On Sampling with Approximate Transport Maps

Louis Grenioux, Alain Oliviero Durmus, Éric Moulines, Marylou Gabrié

On Sampling with Approximate Transport Maps

Louis Grenioux [1] **Alain Oliviero Durmus** [1] **Éric Moulines** [1] **Marylou Gabrié** [1]

Abstract

Transport maps can ease the sampling of distributions with non-trivial geometries by transforming them into distributions that are easier to handle. The potential of this approach has risen with the development of Normalizing Flows (NF) which are maps parameterized with deep neural networks trained to push a reference distribution towards a target. NF-enhanced samplers recently proposed blend (Markov chain) Monte Carlo methods with either (i) proposal draws from the flow or (ii) a flow-based reparametrization. In both cases, the quality of the learned transport conditions performance. The present work clarifies for the first time the relative strengths and weaknesses of these two approaches. Our study concludes that multimodal targets can be reliably handled with flow-based proposals up to moderately high dimensions. In contrast, methods relying on reparametrization struggle with multimodality but are more robust otherwise in high-dimensional settings and under poor training. To further illustrate the influence of target-proposal adequacy, we also derive a new quantitative bound for the mixing time of the Independent Metropolis-Hastings sampler.

1. Introduction

Creating a transport map between an intractable distribution of interest and a tractable reference distribution can be a powerful strategy for facilitating inference. Namely, if a bijective map $T : \mathbb{R}^d \to \mathbb{R}^d$ transports a tractable distribution ρ on $\mathbb{R}^d$ to a target π on the same space, then the expectation of any test function $f : \mathbb{R}^d \to \mathbb{R}$ under the target distribution can also be written as the expectation value under the reference distribution

$$\pi(f) = \int_{\mathbb{R}^d} f(x)\mathrm{d}\pi(x) = \int_{\mathbb{R}^d} f(T(x))|J_T(x)|\mathrm{d}\rho(x)\,,$$

where J_T is the Jacobian determinant of T. However, the intractability of the target is traded here with the difficult task of finding the map T.

According to Brenier's theorem (Brenier, 1991), if ρ is absolutely continous then an exact mapping T between ρ and π always exists. Such a map may be known analytically, as in some field theories in physics (Lüscher, 2009). Otherwise, it can be approximated by optimizing a parameterized version of the map. While learned maps typically suffer from approximation and estimation errors, a sufficiently accurate approximated map is still valuable when combined with a reweighting scheme such as a Markov chain Monte Carlo (MCMC) or Importance Sampling (IS).

The choice of the parametrization must ensure that the mapping is invertible and that the Jacobian determinant remains easy to compute. Among the first works combining approximate transport and Monte Carlo methods, (Parno & Marzouk, 2018) proposed using triangular maps. Over the years, the term Normalising Flow, introduced initially to refer to a Gaussianizing map (Tabak & Vanden-Eijnden, 2010), has become a common name for highly flexible transport maps, usually parameterized with neural networks, that allow efficient computations of inverses and Jacobians (Papamakarios et al., 2021; Kobyzev et al., 2021). NFs were developed in particular for generative modelling and are now also a central tool for Monte Carlo algorithms based on transport maps.

While the issue of estimating the map is of great interest in the context of sampling, it is not the focus of this paper; see e.g. (Rezende & Mohamed, 2015; Parno & Marzouk, 2018; Müller et al., 2019; Noé et al., 2019). Instead, we focus on comparing the performance of algorithmic trends among NF-enhanced samplers developed simultaneously. On the one hand, flows have been used as reparametrization maps that improve the geometry of the target before running local traditional samplers such as Hamiltonian Monte Carlo (HMC) (Parno & Marzouk, 2018; Hoffman et al., 2019; Noé et al., 2019; Cabezas & Nemeth, 2022). We refer to these strategies as *neutra-MCMC* methods. On the other hand, the push-forward of the NF base distribution through

[1]École Polytechnique. Correspondence to: Louis Grenioux <louis.grenioux@polytechnique.edu>, Alain Oliviero Durmus <alain.durmus@polytechnique.edu>, Éric Moulines <eric.moulines@polytechnique.edu>, Marylou Gabrié <marylou.gabrie@polytechnique.edu>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023.

the map has also been used as an independent proposal in IS (Müller et al., 2019; Noé et al., 2019), an approach coined *neural-IS*, and in MCMC updates (Albergo et al., 2019; Gabrié et al., 2022; Samsonov et al., 2022) among others. We refer to the latter as *flow-MCMC* methods.

Despite the growing number of applications of NF-enhanced samplers, such as in Bayesian inference (Karamanis et al., 2022; Wong et al., 2022), Energy Based Model learning (Nijkamp et al., 2021), statistical mechanics, (McNaughton et al., 2020), lattice QCD (Abbott et al., 2022b) or chemistry (Mahmoud et al., 2022), a comparative study of the methods of neutra-MCMC, flow-MCMC and neural-IS is lacking. The present work fills this gap:

- We systematically compare the robustness of algorithms with respect to key performance factors: imperfect flow learning, poor conditioning and complex geometries of the target distribution, multimodality and high-dimensions (Section 3).
- We show that flow-MCMC and neural-IS can handle mutlimodal distributions up to moderately high-dimensions while neutra-MCMC is hindered in mixing between modes by the approximate nature of learned flows.
- For unimodal targets, we find that neutra-MCMC is more reliable than flow-MCMC and neural-IS given low-quality flows.
- We provide a new theoretical result on the mixing time of the independent Metropolis-Hastings (IMH) sampler by leveraging for the first time, to the best of our knowledge, a local approximation condition on the importance weights (Section 4).
- Intuitions formed on synthetic controlled cases are confirmed in real-world applications (Section 6).

The code to reproduce the experiments is available at https://github.com/h2o64/flow_mcmc.

2. Background

2.1. Normalizing flows

Normalizing flows are a class of probabilistic models combining a C^1-diffeomorphism $T : \mathbb{R}^d \to \mathbb{R}^d$ and a fully tractable probability distribution ρ on $\mathbb{R}^d$. The *push-forward* of ρ by T is defined as the distribution of $X = T(Z)$ for $Z \sim \rho$ and has a density given by the change of variable as

$$\lambda_T^\rho(x) = \rho(T^{-1}(x))|J_{T^{-1}}(x)|\,, \tag{1}$$

where $|J_T|$ denotes the Jacobian determinant of T. Similarly, given a probability density π on $\mathbb{R}^d$, the *push-backward* of π through T is defined as the distribution of $Z = T^{-1}(X)$ for $X \sim \pi$ and has a density given by $\lambda_{T^{-1}}^\pi$.

A parameterized family of C^1-diffeomorphisms $\{T_\alpha\}_{\alpha\in\mathbb{A}}$ then defines a family of distributions $\{\lambda_{T_\alpha}^\rho\}_{\alpha\in\mathbb{A}}$. This construction has recently been popularised by the use of neural networks (Kobyzev et al., 2021; Papamakarios et al., 2021) for generative modelling and sampling applications.

In the context of sampling, the flow is usually trained so that the push-forward distribution approximates the target distribution i.e., $\lambda_{T_\alpha}^\rho \approx \pi$. As will be discussed in more detail below, the flow can then be used as a reparametrization map or for drawing proposals. Note that we are interested in situations where samples from π are not available a priori when training flows for sampling applications. In that case, the reverse Kullback-Leiber divergence (KL)

$$\mathrm{KL}(\lambda_{T_\alpha}^\rho||\pi) = \int \log(\lambda_{T_\alpha}^\rho(x)/\pi(x))\lambda_{T_\alpha}^\rho(x)\mathrm{d}x \tag{2}$$

can serve as a training target, since it can be efficiently estimated with i.i.d. samples from ρ. Minimizing the reverse KL amounts to variational inference with a NF candidate model (Rezende & Mohamed, 2015). This objective is also referred to in the literature as *self-training* or *training by energy*; it is notoriously prone to mode collapse (Noé et al., 2019; Jerfel et al., 2021; Hackett et al., 2021). On the other hand, the forward KL

$$\mathrm{KL}(\pi||\lambda_{T_\alpha}^\rho) = \int \log(\pi(x)/\lambda_{T_\alpha}^\rho(x))\pi(x)\mathrm{d}x \tag{3}$$

is more manageable but more difficult to estimate because it is an expectation over π. Remedies include importance reweighting (Müller et al., 2019), adaptive MCMC training (Parno & Marzouk, 2018; Gabrié et al., 2022), and sequential approaches to the target distribution (McNaughton et al., 2020; Arbel et al., 2021; Karamanis et al., 2022; Midgley et al., 2022). Regardless of which training strategy is used, the learned model $\lambda_{T_\alpha}^\rho$ always suffers from approximation and estimation errors with respect to the target π. However, the approximate transport map T_α can be used to produce high quality Monte Carlo estimators using the strategies described in the next Section.

2.2. Sampling with transport maps

Since NFs can be efficiently sampled from, they can easily be integrated in Monte Carlo methods relying on proposal distributions, such as IS and certain MCMCs.

neural-IS Importance Sampling uses a tractable proposal distribution, here denoted by λ, to calculate expected values with respect to π (Tokdar & Kass, 2010). Assuming that the support of π is included in the support of λ, we denote

$$w(x) = \pi(x)/\lambda(x) \tag{4}$$

the importance weight function, and define the self-normalized importance sampling estimator (SNIS) (Robert

& Casella, 2005) of the expectation of f under π as

$$\hat{\pi}_N(f) = \sum_{i=1}^{N} w_N^i f(X^i)$$

where $X^{1:N}$ N are i.i.d. samples from λ and

$$w_N^i = w(X^i) \Big/ \sum_{j=1}^{N} w(X^j) \tag{5}$$

are the self-normalized importance weights. For IS to be effective, the proposal λ must be close enough to π in χ-square distance (see (Agapiou et al., 2017, Theorem 1)), which makes IS also notably affected by the curse of dimensionality (e.g., (Agapiou et al., 2017, Section 2.4.1)).

Adaptive IS proposed to consider parametric proposals adjusted to match the target as closely as possible. NFs are suited to achieve this goal: they define a manageable push-forward density, can be easily sampled, and are very expressive. IS using flows as proposals is known as Neural-IS (Müller et al., 2019) or Boltzmann Generator (Noé et al., 2019). NFs were also used in methods building on IS to specifically estimate normalization constants (Jia & Seljak, 2019; Ding & Zhang, 2021; Wirnsberger et al., 2020).

Flow-based independent proposal MCMCs Another way to leverage a tractable $\lambda^{\rho}_{T_\alpha} \approx \pi$ is to use it as a proposal in an MCMC with invariant distribution π. In particular, the flow can be used as a proposal for IMH.

Metropolis-Hastings is a two-step iterative algorithm relying on a proposal Markov kernel, here denoted by $Q(x^{(k)}, \mathrm{d}x) = q(x^{(k)}, x)\mathrm{d}x$. At iteration $k+1$ a candidate x is sampled from Q conditionally to the previous state $x^{(k)}$ and the next state is set according to the rule

$$x^{(k+1)} = \begin{cases} x & \text{w. prob. } \mathrm{acc}(x^{(k)}, x) \\ x^{(k)} & \text{w. prob. } 1 - \mathrm{acc}(x^{(k)}, x) \end{cases}, \tag{6}$$

where, given a target π, the acceptance probability is

$$\mathrm{acc}(x^{(k)}, x) = \min\left(1, \frac{q(x, x^{(k)})\pi(x)}{q(x, x^{(k)})\pi(x^{(k)})}\right). \tag{7}$$

To avoid vanishing acceptance probabilities, the Markov kernel is usually chosen to propose local updates, as in Metropolis-adjusted Langevin (MALA) (Roberts & Tweedie, 1996) or Hamiltonian Monte Carlo (HMC) (Duane et al., 1987; Neal et al., 2011), which exploit the local geometry of the target. These local MCMCs exhibit a tradeoff between update size and acceptability leading in many cases to a long decorrelation time for the resulting chain. Conversely, NFs can serve as non-local proposals in the *independent* Metropolis Hastings setting $q(x, x') = \lambda^{\rho}_{T_\alpha}(x')$.

Since modern computer hardware allows a high degree of parallelism, it may also be advantageous to consider Markov chains with multiple proposals at each iteration, such as multiple-try Metropolis (Liu et al., 2000; Craiu & Lemieux, 2007) or iterative sampling-importance resampling (i-SIR) (Tjelmeland, 2004; Andrieu et al., 2010). The latter has been combined with NFs in (Samsonov et al., 2022). Setting the number of parallel trials to $N > 1$ in i-SIR, $N-1$ propositions are drawn at iteration $(k+1)$:

$$x_l \sim \lambda^{\rho}_{T_\alpha} \text{ for } l = 2 \cdots N$$

and x_1 is set equal to the previous state $x^{(k)}$. The next state $x^{(k+1)}$ is drawn from the set $\{x_l\}_{l=1}^N$ according to the self-normalized weights w_N^l computed as in (5). If $x^{(k+1)}$ is not equal to $x^{(k)}$, the MCMC state has been fully refreshed.

In what follows, we refer to MCMC methods that use NFs as independent proposals, in IMH or in i-SIR, as *flow-MCMC*. These methods suffer from similar pitfalls as IS. If the proposal is not "close enough" to the target, the importance function $w(x) = \pi(x)/\lambda^{\rho}_{T_\alpha}(x)$ fluctuates widely, leading to long rejection streaks that become all the more difficult to avoid as the dimension increases.

Flow-reparametrized local-MCMCs A third strategy of NF-enhanced samplers is to use the reverse map T_α^{-1} defined by the flow to transport π into a distribution which, it is hoped, will be easier to capture with standard local samplers such as MALA or HMC. This strategy was discussed by (Parno & Marzouk, 2018) in the context of triangular flows and by (Noé et al., 2019) and (Hoffman et al., 2019) with modern normalizing-flows, we keep the denomination of *neutra-MCMC* from the latter. neutra-MCMC amounts to sampling the push-backward of the target $\lambda^{\pi}_{T_\alpha^{-1}}$ with local MCMCs before mapping the samples back through T_α. It can be viewed as a reparametrization of the space or a spatially-dependent preconditioning step that has similarities to Riemannian MCMC (Girolami & Calderhead, 2011; Hoffman et al., 2019). Indeed, local MCMCs notoriously suffer from poor conditioning. For example, one can show that MALA has a mixing time[1] scaling as $O(\kappa\sqrt{d})$ - where κ is the conditioning number of the target distribution - provided the target distribution is log-concave (Wu et al., 2022; Chewi et al., 2021; Dwivedi et al., 2018).

