Manifold Sampling via Entropy Maximization

Cornelius V. Braun[1*] **Tilman Burghoff**[1*] **Marc Toussaint**[1,2]
[1]Technische Universität Berlin
[2]Robotics Institute Germany
{braun,t.burghoff,toussaint}@tu-berlin.de

Abstract

Sampling from constrained distributions has a wide range of applications, including in Bayesian optimization and robotics. Prior work establishes convergence and feasibility guarantees for constrained sampling, but assumes that the feasible set is connected. However, in practice, the feasible set often decomposes into multiple disconnected components, which makes efficient sampling under constraints challenging. In this paper, we propose MAnifold Sampling via Entropy Maximization (MASEM) for uniform sampling on a manifold with an unknown number of disconnected components, implicitly defined by smooth equality and inequality constraints. The presented method uses a resampling scheme to maximize the entropy of the empirical distribution based on k-nearest neighbor density estimation. We show that, in the mean field, MASEM decreases the KL-divergence between the empirical distribution and the maximum-entropy target exponentially in the number of resampling steps. We instantiate MASEM with multiple local samplers and demonstrate its versatility and efficiency on synthetic and robotics-based benchmarks. MASEM enables fast and scalable mixing across a range of constrained sampling problems, improving over alternatives by an order of magnitude in Sinkhorn distance with competitive runtime. Link to code.

1 Introduction

Sampling from constrained distributions is a fundamental problem in machine learning, with applications including Bayesian inference [1] and molecular design [2] as well as robotics and trajectory optimization [3]. In particular, many tasks in robotics require sampling kinematically feasible states or trajectories, for instance, to generate data for behavior cloning or to sample reset states for reinforcement learning [4–7]. The feasible set is usually given implicitly by constraint functions, which can only be evaluated point-wise. This makes sampling from it a challenging problem, as we have no prior knowledge about many of its properties, like the number of connected components.

When no constraints are present, the dominant sampling paradigm is built around Markov Chain Monte Carlo (MCMC) methods such as Metropolis–Hastings, Langevin dynamics, and Hamiltonian Monte Carlo [8]. To handle constraints, these approaches are typically combined with projections or landing mechanisms that ensure samples remain feasible [1, 9–12]. However, all such methods rely on local MCMC kernels, which limits them to moves within a single connected region of the feasible set, and as a result they struggle with infeasibility barriers. As a result, even if such kernels mix well locally, they cannot determine how much probability mass to assign to each component (as illustrated in Figure 1). This makes them ill-suited in many practical scenarios, where such disconnected feasible sets arise naturally [13–15].

We tackle this problem by maximizing sample entropy across the feasible set. This provides empirical coverage of the feasible set, as the samples are distributed approximately uniformly. It enables estimation of component volumes and may serve as unbiased initialization for sampling from

*Equal contribution; Authors in alphabetical order.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

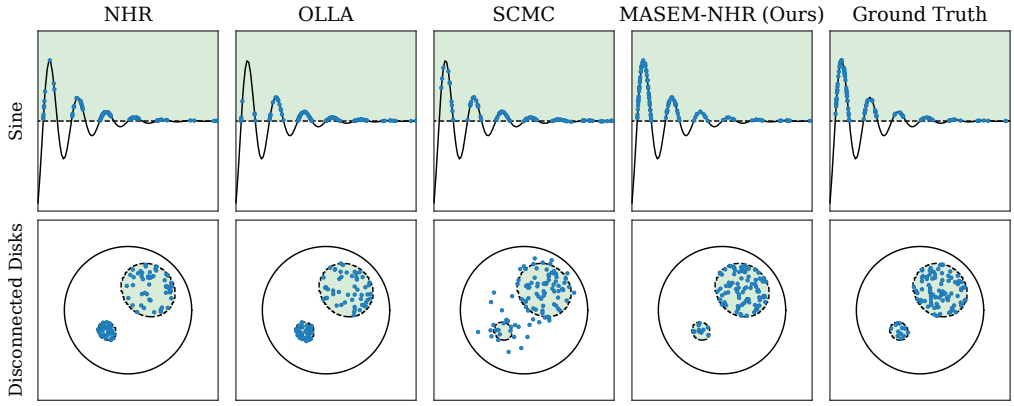


Figure 1: Scatter plots of 100 samples from different samplers on synthetic benchmarks. Black lines show equality constraints, and green shaded areas mark the feasible region(s) by inequality constraints. The bottom row depicts 3d samples on spheres projected into 2d. NHR and OLLA fail to allocate correct sample masses: the larger disk and the leftmost peak of the sine are undersampled.

arbitrary distributions across components. The key idea is to combine local k-nearest-neighbor density estimates [16] with Sequential Monte Carlo (SMC)-inspired resampling: particles in underrepresented components receive higher importance weights, encouraging redistribution of mass toward a uniform allocation. Crucially, the resampling step is modular and can be composed with generic constrained MCMC samplers. Adding this method to a constrained sampler is minimally invasive in the sense that it only incurs a small runtime overhead and does not degrade performance on compact and connected manifolds. Our main contributions are as follows:

- We introduce **Ma**nifold **S**ampling via **E**ntropy **M**aximization (MASEM), an algorithm for uniform sampling on manifolds with disconnected components.
- We show that, under mild assumptions, the induced resampling operator contracts the KL-divergence to the uniform distribution at a geometric rate of $(1 - \tau/p)$ per iteration, where τ is a temperature parameter and p is the intrinsic dimension of the manifold.
- We validate the approach empirically on synthetic and robotics benchmarks and compare performance across different local samplers.

2 Related Work

Constrained Sampling. Sampling under constraints is typically approached by adapting classical MCMC methods to ensure feasibility, for example by projection, reflection, or annealing techniques. Soft-constraint and annealing approaches such as sequentially constrained Monte Carlo (SCMC) [17] improve global exploration by relaxing the constraints, but may produce infeasible samples (see Figure 1) and can still collapse to local constraint minima as the penalty is tightened. In contrast, our method enables mixing across infeasibility barriers and exact constraint feasibility at convergence. For manifold-constrained targets, constrained MCMC [10, 18–21], barrier-based [22] and projected Hamiltonian methods [11], landing-based approaches [9, 1], constrained SVGD [1, 23] and non-linear Hit-and-Run (NHR) [24] operate directly on or near the feasible set and can maintain exact or approximate feasibility. Despite their differences, these methods share the common structural limitation that they rely on local transition kernels and therefore cannot reliably allocate probability mass across disconnected components. Our approach is complementary to this line of work: rather than introducing a new local kernel, it adds a resampling mechanism that can be combined with existing constrained samplers.

Multimodal Sampling. Multimodal (unconstrained) targets where energy barriers separate distant distributional modes is a long-standing challenge in sampling [25]. Despite the absence of constraints, this problem shares structural similarities with constraint sampling, as the main challenge lies in moving samples globally. A popular approach to bridging modes in multimodal distributions is *target relaxation* such that the separating regions acquire non-negligible probability mass. Notably, parallel tempering samples multiple relaxed targets at different temperatures and exchange states across these copies [26], while simulated annealing moves samples between relaxed targets sequentially to ensure

sample movements between modes [27, 28]. Further, diffusion-based jump strategies have recently shown high efficacy at facilitating sample mixing on challenging problems. Specifically, Diffusive Gibbs Sampling (DiGS) convolves the target with a Gaussian kernel and alternates sample noising and denoising in a Metropolis-within-Gibbs scheme [29], which is extended to discrete targets in HiSS via a logistic bridging kernel [30]. While relaxation is highly effective for global moves, it is fundamentally more difficult when the support region is defined by constraints. For one, local optima of constraint functions may induced biased samples. Further, for equality constraints smoothing replaces the singular target with a full-dimensional distribution concentrated in a neighborhood around it, and moving particles from ambient space onto the manifold is challenging. Our method addresses this tension by enabling moves across infeasible barriers via entropy-based resampling while retaining exact constraint feasibility at convergence.

3 Preliminaries & Notation

We consider sampling uniformly from a feasible set

$$\Sigma \coloneqq \{x \in \mathbb{R}^d \mid h(x) = 0,\ g(x) \leq 0\} \subseteq \mathbb{R}^d, \tag{1}$$

where $h\colon \mathbb{R}^d \to \mathbb{R}^m$ and $g\colon \mathbb{R}^d \to \mathbb{R}^\ell$ are differentiable equality and inequality constraint functions, which can only be evaluated point-wise and have Lipschitz-continuous derivatives. We assume that the feasible set Σ is bounded, for example by introducing the additional constraint $\|x\| \leq R$ for some large $R > 0$. Under this assumption, the feasible set and its connected components are compact. Furthermore, we impose the widely used Linear Independent Constraint Qualifier (LICQ).

Assumption 1 (Linear Independent Constraint Qualifier [31])**.** For each point $x \in \Sigma$, the set of gradients of active constraints

$$\{\nabla h_j(x)\} \cup \{\nabla g_i(x) \mid g_i(x) = 0\}$$

is linearly independent.

This assumption ensures that the feasible set Σ is a Riemannian manifold with corners and intrinsic dimension $p = d - m$ [32, Lemma 3.1.12]. Let μ_Σ denote the induced Hausdorff measure and $S = \mu_\Sigma(\Sigma)$ its volume.

We consider the general setting that the manifold decomposes into a finite number of disconnected components $\Sigma = \Sigma_1 \sqcup \cdots \sqcup \Sigma_C$. Importantly, however, we do *not* assume knowledge of the number C of components or any of their structural properties, as they are implicitly defined via h and g. Due to LICQ, each component has measure $\sigma_c = \mu_\Sigma(\Sigma_c) > 0$. Note that if $p = d$, this problem reduces to sampling from the unnormalized density $\hat{\rho}(x) = 1_\Sigma(x)$, where $1_\Sigma(x)$ is the indicator function of Σ. Our target is to maximize the entropy of samples across the constrained set.

Lemma 1 (Informally)**.** *The uniform density is the unique maximizer of the entropy on* Σ*.*

We formalize this in Appendix A. Our method maximizes the entropy by estimating the local density by k-nearest neighbor distances [16]. Let samples $x_1, \ldots, x_N \in \Sigma$ be samples drawn from an underlying distribution parameterized by a density ρ. We define $\varepsilon_{i,k}$ as the distance from x_i to its k-th nearest neighbor among $\{x_j\}_{j \neq i}$. The k-nearest-neighbor density estimator at x_i is

$$\hat{\rho}(x_i) = \frac{k}{N V_p \varepsilon_{i,k}^p}, \tag{2}$$

where V_p is the volume of the unit ball in $\mathbb{R}^p$. Consistency of this estimator, meaning $\hat{\rho}(x_i) \to \rho(x_i)$ in probability as $N \to \infty$ with $k \to \infty$, $k/N \to 0$, is a classical result for interior points of the support [33].

4 Manifold Sampling via Entropy Maximization

We now introduce *Manifold Sampling via Entropy Maximization* (MASEM), a framework for uniform sampling on a constrained set Σ as defined in Section 3. Our goal is to construct a particle system

Algorithm 1 Entropy-based uniform sampling on disconnected manifolds

Input: Initialization scale σ^2, number of samples N, number of iterations T, neighborhood parameter k, temperature τ, rejuvenation kernel K, rejuvenation step number M, constraint functions h, g.

// Initialization
Sample N particles $x_0^i \sim \mathcal{N}(0, \sigma^2)$
Project the particles onto Σ (e.g., with Gauss-Newton method)
Apply a manifold-constrained rejuvenation kernel K
// Iterative entropy maximization
for $t = 1, \ldots, T$ **do**
    Compute weights w_t^i from the entropy-based rule in (3)
    Resample particles according to the normalized weights w_t^i
    Apply a manifold-constrained rejuvenation kernel K for M steps
end for
return final particle set $X^{(T)}$

whose empirical distribution approximates the uniform density u_Σ on Σ. Given an existing strong local sampler that preserves feasibility, the main challenge for constrained uniform sampling is *global* mass allocation. As Σ is only defined implicitly via constraints, this is a hard problem, since the measures of components or even their number are not known a priori, so local samplers cannot calibrate probability mass across disconnected components efficiently. Thus, MASEM introduces a mechanism that redistributes particles across components to ensure that each component receives mass proportional to its measure.

The design of MASEM is intentionally lightweight. Its key mechanism is an importance-resampling step [8] that moves particles towards the maximum entropy target. This target is the uniform density $p = u_\Sigma$ (Lemma 1). Let q denote the current particle density on Σ. Since the target density is uniform, importance-resampling with weights $w_i \propto p(x_i)/q(x_i)$ reduces to resampling according to $w_i \propto q(x)^{-1}$. While the current density itself is unknown, we can estimate it using the k-nearest neighbor estimator (2). This leads to $q(x_i) \propto \varepsilon_{i,k}^{-p}$ in probability in the large particle limit, where $\varepsilon_{i,k}$ denotes the distance of sample x_i to its kth neighbor. Replacing the potentially unknown dimension p by a temperature parameter τ yields

$$\bar{w}_i = \varepsilon_{i,k}^{\tau}, \qquad w_i = \frac{\bar{w}_i}{\sum_j \bar{w}_j}\,. \tag{3}$$

Thus, particles in low-density regions have larger k-NN radii and receive larger resampling weights, encouraging redistribution toward underrepresented connected components. The temperature τ controls the aggressiveness of the resampling step. When τ is small, the update is conservative and moves mass more slowly. Larger values of τ accelerate mass redistribution but can amplify errors in the k-NN density estimate when the number of particles or mixing steps is insufficient.

MASEM, described in pseudocode in Algorithm 1, works by first initializing feasible particles and then iterating resampling followed by manifold sampling rejuvenation. We initialize the particles by random i.i.d. sampling followed by projection onto the feasible set using Gauss-Newton steps on the squared slack

$$\mathrm{Slack}(x) = \frac{1}{2}\left(\|g(x)_+\|^2 + \|h(x)\|^2\right) \tag{4}$$

to ensure approximate constraint feasibility. After initialization, we apply a feasibility-preserving kernel K for rejuvenation. This ensures that the particles are approximately uniformly distributed within components. Then, at each iteration, particles are resampled using the entropy-based resampling weights (3). After resampling, we apply K again to obtain mixed samples within each component. As local kernels are only approximately feasible in practice, we use a slack-penalized variant of the resampling weights to avoid over-replicating particles with large constraint violations in our experiments. These practical modifications are described in Appendix D.5.

5 Theoretical Analysis

Our theoretical guarantees address the global mass-redistribution problem, which cannot be solved by existing local samplers. We analyze Algorithm 1 in the mean-field regime, where the $N \to \infty$

limit reduces the particle system to a deterministic evolution on the simplex of component weights. This parallels the idealized-limit viewpoint taken in related work on particle-based constrained sampling [1, 9]. Our main result (Theorem 1) shows that a single resampling step contracts the KL divergence to the uniform distribution on Σ at the rate $(1-\tau/p)^2$, leading to exponential convergence. All proofs for this section can be found in Appendix A.

5.1 Preliminaries

We need two prerequisites for our main result: Each component must be initially populated, and the sampler has to sample accurately within components. We begin by showing that the initialization step of Algorithm 1 populates all components for N large enough.

