% confidence intervals for the points plotted in Figure 4, i.e. for the 2d Gaussian target experiments.

E.3. Quantization of a Bimodal Gaussian Mixture

In this section, we evaluate the quantization properties of the algorithms at study when the target distribution is a 2-dimensional bimodal Gaussian mixture. This is a non log-concave target distribution, in contrast with the standard Gaussian. When measured with the KSD, all algorithms show faster rates than the i.i.d. samples represented by the black line. In particular, points provided by KSD-lbfgs show the faster rate. However, when measured with the MMD, one can see that the performance of KSD descent is not better than i.i.d. points, while MMD Descent corresponds to the fastest rate. For KSD-LBFGS/Stein Points with grid search, many particles get stuck between the modes, i.e. in a low-probability region (as reported in (Korba et al., 2021)), so the MMD of their final states are large. These results illustrate the sensitivity to evaluation discrepancy, and may be related to the particular way KSD quantifies the difference between measures, see (Gorham & Mackey, 2017). However, one can see that similarly to Figure 5 (for a Gaussian target π), NSVGD quantization rates are quite robust to the evaluation and faster than i.i.d. points (for a bimodal Gaussian mixture π). Notice that to prevent NSVGD from missing one mode of the target distribution, one needs to initialize particles in a large ball, see (Wenliang & Kanagawa, 2020). NSVGD is particularly preferable to SVGD in this setting, since it accelerates the convergence.

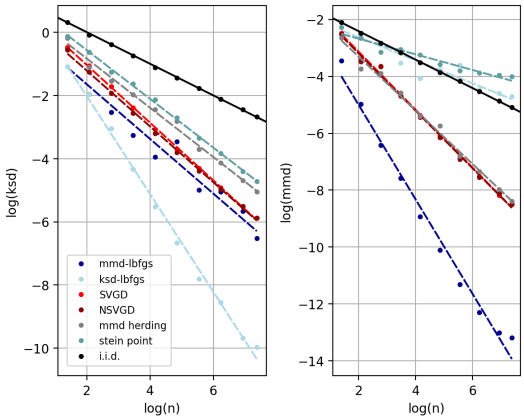


Figure 10. Quantization rates of the algorithms at study when the target distribution is a 2D-Gaussian mixture distribution with variance 0.3, centred at [1,0] and [-1,0]. We evaluate them using MMD and KSD with bandwidth 1. We run algorithms under the same setting as the 2-4D experiments on Figure 4.

E.4. Quantization in Sliced Wasserstein distance.

We also considered how well do particle distributions found by the algorithms is Section 6.2 approximate the target distribution with respect to the Sliced p-Wasserstein distance, which for $\nu, \mu \in \mathcal{P}_p(\mathbb{R}^d)$ is defined as:

$$d_{sw,p}(\nu,\mu) = \int_{\mathbb{S}^{d-1}} W_p(P_{\theta\#}\nu, P_{\theta\#}\mu)d\theta,$$

where $P_\theta : x \mapsto x \cdot \theta$ and $\#$ is the pushforward operator.

Under some further conditions of measure μ it is known that the empirical measure of a random i.i.d. sample approximates the target measure with respect to Sliced Wasserstein distance at parametric rate (see (Manole et al., 2022)):

$$d_{sw,2}(\mu,\mu_n) \lesssim \frac{1}{\sqrt{n}}.$$

In our experiments we find that the particle distributions obtained by SVGD, NSVGD flows and MMD and KSD minimization approximate the target distribution more accurately and at a slightly faster rate, namely for $d_{sw,1}$ the rates for NSVGD are approximately $n^{-0.72}, n^{-0.65}, n^{-0.63}$ in dimensions $d = 2, 3$, and 4, respectively. We note that these are quite close to the rate we theoretically predict for the distance between the measure on a grid in $[0,1]^d$, and the Lebesgue measure: $d_{sw,1} \sim n^{-\frac{1}{2}-\frac{1}{2d}}$, which is $n^{-0.75}, n^{-0.67}, n^{-0.625}$ in dimensions $d = 2, 3$, and 4, respectively. We note that determining the optimal quantization rate with respect to Sliced Wasserstein distances is an interesting open problem.

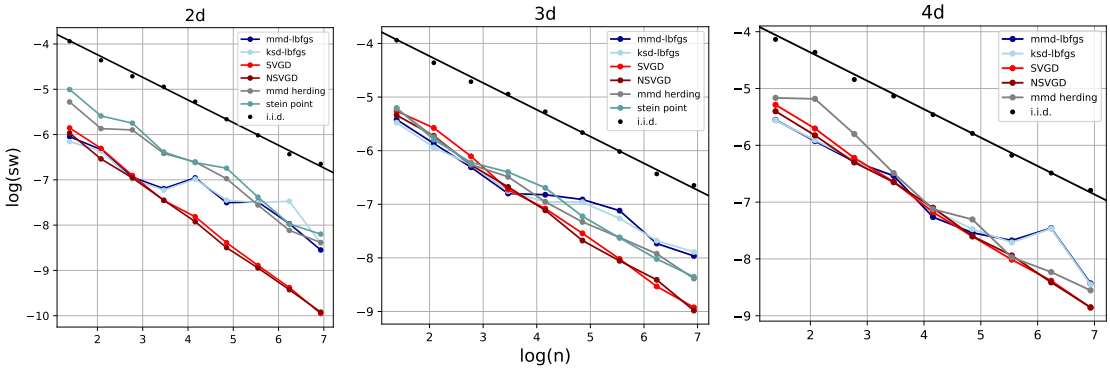


Figure 11. Quantization rates measured in Sliced Wasserstein distance of the algorithms $\pi = \mathcal{N}(0, {}^1/_d I_d)$. The experimental setting is identical to the one of Figure 4 only that instead of MMD we use Sliced 1-Wasserstein distance to evaluate the quantization error. In practice, we use 50 random directions drawn uniformly on $\mathbb{S}^{d-1}$ to discretize the integration.