Nevertheless, neutra-MCMC does not benefit from the fast decorrelation of flow-MCMC methods, since updates remain local. Still, local-updates might be precisely the ingredient making neutra-MCMC escape the curse of dimensionality in some scenarios. Indeed, if the distribution targeted is log-concave, the mixing time of MALA mentioned above depends only sub-linearly on the dimension. The question becomes, when do scaling abilities of neutra-

[1] See Section 4 for a definition of the mixing time.

MCMCs provided by locality allow to beat neural-IS and flow-MCMCs?

In a recent work, (Cabezas & Nemeth, 2022) also used a flow reparametrization for the Elliptical Slice Sampler (ESS) (Murray et al., 2010) which is gradient free and parameter free[2]. Notably, ESS is able to cross energy barriers to tackle multimodal targets more efficiently than MALA or HMC (Natarovskii et al., 2021).

In our experiments we include different versions of neutra-MCMCs and indicate in parentheses the sampler used on the push-backward: either MALA, HMC or ESS.

3. Synthetic case studies

neural-IS, neutra-MCMC and flow-MCMC build on well-studied Monte Carlo scheme with known strengths and weaknesses (see (Rubinstein & Kroese, 2017) for a textbook). Most of their limitations would be lifted if an exact transport between the base distribution and the target were available, as sampling from the flow would directly amounts to sampling from the target distribution. However, learned maps are imperfect, which leaves open a number of questions about the expected performance of NF-enhanced samplers: Which of the methods is most sensitive to the quality of the transport approximation? How do they work on challenging multimodal destinations? And how do they scale with dimension? In this Section, we present systematic synthetic case studies answering the questions above.

In all of our experiments, we selected the samplers' hyperparameters by optimizing case-specific performance metrics. The length of chains was chosen to be twice the number of steps required to satisfy the $\hat{R}$ diagnosis (Gelman & Rubin, 1992) at the 1.1-threshold for the fastest converging algorithm. We used MALA as local sampler as it is suited for the log-concave distributions considered, faster and easier to tune than HMC. Evaluation metrics are fully descibed in App. D.1.

3.1. neutra-MCMCs are robust to imperfect training

Provided an exact transport is available, drawing independent samples from the base and pushing them through the maps generates i.i.d. samples from the target. This is equivalent to running neural-IS and finding that importance weights (4) are uniformly equal. With an approximate transport map, it is not clear which sampling strategy to prefer.

In practice, the quality of the learned flow depends on many parameters: expressiveness, training objective, optimization procedure etc. In the first experiments that we now present, we design a framework in which the quality of the flow can be adjusted manually.

Our first target distribution is a multivariate Gaussian ($d = 128$) with an ill-conditioned covariance. We define an analytical flow with a scalar quality parameter $t \in [0, 1]$ leading to a perfect transport at $t = 0.5$ and an over-concentrated (respectively under-concentrated) push-forward at $t = 0$ (resp. $t = 1$)[3], as shown on Fig. 1. All the experimental details are reported in App D.2.

For the perfect flow, neural-IS yields the most accurate samples as expected, closely followed by flow-MCMCs. More interestingly, neutra-MCMC (MALA) quickly outperforms flow-MCMC as the flow shifts away from the exact transport, both towards over-spreading and over-concentrating the mass (Fig 1). A low-quality flow leads rapidly to zero-acceptance of NF proposals or very poor participation ratios[4] for IS which translates into neural-IS and flow-MCMC being even less efficient that MALA (see Fig. 12 in App. D.4). Conversely, neutra-MCMCs are found to be robust as imperfect pre-conditioning is still an improvement over a simple MALA. These findings are confirmed by repeating the experiment on Neal's Funnel distribution in App. D.3 (Fig. 11), for which the conditioning number of the target distribution highly fluctuates over its support.

Finally note that flow-MCMC methods using a multiple-try scheme, here i-SIR, remain more efficient than neutra-MCMC for a larger set of flow imperfection compared to the single-try IMH scheme. This advantage is understandable: an acceptable proposal is more likely to be available among multiple tries (see Fig. 12 in App. D.2). If multiple-try schemes are more costly per iteration, wall-clock times may be comparable thanks to parallelization (see Fig. 17 in App. F.1).

3.2. neutra-MCMC may not mix between modes

MCMCs with global updates or IS can effectively capture multiple modes, provided the proposal distribution is well matched to the target. MCMCs with local updates, on the other hand, usually cannot properly sample multimodal targets due to a prohibitive mixing time[5]. Therefore,

[2](Cabezas & Nemeth, 2022) uses ESS with a fixed covariance parameter $\Sigma = I_d$. This choice is justified by the fact that using the neutra-MCMC trick amounts to sampling something close to the base of the flow which is $\mathcal{N}(0, I_d)$

[3]This mimics the behavior when fitting the closest Gaussian of type $\mathcal{N}(0, \sigma I_d)$ with the forward KL if $t = 0$ and with the backward KL if $t = 1$.

[4]The participation ratio of a sample of N IS proposal $1/\sum_{i=1}^{N} {w_N^i}^2 \in [1, N]$ tracks the number of samples contributing in practice to the computation of an SNIS estimator.

[5]Indeed, the Eyring-Kramers law shows that the exit time of a bassin of attraction of the Langevin diffusion has expectation which scales exponentially in the depth of the mode; see e.g.

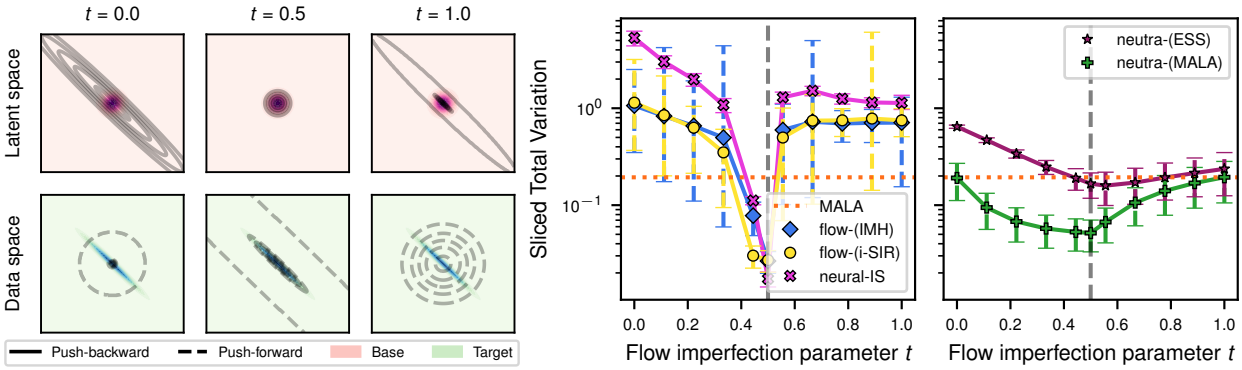

Figure 1. **(Left) Push-backwards $\lambda^{\rho}_{T_t}$ and push-forwards $\lambda^{\pi}_{T_t^{-1}}$ as a function of the flow imperfection parameter t. (Right) Sliced TV distances of different samplers depending on the quality of the flow** t, using 256 chains of length 1400 initialized with draws from NF with T_t. neural-IS was evaluated with 14000 samples. Results were qualitatively unchanged for $d = 16, 32, 64, 256$.

the performance in multimodal environments of neutra-MCMC methods coupled with local samplers depends on the ability of the flow to lower energy barriers while keeping a simple geometry in the push-backward of the target.

We first examined the push-backward of common flow architectures trained by likelihood maximization on 2d target distributions. Both for a mixture of 4 isotropic Gaussians (Fig. 2 left) and for the Two-moons distribution (Fig. 8 in App. C), the approximate transport map is unable to erase energy barriers and creates an intricate push-backward landscape. Visualizing chains in latent and direct spaces makes it is clear that: neutra-MCMC (MALA) mixing is hindered by the complex geometry, the gradient free neutra-MCMC (ESS) mixes more successfully, while flow-MCMC (i-SIR) is even more efficient.

We systematically extended the experiment on 4-Gaussians by training a RealNVP (Dinh et al., 2016) with maximum likelihood for a scale of increasing dimensions (see App. D.5 for all experiment details). We tested the relative ability of the neural-IS, neutra-MCMC and flow-MCMC algorithms to represent the relative weights of each Gaussian component by building histograms of the visited modes within a chain and comparing them with the perfect uniform histogram using a median squared error (Fig. 2 middle). As dimension increases and the quality of the transport map presumably decreases, the ability of neutra-MCMC (MALA) to change bassin worsens. Using neutra-MCMC with ESS enables an approximate mixing between modes but only flow-MCMCs recover the exact histograms robustly up to $d = 256$. In other words, dimension only heightens the performance gap between NF-enhanced samplers observed in small dimension. Note however that the acceptance of independent proposals drops with dimension such that flow-MCMC is also eventually limited by dimension (Fig. 13 in App. D.5). Not included in the plots for readability, neural-IS behaves similarly to flow-MCMC.

In fact, it is expected that exact flows mapping a unmiodal base distribution to a multimodal target distribution are difficult to learn (Cornish et al., 2020). More precisely, Theorem 2.1 of the former reference shows that the bi-Lipschitz constant[6] of a transport map between two distributions with different supports is infinite. Here we provide a complementary result on the illustrative uni-dimensional case of a Gaussian mixture target π and standard normal base ρ:

Proposition 3.1. *Let $\pi = \mathcal{N}(-a, \sigma^2)/2 + \mathcal{N}(a, \sigma^2)/2$ with $a > 0$, $\sigma > 0$ and $\rho = \mathcal{N}(0, 1)$. The unique increasing flow mapping π to ρ denoted $T_{\pi,\rho}$ verifies that*

$$\mathrm{BiLip}(T_{\pi,\rho}) \geq \frac{\mathrm{d}T^{-1}_{\pi,\rho}}{\mathrm{d}z}(0) = \sigma \exp\left(\frac{a^2}{2\sigma^2}\right). \qquad (8)$$

The proof of Proposition 3.1, showing the exponential scaling of the bi-Lipschitz constant in the distance between modes, is postponed to Appendix A.

Overall, these results show that neural-IS and flow-MCMC are typically more effective for multimodal targets than neutra-MCMC. Note further that it has also been proposed to use mixture base distributions (Izmailov et al., 2020) or mixture of NFs (Noé et al., 2019; Gabrié et al., 2022; Hackett et al., 2021) to accommodate for multimodal targets provided that prior knowledge of the modes' structure is available. These mixture models can be easily plugged in neural-IS and flow-MCMC, however it is unclear how to combine them with neutra-MCMC. Finally, mixing neutra-MCMC and flow-MCMC schemes by alternating between global updates (crossing energy barriers) and flow-preconditioned local-steps (robust withing a

(Bovier et al., 2005) and the reference therein.

[6] $\mathrm{BiLip}(f)$ is the bi-Lipschitz constant of f is defined as the infimum over $M \in [1, \infty]$ such that $M^{-1}\|z - z'\| \leq \|f(z) - f(z')\| \leq M\|z - z'\|$ for all z and z' different.

mode) seems promising, in particular when properties of the target distributions are not known a priori. It will be referred below as *neutra-flow-MCMC*. A proposition along these lines was also made in (Grumitt et al., 2022).

3.3. flow-MCMCs are the most impacted by dimension

To investigate the effect of dimension, we ran a systematic experiment on the Banana distribution, a unimodal distribution with complicated geometry (details in App. D.6). We compared NF-enhanced samplers powered by RealNVPs trained to optimize the backward KL and found that a crossover occurs in moderate dimensions : neural-IS and flow-MCMC algorithms are more efficient in small dimension but are more affected by the increase in dimensions compared to neutra-MCMC algorithms (Fig. 2 Right).

4. New mixing rates for IMH

As illustrated by the previous experiment, learning and sampling are expected to be more challenging when dimension increases. To better understand the interplay between flow-quality and dimension, we examined the case of the IMH sampler applied to a strongly log-concave target π and proposal $\mathcal{N}(0, \sigma^2 I_d)$[7], $\sigma > 0$, with density denoted by q_σ.

To this end, we consider the following assumption on the importance weight function $w_\sigma(x) = \pi(x)/q_\sigma(x)$:

Assumption 4.1. For any $R \geq 0$, there exists $C_R \geq 0$ such that for any $x, y \in \mathrm{B}(0, R) = \{z : \|z\| < R\}$:

$$|\log w_\sigma(x) - \log w_\sigma(y)| \leq C_R |x - y| \,. \tag{9}$$

The constant C_R appearing in Assumption 4.1, for $R \geq 0$, represents the quality of the proposal with respect to the target π locally on $\mathrm{B}(0, R)$. Indeed, if $q_\sigma = \pi$ on $\mathrm{B}(0, R)$, this constant is zero. In particular, $q_\sigma \approx \pi$ with q_σ and π smooth and $\nabla \log w_\sigma(x) \approx 0$ on $\mathrm{B}(0, R)$ would result in Assumption 4.1 holding with a small constant C_R. In contrast to existing analyses of IMH which assume w_σ to be uniformly bounded to obtain explicit convergence rates, here we only assume a smooth local condition on $\log(w_\sigma)$, namely that it is locally Lipschitz. Note this latter condition is milder than assuming the former. To the best of our knowledge, it is the first time that such a condition is considered for IMH; a thorough comparison of our contribution with the literature is postponed to Section 5.

While we relax existing conditions on w_σ, we restrict our study to the following particular class of targets:

Assumption 4.2. The target π is positive and $-\log \pi$ is m-strongly convex on $\mathbb{R}^d$ and attains its minimum at 0.

[7]Nevertheless, we develop a theory for a generic proposal in Section E trough Theorem E.4

Denote by P_σ, the IMH Markov kernel with target π and proposal $\mathcal{N}(0, \sigma^2 I_d)$. In our next result, we analyze the mixing time of P_σ, defined for an accuracy $\epsilon > 0$ and an initial distribution μ as

$$\tau_{mix}(\mu, \epsilon) = \inf\{n \in \mathbb{N} \,:\, \|\mu P_\sigma^n - \pi\|_{\mathrm{TV}} \leq \epsilon\} \,. \tag{10}$$

τ_{mix} quantifies the number of MCMC steps needed to bring the total variation distance [8] between the Markov chain and its invariant distribution bellow ϵ.