Lemma 2 (Component coverage at initialization)**.** *Suppose h, g have Lipschitz-continuous derivatives and satisfy LICQ, and that the feasible set Σ is bounded and decomposes into a finite number of connected components. We draw N points $x_i \sim \mathcal{N}(0, \sigma^2)$ for arbitrary $\sigma > 0$ and apply any descent method on the squared slack* (4) *satisfying sufficient decrease. Then $\mathbb{P}$(every component is hit) $\to 1$ as $N \to \infty$.*

Proof sketch. Take $x^* \in \Sigma_c$ arbitrary. The local error bound $d(x, \Sigma) \leq \kappa\,\mathrm{Slack}(x)^{1/2}$ holds for some $\kappa > 0$ due to LICQ [34]. Since Σ is compact and has finitely many components, the inter-component distance $\delta := \min_{i \neq j} d(\Sigma_i, \Sigma_j)$ is strictly positive [35, Ch. XI 4.4]. Taking $L := \{\mathrm{Slack}(x) \leq (\delta/(3\kappa))^2\}$ ensures that all $x \in L \cap U_{\delta/3}(\Sigma_c)$ converge to Σ_c under any descent method satisfying sufficient decrease [36]. Therefore, the basin of attraction has positive Lebesgue measure, and standard coverage arguments yield the claim. □

To consider component-level convergence properties, we assume that the local kernel perfectly mixes chains within each component. This assumption decouples the analysis of the resampling in Algorithm 1 from any specific local sampler.

Assumption 2 (Mixing Rejuvenation)**.** The rejuvenation kernel K is u_c-invariant on each Σ_c (i.e., $u_c K = u_c$, where $u_c = \sigma_c^{-1}\mathbf{1}_{\Sigma_c}$), and mixes sufficiently, such that after application the particles within Σ_c are approximately i.i.d. draws from u_c.

This assumption ensures that chains stay within their component during mixing. In practice, if components are close together, the chain might be able to jump between components. For the sake of our analysis, we treat these clusters of components as one "meta-component", since, given enough time, the chains will mix uniformly within this cluster.

This assumption allows us to consider component-level weights instead of individual particles. Let $\alpha_c := N_c/N$ denote the fraction of particles in Σ_c. Then the empirical density after rejuvenation takes the mixture form $p_\alpha(x) = \alpha_{c(x)}/\sigma_{c(x)}$, where $c(x)$ denotes the component containing x. Let $\alpha = (\alpha_1, \ldots, \alpha_C)$ be the vector of component weights that induces the distribution p_α. The sampling problem thus reduces to steering $\alpha \in \Delta^{C-1}$ towards the maximum entropy distribution $\alpha^* := (\sigma_1/S, \ldots, \sigma_C/S)$.

5.2 Convergence Guarantees

We first derive the deterministic map that governs the evolution of α under resampling, then show that its iterates contract in KL divergence.

Proposition 1 (Mean-field resampling map)**.** *Under Assumption 2, the k-NN radius of a particle $x_i \in \Sigma_c$ concentrates at $\varepsilon_{i,k} \asymp (k\sigma_c/(NV_p\alpha_c))^{1/p}$ in the limit $N \to \infty$ with $k_N \to \infty$, $k_N/N \to 0$ [33, Ch. 2]. Consequently, the resampling weights $w_i \propto \varepsilon_{i,k}^\tau$ induce the map*

$$(\Phi\alpha)_c := \frac{\alpha_c^{1-\beta}\,(\alpha_c^*)^\beta}{\sum_{j=1}^C \alpha_j^{1-\beta}\,(\alpha_j^*)^\beta}, \qquad \beta := \tau/p. \tag{5}$$

The map (5) is a geometric mean of the current distribution α and the target α^*. Its *unique* fixed point on Δ^{C-1} is α^*. Thus, iterating Φ contracts α towards α^*, with the parameter $\beta = \tau/p$ controlling the aggressiveness of rebalancing. We show that iterating Φ drives α to α^* at a geometric rate.

Theorem 1. *Consider constraints h, g with Lipschitz-continuous derivative fulfilling LICQ, which define a bounded constrained set Σ that decomposes into a finite number of connected components. Suppose all components have positive measure with respect to the induced Hausdorff measure on Σ. Under Assumption 2 for any $\tau \in (0, p)$ and any schedule $k_N \to \infty$, $k_N/N \to 0$, the mean-field iterates $\alpha_{t+1} = \Phi(\alpha_t)$ of Algorithm 1 satisfy the following bound*

$$D_{\mathrm{KL}}^{\Sigma}(p_{\alpha_t} \,|\, p_{\alpha^*}) \leq C_0 \left(1 - \frac{\tau}{p}\right)^{2t} \tag{6}$$

with

$$C_0 = \frac{1}{4}\left(\max_c \log \frac{\alpha_c^{(0)}}{\alpha_c^*} - \min_c \log \frac{\alpha_c^{(0)}}{\alpha_c^*}\right)^2 \tag{7}$$

for the initial component weights $\alpha_c^{(0)}$ after the projection step of the algorithm. In particular, $p_{\alpha_t} \to u_\Sigma$ in KL as $t \to \infty$.

Proof sketch. We define the relative estimation error $y_{t,c} \coloneqq \frac{\alpha_{t,c}}{\alpha_c^*}$ as well as and $z_c \coloneqq \log y_{0,c}$ and show using induction that $y_{t,c} = y_{0,c}^{(1-\beta)^t} / \sum_j \alpha_j^* y_{0,j}^{(1-\beta)^t}$. Writing $a_t \coloneqq (1-\beta)^t$, this admits the exponential closed form

$$\alpha_{t,c} \coloneqq \frac{\alpha_c^* \, e^{a_t z_c}}{Z(a_t)}, \qquad Z(a) \coloneqq \sum_j \alpha_j^* e^{a z_j} = \mathbb{E}_{\alpha^*}[e^{az}], \tag{8}$$

from which we can see α_t is an exponential tilt of α^*, with inverse temperature a_t, which shrinks geometrically to zero.

Substituting (8) into the definition of KL gives the decomposition

$$D_{\mathrm{KL}}^{\Sigma}(p_{\alpha_t} \,|\, p_{\alpha^*}) = a_t \sum_c \alpha_{t,c} \log y_{0,c} - \log Z(a_t),$$

so it suffices to upper-bound $\bar{z}_t$ and lower-bound $\log Z(a_t)$. Both bounds follow from properties of the cumulant generating function (CGF) $\Lambda(a) \coloneqq \log Z(a)$ of z under α^*. For the lower bound, Jensen's inequality applied to the convex function e^{az} gives $\log Z(a_t) \geq a_t \sum_c \alpha^* z_c$. For the upper bound, we use basic properties of the CGF and Popoviciu's variance inequality [37] to obtain a bound of $\Lambda''(a) = \mathbb{V}_{\nu_a}[z] \leq R^2/4$, where $R \coloneqq \max_c \log(\alpha_c^{(0)}/\alpha_c^*) - \min_c \log(\alpha_c^{(0)}/\alpha_c^*)$. Integrating then gives $\bar{z}_t \leq \bar{z}_{\alpha^*} + a_t R^2/4$. As the linear terms cancel, we end with $D_{\mathrm{KL}}^{\Sigma}(p_{\alpha_t} \,|\, p_{\alpha^*}) \leq (R^2/4)(1-\beta)^{2t}$, which is the bound in (6). □

Using this result we derive an asymptotic bound on the worst case number of iterations.

Corollary 1. *Assume the conditions of Theorem 1 hold. In the worst case, the number of iterations t to reach a KL-divergence $D_{\mathrm{KL}}^{\Sigma}(p_{\alpha_t} \,|\, p_{\alpha^*}) \leq \varepsilon$ is*

$$t \in \mathcal{O}\left(\frac{\log(1/\varepsilon) + \log\log N}{\tau}\right).$$

Proof sketch. We show $C_0 \in \mathcal{O}\left((\log N)^2\right)$. Bounding (6) from above by ε lets us derive

$$t \approx \frac{\log(\varepsilon/C_0)}{2\log\left(1 - \frac{\tau}{p}\right)} = \frac{\log(1/\varepsilon) + \log C_0}{-2\log(1 - \frac{\tau}{p})} \in \mathcal{O}\left(\frac{\log(1/\varepsilon) + \log\log N}{\tau}\right). \qquad \square$$

Component loss We note that Theorem 1 is only stated in the mean field. It might happen that no particles of a particular component are sampled, resulting in that component vanishing. The

probability of this extinction is $P_c = (1 - (\Phi\alpha)_c)^N \leq \exp\big(-N(\Phi\alpha)_c\big)$. Using Proposition 1 we get $(\Phi\alpha)_c \propto \alpha_c^{1-\beta}(\alpha_c^*)^\beta$. By Lemma 2, each component contains at least one sample, bounding $\alpha_c = N_c/N \geq N^{-1}$. Since α_c^* is constant, we can derive an asymptotic upper bound on the extinction-probability as $P_c \in \mathcal{O}\big(\exp(-N\alpha_c^{1-\beta})\big) = \mathcal{O}\big(\exp(-N^\beta)\big)$. We investigated this effect and found that, in practice, just four chains per component is enough (see Appendix C.4).

6 Experimental Analysis

To demonstrate how MASEM improves existing manifold samplers, we consider Non-Linear Hit-&-Run (NHR) [24] and OLLA [9] with- and without the proposed resampling logic. In addition, we compare against Sequentially Constrained MC (SCMC) [17] which anneals soft constraints. To understand how MASEM compares against explicit component discovery and importance sampling, we implement Cluster-NHR, which clusters samples and computes resampling weights based on the estimated cluster volumes. We report the squared Sinkhorn distance and the averaged maximum slack violation. Since SCMC cannot guarantee exact constraint feasibility, we perform an additional projection step before reporting the metrics. Implementation details are listed in Appendix D and additional figures can be found in Appendix B.

Table 1: Final $\mathcal{W}_2^2$ (Sinkhorn) distance performance across five seeds. Entries are mean $\pm$ 95% confidence interval. Bold indicates better than counterpart for t-test with $p < 0.05$ and Holm-Bonferroni correction. MASEM-Sampler outperform their counterpart by an order of magnitude on all non-connected problems.

Problem	NHR	MASEM-NHR	OLLA	MASEM-OLLA
Connect. Disks ($3d$)	$.06 \pm .01$	$\mathbf{.00 \pm .00}$	$.04 \pm .00$	$\mathbf{.00 \pm .00}$
Disconn. Disks ($3d$)	$1.98 \pm .12$	$\mathbf{.01 \pm .01}$	$1.95 \pm .11$	$\mathbf{.02 \pm .00}$
Seven Lobes ($2d$)	$.37 \pm .02$	$\mathbf{.05 \pm .02}$	$.89 \pm .08$	$\mathbf{.05 \pm .02}$
Sine ($2d$)	100.21 ± 4.09	$\mathbf{.13 \pm .06}$	128.38 ± 3.79	$\mathbf{.13 \pm .07}$
Swiss Roll ($2d$)	$6.08 \pm .75$	$\mathbf{.08 \pm .02}$	18.96 ± 1.15	$\mathbf{.29 \pm .09}$

6.1 Synthetic Benchmarks

We first evaluate sampling on low-dimensional manifolds: (1)-(2) two disks that are embedded on a sphere in $\mathbb{R}^3$ and we vary whether the disks are connected or not, (3) the seven lobes density from Jeon et al. [9] with constant f, (4) a sine equality constraint with reducing amplitude cut into disconnected components inequality constraint, and (5) a product of an Archimedean spiral manifold and randomly sampled circles combined with a nonlinear inequality we label swiss roll. The benchmarks are visualized in Figure 6, and we report the specific constraints and ground truth sampling method in Appendix D.2. For each problem, we run 2 000 independent chains in parallel with a budget of 5 000 target evaluations and collect the last sample per chain.

Table 2: Final $\mathcal{W}_2^2$ (Sinkhorn) distance and slack across five seeds. Entries are mean $\pm$ 95% confidence interval. Bold indicates best for t-test with $p < 0.05$ and Holm-Bonferroni correction.

$\mathcal{W}_2^2$	SCMC	Cluster-NHR	MASEM-NHR
C. Disks	$.12 \pm .01$	$.05 \pm .03$	$\mathbf{.00 \pm .00}$
D. Disks	$.59 \pm .23$	$.19 \pm .14$	$\mathbf{.01 \pm .01}$
Seven Lobes	$.12 \pm .06$	$.14 \pm .04$	$\mathbf{.05 \pm .02}$
Sine	126.79 ± 10.85	$.91 \pm 1.04$	$.13 \pm .06$
Swiss Roll	$1.10 \pm .42$	$.42 \pm .26$	$\mathbf{.08 \pm .02}$
Slack $\times 10^{-3}$	SCMC	Cluster-NHR	MASEM-NHR
C. Disks	20.46 ± 1.12	$.02 \pm .03$	$.01 \pm .01$
D. Disks	29.97 ± 13.61	$.02 \pm .03$	$.01 \pm .01$
Seven Lobes	3.00 ± 1.89	$.00 \pm .00$	$.00 \pm .00$
Sine	9.38 ± 1.48	$\mathbf{.00 \pm .00}$	$.01 \pm .01$
Swiss Roll	$2.77 \pm .71$	$.00 \pm .00$	$\mathbf{.00 \pm .00}$

Local Kernels Fail Under Disconnected Components. In Figure 1 we identify a common failure mode of standard manifold samplers: In the presence of disconnected components, these local methods fail to correctly allocate mass; both methods over-sample the lower disk on the sphere and under sample the left-most arc of the sine. While the connected disks problem permits the chains to

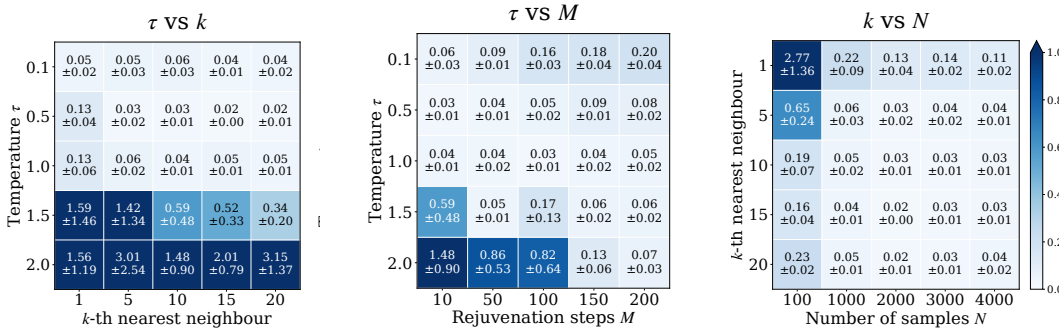


Figure 2: Influence of hyperparameters for MASEM-NHR on the 7 lobes problem. The parameters not in the grid are fixed to parameters $M = 10$, $\tau = 0.5$, $k = 10$ and $N = 2000$.

mix well in the single feasible level set, separating the two disks on the manifold leads to an increase in sampling error, as shown in Table 1.

Sampling Accuracy & Constraint Violation of MASEM. As shown in Table 1, MASEM-NHR and MASEM-OLLA outperform their counterparts across all problems with a disconnected feasible set by an order of magnitude in the Sinkhorn distance $\mathcal{W}_2^2$. Further, MASEM minimizes the constraints, matching (or even outperforming) their local counterparts (Figure 5). We also see in Table 2 that MASEM yields superior performance compared to the global baselines. In particular, we observe that while SCMC effectively yields constraint satisfying samples on the seven lobes problem, it fails to do so in particular on problems with highly non-linear constraints such as the swiss roll problem. Cluster-NHR, in contrast, achieves low constraint violations but higher sampling error than MASEM on all problems.

Effect of Hyperparameters We further examine the influence of the hyperparameters τ, k, the number of rejuvenation steps M and the number of samples N on the seven lobes problem. As shown in Figure 2, our method is robust to variations of M and N. However, setting τ greater than the intrinsic dimension leads to high Sinkhorn divergences to ground truth samples. Increasing the number of rejuvenation steps or increasing k recovers some of the error at high τ. Conversely, using small values of k risks higher Sinkhorn divergences.

We note that MASEM method is relatively robust to the exact choice of these hyperparameters. To aid practitioners, we provide a standard recommendation for the parameters. Setting

$$\tau = 0.3, \; M = 20, \; \text{and } k = 20 \tag{9}$$

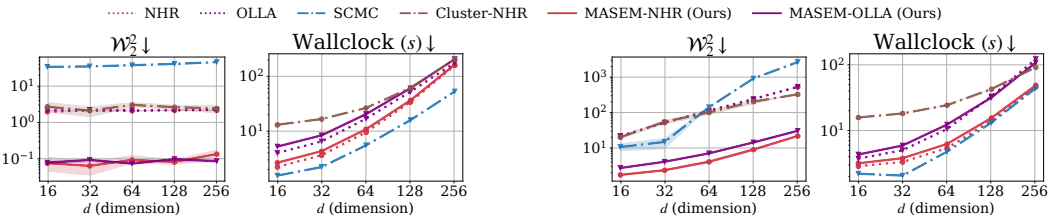
yields good results across all benchmarks, achieving a final $\mathcal{W}_2^2$ in the same order of magnitude as the tuned version, as we show in Appendix C.1.

6.2 Scaling Under High-Dimensionality and Large Number of Constraints

We assess the robustness and scalability of MASEM-NHR and MASEM-OLLA using a synthetic *stress-test problem* that enables explicit control of ambient dimension d as well as the number of equality and inequality constraints and number of disconnected components $(m, l, |C|)$. In our case, we guarantee different components by sampling a set of $|C|$ disjoint spheres, which we embed into a subspace of $\mathbb{R}^d$ by random linear projections defined by the m equality constraints.

Scaling Under Increasing Manifold Dimension. With fixed numbers of constraints $m = l = 5$ we study the effect of an increasing manifold dimension as d increases. Figure 3b shows that MASEM consistently outperforms NHR and OLLA as well as Cluster-NHR by an order of magnitude. In addition, we note that the penalty-based approach of SCMC deteriorates strongly as d increases. Further results in Figure 7 show a similar trend for the KL divergence while the maximum constraint violation stays below 0.1 for MASEM.

Scaling Under Number of Constraints. Figure 3a illustrates the performances as the ambient dimension d increases while the manifold dimension is fixed to 2. We see that the gap between MASEM and its counterparts remains constant across dimensions while the $\mathcal{W}_2^2$ distance only slightly increases with d. The wallclock times are competitive compared to the baselines that evaluate



(a) Increasing ambient and fixed manifold dimension with increasing $m = d - 3$.

(b) Increasing ambient and manifold dimension with fixed $m = 5$.

Figure 3: Sampling performance and wallclock time as the dimension d increases (with $l = |C| = 5$). MASEM achieves lowest $\mathcal{W}_2^2$ (Sinkhorn) distances.

the constraint Jacobian (OLLA, NHR, Cluster-NHR). This shows that our method adds only little overhead in practice. Additional plots for the KL divergence and constraint violation in Figure 7 further support these results.

6.3 Robotics Applications

Motion Planning. We consider trajectory sampling, where we sample constraint satisfying trajectories through an obstacle course. We sample the trajectories of a 2d pointmass across two different obstacle courses, one that is a regular 4×4 grid and another one with 20 randomly placed obstacles (Figure 4). Each trajectory is parameterized by a spline defined by 3 waypoints, resulting in a 6 dimensional problem. Random obstacle motion planning has 934 constraints in total, while grid based planning has 774 constraints.

Grasp Sampling. Secondly, we evaluate MASEM for grasping. We optimize grasps on a capsule geometry (a common shape for collision checking) with three fingers. A band around the capsule shall not be touched, mirroring grasping in the real world where not all objects can be grasped at arbitrary points, e.g., a robot grasping a plate that must not put its finger into the food. Each finger applies a force, such that in the end the fingers counteract gravity pulling on the object. This results in a 18-dimensional problem, with 9 equality and 48 inequality constraints.

Table 3: Performance on robotics problems across seeds. Entries are mean ± 95% confidence interval. Column-wise bold highlights indicate better than counterpart for t-test with p < 0.05 and Holm-Bonferroni correction.

Method	Planning Random Obst.		Planning Grid Obst.		Grasping	
	Feas. Ent. ↑	Slack ↓	Feas. Ent. ↑	Slack ↓	Feas. Ent. ↑	Slack ↓
NHR	6.85 ± .37	.00 ± .00	6.50 ± .14	.00 ± .00	3.61 ± .23	.00 ± .00
Cluster-NHR	6.49 ±1.43	.00 ± .00	4.41 ±1.27	.00 ± .00	3.60 ±1.05	.00 ± .00
MASEM-NHR	**7.93 ± .22**	.00 ± .00	**7.49 ± .16**	.00 ± .00	**4.79 ± .16**	.00 ± .00
OLLA	.86 ± .15	.07 ± .01	1.47 ± .24	.03 ± .01	1.61 ± .09	.37 ± .01
MASEM-OLLA	6.11 ± .47	.01 ± .00	5.35 ± .38	.01 ± .00	4.09 ± .18	.01 ± .00
SCMC	1.43 ± .69	.05 ± .04	2.26 ± .26	.02 ± .01	.22 ± .04	.08 ± .00

Results. Due to the lack of ground truth samples, we report the feasible entropy instead of the Sinkhorn distance. We compute it by multiplying the entropy of the feasible samples and the fraction of feasible samples (made precise in Appendix D.1). Table 3 shows an increase of sample entropy when applying MASEM, regardless of sampler. This is illustrated in Figure 4 for random obstacles. We observe that MASEM-NHR samples trajectories through the narrow gaps at the center of the course, which are missed by standard NHR. For OLLA the performance gap is even larger, as the default sampler only generates a locally distributed sample sets but fails to sample the outer regions of the domain. Moreover, OLLA generates many paths crossing over obstacles. While MASEM-NHR outperforms MASEM-OLLA on all problems, OLLA benefits more from the resampling applied by MASEM. This suggests two things: first, NHR-based samplers are more suited for robotics than OLLA-based samplers (at least within this evaluation set), and secondly, MASEM greatly improves previously unsuited samplers and makes them competitive in this domain. We believe

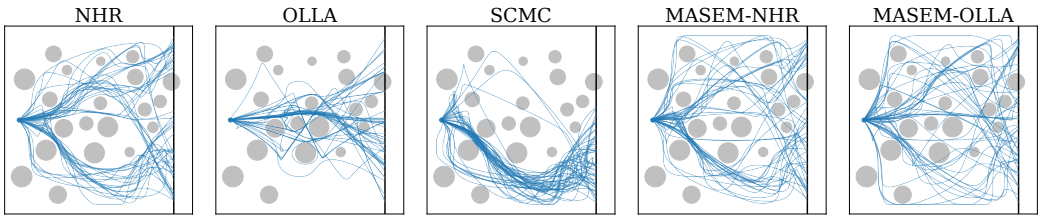


Figure 4: Plots of 50 samples on the random motion planning problem. MASEM-based approaches yield the highest sample entropy across all methods.

these are valuable for future research on sampling in robotics. Further, our approach outperforms alternative global sample allocation strategies: SCMC performs poorly on all tasks; MASEM-NHR outperforms its clustering counterpart on all tasks as well, with a large gap on the regular grid design. The performance of MASEM on the motion planning problems demonstrates its ability to scale to problems with a high number of constraints.

7 Conclusion & Future Work

Summary We presented Manifold Sampling via Entropy Maximization (MASEM), a resampling-based approach that can be combined with any constrained MCMC sampler to enable uniform sampling on disconnected manifolds implicitly defined by constraints. We proved that MASEM minimizes the KL divergence to the uniform target distribution exponentially in the number of resampling steps and provided worst-case bounds in the mean-field. Building on this, we analyzed two instantiations of MASEM on several synthetic and real world problems, where it results in a significantly lower Sinkhorn distance to the ground truth distribution compared to the baselines.

Limitations While we believe that MASEM is a useful constrained sampling framework, it is not without limitations. Some results, namely Lemma 2, Theorem 1, and Corollary 1, are valid only in the large-scale particle limit, i.e. for $N \to \infty$. Further analysis is warranted to describe the algorithms convergence under finite N. Beyond theoretical limitations, there are two practical considerations when using MASEM in combination with an existing sampler. First, the resampling step introduces a small runtime overhead, as one needs to compute the pairwise distances to compute the resampling weights. In practice however, we discovered that on our benchmarks this overhead is small, while applying MASEM greatly enhances sample quality. Secondly, using MASEM introduces additional parameters that need tuning, namely τ, M and k. We provide starting values for these parameters (9). We refer to Appendix E for a more in-depth discussion on the limitations of our method.

Future Work In this work we focus on entropy maximization, which leads to uniform sampling across the manifold. In the future we aim to extend MASEM to non-uniform distributions. We see potential in applying MASEM as a data generation method in robotics and other application domains. An intriguing avenue of research is investigating practical design choices and comparing them to current heuristic approaches in practical applications.

Acknowledgments and Disclosure of Funding

The authors sincerely thank Paula Cordero Encinar for discussions during early stages of the project, and Eckart Cobo-Briesewitz for his feedback on the manuscript. This research was funded by the Amazon Fulfillment Technologies and Robotics team. This work has been supported by the German Federal Ministry of Research, Technology and Space (BMFTR) under the Robotics Institute Germany (RIG).

References

[1] Ruqi Zhang, Qiang Liu, and Xin Tong. Sampling in Constrained Domains with Orthogonal-Space Variational Gradient Descent. *Advances in Neural Information Processing Systems*, 35: 37108–37120, 2022.

[2] D. C. Rapaport. *The Art of Molecular Dynamics Simulation*. Cambridge University Press, Cambridge, 2nd edition, 2004.

[3] Zachary Kingston, Mark Moll, and Lydia E. Kavraki. Sampling-Based Methods for Motion Planning with Constraints. *Annual Review of Control, Robotics, and Autonomous Systems*, 1(1): 159–185, 2018.

[4] Thomas Power, Rana Soltani-Zarrin, Soshi Iba, and Dmitry Berenson. Sampling Constrained Trajectories Using Composable Diffusion Models. In *IROS 2023 Workshop on Differentiable Probabilistic Robotics: Emerging Perspectives on Robot Learning*, 2023.

[5] Carlos Florensa, David Held, Markus Wulfmeier, Michael Zhang, and Pieter Abbeel. Reverse Curriculum Generation for Reinforcement Learning. In *Conference on Robot Learning*, pages 482–495. PMLR, 2017.

[6] Marc Toussaint, Cornelius V. Braun, Armand Jordana, Sayantan Auddy, Eckart Cobo-Briesewitz, Denis Shcherba, Tilman Burghoff, and Justin Carpentier. Combining constrained sampling and reinforcement learning for manipulation. In *10th Annual Conference on Robot Learning*, 2026.

[7] Eckart Cobo-Briesewitz, Tilman Burghoff, Denis Shcherba, Armand Jordana, and Marc Toussaint. Stability-Guided Exploration for Diverse Motion Generation. *arXiv preprint arXiv:2603.06773*, 2026.

[8] Christophe Andrieu, Nando De Freitas, Arnaud Doucet, and Michael I Jordan. An Introduction to MCMC for Machine Learning. *Machine learning*, 50(1):5–43, 2003.

[9] Kijung Jeon, Michael Muehlebach, and Molei Tao. Fast Non-Log-Concave Sampling under Nonconvex Equality and Inequality Constraints with Landing. In *The Thirty-Ninth Annual Conference on Neural Information Processing Systems*, 2025.

[10] Emilio Zappa, Miranda Holmes-Cerfon, and Jonathan Goodman. Monte Carlo on Manifolds: Sampling Densities and Integrating Functions. *Communications on Pure and Applied Mathematics*, 71(12):2609–2647, 2018.

[11] Tony Lelièvre, Mathias Rousset, and Gabriel Stoltz. Hybrid Monte Carlo methods for sampling probability measures on submanifolds. *Numerische Mathematik*, 143(2):379–421, 2019.

[12] David E Kaufman and Robert L Smith. Direction Choice for Accelerated Convergence in Hit-and-Run Sampling. *Operations Research*, 46(1):84–95, 1998.

[13] An T Le, Georgia Chalvatzaki, Armin Biess, and Jan R Peters. Accelerating Motion Planning via Optimal Transport. *Advances in Neural Information Processing Systems*, 36:78453–78482, 2023.

[14] Zechu Li, Rickmer Krohn, Tao Chen, Anurag Ajay, Pulkit Agrawal, and Georgia Chalvatzaki. Learning Multimodal Behaviors from Scratch with Diffusion Policy Gradient. *Advances in Neural Information Processing Systems*, 37:38456–38479, 2024.

[15] Natalie Maus, Kaiwen Wu, David Eriksson, and Jacob Gardner. Discovering Many Diverse Solutions with Bayesian Optimization. In *International Conference on Artificial Intelligence and Statistics*, pages 1779–1798. PMLR, 2023.

[16] Don O Loftsgaarden and Charles P Quesenberry. A Nonparametric Estimate of a Multivariate Density Function. *The Annals of Mathematical Statistics*, 36(3):1049–1051, 1965.

[17] Shirin Golchi and David A Campbell. Sequentially Constrained Monte Carlo. *Computational Statistics & Data Analysis*, 97:98–113, 2016.

[18] Tony Lelievre, Mathias Rousset, and Gabriel Stoltz. Langevin dynamics with constraints and computation of free energy differences. *Mathematics of computation*, 81(280):2071–2125, 2012.

[19] Marcus Brubaker, Mathieu Salzmann, and Raquel Urtasun. A Family of MCMC Methods on Implicitly Defined Manifolds. In *Artificial Intelligence and Statistics*, pages 161–172. PMLR, 2012.

[20] Simon Byrne and Mark Girolami. Geodesic Monte Carlo on Embedded Manifolds. *Scandinavian Journal of Statistics*, 40(4):825–845, 2013.

[21] Sébastien Bubeck, Ronen Eldan, and Joseph Lehec. Sampling from a Log-Concave Distribution with Projected Langevin Monte Carlo. *Discrete & Computational Geometry*, 59(4):757–783, 2018.

[22] Yunbum Kook, Yin-Tat Lee, Ruoqi Shen, and Santosh Vempala. Sampling with Riemannian Hamiltonian Monte Carlo in a Constrained Space. *Advances in neural information processing systems*, 35:31684–31696, 2022.

[23] Thomas Power and Dmitry Berenson. Constrained Stein Variational Trajectory Optimization. *IEEE Transactions on Robotics*, 40:3602–3619, 2024.

[24] Marc Toussaint, Cornelius V Braun, and Joaquim Ortiz-Haro. NLP Sampling: Combining MCMC and NLP Methods for Diverse Constrained Sampling. *arXiv preprint arXiv:2407.03035*, 2024.

[25] Krzysztof Łatuszyński, Matthew T. Moores, and Timothée Stumpf-Fétizon. *MCMC Methods for Multi-Modal Distributions*, pages 368–396. Chapman and Hall/CRC, 2 edition, 2026.

[26] Robert H Swendsen and Jian-Sheng Wang. Replica monte carlo simulation of spin-glasses. *Physical review letters*, 57(21):2607, 1986.

[27] Pierre Del Moral, Arnaud Doucet, and Ajay Jasra. Sequential monte carlo samplers. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 68(3):411–436, 2006.

[28] Radford M Neal. Annealed importance sampling. *Statistics and computing*, 11(2):125–139, 2001.