Theorem 4.3 (Explicit mixing time bounds for IMH)**.** *Assume Assumption-4.1-4.2 hold. Let $0 < \epsilon < 1$ and μ a β-warm distribution with respect to π [9]. Suppose in addition that $C_R \leq \log 2\sqrt{m}/32$ with*

$$R \geq C\sqrt{d} \max\left(\sigma, \frac{1}{\sqrt{m}}\right) \left(1 + |\log^\alpha(\epsilon/\beta)| \,/ d^{\alpha/2}\right), \tag{11}$$

for some explicit numerical constant $C \geq 0$ and exponent $\alpha > 0$. Then the mixing time of IMH is bounded as

$$\tau_{mix}(\mu, \epsilon) \leq 128 \log\left(\frac{2\beta}{\epsilon}\right) \max\left(1, \frac{128^2 C_R^2}{\log(2)^2 m}\right) \,. \tag{12}$$

The proof of Theorem 4.3 is postponed to App. E. It shows that if C_R is bounded by a constant independent of the dimension for R of order at least $\sqrt{d}$, then the mixing time is also independent of the dimension, which recovers easy consequences of existing analyses (Roberts & Rosenthal, 2011; Wang, 2022). In contrast to these works, Theorem 4.3 can be applied to the illustrative case where $\pi = \mathcal{N}(0, I_d)$ and $\sigma = 1 + \lambda$ considering the error term λ either positive *or* negative (for which w_σ is not uniformly bounded). In that case, Theorem 4.3 shows that reaching a precision ϵ with a fixed number of MCMC steps n requires λ to decrease as $\mathcal{O}(1/d)$ (the detailed derivation is postponed to App E). Finally, note that we do not assume that π is L-smooth, i.e., $\nabla \log \pi$ is Lipschitz in Theorem 4.3. This condition is generally considered in existing results on MALA and HMC for strongly log-concave target distributions; see (Dwivedi et al., 2018; Chen et al., 2020).

5. Related Works

Comparison of NF-enhanced samplers Several papers have investigated the difficulty of flow-MCMC algorithms in scaling with dimension (Del Debbio et al., 2021; Abbott et al., 2022a). Hurdles arising from multimodality were also discussed in (Hackett et al., 2021; Nicoli et al., 2022) in the context of flow-MCMC methods. Meanwhile, the authors of (Hoffman et al., 2019) argued that the success of

[8]For two distribution μ, ν on $\mathbb{R}^d$, $\|\mu - \nu\|_{\mathrm{TV}} = \sup_{\mathsf{A} \in \mathcal{B}(\mathbb{R}^d)} |\mu(\mathsf{A}) - \nu(\mathsf{A})|$.

[9]For any Borel set E, $\mu(\mathsf{E}) \leq \beta \pi(\mathsf{E})$

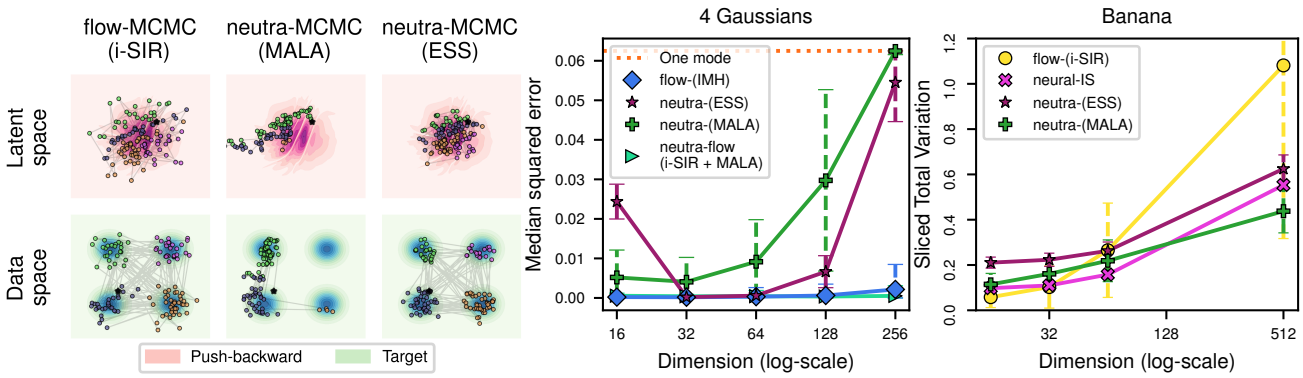

Figure 2. **(Left) Example chains of NF-enhanced walkers with a 2d target mixture of 4 Gaussians.** The 128-step MCMC chain is colored according to the closest mode in the data space (bottom row) with corresponding location in the latent space (top row). The complex geometry of the push-backward $\lambda^{\pi}_{T_\alpha}$ hinders the mixing of local-update algorithms. MALA's step-size was chosen to reach 75% of acceptance. **(Middle) Median squared error of the histograms of visited modes of 4 Gaussians per chain against the perfect uniform histogram as a function of dimension.** 512 chains of 1000-steps on average were used. **(Right) Sliced total variation in sampling the Banana distribution in increasing dimension** using a RealNVP. 128 chains of 1024-steps were used.

their neutra-MCMC was bound to the quality of the flow but did not provide experiments in this direction. To the best of our knowledge, no thorough comparative study of the different NF-enhanced samplers was performed prior to this work.

As previously mentioned, (Grumitt et al., 2022) proposed to mix local NF-preconditioned steps with NF Metropolis-Hastings steps, i.e., to combine neutra-MCMC and flow-MCMC methods. However, the focus of these authors was on the aspect of performing deterministic local updates using an instantaneous estimate of the density of walkers provided by the flow. More related to the present work, they present a rapid ablation study in their Appendix D.

Enhancing Sequential Monte Carlo (Del Moral et al., 2006) with NFs has also been investigated by (Arbel et al., 2021; Karamanis et al., 2022). These methods are more involved and require the definition of a collection of target distributions approaching the final distribution of interest. They could not be directly compared to neural-IS, neutra-MCMC and flow-MCMC. We also note that (Invernizzi et al., 2022) recently proposed another promising method to assist sampling with flows, in the context of replica exchange MCMCs.

IMH analysis Most analyses establishing quantitative convergence bounds rely on the condition that the ratio π/q be uniformly bounded (Yang & Liu, 2021; Brown & Jones, 2021; Wang, 2022). In these works, it is shown that IMH is uniformly geometric in total variation or Wasserstein distances. Our contribution relaxes the uniform boundedness condition on π/q by restricting our study to the class of strongly log-concave targets.

The analysis of local MCMC samplers, such as MALA or HMC for sampling from a strongly log-concave target is now well developed; see e.g., (Dwivedi et al., 2018; Chen et al., 2020; Chewi et al., 2021; Wu et al., 2022). These works rely on the notion of s-conductance for a reversible Markov kernel and on the results developed in (Lovász & Simonovits, 1993) connecting this notion to the kernel's mixing time. This strategy has been successively applied to numerous MCMC algorithm since then; e.g., (Lovász, 1999; Vempala, 2005; Lovász & Vempala, 2007; Chandrasekaran et al.; Mou et al., 2019; Cousins & Vempala; Laddha & Vempala, 2020; Narayanan & Srivastava, 2022). We follow the same path in the proof of Theorem 4.3.

Finally, while (Roberts & Rosenthal, 2011) establish a general convergence for IMH under mild assumption, exploiting this result turns out to be difficult. In particular, we believe it cannot be made quantitative if π/q is unbounded since their main convergence result involves an intractable expectation with respect to the target distribution.

6. Benchmarks on real tasks

In this Section we compare NF-enhanced samplers beyond the previously discussed synthetic examples. Our main findings hold for real world use-cases. An extra experiment on high dimension with image dataset is available in App. G.

6.1. Molecular system

Our first experiment is the alanine dipeptide molecular system, which consists of 22 atoms in an implicit solvent. Our goal is to capture the Boltzmann distribution at temperature $T = 300K$ of the atomic 3D coordinates, which is known to be multimodal. We have used the flow trained

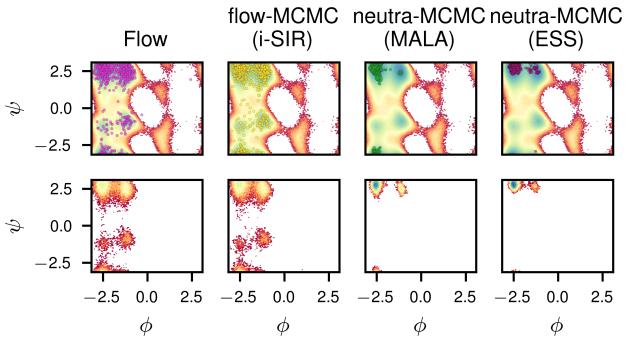

Figure 3. **Sampled configurations of alanine-dipeptide projected from 66 Cartesian coordinates to 2 dihedral angles ϕ and ψ** (see App. F.3). **(Top)** Samples from the flow (left) and samples from a single MCMC chain of the different NF-samplers are shown as bright-colored points on colored background displaying the log histogram of exact samples at $T = 300K$ obtained by a Replica Exchange Molecular Dynamics simulation of (Stimper et al., 2022). **(Bottom)** Log-histograms of samples from the flow (left) and from 256 MCMC chains started at the same location.

Table 1. Predictive posterior distribution for Bayesian sparse logistic regression on the German credit dataset.

Sampler	Average predictive log-posterior distribution
neutra-MCMC (HMC)	-191.1 $\pm$ 0.1
neutra-flow-MCMC (i-SIR + HMC)	-194.1 $\pm$ 1.6
flow-MCMC (i-SIR)	-208.5 $\pm$ 2.1
HMC	-209.7 $\pm$ 1.0

in (Midgley et al., 2022) to drive the samplers and generated 2d projections of the outputs in Figure 3. neutra-MCMC methods are not perfectly mixing between modes, while flow-MCMC properly explores the weaker modes. For more details, see App. F.3.

6.2. Sparse logistic regression

Our second experiment is a sparse Bayesian hierarchical logistic regression on the German credit dataset (Dua & Graff, 2017), which has been used as a benchmark in recent papers (Hoffman et al., 2019; Grumitt et al., 2022; Cabezas & Nemeth, 2022). We trained an Inverse Autoregressive Flow (IAF) (Papamakarios et al., 2017) using the procedure described in (Hoffman et al., 2019). More details about the sampled distribution and the construction of the flow are given in App. F.2. We sampled the posterior predictive distribution on a test dataset and reported the log-posterior predictive density values for these samples in Table 1. neutra-MCMC methods achieve higher posterior predictive values compared to flow-MCMC methods, which differ little from HMC. Note that neutra-flow-MCMC, alternating between flow-MCMC and neutra-MCMC, does not improve upon neutra-MCMC.

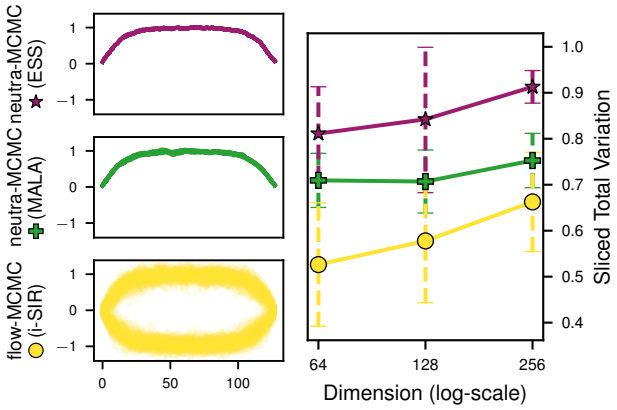

Figure 4. **(Left) Sampled ϕ^4 configurations** in dimension 128. **(Right)** Within-mode Sliced TV as a function of dimension.

6.3. Field system

In our last experiment we investigate the 1-d ϕ^4 model used as a benchmark in (Gabrié et al., 2022). This field system has two well-separated modes at the chosen temperature. Defined at the continuous level, the field can be discretized with different grid sizes, leading to practical implementations in different dimensions. We trained a RealNVP in 64, 128, and 256 dimensions by minimizing an approximated forward KL (more details on this procedure in App. F.4). Consistent with the results of Section 3.2, neutra-MCMC (MALA) chains remain in the modes in which they were initialized, neutra-MCMC (ESS) crosses over to the other mode rarely while flow-MCMC is able to mix properly (see Fig. 4 left). To further examine performance as a function of dimension, we considered the distribution restricted to the initial mode only and calculated the sliced total variation of samplers' chain compared to exact samples (Fig. 4 right). neutra-MCMC methods appear to be less accurate here than flow-MCMC. Even within a mode, the global updates appear to allow for more effective exploration. Both approaches suffer as dimensions grow.

6.4. Run time considerations

In all experiments, algorithms were compared with a fixed sample-size, yet wall-clock time and computational costs per iteration vary between samplers: neural-IS and single-try flow-MCMC require two passes through the flow per iteration, neutra-MCMCs require typically more. Multiple-try flow-MCMC computational cost scales linearly with the number of trials yet can be parallelized. In App. F.1 we report the run-time per iteration for the experiments of this section. Results show that neutra-MCMCs are usually significantly slower per iteration than other methods. Nevertheless, expensive target evaluation such as in Molecular

Table 2. **Informal summary of findings.** The symbol ✓ indicates where algorithms can be expected to produce reliable samples, the symbol is starred ✓* when faster than another suited algorithm for the task. The symbol $\sim$ is used when the algorithm reaches its limits and may fail. A cross ✗ indicates failure to sample the target.

	Unimodal target			Multimodal target		
	low-dim good map	mid-dim fair map	high-dim poor map	low-dim good map	mid-dim fair map	high-dim poor map
flow-MCMC neural-IS	✓*	✓	✗	✓	✓	✗
neutra-MCMC	✓	✓*	$\sim$	$\sim$	✗	✗

Dynamics impact in particular multiple-try flow-MCMC.

7. Conclusion

As a conclusion, we gather our findings in an informal summary table (2) of heuristics depending on the type of target and with respect to the quality of the map and dimension of the problem which typically go together (the lower the dimension, the better the learned map and vice-versa). In synthetic and real experiments, we show that NF provide significant advantage in sampling provided the method is chosen accordingly to the properties of the target. However, high-dimensional multimodal targets remain a challenge for NF-assisted samplers.

Acknowledgements

We thank the anonymous reviewers for their valuable feedback on the paper. L.G. and M.G. acknowledge funding from Hi! Paris. The work was partly supported by ANR-19-CHIA-0002-01 “SCAI”. Part of this research has been carried out under the auspice of the Lagrange Center for Mathematics and Computing. A.O.D. would like to thank the Isaac Newton Institute for Mathematical Sciences for support and hospitality during the programme *The mathematical and statistical foundation of future data-driven engineering* when work on this paper was undertaken. This work was supported by: EPSRC grant number EP/R014604/1

References

Abbott, R., Albergo, M. S., Botev, A., Boyda, D., Cranmer, K., Hackett, D. C., Matthews, A. G. D. G., Racanière, S., Razavi, A., Rezende, D. J., Romero-López, F., Shanahan, P. E., and Urban, J. M. Aspects of scaling and scalability for flow-based sampling of lattice QCD, November 2022a. URL http://arxiv.org/abs/2211.07541. arXiv:2211.07541 [cond-mat, physics:hep-lat].

Abbott, R., Albergo, M. S., Boyda, D., Cranmer, K., Hackett, D. C., Kanwar, G., Racanière, S., Rezende, D. J., Romero-López, F., Shanahan, P. E., Tian, B., and Urban, J. M. Gauge-equivariant flow models for sampling in lattice field theories with pseudofermions. *Physical Review D*, 106(7):074506, October 2022b. ISSN 2470-0010, 2470-0029. doi: 10.1103/PhysRevD.106.074506. URL https://link.aps.org/doi/10.1103/PhysRevD.106.074506.

Agapiou, S., Papaspiliopoulos, O., Sanz-Alonso, D., and Stuart, A. M. Importance sampling: Intrinsic dimension and computational cost. *Statistical Science*, 32(3):405–431, 2017. ISSN 08834237, 21688745. URL http://www.jstor.org/stable/26408299.

Albergo, M. S., Kanwar, G., and Shanahan, P. E. Flow-based generative models for Markov chain Monte Carlo in lattice field theory. *Physical Review D*, 100(3):034515, aug 2019. ISSN 2470-0010. doi: 10.1103/PhysRevD.100.034515. URL http://arxiv.org/abs/2106.05934https://link.aps.org/doi/10.1103/PhysRevD.100.034515.

Andrieu, C., Doucet, A., and Holenstein, R. Particle Markov chain Monte Carlo methods. *Journal of the Royal Statistical Society: Series B*, 72(3):269–342, 2010.

Arbel, M., Matthews, A., and Doucet, A. Annealed Flow Transport Monte Carlo. In *Proceedings of the 38th International Conference on Machine Learning*, pp. 318–330. PMLR, July 2021. URL https://proceedings.mlr.press/v139/arbel21a.html. ISSN: 2640-3498.

Bingham, E., Chen, J. P., Jankowiak, M., Obermeyer, F., Pradhan, N., Karaletsos, T., Singh, R., Szerlip, P. A., Horsfall, P., and Goodman, N. D. Pyro: Deep universal probabilistic programming. *J. Mach. Learn. Res.*, 20:28:1–28:6, 2019. URL http://jmlr.org/papers/v20/18-403.html.

Bobkov, S. G. Isoperimetric and Analytic Inequalities for Log-Concave Probability Measures. *The Annals of Probability*, 27(4):1903 – 1921, 1999. doi: 10.1214/aop/1022874820. URL https://doi.org/10.1214/aop/1022874820.

Bogachev, V. I., Kolesnikov, A. V., and Medvedev, K. V. Triangular transformations of measures. *Sbornik: Mathematics*, 196(3):309, apr 2005. doi: 10.1070/SM2005v196n03ABEH000882. URL https://dx.doi.org/10.1070/SM2005v196n03ABEH000882.

Bovier, A., Klein, M., and Gayrard, V. Metastability in reversible diffusion processes ii. precise asymptotics for small eigenvalues. *Journal of the European Mathematical Society*, 7:69–99, 10 2005. doi: 10.4171/JEMS/22.

Brenier, Y. Polar factorization and monotone rearrangement of vector-valued functions. *Communications on Pure and Applied Mathematics*, 44(4):375–417, 1991. doi: https://doi.org/10.1002/cpa.3160440402. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/cpa.3160440402.

Brown, A. and Jones, G. L. Exact convergence analysis for metropolis-hastings independence samplers in wasserstein distances, 2021.

Cabezas, A. and Nemeth, C. Transport elliptical slice sampling, 2022. URL https://arxiv.org/abs/2210.10644.

Carvalho, C. M., Polson, N. G., and Scott, J. G. Handling sparsity via the horseshoe. In van Dyk, D. and Welling, M. (eds.), *Proceedings of the Twelth International Conference on Artificial Intelligence and Statistics*, volume 5 of *Proceedings of Machine Learning Research*, pp. 73–80, Hilton Clearwater Beach Resort, Clearwater Beach, Florida USA, 16–18 Apr 2009. PMLR. URL https://proceedings.mlr.press/v5/carvalho09a.html.

Chandrasekaran, K., Dadush, D., and Vempala, S. *Thin Partitions: Isoperimetric Inequalities and a Sampling Algorithm for Star Shaped Bodies*, pp. 1630–1645. doi: 10.1137/1.9781611973075.133. URL https://epubs.siam.org/doi/abs/10.1137/1.9781611973075.133.

Che, T., ZHANG, R., Sohl-Dickstein, J., Larochelle, H., Paull, L., Cao, Y., and Bengio, Y. Your gan is secretly an energy-based model and you should use discriminator driven latent sampling. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 12275–12287. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/90525e70b7842930586545c6f1c9310c-Paper.pdf.

Chen, R. T. Q., Behrmann, J., Duvenaud, D. K., and Jacobsen, J.-H. Residual flows for invertible generative modeling. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper/2019/file/5d0d5594d24f0f955548f0fc0ff83d10-Paper.pdf.

Chen, Y., Dwivedi, R., Wainwright, M. J., and Yu, B. Fast mixing of metropolized hamiltonian monte carlo: Benefits of multi-step gradients. *Journal of Machine Learning Research*, 21(92):1–72, 2020. URL http://jmlr.org/papers/v21/19-441.html.

Chewi, S., Lu, C., Ahn, K., Cheng, X., Gouic, T. L., and Rigollet, P. Optimal dimension dependence of the metropolis-adjusted langevin algorithm. In Belkin, M. and Kpotufe, S. (eds.), *Proceedings of Thirty Fourth Conference on Learning Theory*, volume 134 of *Proceedings of Machine Learning Research*, pp. 1260–1300. PMLR, 15–19 Aug 2021. URL https://proceedings.mlr.press/v134/chewi21a.html.

Cornish, R., Caterini, A., Deligiannidis, G., and Doucet, A. Relaxing bijectivity constraints with continuously indexed normalising flows. In III, H. D. and Singh, A. (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 2133–2143. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/cornish20a.html.

Cousins, B. and Vempala, S. *A Cubic Algorithm for Computing Gaussian Volume*, pp. 1215–1228. doi: 10.1137/1.9781611973402.90. URL https://epubs.siam.org/doi/abs/10.1137/1.9781611973402.90.

Craiu, R. V. and Lemieux, C. Acceleration of the multiple-try metropolis algorithm using antithetic and stratified sampling. *Statistics and Computing*, 17(2):109–120, Jun 2007. ISSN 1573-1375. doi: 10.1007/s11222-006-9009-4. URL https://doi.org/10.1007/s11222-006-9009-4.

Del Debbio, L., Rossney, J. M., and Wilson, M. Efficient Modelling of Trivializing Maps for Lattice ϕ^4 Theory Using Normalizing Flows: A First Look at Scalability. *Physical Review D*, 104(9):094507, November 2021. ISSN 2470-0010, 2470-0029. doi: 10.1103/PhysRevD.104.094507. URL http://arxiv.org/abs/2105.12481. arXiv:2105.12481 [hep-lat].

Del Moral, P., Doucet, A., and Jasra, A. Sequential Monte Carlo samplers. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(3):411–436, June 2006. ISSN 1369-7412, 1467-9868. doi: 10.1111/j.1467-9868.2006.00553.x. URL https://onlinelibrary.wiley.com/doi/10.1111/j.1467-9868.2006.00553.x.

Ding, X. and Zhang, B. DeepBAR: A Fast and Exact Method for Binding Free Energy Computation. *The Journal of Physical Chemistry Letters*, 12(10):2509–2515, March 2021. ISSN 1948-7185, 1948-7185. doi: 10.1021/acs.jpclett.1c00189. URL https://pubs.acs.org/doi/10.1021/acs.jpclett.1c00189.

Dinh, L., Sohl-Dickstein, J., and Bengio, S. Density estimation using real nvp, 2016. URL https://arxiv.org/abs/1605.08803.

Dua, D. and Graff, C. UCI machine learning repository, 2017. URL http://archive.ics.uci.edu/ml.

Duane, S., Kennedy, A. D., Pendleton, B. J., and Roweth, D. Hybrid Monte Carlo. *Physics Letters B*, 195(2):216–222, September 1987. ISSN 0370-2693. doi: 10.1016/0370-2693(87)91197-X. URL https://www.sciencedirect.com/science/article/pii/037026938791197X.

Dwivedi, R., Chen, Y., Wainwright, M. J., and Yu, B. Log-concave sampling: Metropolis-hastings algorithms are fast! In Bubeck, S., Perchet, V., and Rigollet, P. (eds.), *Proceedings of the 31st Conference On Learning Theory*, volume 75 of *Proceedings of Machine Learning Research*, pp. 793–797. PMLR, 06–09 Jul 2018. URL https://proceedings.mlr.press/v75/dwivedi18a.html.

Gabrié, M., Rotskoff, G. M., and Vanden-Eijnden, E. Adaptive monte carlo augmented with normalizing flows. *Proceedings of the National Academy of Sciences*, 119(10):e2109420119, 2022. doi: 10.1073/pnas.2109420119. URL https://www.pnas.org/doi/abs/10.1073/pnas.2109420119.

Gelman, A. and Rubin, D. B. Inference from Iterative Simulation Using Multiple Sequences. *Statistical Science*, 7(4):457 – 472, 1992. doi: 10.1214/ss/1177011136. URL https://doi.org/10.1214/ss/1177011136.

Girolami, M. and Calderhead, B. Riemann manifold langevin and hamiltonian monte carlo methods. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 73(2):123–214, 2011. ISSN 13697412, 14679868.

Grumitt, R. D. P., Dai, B., and Seljak, U. Deterministic langevin monte carlo with normalizing flows for bayesian inference, 2022. URL https://arxiv.org/abs/2205.14240.

Hackett, D. C., Hsieh, C.-C., Albergo, M. S., Boyda, D., Chen, J.-W., Chen, K.-F., Cranmer, K., Kanwar, G., and Shanahan, P. E. Flow-based sampling for multimodal distributions in lattice field theory, July 2021. URL http://arxiv.org/abs/2107.00734. arXiv:2107.00734 [cond-mat, physics:hep-lat].

Hoffman, M. D., Sountsov, P., Dillon, J. V., Langmore, I., Tran, D., and Vasudevan, S. NeuTra-lizing Bad Geometry in Hamiltonian Monte Carlo Using Neural Transport. In *1st Symposium on Advances in Approximate Bayesian Inference, 2018 1–5*, 2019. URL http://arxiv.org/abs/1903.03704.

Invernizzi, M., Krämer, A., Clementi, C., and Noé, F. Skipping the Replica Exchange Ladder with Normalizing Flows. *The Journal of Physical Chemistry Letters*, 13(50):11643–11649, December 2022. doi: 10.1021/acs.jpclett.2c03327. URL https://doi.org/10.1021/acs.jpclett.2c03327. Publisher: American Chemical Society.

Izmailov, P., Kirichenko, P., Finzi, M., and Wilson, A. G. Semi-Supervised Learning with Normalizing Flows. *Proceedings of the 37 th International Conference on Machine Learning*, PMLR 119, 2020.

Jerfel, G., Wang, S., Wong-Fannjiang, C., Heller, K. A., Ma, Y., and Jordan, M. I. Variational refinement for importance sampling using the forward kullback-leibler divergence. In *Uncertainty in Artificial Intelligence*, pp. 1819–1829. PMLR, 2021.

Jia, H. and Seljak, U. Normalizing Constant Estimation with Gaussianized Bridge Sampling, December 2019. URL http://arxiv.org/abs/1912.06073. arXiv:1912.06073 [astro-ph, stat].

Karamanis, M., Beutler, F., Peacock, J. A., Nabergoj, D., and Seljak, U. Accelerating astronomical and cosmological inference with Preconditioned Monte Carlo. *Monthly Notices of the Royal Astronomical Society*, 516(2):1644–1653, September 2022. ISSN 0035-8711, 1365-2966. doi: 10.1093/mnras/stac2272. URL http://arxiv.org/abs/2207.05652. arXiv:2207.05652 [astro-ph, physics:physics].

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization, 2014. URL https://arxiv.org/abs/1412.6980.

Kobyzev, I., Prince, S. J., and Brubaker, M. A. Normalizing flows: An introduction and review of current methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(11):3964–3979, nov 2021. doi: 10.1109/tpami.2020.2992934. URL https://doi.org/10.1109%2Ftpami.2020.2992934.

Laddha, A. and Vempala, S. Convergence of gibbs sampling: Coordinate hit-and-run mixes fast, 2020. URL https://arxiv.org/abs/2009.11338.

Liu, J. S., Liang, F., and Wong, W. H. The multiple-try method and local optimization in metropolis sampling. *Journal of the American Statistical Association*, 95(449):121–134, 2000. ISSN 01621459. URL http://www.jstor.org/stable/2669532.

Lovász, L. Hit-and-run mixes fast. *Mathematical Programming*, 86(3):443–461, Dec 1999. ISSN 1436-4646. doi: 10.1007/s101070050099. URL https://doi.org/10.1007/s101070050099.

Lovász, L. and Simonovits, M. Random walks in a convex body and an improved volume algorithm. *Random Structures & Algorithms*, 4(4):359–412, 1993. doi: https://doi.org/10.1002/rsa.3240040402. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/rsa.3240040402.

Lovász, L. and Vempala, S. The geometry of log-concave functions and sampling algorithms. *Random Structures & Algorithms*, 30(3):307–358, 2007. doi: https://doi.org/10.1002/rsa.20135. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/rsa.20135.

Lüscher, M. Trivializing Maps, the Wilson Flow and the HMC Algorithm. *Communications in Mathematical Physics*, 293(3):899, November 2009. ISSN 1432-0916. doi: 10.1007/s00220-009-0953-7. URL https://doi.org/10.1007/s00220-009-0953-7.

Mahmoud, A. H., Masters, M., Lee, S. J., and Lill, M. A. Accurate Sampling of Macromolecular Conformations Using Adaptive Deep Learning and Coarse-Grained Representation. *Journal of Chemical Information and Modeling*, 62(7):1602–1617, April 2022. ISSN 1549-9596, 1549-960X. doi: 10.1021/acs.jcim.1c01438. URL https://pubs.acs.org/doi/10.1021/acs.jcim.1c01438.

McNaughton, B., Milošević, M. V., Perali, A., and Pilati, S. Boosting Monte Carlo simulations of spin glasses using autoregressive neural networks. *Physical Review E*, 101(Mc):1–13, 2020. ISSN 24700053. doi: 10.1103/PhysRevE.101.053312. URL http://arxiv.org/abs/2002.04292.