[29] Wenlin Chen, Mingtian Zhang, Brooks Paige, José Miguel Hernández-Lobato, and David Barber. Diffusive Gibbs sampling. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp, editors, *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 7731–7747. PMLR, 21–27 Jul 2024.

[30] Pinaki Mohanty and Ruqi Zhang. Slithering through gaps: Capturing discrete isolated modes via logistic bridging. In Emtiyaz Khan, Yingzhen Li, Arno Solin, and Aaditya Ramdas, editors, *Proceedings of The 29th International Conference on Artificial Intelligence and Statistics*, volume 300 of *Proceedings of Machine Learning Research*, pages 4600–4608. PMLR, 02–05 May 2026.

[31] Francisco J Aragón, Miguel A Goberna, Marco A López, and Margarita ML Rodríguez. *Nonlinear Optimization*. Springer, 2019.

[32] Hubertus Th. Jongen, Peter Jonker, and Frank Twilt. *Nonlinear Optimization in Finite Dimensions: Morse Theory, Chebyshev Approximation, Transversality, Flows, Parametric Aspects*. Nonconvex Optimization and Its Applications. Springer New York, 2000.

[33] Gérard Biau and Luc Devroye. *Lectures on the Nearest Neighbor Method*, volume 246. Springer, 2015.

[34] R Andreani, G Haeser, RW Prado, and LD Secchin. Primal-dual global convergence of an augmented Lagrangian method under the error bound condition. Optimization Online preprint, 2025. URL https://optimization-online.org/?p=31199.

[35] James Dugundji. *Topology*. Allyn and Beacon, Boston, 1966.

[36] Hamed Karimi, Julie Nutini, and Mark Schmidt. Linear Convergence of Gradient and Proximal-Gradient Methods Under the Polyak-Łojasiewicz Condition. In Paolo Frasconi, Niels Landwehr, Giuseppe Manco, and Jilles Vreeken, editors, *Machine Learning and Knowledge Discovery in Databases*, pages 795–811. Springer International Publishing, 2016.

[37] Tiberiu Popoviciu. Sur les équations algébriques ayant toutes leurs racines réelles. *Mathematica*, 9(129-145):20, 1935.

[38] Martin J Wainwright and Michael I Jordan. Graphical Models, Exponential Families, and Variational Inference. *Foundations and Trends® in Machine Learning*, 1(1-2):1–305, 2008.

[39] James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Yash Katariya, Chris Leary, Dougal Maclaurin, George Necula, Adam Paszke, Jake VanderPlas, Skye Wanderman-Milne, and Qiao Zhang. JAX: Composable transformations of Python+NumPy programs, 2018.

[40] L. F. Kozachenko and N. N. Leonenko. Sample Estimate of the Entropy of a Random Vector. *Problems of Information Transmission*, 23(1-2):95–101, 1987.

[41] Richard M. Murray, Zexiang Li, and S. Shankar Sastry. *A Mathematical Introduction to Robotic Manipulation*. CRC Press, Boca Raton, 1994. ISBN 978-1-315-13637-0.

[42] Michael Laskin, Hao Liu, Xue Bin Peng, Denis Yarats, Aravind Rajeswaran, and Pieter Abbeel. Unsupervised reinforcement learning with contrastive intrinsic control. In *Advances in Neural Information Processing Systems*, 2022.

[43] Thomas B. Berrett, Richard Samworth, and Ming Yuan. Efficient multivariate entropy estimation via k-nearest neighbour distances. *The Annals of Statistics*, 47(1):288–318, 2019.

[44] Jasper Snoek, Hugo Larochelle, and Ryan Adams. Practical Bayesian Optimization of Machine Learning Algorithms. In F. Pereira, C.J. Burges, L. Bottou, and K. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012.

[45] Ben Harwood, Amir Dezfouli, Iadine Chades, and Conrad Sanderson. Approximate Nearest Neighbour Search on Dynamic Datasets: An Investigation. In Mingming Gong, Yiliao Song, Yun Sing Koh, Wei Xiang, and Derui Wang, editors, *AI 2024: Advances in Artificial Intelligence*, volume 15443, pages 95–106. Springer Nature Singapore, 2025.

[46] 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, editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016.

A Formal Lemmas and Proofs

We begin by formalizing Lemma 1 introduced in Section 3.

Lemma 1. *Let Σ be measurable with $0 < \sigma_\Sigma(\Sigma) < \infty$, and let*

$$u_\Sigma(x) \coloneqq \frac{1}{S} \quad \textit{for } S = \sigma_\Sigma(\Sigma)$$

denote the uniform density on Σ with respect to σ_Σ. Then it holds that $\log S = H(u_\Sigma) \geq H(\rho_\Sigma)$ for all densities ρ_Σ supported on Σ with equality iff $\rho = u_\Sigma$ almost everywhere.

Proof. We compute the entropy of the uniform density as

$$\mathrm{H}(u_\Sigma) = -\int_\Sigma \frac{1}{S} \log\left(\frac{1}{S}\right) d\sigma_\Sigma = \log S$$

Next, consider the relative entropy of ρ w.r.t. u_Σ:

$$\begin{aligned} D_{KL}(\rho \,|\, u_\Sigma) &= \int_\Sigma \rho \log \frac{\rho}{u_\Sigma} d\sigma_\Sigma = \int_\Sigma \rho \log S \; d\sigma_\Sigma + \int_\Sigma \rho \log \rho \; d\sigma_\Sigma \\ &= \log S - \mathrm{H}(\rho) = \mathrm{H}(u_\Sigma) - \mathrm{H}(\rho). \end{aligned}$$

Now, by Gibbs' inequality $0 \leq D_{KL}(\rho \,|\, u_\Sigma)$ with equality iff $\rho = u_\Sigma$ almost everywhere, proving the statement. □

Next, we restate the results of Section 5 and provide the complete proofs.

Lemma 2 (Component coverage at initialization)**.** *Suppose h, g have Lipschitz-continuous derivatives and satisfy LICQ, and that the feasible set Σ is bounded and decomposes into a finite number of connected components. We draw N points $x_i \sim \mathcal{N}(0, \sigma^2)$ for arbitrary $\sigma > 0$ and apply any descent method on the squared slack* (4) *satisfying sufficient decrease. Then $\mathbb{P}($every component is hit$) \to 1$ as $N \to \infty$.*

Proof. Consider gradient descent on the squared constraint violations

$$\mathrm{Slack}(x) = \frac{1}{2}\left(\|g(x)_+\|^2 + \|h\|^2\right) .$$

We will show that each component Σ_c admits a basin of attraction A_c under gradient descent on $\mathrm{Slack}(x)$ with $\mu(A_c) > 0$. Then, drawing i.i.d. from an uncorrelated Gaussian results in each set of positive measure having a positive probability of being hit. Let the probability of hitting A_c be $p_c > 0$. The probability that no particle among N i.i.d. draws lands in A_c is $(1 - p_c)^N \to 0$, where M_0 is the volume of the sampling region. A union bound over the finitely many components gives the claim.

It is left to show that each component admits a basin of attraction of positive measure. To that end, we show that for each component there exists a nonempty open set U_c, such that gradient descent on points in this set ends up in Σ_c. Then $U_c \subseteq A_c$ and therefore $\mu(A_c) > 0$.

Fix Σ_c and $x^* \in \Sigma_c$. By LICQ, the squared slack $\mathrm{Slack}(x)$ fulfills the error bound [34]. That means there exists $\varepsilon, \kappa > 0$ such that

$$d(x, \Sigma) \leq \kappa \; \mathrm{Slack}(x)^{1/2} \quad \text{for all } x \in U_\varepsilon(x^*) \,. \tag{10}$$

Since the components are compact, there exists

$$\delta \coloneqq \min_{i \neq j} d(\Sigma_i, \Sigma_j) > 0$$

by standard topological arguments [35, Ch. XI 4.4]. We choose $\alpha > 0$ such that $\sqrt{\alpha} \leq \delta/(3\kappa)$, and define $L \coloneqq \{x \in U_\varepsilon(x^*) \mid \mathrm{Slack}(x) < \alpha\}$. For any $x \in L$,

$$d(x, \Sigma) \leq \kappa \, \mathrm{Slack}(x)^{1/2} < \kappa\sqrt{\alpha} \leq \delta/3.$$

By Karimi et al. [36], gradient descent from any $x_0 \in L$ converges to Σ, and, since $\mathrm{Slack}(x_{k+1}) \leq \mathrm{Slack}(x_k)$, iterates remain in L. Therefore any $x_0 \in U_c \coloneqq L \cap U_{\delta/3}(\Sigma_c)$ converges into Σ_c. The set U_c is open and nonempty, hence $\mu(A_c) \geq \mu(U_c) > 0$, which completes the proof. □

Proposition 1 (Mean-field resampling map)**.** *Under Assumption 2, the k-NN radius of a particle $x_i \in \Sigma_c$ concentrates at $\varepsilon_{i,k} \asymp (k\sigma_c/(NV_p\alpha_c))^{1/p}$ in the limit $N \to \infty$ with $k_N \to \infty$, $k_N/N \to 0$ [33, Ch. 2]. Consequently, the resampling weights $w_i \propto \varepsilon_{i,k}^{\tau}$ induce the map*

$$(\Phi\alpha)_c := \frac{\alpha_c^{1-\beta}\,(\alpha_c^*)^{\beta}}{\sum_{j=1}^{C} \alpha_j^{1-\beta}\,(\alpha_j^*)^{\beta}}, \qquad \beta := \tau/p. \tag{5}$$

Proof. The mass assigned to component c after normalization is

$$(\Phi\alpha)_c = \frac{\sum_{x_i\in\Sigma_c} w_i}{\sum_j \sum_{x_k\in\Sigma_j} w_k} = \frac{N_c w_c}{\sum_j N_j w_j} = \frac{\alpha_c\,(\sigma_c/\alpha_c)^{\beta}}{\sum_j \alpha_j\,(\sigma_j/\alpha_j)^{\beta}} = \frac{\alpha_c^{1-\beta}\sigma_c^{\beta}}{\sum_j \alpha_j^{1-\beta}\sigma_j^{\beta}},$$

where we used $N_c = \alpha_c N$ and the concentration $w_c \propto (\sigma_c/\alpha_c)^{\beta}$. Substituting $\sigma_c = S\alpha_c^*$ and cancelling S^{β} gives (5). □

Theorem 1. *Consider constraints h, g with Lipschitz-continuous derivative fulfilling LICQ, which define a bounded constrained set Σ that decomposes into a finite number of connected components. Suppose all components have positive measure with respect to the induced Hausdorff measure on Σ. Under Assumption 2 for any $\tau \in (0, p)$ and any schedule $k_N \to \infty$, $k_N/N \to 0$, the mean-field iterates $\alpha_{t+1} = \Phi(\alpha_t)$ of Algorithm 1 satisfy the following bound*

$$D_{\mathrm{KL}}^{\Sigma}(p_{\alpha_t} \,|\, p_{\alpha^*}) \le C_0 \left(1 - \frac{\tau}{p}\right)^{2t} \tag{6}$$

with

$$C_0 = \frac{1}{4}\left(\max_c \log\frac{\alpha_c^{(0)}}{\alpha_c^*} - \min_c \log\frac{\alpha_c^{(0)}}{\alpha_c^*}\right)^2 \tag{7}$$

for the initial component weights $\alpha_c^{(0)}$ after the projection step of the algorithm. In particular, $p_{\alpha_t} \to u_{\Sigma}$ in KL as $t \to \infty$.

Proof. We prove the claim in three steps: (i) derive a closed form for the iterates, (ii) decompose the KL, and (iii) bound each term.

Step 1: Closed form of the iterates. We recall that under Assumption 2, the k-NN radius of a particle $x_i \in \Sigma_c$ concentrates at $\varepsilon_{i,k} \asymp (k\sigma_c/(NV_p\alpha_c))^{1/p}$ in the limit $N \to \infty$ with $k_N \to \infty$, $k_N/N \to 0$ [33, Ch. 2]. Hence, our weights concentrate as

$$w_i^t = \varepsilon_{i,k}^{\tau} \asymp \left(\frac{k}{NV_p}\frac{\sigma_c}{\alpha_c}\right)^{\tau/p} \propto \left(\frac{\sigma_c}{\alpha_c}\right)^{\tau/p}.$$

Define $y_{t,c} := \alpha_{t,c}/\alpha_c^*$. Substituting $\alpha_{t,c} = y_{t,c}\alpha_c^*$ into (5) then yields:

$$(\Phi\alpha_t)_c = \frac{(y_{t,c}\alpha_c^*)^{1-\beta}(\alpha_c^*)^{\beta}}{\sum_j (y_{t,j}\alpha_j^*)^{1-\beta}(\alpha_j^*)^{\beta}} = \frac{\alpha_c^* y_{t,c}^{1-\beta}}{\sum_j \alpha_j^* y_{t,j}^{1-\beta}}.$$

Using this expression, we can characterize how the gap $y_{t,c}$ changes over time:

$$y_{t+1,c} = \frac{\alpha_{t+1,c}}{\alpha_c^*} = \frac{\cancel{\alpha_c^*}\, y_{t,c}^{1-\beta}}{\cancel{\alpha_c^*}\sum_j \alpha_j^* y_{t,j}^{1-\beta}} = y_{t,c}^{1-\beta}/Z_t \tag{11}$$

with $Z_t := \sum_j \alpha_j^* y_{t,j}^{1-\beta}$ independent of component c. This implies the following recurrence which we prove by induction:

$$y_{t,c} = \frac{y_{0,c}^{a_t}}{\tilde{Z}_t}, \qquad a_t := (1-\beta)^t, \qquad \tilde{Z}_t := \sum_j \alpha_j^* y_{0,j}^{a_t}. \tag{12}$$

Base case $t = 0$. We have $a_0 = 1$ so $\tilde{Z}_0 = \sum_j \alpha_j^{(0)} = 1$. This reduces the statement to $y_{0,c} = y_{0,c}$ which is trivially true.

Inductive step. We have

$$\begin{aligned}
y_{t+1,c} &= \frac{y_{t,c}^{1-\beta}}{\sum_j \alpha_j^* {y_{t,j}}^{1-\beta}} && \text{By (11)} \\
&= \frac{\left(\frac{y_{0,c}^{a_t}}{Z_t}\right)^{1-\beta}}{\sum_j \alpha_j^* \left(\frac{y_{0,j}^{a_t}}{Z_t}\right)^{1-\beta}} && \text{By (12)} \\
&= \frac{\frac{y_{0,c}^{a_t(1-\beta)}}{\cancel{Z_t^{1-\beta}}}}{\frac{1}{\cancel{Z_t^{1-\beta}}} \sum_j \alpha_j^* y_{0,j}^{a_t(1-\beta)}} \\
&= \frac{y_{0,c}^{a_{t+1}}}{\sum_j \alpha_j^* y_{0,j}^{a_{t+1}}}, && \text{By } a_t(1-\beta) = (1-\beta)^{t+1} = a_{t+1}
\end{aligned}$$

which completes the induction.

Now we can express this in tilted exponential form by writing $z_c := \log y_{0,c}$ and $Z(a) := \sum_j \alpha_j^* e^{a z_j}$. Thus (12) becomes

$$\alpha_{t,c} = \frac{\alpha_c^* \, e^{a_t z_c}}{Z(a_t)}. \tag{13}$$

This expression quantifies how much component c was initially over- or under-represented relative to the target and still is at time t.