Midgley, L. I., Stimper, V., Simm, G. N. C., Schölkopf, B., and Hernández-Lobato, J. M. Flow annealed importance sampling bootstrap, 2022. URL https://arxiv.org/abs/2208.01893.

Miyato, T., Kataoka, T., Koyama, M., and Yoshida, Y. Spectral normalization for generative adversarial networks. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=B1QRgziT-.

Mou, W., Ho, N., Wainwright, M. J., Bartlett, P. L., and Jordan, M. I. Sampling for bayesian mixture models: Mcmc with polynomial-time mixing, 2019. URL https://arxiv.org/abs/1912.05153.

Murray, I., Adams, R., and MacKay, D. Elliptical slice sampling. In Teh, Y. W. and Titterington, M. (eds.), *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, volume 9 of *Proceedings of Machine Learning Research*, pp. 541–548, Chia Laguna Resort, Sardinia, Italy, 13–15 May 2010. PMLR. URL https://proceedings.mlr.press/v9/murray10a.html.

Müller, T., Mcwilliams, B., Rousselle, F., Gross, M., and Novák, J. Neural Importance Sampling. *ACM Transactions on Graphics*, 38(5):1–19, October 2019. ISSN 0730-0301, 1557-7368. doi: 10.1145/3341156. URL https://dl.acm.org/doi/10.1145/3341156.

Narayanan, H. and Srivastava, P. On the mixing time of coordinate hit-and-run. *Combinatorics, Probability and Computing*, 31(2):320–332, 2022. doi: 10.1017/S0963548321000328.

Natarovskii, V., Rudolf, D., and Sprungk, B. Geometric convergence of elliptical slice sampling. In Meila, M. and Zhang, T. (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 7969–7978. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/natarovskii21a.html.

Neal, R. M. et al. Mcmc using hamiltonian dynamics. *Handbook of markov chain monte carlo*, 2(11):2, 2011.

Nicoli, K. A., Anders, C., Funcke, L., Hartung, T., Jansen, K., Kessel, P., Nakajima, S., and Stornati, P. Machine Learning of Thermodynamic Observables in the Presence of Mode Collapse. In *Proceedings of The 38th International Symposium on Lattice Field Theory — PoS(LATTICE2021)*, pp. 338, May 2022. doi: 10.22323/1.396.0338. URL http://arxiv.org/abs/2111.11303. arXiv:2111.11303 [hep-lat].

Nijkamp, E., Gao, R., Sountsov, P., Vasudevan, S., Pang, B., Zhu, S.-C., and Wu, Y. N. Mcmc should mix: Learning energy-based model with neural transport latent space mcmc. In *International Conference on Learning Representations*, 2021.

Noé, F., Olsson, S., Köhler, J., and Wu, H. Boltzmann generators: Sampling equilibrium states of many-body systems with deep learning. *Science*, 365 (6457), 2019. ISSN 10959203. doi: 10.1126/science.aaw1147. URL https://www.science.org/doi/10.1126/science.aaw1147.

Papamakarios, G., Pavlakou, T., and Murray, I. Masked autoregressive flow for density estimation. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper/2017/file/6c1da886822c67822bcf3679d04369fa-Paper.pdf.

Papamakarios, G., Nalisnick, E., Rezende, D. J., Mohamed, S., and Lakshminarayanan, B. Normalizing flows for probabilistic modeling and inference. *Journal of Machine Learning Research*, 22(57):1–64, 2021. URL http://jmlr.org/papers/v22/19-1028.html.

Parno, M. D. and Marzouk, Y. M. Transport map accelerated markov chain monte carlo. *SIAM/ASA Journal on Uncertainty Quantification*, 6(2):645–682, jan 2018. doi: 10.1137/17m1134640. URL https://doi.org/10.1137%2F17m1134640.

Rezende, D. and Mohamed, S. Variational Inference with Normalizing Flows. In *Proceedings of the 32nd International Conference on Machine Learning*, pp. 1530–1538. PMLR, June 2015. URL https://proceedings.mlr.press/v37/rezende15.html. ISSN: 1938-7228.

Robert, C. P. and Casella, G. *Monte Carlo Statistical Methods (Springer Texts in Statistics)*. Springer-Verlag, Berlin, Heidelberg, 2005. ISBN 0387212396.

Roberts, G. O. and Rosenthal, J. S. Quantitative non-geometric convergence bounds for independence samplers. *Methodology and Computing in Applied Probability*, 13(2):391–403, Jun 2011. ISSN 1573-7713. doi: 10.1007/s11009-009-9157-z. URL https://doi.org/10.1007/s11009-009-9157-z.

Roberts, G. O. and Tweedie, R. L. Exponential convergence of langevin distributions and their discrete approximations. *Bernoulli*, 2(4):341–363, 1996.

ISSN 13507265. URL http://www.jstor.org/stable/3318418.

Rubinstein, R. Y. and Kroese, D. P. *Simulation and the Monte Carlo method*. Wiley series in probability and statistics. John Wiley & Sons, Inc, Hoboken, New Jersey, third edition edition, 2017. ISBN 978-1-118-63220-8 978-1-118-63238-3.

Samsonov, S., Lagutin, E., Gabrié, M., Durmus, A., Naumov, A., and Moulines, E. Local-global mcmc kernels: the best of both worlds. In *Advances in Neural Information Processing Systems*, 2022.

Stimper, V., Midgley, L. I., Simm, G. N. C., Schölkopf, B., and Hernández-Lobato, J. M. Alanine dipeptide in an implicit solvent at 300k, August 2022. URL https://doi.org/10.5281/zenodo.6993124.

Tabak, E. G. and Vanden-Eijnden, E. Density estimation by dual ascent of the log-likelihood. *Communications in Mathematical Sciences*, 8(1):217–233, 2010. ISSN 15396746. doi: 10.4310/CMS.2010.v8.n1.a11.

Tjelmeland, H. Using all metropolis–hastings proposals to estimate mean values. Technical report, 2004.

Tokdar, S. T. and Kass, R. E. Importance sampling: a review. *WIREs Computational Statistics*, 2 (1):54–60, 2010. doi: https://doi.org/10.1002/wics.56. URL https://wires.onlinelibrary.wiley.com/doi/abs/10.1002/wics.56.

Vempala, S. Geometric random walks: a survey. *Combinatorial and computational geometry*, 52(573-612):2, 2005.

Wang, G. Exact convergence analysis of the independent Metropolis-Hastings algorithms, 2022. URL https://doi.org/10.3150/21-BEJ1409.

Wirnsberger, P., Ballard, A. J., Papamakarios, G., Abercrombie, S., Racanière, S., Pritzel, A., Jimenez Rezende, D., and Blundell, C. Targeted free energy estimation via learned mappings. *The Journal of Chemical Physics*, 153(14):144112, October 2020. ISSN 0021-9606, 1089-7690. doi: 10.1063/5.0018903. URL https://aip.scitation.org/doi/10.1063/5.0018903.

Wong, K. W. K., Gabrié, M., and Foreman-Mackey, D. flowMC: Normalizing-flow enhanced sampling package for probabilistic inference in Jax, November 2022. URL http://arxiv.org/abs/2211.06397. arXiv:2211.06397 [astro-ph].

Wu, K., Schmidler, S., and Chen, Y. Minimax mixing time of the metropolis-adjusted langevin algorithm for log-concave sampling. *Journal of Machine Learning Research*, 23(270):1–63, 2022. URL http://jmlr.org/papers/v23/21-1184.html.

Yang, X. and Liu, J. S. Convergence rate of multiple-try metropolis independent sampler, 2021. URL https://arxiv.org/abs/2111.15084.

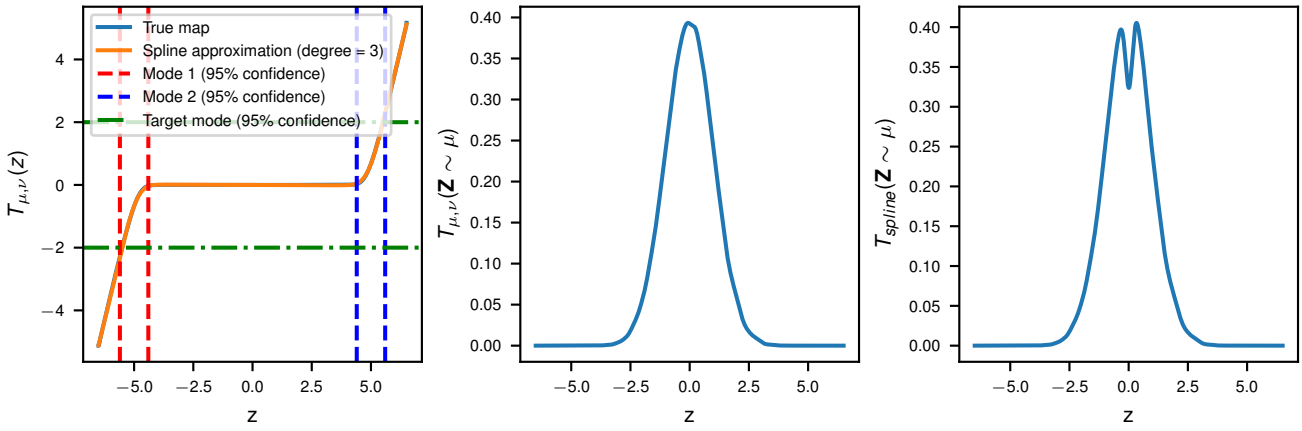

Figure 5. Flow $T_{\mu,\nu}$ in 1D - **(Left)** Map $T_{\mu,\nu}$ from the latent space (on the x-axis) to the data space (on the y-axis). Each pair of dotted lines highlight a mode in the latent space while the horinzontal line show the mode in the data space. **(Middle)** Kernel density estimation of the push forward of μ through the flow $T_{\mu,\nu}$ **(Right)** Kernel density estimation of the push forward of the base ($\mathcal{N}(0,1)$) through the flow a smooth cubic spline approximation of $T_{\mu,\nu}$

A. Perfect flows between multimodal distributions and unimodal distributions

In (Bogachev et al., 2005), the authors provide a recipe to build a triangular mapping [10] from one distribution to another. They do this by using the inverse CDF method for the first coordinate and then iterate it on the conditional distribution of the other coordinates. (Bogachev et al., 2005) shows that this bijection is the unique increasing one. We will illustrate this method by building a triangular map between a mixture of two Gaussians and a single Gaussian in 1D and 2D.

Unidimensional example Consider $\mu = \mathcal{N}(-a, \sigma^2)/2 + \mathcal{N}(a, \sigma^2)/2$ with $a > 0$ and $\sigma > 0$ and $\nu = \mathcal{N}(0, \tilde{\sigma}^2)$ with $\tilde{\sigma} > 0$. We can build a bijective mapping $T_{\mu,\nu}$ between the two distribution by taking $T = F_\nu^{-1} \circ F_\mu$ where F_ν and F_μ are the cumulative distribution functions of ν and μ respectively. In our case, we have

$$F_\mu(z) = \frac{1}{4}\left(2 + \operatorname{erf}\left(\frac{z+a}{\sigma\sqrt{2}}\right) + \operatorname{erf}\left(\frac{z-a}{\sigma\sqrt{2}}\right)\right) \qquad F_\nu(z) = \frac{1}{2}\left(1 + \operatorname{erf}\left(\frac{z}{\tilde{\sigma}\sqrt{2}}\right)\right)$$
$$F_\nu^{-1}(y) = \tilde{\sigma}\sqrt{2}\operatorname{erf}^{-1}(2y-1) \qquad T_{\mu,\nu}(z) = F_\nu^{-1}(F_\mu(z))$$

Figure 5 shows $T_{\mu,\nu}$: it takes the mass from the left and push it to the middle and does the symmetric task on the other side. Note that at the edge of the modes, the curve is very sharp. This sharpness increases as the modes are getting further apart (i.e., $a \to \infty$) which makes smooth approximation (like the spline approximation shown here) more and more difficult. Those small errors on the edges of the mode lead to multimodality in the push-forward.

Bidimensional example We take a similar setup in 2 dimensions. We consider $\mu = \mathcal{N}(-a\mathbf{1}_2, \sigma^2 I_2)/2 + \mathcal{N}(a\mathbf{1}_2, \sigma^2 I_2)/2$ with $a > 0$ and $\sigma > 0$ and $\nu = \mathcal{N}(0, \tilde{\sigma}^2 I_2)$ with $\tilde{\sigma} > 0$. Following (Bogachev et al., 2005) protocol, we define $T^{(1)}_{\mu,\nu}$ as our previous flow

$$T^{(1)}_{\mu,\nu}(z) = \tilde{\sigma}\sqrt{2}\operatorname{erf}^{-1}\left(\frac{1}{2}\left(2 + \operatorname{erf}\left(\frac{z+a}{\sigma\sqrt{2}}\right) + \operatorname{erf}\left(\frac{z-a}{\sigma\sqrt{2}}\right)\right) - 1\right),$$

because it is the canonical mapping between μ_1 and ν_1 which are the projections of μ and ν on the first axis. Let μ_x and

[10] A function $T : \mathbb{R}^n \to \mathbb{R}^n$ is triangular if $T_i(x)$ only depends on $x_1, \dots, x_i$ for $x \in \mathbb{R}^n$ and $1 \leq i \leq n$

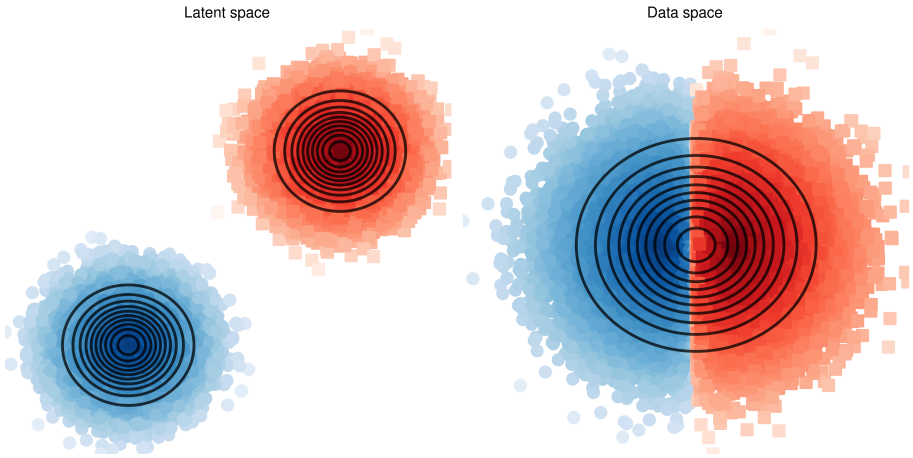

Figure 6. Flow $T_{\mu,\nu}$ in 2D - **(Left)** Samples from μ in the latent-space colored. The color of the samples is based on the closest mode. **(Right)** Samples from μ pushed through the flow $T_{\mu,\nu}$. The color of the samples correspond to their origin mode in the latent space.