Step 2: KL decomposition. Using (13), we decompose the KL divergence as follows:

$$D_{\mathrm{KL}}^{\Sigma}(p_{\alpha_t} \,|\, p_{\alpha^*}) = \sum_c \alpha_{t,c} \log \frac{\alpha_c^* e^{a_t z_c}}{\alpha_c^*} = \sum_c \alpha_{t,c} \left(a_t z_c - \log Z(a_t)\right) = a_t \bar{z}_t - \log Z(a_t), \tag{14}$$

where $\bar{z}_t := \sum_c \alpha_{t,c} z_c$ is the first moment of z under α^t and $Z(a) := \sum_j \alpha_j^* e^{a z_j}$ is the normalizing constant.

Step 3: Bounds on each term. Note that α_t is itself a tilted distribution from the exponential form $\alpha_c^t = \alpha_c^* e^{a_t z_c} / Z(a_t)$ and that $Z(a)$ is its moment generating function. We can therefore study how its mean $\bar{z}_t$ deviates from the untilted mean $\bar{z}_{\alpha^*} = \sum_c \alpha_c^* z_c$ by treating the tilt parameter a as continuous and tracking the mean $\mu(a) := \sum_c \nu_{a,c} z_c$ as a varies from 0 (where $\nu_0 = \alpha^*$) to a_t (where $\nu_{a_t} = \alpha_t$).

Upper Bound via Popoviciu's Inequality. Define the log-partition function $\Lambda(a) := \log Z(a)$. Differentiation of this cumulant generating function yields the first moment of the tilted distribution:

$$\Lambda'(a) = \frac{\sum_c \alpha_c^* z_c e^{a z_c}}{Z(a)} = \sum_c \nu_c^{(a)} z_c = \mathbb{E}_{\nu_a}[z] = \mu(a). \tag{15}$$

Differentiating once more yields the variance thereof [38]:

$$\mu'(a) = \Lambda''(a) = \sum_c \nu_{a,c} z_c^2 - \left(\sum_c \nu_{a,c} z_c\right)^2 = \mathbb{V}_{\nu_a}[z]. \tag{16}$$

Since z only takes values in $[\min_c z_c, \max_c z_c]$, we can apply Popoviciu's variance inequality [37], which yields $\mathbb{V}_{\nu_a}(z) \le R^2/4$ for all a. In the last step we define $R := \max_c \log(\alpha_c^{(0)}/\alpha_c^*) - \min_c \log(\alpha_c^{(0)}/\alpha_c^*)$ for clarity.

Integrating this bound over time from 0 to a_t yields a bound on the gap $\mu(a_t) - \mu(0) = \int_0^{a_t} \mu'(a)\, da \le a_t \frac{R^2}{4}$. This establishes a bound for the first term in (14) since

$$a_t \bar{z}_t = a_t \mu(a_t) \le a_t \mu(0) + a_t^2 R^2/4. \tag{17}$$

Lower Bound via Jensen's Inequality. We apply Jensen's inequality directly to the convex exponential function in the log-partition function:

$$Z(a) = \sum_c \alpha_c^* e^{a z_c} \geq \exp\left(a \sum_c \alpha_c^* z_c\right) = e^{a \bar{z}_{\alpha^*}}.$$

Hence, we get

$$\log Z(a_t) \geq a_t \, \bar{z}_{\alpha^*}. \tag{18}$$

The KL Bound. Substituting (17) and (18) into (14):

$$D_{\mathrm{KL}}^{\Sigma}(p_{\alpha_t} \,|\, p_{\alpha^*}) \leq a_t\big(\bar{z}_{\alpha^*} + a_t R^2/4\big) - a_t \bar{z}_{\alpha^*} = \tfrac{R^2}{4}\,(1-\beta)^{2t}.$$

Since p_{α_t} and u_Σ differ only in their component weights and Assumption 2 ensures within-component uniformity, $D_{\mathrm{KL}}^{\Sigma}(p_{\alpha_t} \,|\, u_\Sigma) = D_{\mathrm{KL}}^{\Sigma}(p_{\alpha_t} \,|\, p_{\alpha^*})$, yielding (6). □

Corollary 1. *Assume the conditions of Theorem 1 hold. In the worst case, the number of iterations t to reach a KL-divergence $D_{\mathrm{KL}}^{\Sigma}(p_{\alpha_t} \,|\, p_{\alpha^*}) \leq \varepsilon$ is*

$$t \in \mathcal{O}\left(\frac{\log(1/\varepsilon) + \log\log N}{\tau}\right).$$

Proof. We start by providing an upper bound on the initialization constant C_0 (Eq. (7)) by considering worst-case initialization. We assume using Lemma 2 that each component contains at least one sample, therefore $\alpha_c^{(0)} \in \left[\frac{1}{N}, 1\right)$. This lets us derive

$$\max_c \log \frac{\alpha_c^{(0)}}{\alpha_c^*} \leq \max_c \log \frac{1}{\alpha_c^*} = -\min_c \log \alpha_c^*$$

and similarly

$$\min_c \log \frac{\alpha_c^{(0)}}{\alpha_c^*} \geq \min \log \frac{1/N}{\alpha_c^*} = -\log(N) - \max_c \log \alpha_c^*.$$

Therefore

$$R = \max_c \log \frac{\alpha_c^{(0)}}{\alpha_c^*} - \min_c \log \frac{\alpha_c^{(0)}}{\alpha_c^*} \leq \log N + \max_c \log \alpha_c^* - \min_c \log \alpha_c^*$$

and, since $R > 0$, we know that C_0 fulfills

$$C_0 = \frac{1}{4}R^2 \leq \frac{1}{4}\left(\log N + \max_c \log \alpha_c^* - \min \log \alpha_c^*\right)^2$$

and therefore $C_0 \in \mathcal{O}\left((\log N)^2\right)$ since $\alpha_c^* = \sigma_c$ is independent of N.

By solving $C_0(1 - \frac{\tau}{p})^{2t} = \varepsilon$ for t we then get

$$t \approx \frac{\log(\varepsilon/C_0)}{2\log\left(1-\frac{\tau}{p}\right)} = \frac{\log(1/\varepsilon) + \log C_0}{-2\log(1-\frac{\tau}{p})} \in \mathcal{O}\left(\frac{\log(1/\varepsilon) + \log\log N}{\tau}\right)$$

since $\beta = \frac{\tau}{p} \in (0,1)$ and therefore $\beta \leq -\log(1-\beta)$. □

B Additional Figures

B.1 Synthetic Benchmarks Convergence and Qualitative Results

We list the full learning curves in Figure 5. These curves illustrate the evolution of Sinkhorn and KL divergence as well as constraint violation. More details on the metrics can be found in Appendix D.1. Overall we observe strong performance and fast convergence of both MASEM variants on all problems. While KL divergence is low for all methods on the connected disk problem, separating the disks as depicted in Figure 6 leads to a performance deterioration of all baselines as shown in Figure 5. Even for the connected disks, MASEM leads to faster convergence.

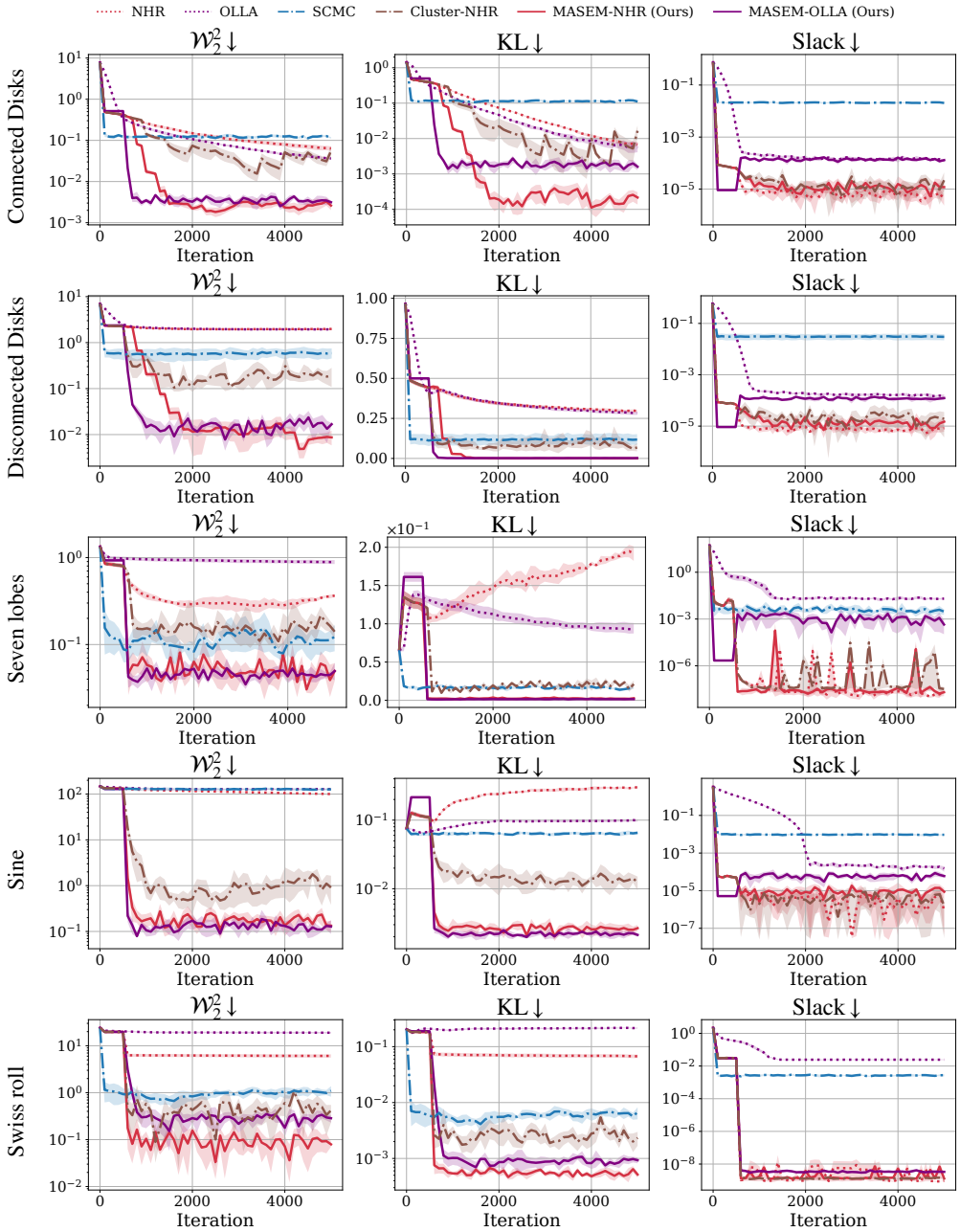


Figure 5: Convergence over 5000 iterations on synthetic benchmarks. From left to right: (1) squared Sinkhorn distance to ground truth samples, (2) KL divergence to pairwise distances of ground truth samples, and (3) mean maximum constraint violation $\mathbb{E}_x[\max\{|h(x)|, g^+(x)\}]$. Solid lines and shaded bands show the mean and 95% CI over five independent runs. Both MASEM-NHR and MASEM-OLLA quickly decrease constraint violation and maintain it there, while achieving the lowest $\mathcal{W}_2^2$ and KL values.

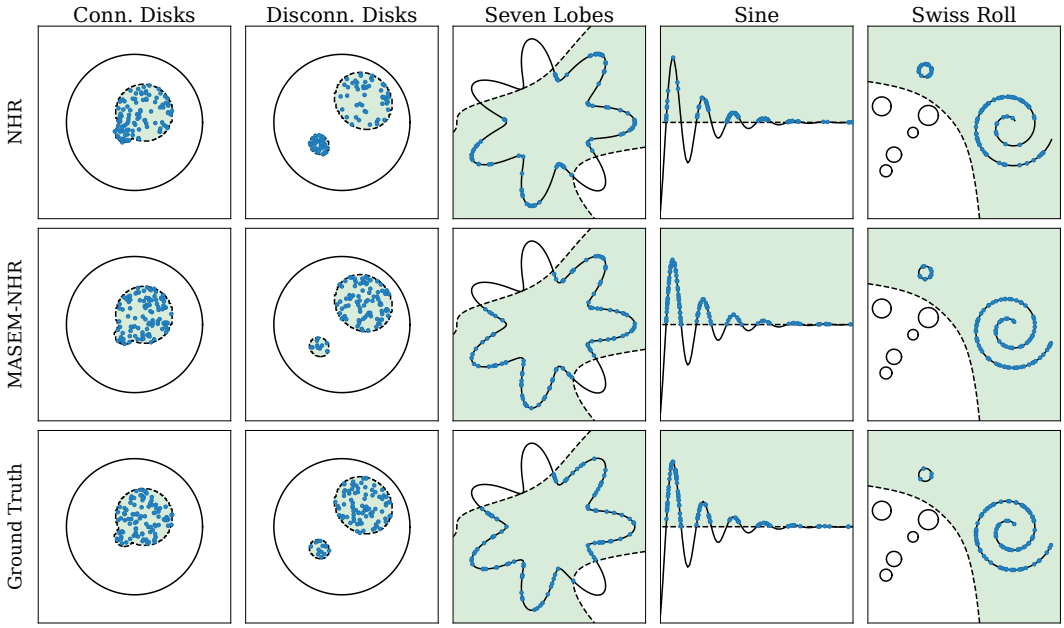


Figure 6: Qualitative results for NHR and MASEM-NHR on all 2d benchmark problems. Black lines show equality constraints, and green shaded areas mark the feasible region(s) by inequality constraints. NHR struggles with moving across components, as can be seen in the disconnected disk and the sine benchmark. Moreover, on manifolds with high curvature, NHR also struggles to distribute particles correctly, since it works by locally linearizing the equality constraints. This can be seen with the seven lobes and the swiss roll benchmark. In both cases, MASEM is able to improve the distribution of particles.

B.2 Scaling Analysis

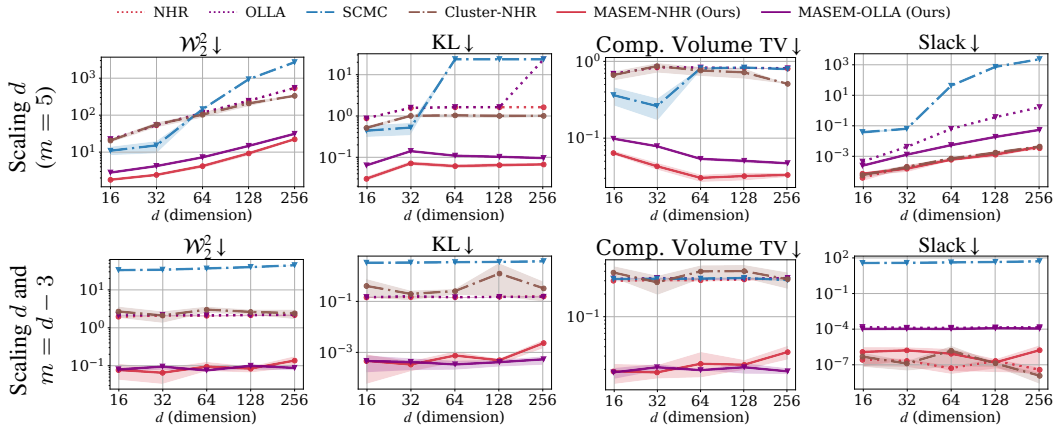


Figure 7: Stress test for scaling ambient and manifold dimension (first row) and scaling the ambient dimension while the manifold dimension stays fixed. (second row). Curves depict mean and 95% CI across 5 seeds. We see that MASEM outperforms alternative approaches by an order of magnitude. Performances are stable across ambient dimensions, indicating that manifold dimension is the main difficulty for samplers. Soft constraint approaches suffer in particular with high-dimensional manifolds.