ν_x be the projections of μ and ν on the second axis. We first compute $\rho_\mu^{x_1}$ which is the density of μ_{x_1},

$$\begin{aligned}\rho_\mu^{x_1}(x_2) &= \frac{\rho_{\mu,\nu}(x_1,x_2)}{\int_{\mathbb{R}}\rho_{\mu,\nu}(x_1,x_2)dx_2}\\ &= \frac{\mathcal{N}(x_1,-a,\sigma)\mathcal{N}(x_2,-a,\sigma)/2+\mathcal{N}(x_1,a,\sigma)\mathcal{N}(x_2,a,\sigma)/2}{\left(\mathcal{N}(x_1,-a,\sigma)+\mathcal{N}(x_1,-a,\sigma)\right)/2}\\ &= w_{x_1,a,\sigma}\mathcal{N}(x_2,-a,\sigma)+(1-w_{x_1,a,\sigma})\mathcal{N}(x_2,a,\sigma)\,,\end{aligned}$$

where $w_{x_1,a,\sigma} = \mathcal{N}(x_1,-a,\sigma)/(\mathcal{N}(x_1,-a,\sigma)+\mathcal{N}(x_1,-a,\sigma))$. So μ_{x_1} is a mixture between two Gaussians $\mathcal{N}(-a,\sigma)$ and $\mathcal{N}(a,\sigma)$ with weight $w_{x_1,a,\sigma}$. On the other hand, ν_{x_1} is simply $\mathcal{N}(0,\tilde{\sigma}^2)$ because it is isotropic. Following the same recipe as the unidimensional example, we can build $T^{(2)}_{\mu,\nu} = T_{\mu_{x_1},\nu_{x_1}}$ to map μ_{x_1} on ν_{x_1}. Finaly, we define the bidimensional map $T_{\mu,\nu}$ to be $T_{\mu,\nu}(x,y) = (T^{(1)}_{\mu,\nu}(x), T^{(2)}_{\mu,\nu}(y))$. Figure 6 show how the samples from μ are transported to ν. Again, we find a very sharp border between the two modes which would lead to multimodality if not well approximated.

Theoretical explanation In (Cornish et al., 2020, Theorem 2.1), the authors explain that if the support of the base and the target of a flow are different then in order to have a sequence T_{α_n} of homeomorphisms such $\lambda^\rho_{T_{\alpha_n}} \xrightarrow{\mathcal{D}} \pi$ (with the notations of Sec. 2.1) then

$$\lim_{n\to\infty}\mathrm{BiLip}(T_{\alpha_n}) = \infty\,,$$

where $\mathrm{BiLip}(f)$ is the bi-Lipschitz constant of f is defined as the infemum over $M\in[1,\infty]$ such that

$$M^{-1}\left\|z-z'\right\| \leq \left\|f(z)-f(z')\right\| \leq M\left\|z-z'\right\|\,.$$

Because classic deep normalizing flows use contraction mappings like ReLu as activation functions, their bi-Lipschitz function is necessarily bounded thus forbidding a perfect approximation of π. As mentionned by (Cornish et al., 2020), this is especially true for ResFlows (Chen et al., 2019) which are built upon Lipschitz transformations.

The intuition of this theorem can be seen in the previous example, as the sharp edges at the border of the modes would require unbounded derivatives.

Proof of Proposition 3.1 Because $d=1$, all real functions are triangular. Using (Bogachev et al., 2005, Lemma 2.1) we deduce the transport map $T_{\mu,\nu}$ that we built in the previous paragraph is actually the unique increasing flow. We now

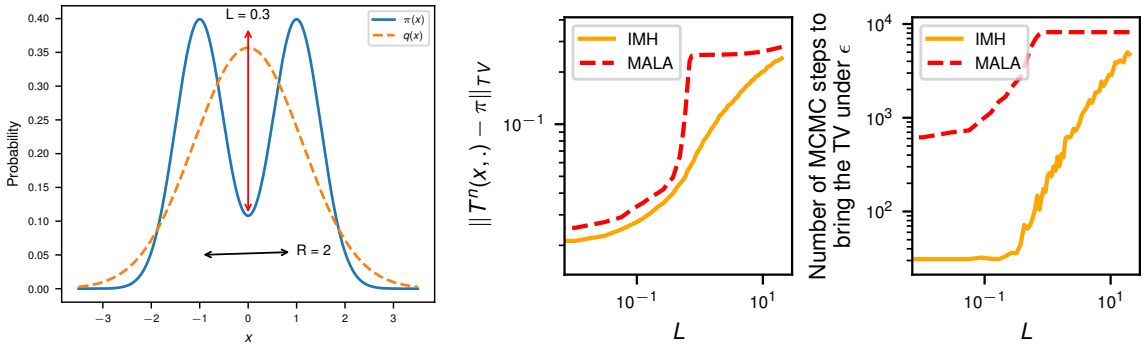

Figure 7. Toy experiment with a mixture of two uni-dimensional Gaussians - **(Left)** Target distribution π and proposal $Q = \mathcal{N}(0, \sqrt{\sigma^2 + a^2})$. The distance L is the depth of the gap between the two modes. **(Middle)** Total variation metrics of 256 MCMC chains sampling π with different values of L after 8192 steps. **(Right)** Number of steps needed to bring the total variation distance of the chain being built under $\epsilon = 8 \times 10^{-2}$ for different values of L. Note that curves eventually plateau due to the fact that we can't sample infinitely long MCMC chains.

compute its bi-Lipschitz constant. We have that

$$\frac{\mathrm{d}F_\mu}{\mathrm{d}z}(z) = \frac{1}{2\sigma\sqrt{2\pi}}\left(\exp\left(-\left(\frac{z+a}{\sigma\sqrt{2}}\right)^2\right) + \exp\left(-\left(\frac{z-a}{\sigma\sqrt{2}}\right)\right)^2\right), \tag{13}$$

$$\frac{\mathrm{d}F_\nu^{-1}}{\mathrm{d}y}(y) = \tilde{\sigma}\sqrt{2\pi}\exp([\mathrm{erf}^{-1}(2y-1)]^2), \tag{14}$$

$$\frac{\mathrm{d}T_{\mu,\nu}}{\mathrm{d}z}(z) = \frac{\mathrm{d}F_\nu^{-1}}{\mathrm{d}y}(F_\mu(z))\frac{\mathrm{d}F_\mu}{\mathrm{d}z}(z). \tag{15}$$

Using equations (13)-(15) with $z = 0$, we have that

$$\begin{aligned}\frac{\mathrm{d}T_{\mu,\nu}}{\mathrm{d}z}(0) &= \tilde{\sigma}\sqrt{2\pi}\exp\left(\left[\mathrm{erf}^{-1}\left(\frac{2}{4}\left(2 + \mathrm{erf}\left(\frac{a}{\sigma\sqrt{2}}\right) + \mathrm{erf}\left(\frac{a}{\sigma\sqrt{2}}\right)\right) - 1\right)\right]^2\right) \\ &\quad \times \frac{1}{2\sigma\sqrt{2\pi}}\left(\exp\left(-\left(\frac{a}{\sigma\sqrt{2}}\right)^2\right) + \exp\left(-\left(\frac{a}{\sigma\sqrt{2}}\right)\right)^2\right) \\ &= \frac{\tilde{\sigma}}{\sigma}\exp\left(\frac{-a^2}{2\sigma^2}\right).\end{aligned}$$

Using this result, we can derive a lower bound on $\mathrm{BiLip}(T_{\mu,\nu})$

$$\begin{aligned}\mathrm{BiLip}(T_{\mu,\nu}) \geq \mathrm{Lip}(T_{\mu,\nu}^{-1}) = 1/\mathrm{Lip}(T_{\mu,\nu}) &\geq \left(\frac{\mathrm{d}T_{\mu,\nu}}{\mathrm{d}z}(0)\right)^{-1} \\ &= \frac{\sigma}{\tilde{\sigma}}\exp\left(\frac{a^2}{2\sigma^2}\right).\end{aligned}$$

This shows that $\lim_{a\to\infty}\mathrm{BiLip}(T_{\mu,\nu}) = +\infty$. Proposition 3.1 can be recovered by taking $\tilde{\sigma} = 1$.

B. Local samplers can't cross energy barriers

We can illustrate that local samplers can't cross energy barriers by taking a simple uni-dimensional mixture of Gaussians as in Fig. 7. Figure 7 shows that, as the energy barrier increases, it gets more and more difficult for a local sampler to sample the mixture. Independent proposal methods are able to overcome this issue despite a poorly chosen proposal.

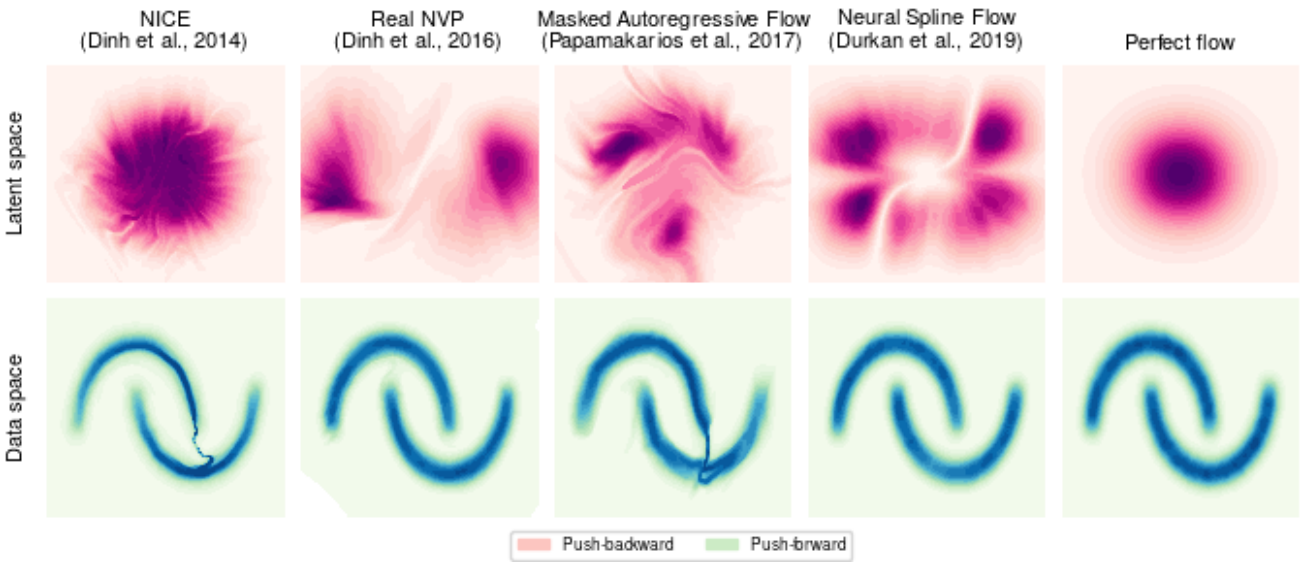

Figure 8. Push-backward and push-forward of popular deep normalizing flows targeting the two moons distribution

C. Flows typically do not erase potential barriers in the latent space

The toy experiment from App. B explained the difficulty of sampling multi-modal distributions directly in the data space, but could the flow kill this multi-modality in the latent space like it killed the bad conditioning before ? We trained popular deep normalizing flows on the two moons multimodal target and observed the push-forward and push-backward space. Figure 8 shows that the flows are not erasing energy barriers and can even worsen the conditioning of the modes in the latent space compared to the data space. However, they successfully put all the mass under the support of the base of the flow which should improve the quality flow proposal based methods. The intuition behind this difficulty is hinted in App. A.

D. Additional details on synthetic examples

D.1. Computation of sliced distribution metrics

In section 3, we used two different distances : the Total Variation (TV) distance and the Kolmogorov-Smirnov (KS) distance. They are computed between distributions μ and ν as follows

$$D_{\mathrm{TV}}(\mu,\nu) = \sup_{A\in\mathcal{F}} |\mu(A) - \nu(A)| \ , \tag{16}$$

$$D_{\mathrm{KS}}(\mu,\nu) = \sup_{x} |F_\mu(x) - F_\nu(x)| \ , \tag{17}$$

where F_μ and F_ν are the cumulative distribution functions of μ and ν respectively. If we consider sliced versions of those metrics in Sec. 3, it's because both of them cannot be computed in high dimension : (16) require the computation of an intractable integral and (17) require the multidimensional generalisation of the cumulative distribution function which is also intractable. Moreover, since μ will always be the distribution of the MCMC samples which is unknown it requires density estimation which is notoriously hard in high dimension.

To solve this problem, we project the samples in 1D before computing the metric. Let $(X_i)_{i=1}^N$ and $(Y_i)_{i=1}^M$ be samples from μ and ν respectively, for every random normal projection $P : \mathbb{R}^d \to \mathbb{R}$ (i.e., $P_i(x) = p_i x$ where $p_i \sim \mathcal{N}(0,1)$), we compute $\tilde{X}_i = P(X_i)$ and $\tilde{Y}_i = P(Y_i)$ and then

- *Sliced* Total Variation
  - we perform density estimation on the obtained samples leading to densities $\tilde{\mu}$ and $\tilde{\nu}$;

  - we compute $D_{\mathrm{TV}}(\tilde{\mu}, \tilde{\mu})$ by doing

$$D_{\mathrm{TV}}(\tilde{\mu}, \tilde{\mu}) = \frac{1}{2} \int |\tilde{\mu}(x) - \tilde{\nu}(x)| \, dx \, .$$

- *Sliced* Kolmogorov-Smirnov
  - we compute the empirical cumulative distributions of those projected samples $\hat{F}_{\mu_{proj}}(x) = 1/n \sum_{i=1}^{n} 1_{(-\infty,x]}(x_i)$ where x browse the union support of μ_{proj} and ν_{proj} (same goes for $\hat{F}_{\nu_{proj}}(x)$);
  - we compute the supremum of $\left| \hat{F}_{\mu_{proj}} - \hat{F}_{\nu_{proj}} \right|$.

To be accurate this procedure requires many random projections. In this work, we always use 128 random projections. Moreover, the density estimation task needed for the sliced total variation is performed using kernel density estimation (with a gaussian kernel) where the bandwitch is selected with Scott method for all sampling algorithms except IMH and IS which use the Sheather-Jones algorithm [11]. This density estimation task (just like the estimation of the cumulative distribution function) require to have $M \gg N$ if ν represent the true distribution and μ is an approximate distribution. In pratice, we use $M = 10N$.