We report the full scaling results in Figure 7. The plots report Sinkhorn distance and pairwise distance KL divergence w.r.t. to ground truth samples. In addition, we estimate the total variation distance (TV) to the ground truth component volume estimates of the resulting empirical distributions as well as expected maximum constraint violations. We observe that soft constraint approaches scale poorly

on problems with high ambient dimension, likely due to the high number of constraints that must be annealed. Across dimensions MASEM outperforms its NHR and OLLA counterparts for both problems.

B.3 Robotics Problems Convergence Curves

We provide the convergence curves for the robotics problems. In Figure 8 we report the convergence for motion planning. In addition to the feasible entropy and the slack, we report the entropy of path homotopy classes. This corresponds approximately to the entropy of the component weights (for more details see Appendix D.1). In Figure 9 we report the convergence curves for the grasping problem.

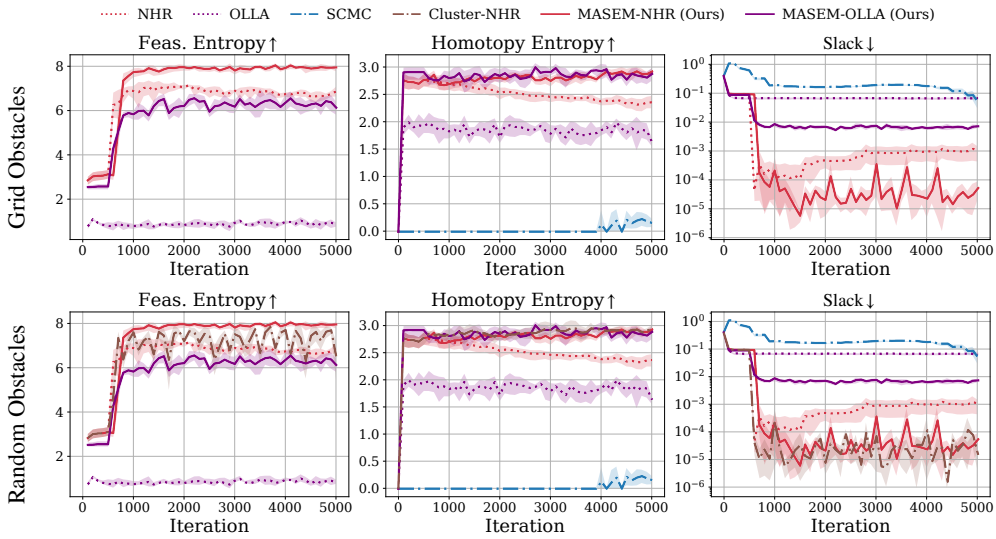


Figure 8: Convergence curves on the motion planning problems with grid obstacles (first row) and random obstacles (second row). Curves depict mean and 95% CI across 5 seeds.

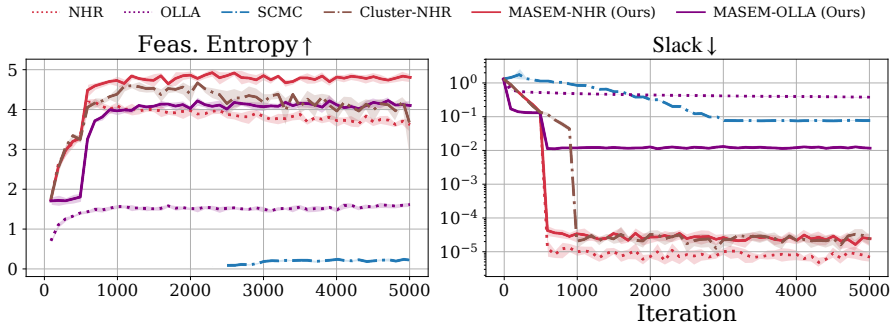


Figure 9: Convergence on the grasping problem. Curves depict mean and 95% CI across 5 seeds.

Table 5: Comparison of sampling using untuned resampling with the default parameters (9) against the tuned sampler.

Problem	MASEM-NHR		MASEM-OLLA	
	Fixed	Tuned	Fixed	Tuned
Connected Disks	0.00 ± 0.00	0.00 ± 0.00	0.01 ± 0.00	0.00 ± 0.00
Disconnected Disks	0.01 ± 0.01	0.01 ± 0.01	0.03 ± 0.01	0.02 ± 0.00
Seven Lobes	0.03 ± 0.01	0.05 ± 0.02	0.04 ± 0.01	0.05 ± 0.02
Sine	0.50 ± 0.07	0.13 ± 0.06	0.13 ± 0.06	0.13 ± 0.07
Swiss Roll	0.25 ± 0.12	0.08 ± 0.02	0.23 ± 0.08	0.29 ± 0.09
Stress Test ($m = 5, d = 256$)	25.21 ± 1.38	21.86 ± 23.40	28.10 ± 2.60	30.50 ± 31.52
Stress Test ($m = d - 3, d = 256$)	0.86 ± 0.08	0.14 ± 0.18	0.10 ± 0.02	0.09 ± 0.10

C Additional Results

C.1 Analysis of Recommended Parameters

We recommend the standard parameters

$$\tau = 0.3, \ M = 20, \text{ and } k = 20 \tag{9}$$

as an initial guess for using our framework. To demonstrate their use, we compare the Sinkhorn divergence on all benchmarks that have a ground truth against the tuned sampler. We note that this analysis was performed post-hoc: at the time of choosing these parameters, the benchmarks and the tuned parameters were already known. Nevertheless, we believe this is a useful validation of our recommended parameters.

Table 5 shows the results of this analysis. We see that for all benchmarks, the fixed parameters land within one order of magnitude of the tuned sampler.

C.2 Performance on High Curvature Manifolds

To test the performance on high-curvature manifolds, we evaluate MASEM on a variant of the high-dimensional stress test. We embed two-dimensional disks with radius r in a $d = 16$-dimensional ambient euclidean space. For each disk, we sample two distorting polynomials p_i, q_i for $i = 1, \dots, d-2$ for the two axes of the disk with coefficients uniform in $[-1, 1]$, and embed the graph of the disk under the product of these distortions $f \colon \mathbb{R}^2 \to \mathbb{R}^{d-2}, f(x, y)_i = p_i(x)q_i(y)$. Last, we rotate the distorted disk by multiplying its points with a randomly generated orthonormal matrix. To measure the effect of the polynomial degree on the curvature, we use the Frobenius norm of the Hessian $\|H_f(x, y)\|_F$ as a measure of local curvature and calculate the mean and max curvature of the disk by Monte Carlo Estimation (see Table 4). Table 6 shows the final Sinkhorn divergence when using a growing number of components and different polynomial degrees. We use MASEM-NHR with the standard parameters (9). Additionally, we provide the baseline performance of tuned NHR for 5 components in Table 4. We see that on five components, MASEM-NHR performs worse at higher curvature, but still beats the baseline. Performance degrades gently as the number of components and the curvature grows.

Table 4: Additional results for the curvature scaling experiment at $|C| = 5$. For polynomials p and q of degree $1, \dots, 6$, we provide the mean of the mean and max curvature, and the baseline Sinkhorn divergence measured by (tuned) NHR and averaged over five runs.

Deg.	mean curv.	max curv.	NHR $\mathcal{W}_2^2 \downarrow$
1	1.42	1.42	$2.57 \pm .24$
2	3.48	4.14	$1.54 \pm .24$
3	5.73	8.08	$3.81 \pm .40$
4	5.86	10.01	4.14 ± 1.05
5	5.95	11.77	$3.77 \pm .42$
6	8.00	19.12	$3.98 \pm .39$

C.3 Comparison of Theoretical and Empirical Convergence Rate

We plot the empirical reduction of component weight error for MASEM and compare it to the mean field bound $C_0(1 - \tau/p)^{2t}$ from Theorem 1 in Figure 10. We observe across all dimensions

Table 6: Final $\mathcal{W}_2^2$ when using MASEM-NHR standard parameters (9) on varying number of components and curvature. Values are mean over 5 seeds ± 95% CI.

$\mathcal{W}_2^2 \downarrow$	polynomial degree of p and q					
	1	2	3	4	5	6
$\lvert C\rvert = 5$	.12 ±.04	.17 ±.01	.69 ±.10	.60 ±.06	1.41 ±.11	1.58 ±.08
$\lvert C\rvert = 10$	.16 ±.04	.25 ±.03	.64 ±.04	.94 ±.06	1.57 ±.12	1.34 ±.04
$\lvert C\rvert = 15$	.13 ±.03	.77 ±.07	.82 ±.05	1.82 ±.07	2.76 ±.07	6.71 ±.04
$\lvert C\rvert = 20$	.26 ±.04	1.05 ±.08	1.71 ±.14	2.71 ±.08	3.56 ±.08	6.75 ±.08

that MASEM rapidly reduces the initial component-weight error geometrically. In $d = 64$ and $d = 256$, the empirical KL remains below the mean-field upper bound throughout the evaluated horizon. In $d = 16$, the empirical curve eventually crosses the idealized bound after approximately $1500 - 2000$ iterations and stabilizes around for MASEM-NHR and for OLLA. These results show that the mean-field contraction accurately describes the initial correction, after which finite-particle and approximate-mixing effects determine the residual accuracy. Importantly, the curves remain stable and most of the initial component imbalance has already been removed before this regime is reached. In addition, we see that the feasibility correction has little effect in lower dimensions. In the more challenging setting, it substantially improves the final KL for both MASEM variants, which supports its role as a practical robustness mechanism when approximate feasibility becomes more consequential. We also observe that the NHR base sampler does not move samples into infeasible regions by design. Therefore, MASEM-NHR does not benefit from the infeasibility-reweighting as much as MASEM-OLLA.

C.4 Component Loss

As noted in Section 5.2, there exists a chance that during resampling, none of the chains in a particular component are sampled, meaning this component is lost. Here, we analyze this phenomenon empirically using a synthetic 2-dimensional benchmark, containing 100 disks in a 10×10 grid. Two neighboring disk-centers are 5 units apart; the disks have radius 1. We run MASEM-NHR for $T = 10$ iterations and count the number of components with at least one chain, averaged over ten runs. Importantly, the maximum step size of NHR is set to 1 to ensure that the chains only jump between components due to resampling. To initialize, we either distribute the chains uniformly across all components (uniform initialization), or populate every component except for one chosen at random starts with a single chain (worst case initialization).

Figure 11 shows the results of this experiment. For a reasonable choice of $\tau \in [0.5, 1.5]$, even just 4 times the amount of components reduces the chance of component loss over ten iterations to 0.1% at

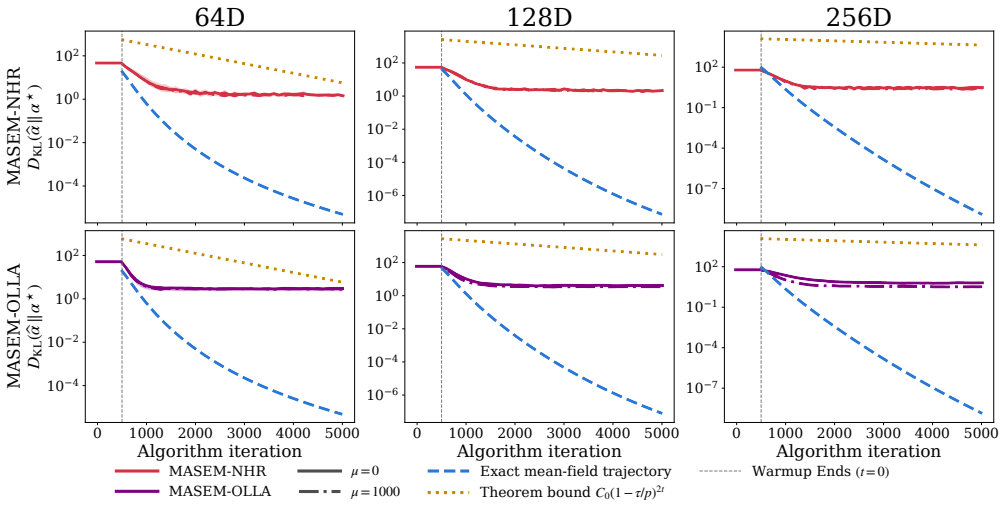


Figure 10: Comparison of empirically observed convergence and exact meanfield rate on the five-component disk stress test problem. Lines show mean and 95% confidence interval across 5 seeds.

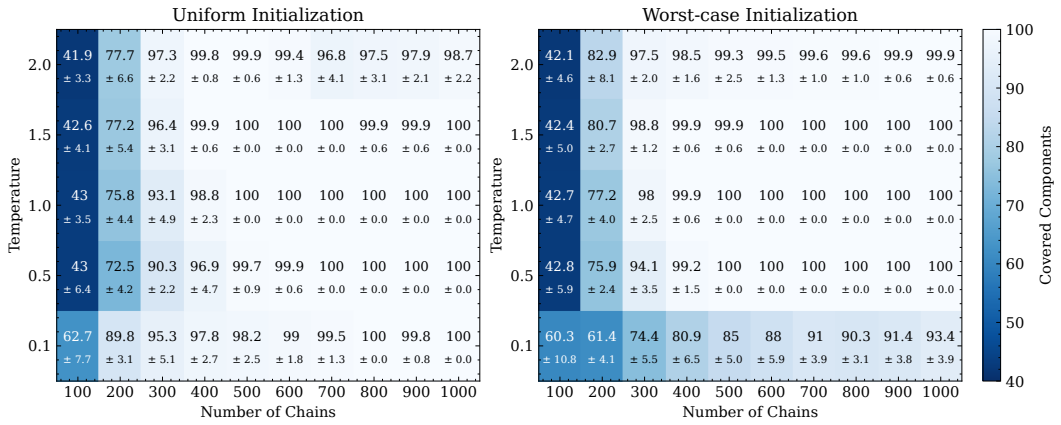


Figure 11: Number of covered components (out of 100), depending on temperature τ and number of chains N. Numbers include 95% confidence interval.

both initializations. Uniform initialization and worst case initialization perform similarly, with the only notable difference being at $\tau = 0.1$. There, uniform initialization achieves 99% coverage with 600 chains, while worst-case initialization does not reach this coverage even at 1000 chains.

C.5 The Pairwise Distance KL-Divergence as Metric

In our reports, we use the (to our knowledge) novel metric of the KL-divergence computed on the histogram of pairwise distances. We define this metric in Appendix D.1. Here, we demonstrate the ability of this metric to distinguish between distributions with slightly different densities. To that end we sample from a disk (ρ_{GT}), a disk with a higher weight in the center (ρ_{center}) and a disk with higher weight at the border (ρ_{edge}). For a divergence $\mathcal{D}$ we pose the null-hypothesis that the divergence does not distinguish between two instances of the ground truth and a non ground truth density and ground truth. More precisely, our hypothesis $H^{\mathcal{D}}_{\text{center|edge}}$ is that $\mathcal{D}(\rho_{\text{GT}}, \rho_{\text{GT}}) \sim \mathcal{D}(\rho_{\text{GT}}, \rho_{\text{center|edge}})$ holds.

Table 7: The table shows the probability that the null hypothesis "the metric cannot distinguish between a ground truth and a non-ground truth distribution" holds. It is computed with a Welch t-test over 1000 runs. Note that only KL rejects the null-hypothesis with $p < 0.01$.