Finally, we have chosen the use of the Kolmogorov-Smirnov distance for the experiments involving Neal's Funnel as we wanted to highlight the sampling behavior in the tails of the distribution.

To circumvent the need of slicing dimensions, a reviewer suggested to use the *Maximum Mean Discrepancy* (MMD). Given a feature map $\phi : \mathbb{R}^d \to \mathcal{F}$ and two distributions P and Q, the MMD is defined as

$$\mathrm{MMD}^2(P, Q) = \|\mu_P - \mu_Q\|_{\mathcal{F}}^2 \, ,$$

where $\mu_P = (\mathbb{E}_P[\phi(X)_1], \ldots, \mathbb{E}_P[\phi(X)_m])^T$. Using the kernel trick, one can find a kernel $k : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ that satisfies

$$< \mu_P, \mu_Q >_{\mathcal{F}} = \mathbb{E}_{P,Q}[k(X, Y)] \, .$$

In the following, we'll empirically compute the MMD between two datasets X and Y (as feature matrix) with the following formula

$$MMD^2(X, Y) = \frac{1}{m(m-1)} \sum_i \sum_{j \neq i} k(x_i, x_j) - 2 \frac{1}{m.m} \sum_i \sum_j k(x_i, y_j) + \frac{1}{m(m-1)} \sum_i \sum_{j \neq i} k(y_i, y_j)$$

with a Gaussian kernel $k(x, y) = \exp(-\|x - y\|^2 / (2\sigma^2))$. In practice, we choose the bandwidth σ as the mean of the pairwise distances between each sample of X and Y.

D.2. Experimental details on the interpolated Gaussians

Design of the target π The target π is a unimodal centered Gaussian $\mathcal{N}(0, \Sigma)$ with a badly conditioned covariance matrix Σ. We take $\Sigma = R_{\pi/4} \mathrm{diag}(\sigma_1^2, \ldots, \sigma_d^2) R_{\pi/4}^T$ where $R_{\pi/4}$ is the rotation of angle $\pi/4$ in the plan (x_1, x_d) (first and last axis) and σ_i are logarithmically evenly distributed between 10^{-1} and 10^1.

Design of T_t The imperfect flows are built with a piecewise linear interpolation T_t using multiple canonical transformations between different multivariate Gaussians. If $\sqrt{A}$ denotes the Cholesky factor of a matrix A (i.e., $\sqrt{A}\sqrt{A}^T = A$), then the flow indexed by t can be expressed as

$$T_t : z \mapsto \begin{cases} ((1-2t)\sqrt{\sigma_1^2 I_d} + 2t\sqrt{\Sigma})z & \text{if } t < 1/2 \\ \sqrt{\Sigma} z & \text{if } t = 1/2 \\ ((2t-1)\sqrt{\sigma_d^2 I_d} + 2(1-t)\sqrt{\Sigma})z & \text{if } t > 1/2 \end{cases} ,$$

[11] This is because when the flow is very far from the target, those samples tend to stagnate breaking Scott or Silverman rules.

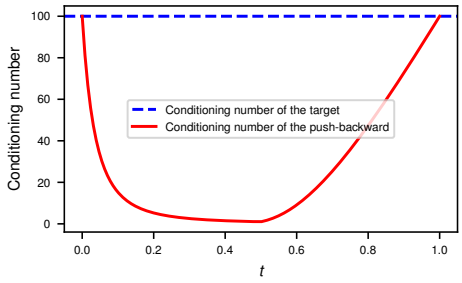

Figure 9. Conditioning of the push-backward of π through T_t

Table 3. Sampling hyper-parameters for the interpolated Gaussians experiment - n is the number of MCMC steps in the chain, n_{local} is the number of interleaved local steps in global/local samplers and N is the number of particles used in importance sampling.

Dimension	n	N (flow-MCMC)	N (neutra-flow-MCMC)	N (IS)	n_{local} (flow-MCMC)	n_{local} (neutra-flow-MCMC)
16	1100	80	80	22000	50	50
32	1200	80	80	24000	50	50
64	1300	80	80	26000	50	50
128	1400	80	80	28000	50	50
256	1500	80	80	30000	50	50

$$T_t^{-1} : x \mapsto \begin{cases} [(1-2t)\sqrt{\sigma_1^2 I_d} + 2t\sqrt{\Sigma}]^{-1} x & \text{if } t < 1/2 \\ \sqrt{\Sigma}^{-1} x & \text{if } t = 1/2 \\ [(2t-1)\sqrt{\sigma_d^2 I_d} + 2(1-t)\sqrt{\Sigma}]^{-1} x & \text{if } t > 1/2 \end{cases} .$$

The flow T_t is a linear map and its jacobians are the determinant of the scaling factors. This flow is built so that the conditioning number of the push-backward is canceled when the flow is perfect and can be as high as the one from the target if $t \neq 0.5$ (Fig. 9).

Sampling details Sampling hyperparameters where chosen by doing a grid-search for each algorithm and maximizing the median (over different quality of flows) of the sliced TV. The number of MCMC steps n was also chosen so that n is twice the number of steps needed to bring the $\hat{R}$ diagnostic of the faster algorithm (to converge) bellow 1.1. The number of particles used in importance sampling is a quarter of the total number of particles used by i-SIR. This division was chosen to avoid memory problems when using non-sequential importance sampling. All details are available in table 3. 256 chains were sampled in parallel to compute the metrics and were started by samples from the flow. 128 random projections are used to compute the sliced total variation (more details about this metric in App. D.1). The step size of the local steps was chosen to maintain the acceptance rate at 75%.

Maximum Mean Discrepancy distance As suggested by one of the reviewer, we computed the MMD distance - see end of App. D.1 - instead of the Sliced TV - see Fig. 1 (Left). Fig. 10 shows no apparent differences.

D.3. Experimental details on Neal's Funnel

Design of the target π Neal's funnel is a distribution which closely resembles the posterior of hierarchical Bayesian models. It depends on 2 positive parameters a, b and is defined as follows

$$X \sim \pi(a,b) \iff \begin{cases} X_1 & \sim \mathcal{N}(0,a) \\ X_i & \sim \mathcal{N}(0, \mathrm{e}^{bX_1}), \forall i \in \{2,\dots,d\} \end{cases} .$$

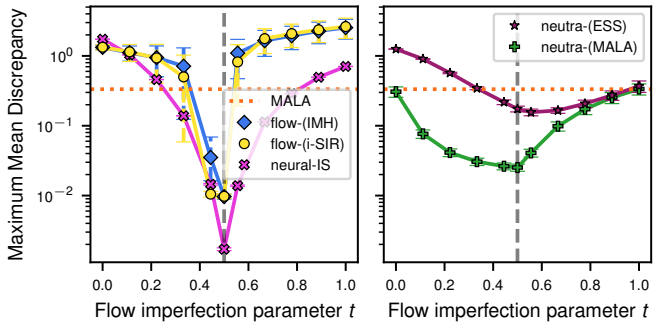

Figure 10. Replicate of Fig. 1 (Left) on the interpolated Gaussians with MMD distance.

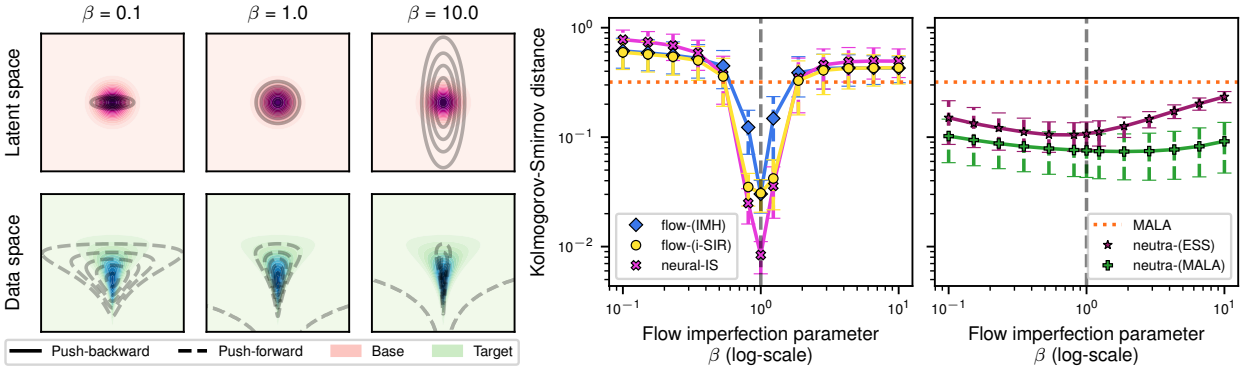

Figure 11. **(Left) Push-backwards $\lambda^{\rho}_{T_t}$ and push-forwards $\lambda^{\pi}_{T_t^{-1}}$ as a function of the flow imperfection parameter t.** Bottom row compares ill-conditioned Gaussian target (blue levels) and flow push-forward (black lines) projected on smallest and largest variance axes. Top row compares base distribution $\mathcal{N}(0, I_d)$ (pink level) with target push-backward (black lines). **(Right) Kolmogorov-Smirnov (KS) distances between target and empirical samples depending on the quality of the flow** β using 512 chains of length 600 initialized with the flow T_β. neural-IS was evaluated with 9750 samples.

Design of the flow T_β We define an analytical parametric flow $T_{a,b,\alpha}$ as follows

$$T_{a,b,\alpha} : z \mapsto \begin{pmatrix} \sqrt{\frac{a}{\alpha}} z_1 \\ \vdots \\ \frac{1}{\sqrt{\alpha}} \exp(\frac{b\sqrt{\frac{a}{\alpha}} z_1}{2}) z_i \\ \vdots \end{pmatrix}, \quad T^{-1}_{a,b,\alpha} : x \mapsto \begin{pmatrix} \sqrt{\frac{\alpha}{a}} x_1 \\ \vdots \\ \sqrt{\alpha} \exp(-\frac{b}{2} x_1) x_i \\ \vdots \end{pmatrix},$$

$$\log \left|\det(J_{T_{a,b,\alpha}}(z))\right| = \log(\sqrt{a}) - \frac{d}{2}\log(\alpha) + \frac{d-1}{2} b \sqrt{\frac{a}{\alpha}} z_1 \,,$$
$$\log \left|\det(J_{T^{-1}_{a,b,\alpha}}(x))\right| = -\log(a) + \frac{d}{2}\log(\alpha) - \frac{d-1}{2} b x_1 \,.$$

$T_{a,b,\alpha=1}$ is the natural flow which transports $N(0, I_d)$ to $\pi(a, b)$. The parameter α was introduce for two purposes : control the variance of the push-backward and calibrate the mode of the push-forward. Indeed, it's easy to see that

$$T_{a,b,\alpha}^{-1}(T_{a,b,1}(z)) = T_{a,b,\alpha}^{-1}\left(\begin{pmatrix}\sqrt{a}z_1\\ \vdots \\ \exp(\frac{b\sqrt{a}z_1}{2})z_i \\ \vdots\end{pmatrix}\right) = \begin{pmatrix}\sqrt{\frac{\alpha}{a}}(\sqrt{a}z_1)\\ \vdots \\ \sqrt{\alpha}\exp\left(-\frac{b}{2}\sqrt{a}z_1\right)\left(\exp(\frac{b\sqrt{a}z_1}{2})z_i\right) \\ \vdots\end{pmatrix} = \sqrt{\alpha}z\,.$$

So if $X \sim \pi(a,b)$, then $T_{a,b,\alpha}^{-1}(X) \sim \mathcal{N}(0,\alpha I_d)$. Moreover, we can compute the mode of $T_{a,b,\alpha}(\mathcal{N}(0,I_d))$ using the change of variable formula,

$$\begin{aligned}
\log \lambda_{T_{a,b,\alpha}}^{\mathcal{N}(0,I_d)}(x) &= \log(\mathcal{N}(T_{a,b,\alpha}^{-1}(x);0,I_d)) + \log\det\left|J_{T_{a,b,\alpha}^{-1}}(x)\right| \\
&= -\frac{\left\|T_{a,b,\alpha}^{-1}(x)\right\|^2}{2} - \frac{d}{2}\log(2\pi) - \log(a) + \frac{d}{2}\log(\alpha) - \frac{d-1}{2}bx_1 \\
&= -\frac{1}{2}\left(\left(\sqrt{\frac{\alpha}{a}}x_1\right)^2 + \sum_{i=2}^{d}\left(\sqrt{\alpha}\exp\left(-\frac{b}{2}x_1\right)x_i\right)^2\right) - \frac{d-1}{2}x_1 + K \\
&= -\frac{1}{2}\left(\frac{\alpha}{a}x_1^2 + \alpha\exp(-bx_1)\sum_{i=2}^{d}x_i^2\right) - \frac{d-1}{2}x_1 + K\,,
\end{aligned}$$

where K is a constant. Canceling the gradient of $\log \lambda_{T_{a,b,\alpha}}^{\mathcal{N}(0,I_d)}$ leads to

$$\begin{aligned}
\begin{cases}\frac{\partial \log \lambda_{T_{a,b,\alpha}}^{\mathcal{N}(0,I_d)}(x)}{\partial x_1}(x) = 0 \\ \frac{\partial \log \lambda_{T_{a,b,\alpha}}^{\mathcal{N}(0,I_d)}(x)}{\partial x_j}(x) = 0 \quad ,(j\neq 1)\end{cases} &\iff \begin{cases}-\frac{1}{2}\left(\frac{2\alpha}{a}x_1 - b\alpha\left(\sum_{i=2}^{d}x_i^2\right)\exp(-bx_1)\right) - \frac{d-1}{2}b = 0 \\ -\alpha\exp(-bx_1)x_j = 0 \qquad\qquad (j\neq 0)\end{cases} \\
&\iff \begin{cases}-\frac{\alpha}{a}x_1 = \frac{d-1}{2}b \\ x_j = 0 \qquad\qquad (j\neq 0)\end{cases} \\
&\iff z = \begin{pmatrix}-\frac{ab}{2\alpha}(d-1)\\ 0 \\ \vdots \\ 0\end{pmatrix}.
\end{aligned}$$

Let $\beta > 0$, $a^\star = 3$, $b^\star = 1$ then if $a = \beta a^\star$, $b = b^\star$ and $\alpha = \beta$ the mode of the push-forward of $\mathcal{N}(0,I_d)$ through $T_{a,b,\alpha}$ and the mode of $\pi(a^\star,b^\star)$ will always coincide for any $\beta > 0$. We use this principle to create the flow $T_\beta = T_{\beta a^\star,b^\star,\beta}$ which is a perfect mapping from $\mathcal{N}(0,I_d)$ to $\pi(a^\star,b^\star)$ if $\beta = 1$.