$\mathcal{D}$	$\mathbb{P}(H^{\mathcal{D}}_{\text{center}})$	$\mathbb{P}(H^{\mathcal{D}}_{\text{edge}})$
ED	.068	.68
$\mathcal{W}_2^2$	.013	.36
KL	**4.4 $\times 10^{-34}$**	**2.6 $\times 10^{-7}$**

To measure this, we generate two sets $S_{\text{GT1}}, S_{\text{GT2}}$ as well as S_{center} and S_{edge}. Each set contains 2000 samples. For S_{center}, we sample 50 points of those within a radius of 0.5 of the center instead of the ground truth. For S_{edge}, we instead sample 50 points with radius $r \in [0.5, 1.0]$. With these samples we calculate $\mathcal{D}(S_{\text{GT1}}, S_{\text{GT2}})$, $\mathcal{D}(S_{\text{GT1}}, S_{\text{center}})$, and $\mathcal{D}(S_{\text{GT1}}, S_{\text{edge}})$. We repeat this process 1000 times and use the results to perform a Welch t-test to compute the probability p of rejecting $H^{\mathcal{D}}_{\text{center}}$ and $H^{\mathcal{D}}_{\text{edge}}$ respectively.

We compare the energy distance

$$\text{ED}(X, Y) \coloneqq \left(\frac{2}{nm} \sum_{i,j} \|x_i - y_j\| - \frac{1}{n^2} \sum_{i,j} \|x_i - x_j\| - \frac{1}{m^2} \sum_{i,j} \|y_i - y_j\| \right)^{\frac{1}{2}},$$

used, for example, by Jeon et al. [9], the Sinkhorn distance $\mathcal{W}_2^2$ and our metric (KL). We report the results in Table 7. The KL divergence is the only metric which rejects the hypothesis that it cannot distinguish between ground truth and the other densities with $p < 0.01$.

D Experimental Details

Each experiment is repeated across 5 different seeds. Where applicable, we report the mean across all 5 runs and 95% confidence intervals in our plots. Where reported we test statistical significance

using t-tests and Holm-Bonferroni using SciPy. In the following, we provide details about the benchmarks and implementation that we used in this work. All the experiments are performed on a single workstation with a NVIDIA RTX PRO 6000 Blackwell GPU, a AMD Ryzen Threadripper PRO 7955WX CPU with 16 cores, and 128 gigabyte RAM. All algorithms, benchmarks and metrics are implemented in JAX [39].[2] The 2d problems are depicted in Figure 6. We will provide mathematical and implementation details on the metrics, the benchmarks and their ground truth samplers (if available) below. After that, we provide details and hyperparameters for the method.

D.1 Metrics

The main challenge in comparing different sampling methods is measuring the sampling quality, that is, how well the samples represent the actual distribution. We try to tackle this problem with multiple metrics. For all benchmarks where a ground truth is available, we report the Sinkhorn distance $\mathcal{W}_2^2$. Additionally, we report the average maximum slack violation $\mathbb{E}_x[\max\{|h(x)|, g^+(x)\}]$ for all problems.

For benchmarks with ground truth, we additionally report the KL divergence of pairwise distance histograms. If no ground truth is available, we instead report the feasible entropy. For the scaling benchmarks we calculate the TV distance between the empirical and the uniform component masses. In the path planning problem, we additionally report the entropy of the homotopy-class membership distribution. All these metrics are defined in more detail below.

KL divergence. If a ground truth distribution is available, we report the KL divergence on the histogram of pairwise distances. This is well suited to detect small changes in component mass allocation (Appendix C.5). For this metric, we compute the pairwise distances of the ground truth distribution as well as the approximate distribution. With this, we compute the normalized histogram with 50 bins for values clipped between $[0, b]$, where $b = \max_{x,y\in\Sigma}\|x - y\|_\infty$, i. e. the maximum distance of feasible points along any dimension. We then calculate the forward KL divergence between the ground truth and the approximate distribution.

The pairwise distance KL only compares pairwise distances, and is therefore invariant to relabeling or rotation. To be precise, for any $Q \in \mathrm{SO}(d)$ and $t \in \mathbb{R}^d$ we have $\|(Qx_i+t)-(Qx_j+t)\| = \|x_i-x_j\|$. The distance and therefore the metric itself are in fact invariant to translations, rotations, and reflections. As the metric is also invariant under relabeling, we do not require any information about the components themselves. For samplers that miss some components, the distance distribution changes, which will therefore generally increase the KL with respect to the ground truth.

Feasible Entropy If no ground truth is available, we report the feasible entropy, which is the estimated entropy of the feasible samples multiplied with the fraction of feasible samples. This metric serves as a proxy to measure whether we achieve our goal of entropy maximization. We compute

$$\hat{H}_{\text{feas}} = \frac{\sum 1[\text{Slack}(x) < \text{tol}]}{N} \cdot \frac{1}{30} \sum_{k=2,4,8} \sum_{C_i \subseteq \mathfrak{C}_{\text{feas}}} \hat{H}(C_i, k)\,,$$

where $\mathfrak{C}_{\text{feas}}$ is the set of feasible samples, C_i contains 100 samples drawn without replacement from $\mathfrak{C}_{\text{feas}}$ for $i = 1, \dots 10$ and $\hat{H}(C_i, k)$ denotes the Kozachenko-Leonenko Entropy estimate of C_i, estimated using the k-th nearest neighbor [40]. If less than 100 feasible samples are available, we do not compute this metric. We choose to subsample 100 points to compare entropy across methods with potentially different numbers of feasible samples, since the estimator depends on the number of samples.

TV-Distance We introduce a new metric for the scaling problems, since the other metrics might struggle with high-dimensional samples. For this metric we count the number of samples contained in each component (up to a small tolerance of 10^{-5}) and normalize it. This yields the component weights α_c as introduced in Section 5. We then compute the total variation distance to the ground truth α_c^* as

$$\text{TV} = \max_{c=1,\dots,C} |\alpha_c - \alpha_c^*|$$

[2]The code is published in https://github.com/LIS-TU-Berlin/masem.

Homotopy-Entropy. For the motion-planning problems, we additionally measure the entropy of the distribution of paths among their homotopy-classes. Two paths belong to different homotopy classes if they cross an obstacle on different sides. This directly corresponds to the components of the manifold. More formally, we consider the space $W = ([-4, 4]^2 \setminus O)/\sim$, where $O = \bigcup_i O_i$ is the union of obstacles and $(x_i, y_i) \sim (x_j, y_j)$ iff $x_i = x_j = 3.6$, which is the vertical goal line. We then consider homotopy classes of paths on this space. Note that, since we constrain paths to have increasing x-coordinates, we only observe a finite subset of the homotopy classes. In practice, we compute a signature for each path by testing whether it passes above or below for each obstacle. This results in a $|\text{obs}|$ dimensional vector. We count the occurrences of each signature and compute the entropy of their distribution. Note that while this is a useful metric, maximizing it does not directly correspond to maximizing the entropy of paths. By computing the entropy over the signatures, we implicitly assume that each component has the same weight. While not true in practice, this is still a useful proxy, since the actual component weights are unknown a priori.

D.2 Synthetic Problems

For each low-dimensional example, we run 2 000 independent chains for 5 000 function evaluations. From each chain, we retain only the last state, yielding 2 000 samples.

Connected & Disconnected Disks We consider two randomly rotated disks embedded on a sphere in $\mathbb{R}^3$.

$$\begin{aligned} h(x) &= \|x\|_2 - R, \qquad R = 2.5; \\ g(x) &= \min_{i \in \{1,2\}} \left\{ \cos(\rho_i) - R\langle x, \mu_i \rangle \right\} . \end{aligned}$$

We use two different parameters $\rho_1 = 0.2$, $\rho_2 = 0.6$ such that both disks have different areas. In the connected case the disks have a arc-length of $\delta = 0.6$ between their centers; in the disconnected case we use $\delta = 1.35$. We choose the disk centers

$$\begin{aligned} \mu_1 &= R_x \left(\cos -\tfrac{\delta}{2}, \sin -\tfrac{\delta}{2}, 0\right)^T \\ \mu_2 &= R_x \left(\cos \tfrac{\delta}{2}, \sin \tfrac{\delta}{2}, 0\right)^T \end{aligned} \quad \text{where } R_x = \begin{bmatrix} 1 & 0 & 0 \\ 0 & t & -t \\ 0 & t & t \end{bmatrix} \in \mathrm{SO}(3), \quad t = \frac{\sqrt{2}}{2}$$

is the rotation of 45 degrees around the x-axis. In addition we impose bound constraints $x \in [-5, 5]^3$.

Ground-truth samples are generated by exact surface-uniform sampling from the union of the two disks, with a correction for possible overlap. We first choose disk $i \in \{1, 2\}$ with probability proportional to its spherical area, $\mathbb{P}(i) = \frac{1-\cos(\rho_i)}{\sum_{j=1}^{2}(1-\cos(\rho_j))}$. Given disk i, we sample uniformly on its area by drawing samples from

$$\phi \sim \mathrm{Unif}(0, 2\pi), \qquad \cos\alpha \sim \mathrm{Unif}(\cos\rho_i, 1),$$

and setting $x = R\left[\cos\alpha\, \mu_i + \sin\alpha\, (\cos\phi\, e_{i,1} + \sin\phi\, e_{i,2})\right]$. To correct for overlap, proposals are accepted with probability $1/m(x)$, where $m(x)$ is the number of disks containing x.

Seven Lobes We adapt this problem from Jeon et al. [9] and replace the Gaussian mixture target with constant target $f(x) = 0$ to sample uniformly.

$$\begin{aligned} h(x) &= \sqrt{x_1^2 + x_2^2} - (3 + \cos(7\theta)), \qquad \theta = \mathrm{atan2}(x_2, x_1); \\ g(x) &= (x_1 - 2)^2 - 5x_1x_2^3 + \frac{1}{2}x_2^5 - 40. \end{aligned}$$

Additionally, we impose bound constraints such that $x \in [-4.5, 4.5]^2$. We use rejection sampling from the polar parametrization of the equality constraint to generate ground truth samples. Specifically, we sample

$$\theta \sim \mathrm{Unif}(0, 2\pi), \qquad r(\theta) = 3 + \cos(7\theta), \qquad x(\theta) = r(\theta)(\cos\theta, \sin\theta).$$

We correct the acceptance rate with a Jacobian correction that accepts samples with probability proportional to

$$\frac{ds}{d\theta} = \sqrt{r(\theta)^2 + \left(\frac{dr}{d\theta}\right)^2}, \qquad \frac{dr}{d\theta} = -7\sin(7\theta).$$

Sine We define a uniform target supported on a one-dimensional sine curve in $\mathbb{R}^2$. The constraints are

$$\begin{aligned} h(x) &= x_2 - \exp(-0.15x_1)\sin(x_1); \\ x_2 &\geq 0. \end{aligned}$$

We additionally impose bound constraints such that $x \in [-20, 20]^2$. We generate ground truth samples using rejection sampling from the parametrization of the equality constraint similar to the seven lobes procedure above. Specifically, we sample

$$t \sim \mathrm{Unif}(-20, 20), \qquad x(t) = (t, \exp(-0.15t)\sin t)\,.$$

Since uniform sampling in t is not uniform w.r.t. arclength, we correct the acceptance probability with the Jacobian. We retain only accepted proposals satisfying the halfspace and bound constraints.

Swiss Roll We define a uniform target on a disconnected one-dimensional manifold in $\mathbb{R}^2$ consisting of six circles and one Archimedean spiral segment. The circle components are

$$h_k(x) = \|x - c_k\|_2 - r_k, \qquad k = 1, \ldots, 6,$$

where the centers c_k and radii r_k are sampled once with layout seed 0, using non-overlapping circles with radii in $[0.45, 1.15]$. The spiral is parametrized by

$$\gamma(t) = c_{\mathrm{sp}} + (a + bt)(\cos t, \sin t), \qquad t \in [t_{\min}, t_{\max}],$$

with

$$c_{\mathrm{sp}} = (4.8, -0.4), \qquad a = 0.45, \qquad b = 0.33, \qquad t_{\min} = 0.9, \qquad t_{\max} = 3.9\pi.$$

The equality constraint is the signed residual of the closest circle or valid spiral branch. We additionally impose

$$g(x) = \mathrm{softplus}(-3 - x_1)\,\mathrm{softplus}(-x_2) - 0.1,$$

as well as box constraints $x \in [-10, 10]^2$. Ground-truth samples are generated by first choosing a circle or the spiral with probability proportional to its parameter-domain length, i.e. 2π for each circle and $t_{\max} - t_{\min}$ for the spiral. For circles we sample $\phi \sim \mathrm{Unif}(0, 2\pi)$ and set $x = c_k + r_k(\cos\phi, \sin\phi)$, correcting by the arclength Jacobian. For the spiral we sample $t \sim \mathrm{Unif}(t_{\min}, t_{\max})$ and set $x = \gamma(t)$, correcting by $\frac{ds}{dt} = \sqrt{(a + bt)^2 + b^2}$.

Scaling Stress Test In the synthetic stress-test benchmark we target an uniform disconnected manifold embedded in $\mathbb{R}^d$. We use m to denote the total number of equality constraints. Of these, $m - 1$ are linear equality constraints, which define a random subspace via

$$Ax = b,$$

where $A \in \mathbb{R}^{(m-1)\times d}$ has orthonormal rows and is sampled randomly once. In all stress-test experiments, we set $b = 0$. Let

$$p = d - (m - 1) = d - m + 1$$

denote the dimension of the nullspace of A, and let $N \in \mathbb{R}^{d\times p}$ be an orthonormal basis of this nullspace. We write the corresponding latent coordinates as

$$z = N^\top x.$$

Within this subspace we then randomly sample $|C| = 5$ disconnected hyperspheres which define the manifold on which we sample via equality constraints. For centers $c_i \in \mathbb{R}^p$ and radii r_i, we use

$$h(x) = \|z - c_{i^\star}\|_2^2 - r_{i^\star}^2, \qquad i^\star = \operatorname*{arg\,min}_{i\in\{1,\ldots,5\}} \left| \|z - c_i\|_2^2 - r_i^2 \right|.$$

Since each sphere has dimension $p - 1$, the intrinsic manifold dimension is

$$p - 1 = d - m.$$

The base radius is set to 0.25, and the component radii are multiplied by independent jitter factors in $[0.5, 1.5]$. The component centers are placed randomly in latent space using rejection sampling to ensure separation between components.

We additionally include exactly $l = 5$ spherical inequality constraints, which act as ball cutouts on the sphere components. Each cutout center is placed on the surface of a randomly selected component, and its radius is set to half of the corresponding component radius. For cutout center o_j and radius ρ_j, the inequality is

$$g_j(x) = \rho_j^2 - \|z - o_j\|_2^2, \qquad j = 1, \dots, 5,$$

so that feasible samples lie outside all cutout balls. Finally, we impose box constraints $x \in [-28, 28]^d$.

We compare two main scaling regimes: increasing ambient dimension with fixed manifold dimension for $m = d - 3$ and increasing ambient dimension with increasing manifold dimension for fixed $m = 5$. In this case, the intrinsic dimension grows with the ambient dimension and is given by $d - m = d - 5$.

Ground-truth samples are generated directly from the latent representation. We first sample a component index with probability proportional to its surface area,

$$\mathbb{P}(i) = \frac{r_i^{p-1}}{\sum_{j=1}^{5} r_j^{p-1}}.$$

Given the component, we sample a direction u uniformly on the unit sphere $\mathbb{S}^{p-1}$ by normalizing a standard Gaussian vector, and set

$$z = c_i + r_i u, \qquad x = Nz.$$

Since all stress-test experiments include ball cutouts, we use rejection sampling from this proposal and retain only feasible samples.

D.3 Motion Planning

We consider 2D motion planning problems in which a point robot must navigate from a fixed start configuration $q_0 \in \mathbb{R}^2$ to a goal region while avoiding circular obstacles. We parametrize each path through n_w waypoints $X = (x_1, \dots, x_{n_w}) \in \mathbb{R}^{n_w \times 2}$, which form the sampling domain. Following standard robotics convention, we recover smooth trajectories as cubic B-splines whose control points are obtained from the start and waypoints via $C = \Phi^{-1}[q_0; X]$, using the pseudoinverse of the interpolation matrix $\Phi \in \mathbb{R}^{T \times (n_w+1)}$. Evaluating the resulting curve at T equally spaced parameter values yields a discretized path $p \in \mathbb{R}^{T \times 2}$. The goal is a vertical line on the right side of the workspace, which we impose as an equality constraint on the first coordinate of the final waypoint:

$$h^{\text{goal}}(X) = x_{n_w,1} - 3.6,$$

leaving the second coordinate free so any point on the goal line is admissible.

The main challenge of this domain is navigating around obstacles, where multiple homotopically distinct paths are admissible. Collision constraints are enforced at each path sample via

$$g_{j,t}^{\text{coll}}(p) = r_j^2 - \|p_t - c_j\|_2^2, \qquad t = 1, \dots, T,$$

for each obstacle j with center c_j and radius r_j. We additionally impose monotone progress along the x axis,

$$g_t^{\text{mon}}(p) = p_{t-1,1} - p_{t,1}, \qquad t = 1, \dots, T,$$

with $p_0 \equiv q_0$, which removes looping and backwards-going trajectories. We further impose discrete velocity and acceleration limits to suppress paths with large gaps or sharp kinks, both of which are known failure modes when the path is reconstructed from sparse waypoints:

$$\begin{aligned} g_t^{\text{vel}}(p) &= \|p_t - p_{t-1}\|_2^2 - s_{\max}^2, & s_{\max} &= 1, \\ g_t^{\text{acc}}(p) &= \|p_{t+1} - 2p_t + p_{t-1}\|_2^2 - a_{\max}^2, & a_{\max} &= 0.65. \end{aligned}$$

Finally, all waypoint coordinates are bound-constrained to $[-4, 4]^2$. Since each obstacle splits the configuration space into a homotopy class passing above and one passing below, the feasible set decomposes into up to $2^{|\text{obs}|}$ connected components (modulo monotonicity and feasibility pruning), which we use to evaluate homotopy-class entropy. For more details see Appendix D.1.

Grid Layout. In the grid problem, obstacles are placed on a regular 4×4 grid, that is centered within the domain. All obstacles share a common radius $r = 0.5$. In total, this induces $m = 3$ equality and $l = 771$ inequality constraints. This setup yields a structured benchmark with a known and combinatorially large number of homotopy classes, which is well suited for testing whether a sampler covers all feasible passages between obstacle columns or collapses onto a few preferred routes.

Random Layout. In the random problem, we place 20 disk obstacles with radii drawn independently from $\mathrm{Unif}[0.2, 0.5]$ and center coordinates drawn uniformly over the domain. We employ rejection sampling to guarantee minimal separation between obstacle pairs to increase the number of feasible paths. In total, this induces $m = 3$ equality and $l = 931$ inequality constraints. The random layout induces an irregular and often narrower set of feasible corridors than the grid, reflecting real world robot navigation challenges.

D.4 Grasping

We want to grasp a capsule, which is a cylinder with a halfsphere with the same radius as the cylinder on each end. This shape is commonly used for collision checking. The capsule is placed at the origin $(0, 0, 0)$. Its orientation is given by a vector along its long axis, pointing towards $o = (1, 1, 1)$. It has the length $l = 1$ and the radius $r = 0.25$. Gravity impacts this capsule with the force-vector $f_g = (0, 0, 1)$. Its friction coefficient is $\mu = 1$. Figure 12 shows the capsule and a grasp schematically.

We grasp the capsule with three fingers. The fingers as hard point-fingers with friction [41]. This means each finger i is parametrized by a point of attack $p_i \in \mathbb{R}^3$ as well as a force-vector $f_i \in \mathbb{R}^3$. To ensure the capsule is grasped and stable, the sum of forces of the fingers has to counter gravity, and the sum of torques at it's center $\tau_i = p_i \times f_i$ has to be zero. This yields the constraints

$$f_1 + f_2 + f_3 = f_g$$
$$\tau_1 + \tau_2 + \tau_3 = 0 \,.$$

For each finger i, we calculate a vector pointing from the middle axis of the capsule towards the finger as

$$d_i = \mathrm{clamp}\big(\langle o, p_i \rangle, -l/2, l/2\big) o - p_i \,.$$

This allows us to ensure that the point of attack lies on the capsules surface by constraining $\|d_i\| = r$. Furthermore, each finger can exert a force deviating from the normal of the capsules at the point of attack $n_i = d_i / r$ as far as friction allows. This is expressed in the force cone constraint

$$\|f_i - n_i \langle n_i, f_i \rangle\| \leq \mu \langle -n_i, f_i \rangle \,.$$

We constrain the force each finger can apply to $\|f_i\| \in [0.1, 1]$. Lastly, we introduce an infeasible region on the capsule. This is meant to model objects that can't be grasped at arbitrary points. We choose a band across the middle of the cylinder with width $w = 0.25$. This results in the constraint $|\langle o, p_i \rangle| \geq w/2$.

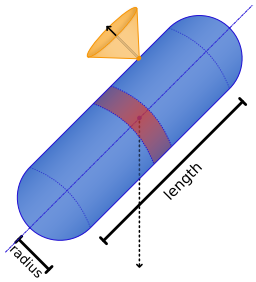


Figure 12: Schematic diagram visualizing the capsule in 2d. The red band in the middle represents the infeasible region. In the top left, a finger contact is shown. The orange dot is the point of attack on the capsule surface. The orange cone shows the friction cone and the black arrow the surface normal. The dotted black arrow in the middle symbolizes gravity.

For each finger, we have 6 dimensions, 1 equality, and 4 inequality constraints. We enforce 6 equalities in the force and torgue constraints. Additionally, we constrain each entry of the points of attacks and forces to be between -1 and 1. Overall, this results in 18 dimensions, 9 equalities and 48 inequalities.

D.5 Implementation Details

All methods are implemented in JAX [39]. We implement all baselines based on their official implementation: NHR is based on the code of Toussaint et al. [24], OLLA based on Jeon et al. [9] and SCMC on Golchi and Campbell [17]. For OLLA, we use the OLLA-H variant which employs Hutchinson's trace estimator.

Initialization. We initialize all sampling runs from a Gaussian in ambient space which we center at the origin and scale by a quarter of the distance between minimum and maximum bound for that problem. Following Toussaint et al. [24], we implement the initial projection for NHR and MASEM-OLLA using the Gauss-Newton method for 500 steps with additional standard Gaussian noise that we scale by $\epsilon = 0.01$. While this is inconsistent with theory, we find that it improves mode coverage of the initialization and yields feasible samples on every problem that we tested on.

MASEM Details. In Section 5, we assume that the local sampling kernel preserves sample feasibility. Since practical samplers tend to break this property resampling naively can increase the number of infeasible samples as those tend to have largest distance to the samples on the manifold. To prevent this from happening, we apply a penalty from the resampling weights:

$$w_i \leftarrow w_i \frac{1}{\exp\big(\mu \operatorname{Slack}(x_i)\big)}$$

We choose $\mu = 1000.$ as default and find that this offers a robust choice. Further, we ensemble multiple values for k to compute the weights [42, 43]. Specifically for $k = 3$ we would compute all distances $\varepsilon_{i,1}, \varepsilon_{i,2}, \varepsilon_{i,3}$ and take the geometric mean over their resulting logarithmic distance $\bar{w}_i = \big(\frac{1}{k}\sum_{j=1}^{k} \log \varepsilon_{i,j}\big)^{\tau}$. Specifically, for fixed k we take the geometric mean the k nearest neighbors distances $\bar{w}_i = \big(\prod_{j=1}^{k} \varepsilon_{i,j}\big)^{\frac{\tau}{k}}$. This approach tends to have lower variance and is less sensitive to the exact choice of k despite introducing bias.

Hyperparameters We tune the main hyperparameters of each method and baseline using Bayesian Optimization [44], implemented in the wandb api.[3] For each method we allocate the same budget of 20 tuning runs which we tune against the Sinkhorn distance on problems with ground truth samples and against the feasible entropy on practical problems. The MASEM hyperparameters for all problems are listed in Table 8.

Table 8: Hyperparameters for MASEM-NHR and MASEM-OLLA. Step sizes and other sampler-related hyperparameters are taken from the corresponding base sampler. Hyperparameters are the temperature τ, the number of mixing steps M and the considered neighbour k.

	MASEM-NHR			MASEM-OLLA		
Problem	τ	M	k	τ	M	k
Connected Disks	1.0	50	4	1.0	50	4
Disconnected Disks	1.0	50	4	1.0	50	4
Seven Lobes	.81	5	8	.57	5	8
Sine	.98	5	16	.94	10	16
Swiss Roll	.65	5	16	.5	50	8
Stress Test ($m = d - 3$)	.3	5	16	.3	5	16
Stress Test ($m = 5$)	.74	10	20	.74	10	20
Motion Planning Grid	.65	50	8	.65	100	8
Motion Planning Random	1.0	50	8	.94	50	4
Grasping	.41	5	19	.94	10	16

E Limitations

E.1 Theoretical Limitations

Some results, namely Lemma 2, Theorem 1, and Corollary 1, are valid only in the large-scale particle limit, i.e. for $N \to \infty$. Furthermore, the main result in Theorem 1 is only stated in the mean-field. It might happen that no particles are sampled for a particular component, resulting in that component vanishing. However, the probability of this occurrence decreases exponentially with the number of samples and our experiments show this effect is negligible even for just 4 chains per component, as we demonstrated in Appendix C.4.

[3] https://docs.coreweave.com/products/wandb/sweeps

E.2 Runtime Overhead

While small, our method introduces some runtime overhead compared to only using constrained samplers like NHR: every M steps, we compute the k-nearest neighbor distances and resample. This is also reflected in the asymptotic runtime. Assuming each iteration of the constrained sampler is constant in the number of samples, we increase the asymptotic runtime from $\mathcal{O}(NTM)$ to $\mathcal{O}(NT(M + \log N)$ when using the proposed resampling.[4] In practice however, we discovered that on our benchmarks this overhead is small, while applying MASEM greatly enhances sample quality.

E.3 Non-Uniform Sampling

This work focuses on entropy maximization, which results in uniformly sampling the feasible set. This limits its usefulness when non-uniform samples are needed. If that is the case, we suggest either applying MASEM and using rejection sampling afterwards, or using the output of MASEM as an unbiased initialization for a Sequential Monte Carlo sampler [8].

E.4 Parameter Selection

This framework introduces additional parameters (k, τ, M), which require tuning. Additionally, one needs to set τ to be smaller than the intrinsic dimension p to ensure the convergence guarantees of Theorem 1 hold. Under LICQ, we know that $p = d - m$. If LICQ doesn't hold, we recommend setting conservative values $\tau < 1$ and increasing T to compensate. Alternatively, one could compute the rank of the Jacobian of the feasible samples after projection to estimate the intrinsic dimension. In general, if the available compute resources allow, we recommend tuning the parameters to maximize the feasible entropy, as we demonstrated on the robotics problems. If this is not feasible, we provide a set of standard parameters (9) as a starting point. These show competitive performance compared to the tuned parameters on the benchmarks in this paper.

F Broader Impact

This work is foundational, and we expect no direct societal impact. Nevertheless, as described in Section 1, constrained sampling has many applications in robotics. An important concern when deploying robots in the real world is safety, especially when they are controlled by a neural network. We believe safe operation can be encouraged during training by sampling from a constrained distribution, where the constraints ensure, for example, that the robot does not collide with the environment. The method presented here should be only one of many components of a holistic system for safe robot operation. While we hope it can play a small part in improving robotic operations, we believe that its direct broader impact is limited.

G List of Acronyms

CGF	Cumulant Generating Function
LICQ	Linear Independent Constraint Qualifier (see Assumption 1)
MASEM	Manifold Sampling via Entropy Maximization (introduced in this paper)
MCMC	Markov Chain Monte Carlo
NHR	Non-linear hit & run (constrained sampler, introduced in [24])
OLLA	Overdamped Langevin with Landing (constrained sampler, introduced in [9])
SCMC	Sequentially Constrained Monte Carlo (constrained sampler, introduced in [17])
SMC	Sequential Monte Carlo
SVGD	Stein Variational Gradient Descent (introduced in [46])

[4] Using approximate nearest neighbors, one can determine the k-nearest neighbor distance in $\mathcal{O}(N \log N)$ [45]. Furthermore, with binary search, the resampling step can be executed in $\mathcal{O}(N \log N)$ as well

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: Summarized contributions in Section 1.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: We briefly mention limitations in the conclusion and provide a more in-depth discussion in Appendix E.

   Guidelines:

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

3. **Theory assumptions and proofs**

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

   Answer: [Yes]

Justification: We provide a proof for each result in Appendix A. In the main test of the paper all theoretical results include a short proof sketch as well as supporting text to guide intuition.

Guidelines:

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

4. **Experimental result reproducibility**

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

Answer: [Yes]

Justification: Appendix D provides extensive details on the implementation details of the method, the metrics and the synthetic benchmarks, aimed to make the experiments reproducible.

Guidelines:

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

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

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

   Answer: [No]

   Justification: Since the framework is described extensively in Section 4, Algorithm 1, and Appendix D.5, we believe it is easily reproducible. Furthermore, since it is designed to work with various constrained samplers, we believe a thorough description of the method is more beneficial to researchers, to enable them to integrate the framework into existing samplers. We will release our code upon publication.

   Guidelines:

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

6. **Experimental setting/details**

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

   Answer: [Yes]

   Justification: See Sections 4 and 6 and Appendix D.

   Guidelines:

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

7. **Experiment statistical significance**

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

   Answer: [Yes]

   Justification: We plot error bars or report $\pm$ for 95% confidence interval for all measured metrics. We omit the error bars for the wallclock plots, since the deviation is so minimal that it is not visible in the final plot.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.

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

8. **Experiments compute resources**

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

   Answer: [Yes]

   Justification: We provide the necessary specifications in Appendix D. We report runtime and scaling in Section 6.2.

   Guidelines:

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

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: The work doesn't requiere human participants or contract work. Since all benchmarks are generated synthetically, no datasets were used or created. We discuss broader impact in Appendix F. No research artefacts are released at this time, and all details necessary to reproduce the results are included in the paper.

   Guidelines:

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

10. **Broader impacts**

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

Answer: [Yes]

Justification: While the research is fundamental, we still discuss potential impact in Appendix F.

Guidelines:

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

11. **Safeguards**

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

Answer: [N/A]

Justification: This work doesn't pose a high risk of misuse.

Guidelines:

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

12. **Licenses for existing assets**

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

Answer: [Yes]

Justification: All methods and benchmarks were implemented by the authors, with references to their publication when applicable.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.

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

13. **New assets**

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

    Answer: [N/A]

    Justification: We do not release assets at this time.

    Guidelines:

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

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

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

    Answer: [N/A]

    Justification: No crowdsourcing or or research with human subjects was conducted.

    Guidelines:

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

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

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

    Answer: [N/A]

    Justification: No crowdsourcing or or research with human subjects was conducted.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

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

    Answer: [N/A]

    Justification: No LLMs were used for any important, original or non-standard components. The authors take responsibility for the content and its accuracy.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.