Sampling details Sampling procedure is the same as in Sec. D.2 with the Kolmogorov-Smirnov distance as target metric, only some hyper-parameters change (see table 4).

D.4. High acceptance rates of pure flow-MCMC algorithms

Figure 12 shows how global flow-MCMC samplers - IMH and i-SIR - behave depending on the quality of the flow. The middle and right columns show that the acceptance of IMH/i-SIR and the participation ratio of IS are crashing as soon as π and ρ aren't aligned perfectly. Note the the crashing rate of i-SIR is slower than the one of IMH. Fig. 12 also highlights that flow-MCMC algorithms as well as neutra-MCMC with ESS produce samples with very different likelihoods compared to neutra-MCMC methods which are known to perform better (see Fig. 1 and Fig. 11 of the main paper).

Table 4. Sampling hyper-parameters for Neal's Funnel experiment.

Dimension	n	N (flow-MCMC)	N (neutra-flow-MCMC)	N (IS)	n_{local} (flow-MCMC)	n_{local} (neutra-flow-MCMC)
16	500	60	60	7500	10	40
32	550	60	100	8250	10	40
64	600	60	100	9000	10	40
128	650	60	100	9750	10	40
256	700	60	120	10500	10	40

Table 5. RealNVP training hyperparameters for the Gaussian mixture experiment

Dimension	# Iteration	Patience of learning rate scheduler	Learning rate	Size of hidden layers	# RealNVP blocks
16	2500	100	10^{-2}	64	4
32	2640	106	7.36×10^{-3}	76	4
64	2900	120	3.98×10^{-3}	102	4
128	3440	146	1.17×10^{-3}	153	5
256	11250	200	10^{-4}	256	8

D.5. Experimental details for the high dimensional Gaussian mixture

Design of π π is a mixture of 4 isotropic Gaussians i.e., $\pi = 1/4 \sum_{i=1}^{4} \mathcal{N}(\mu_i, I_d)$ where the μ_i are defined as $\mu_1 = a \times (1, 1, 1, \ldots, 1, 1, 1)$, $\mu_2 = a \times (-1, -1, -1, \ldots, 1, 1, 1)$, $\mu_2 = -\mu_2$ and $\mu_4 = -\mu_1$ and $a = 0.5919$. This specific value of a guarantees that if $X \sim \mathcal{N}(\mu_i, I_d)$ then $\forall j \neq i, \mathbb{P}(\|X - \mu_j\| < \|X - \mu_i\|) \leq 10^{-10}$ for any dimension d.

Training the flow The flows used here are RealNVPs. The base of the flow is $\rho = \mathcal{N}(0, \sigma^2 I_d)$ where σ^2 is the maximum variance of π along each dimension [12]. The flow was trained using Adam (Kingma & Ba, 2014) optimizer at a progressive learning rate. All the coupling layers had 3 hidden layers initialized with very small weights ($\simeq 10^{-6}$). The batch size was 8192 and the decay rate was 0.99. The other hyper-parameters can be found in table 5. An indicator of the flows qualities can be found in Fig. 13.

More neutra-MCMC algorithms for the 2D mixture Figure 14 extends Fig. 2 with more neutra-MCMC methods. It shows that changing the reparametrized sampler could allow switching modes in the latent space. For instance, ESS is able to do so. Our conclusion is that using local samplers in this pathological latent space make mixing between modes much longer and less accurate compared to using flow-MCMC methods.

The case of mode collapse As recalled in the introduction, normalizing flows trained with the backward KL objective notoriously suffer from mode collapse. Based on this observation, we extended the conclusions of Fig. 2 Left using a flow which suffered from mode collapse. Fig. 15 shows that flow-MCMC methods sometimes reach the uncovered mode while neutra-MCMC never do. One can also think of the following thought experiment: consider a target as a two-component mixture of Gaussian and a Gaussian base distribution. In a mode-collapsed situation, we can imagine that an affine flow sends the base distribution perfectly to one of the modes. All types of NF-enhanced samplers will fail. On the one hand, flow-MCMC will never manage to propose in the other mode. On the other hand, the push-backward of the target sampled by neutra-MCMC will be an affine transformation of the original Gaussian mixture, which a local sampler cannot properly sample.

flow-MCMC methods exploit modes If figure 2 shows the exploration capability that flow-MCMC methods have and neutra-MCMC methods lack, it doesn't show how much the mass within the modes is exploited. By cutting the Markov chains depending on the closest mode and computing the mean and covariance of each part, we can compute the forward KL between each mode and the Gaussian approximated in each part. Averaging those metrics leads to Fig. 13 which shows

[12] Using the total variation formula $\forall i \in \{1, \ldots, d\}, \sigma^2 = \sigma_i^2 = 1 + \sum_{j=1}^{4} \frac{1}{4} (\mu_j)_i^2$

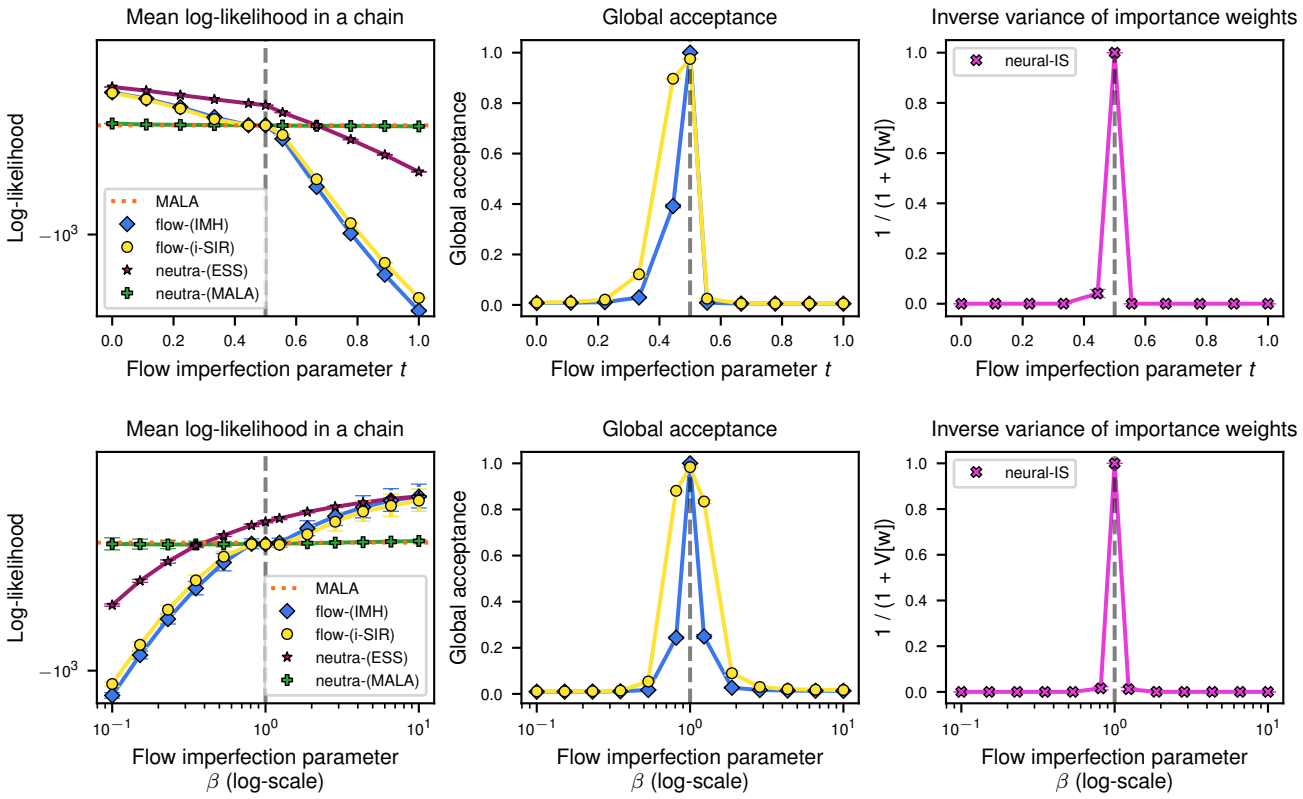

Figure 12. (Left) Likelihoods levels *(Middle)* / Global acceptance of pure flow-MCMC methods / *(Right)* Participation ratio of importance sampling, depending on the quality of the flow - **(Top)** Interpolated gaussians flow **(Bottom)** Neal's Funnel flow

that in moderate dimensions the flow is good enough so purely global methods can exploit the modes. Here, we can see the dimension dependence of all methods.

Sampling details Sampling procedure is the same as in Sec. D.2 and targets the mean score between the average KL and the mode mixing metric. Note that unlike Sec. D.2, 1024 chains were used. Only some hyper-parameters change and can be found in table 6.

D.6. Experimental details on the banana distribution

Design of π The banana distribution depends on two positive parameters a and b and is expressed as follows

$$X \sim \pi(a,b) \iff \forall i \in \{0,\dots,d/2\}, \begin{cases} X_{2i} & = aZ_{2i} \\ X_{2i+1} & = Z_{2i+1} + ba^2 Z_{2i}^2 - a^2 b \end{cases},$$

where $Z \sim \mathcal{N}(0, I_d)$ and d is even. This leads to a natural bijection $T_{a,b}$ between $\mathcal{N}(0, I_d)$ and $\pi(a,b)$

$$T_{a,b} : z \mapsto \begin{pmatrix} \vdots \\ az_{2i} \\ z_{2i+1} + ba^2 z_{2i}^2 - a^2 b \\ \vdots \end{pmatrix}, \quad T_{a,b}^{-1} : x \mapsto \begin{pmatrix} \vdots \\ x_{2i}/a \\ x_{2i+1} - bx_{2i}^2 + a^2 b \\ \vdots \end{pmatrix},$$

$$\log|\det(J_{T_{a,b}}(z))| = \frac{d}{2}\log(a), \quad \log|\det(J_{T_{a,b}^{-1}}(x))| = -\frac{d}{2}\log(a).$$

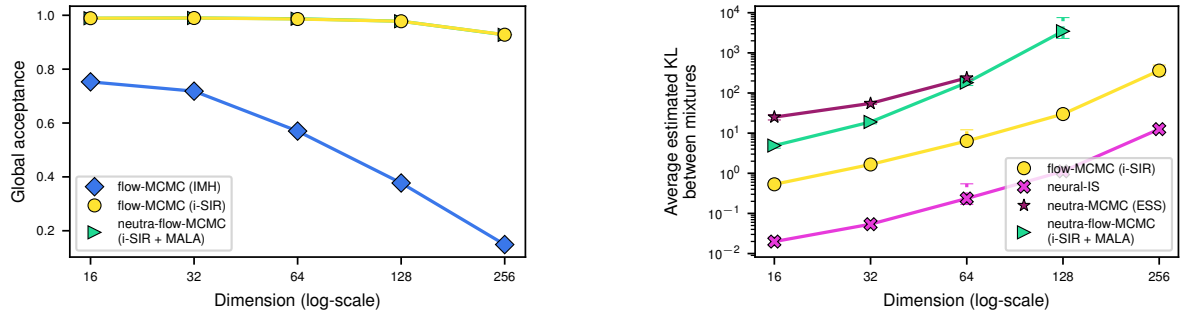

Figure 13. **(Left)** Global acceptance of the RealNVP flow obtained after training on the mixture of Gaussians in increasing dimension **(Right)** Averaged forward Kullback-Leiber for the Gaussian mixture.

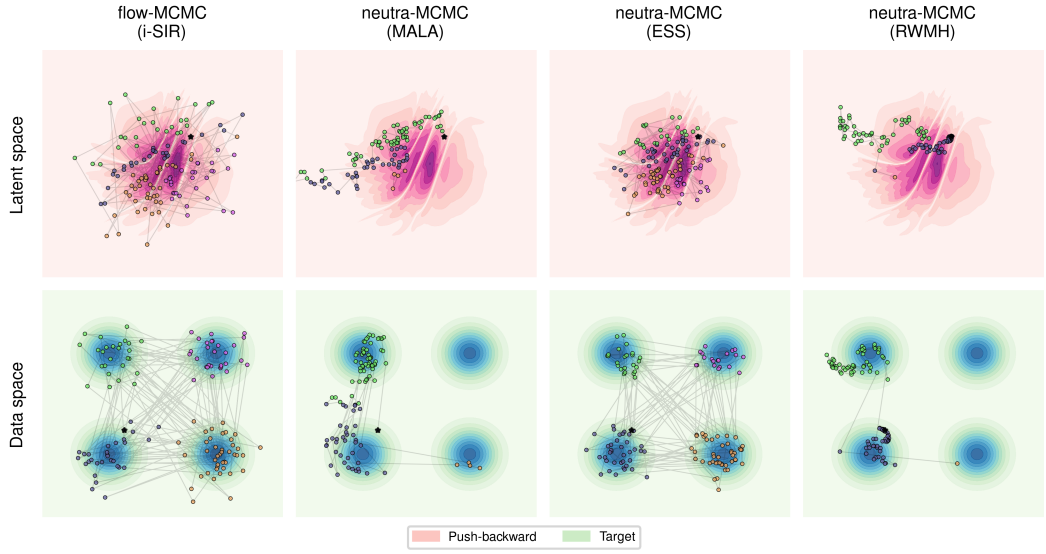

Figure 14. Extended version of Fig. 2

In the following, we take $a = 10$ and $b = 0.02$ (see Fig. 16).

Flows, training and sampling The normalizing flows used are RealNVPs (Dinh et al., 2016) with 3 layers deep neural networks initialized with small weights ($\simeq 10^{-6}$). The specific architecture of the neural networks and the training hyperparameters depending on the dimension of the problem can be found in table 7 (the learning rate scaling is 0.98). The sampling algorithms with i-SIR using $N = 60$ particles, $n_{local} = 10$ and the target acceptance of local steps is 75%. 128 random projections where used to compute the sliced total variation. There are 128 MCMC chains of length 1024 sampled in parallel.

E. Mixing time for IMH for log concave distribution

E.1. General Theorem

Here we consider Independent Metropolis-Hastings (IMH) with a target π on $\mathbb{R}^d$ and a proposal Q with density denoted by q with respect to the Lebesgue measure. We denote by P the resulting Markov kernel given for any $x \in \mathbb{R}^d$ and Borel set A by

$$P(x, \mathsf{A}) = \int_{\mathsf{A}} Q(\mathrm{d}y)\mathrm{acc}(x, y) + \left(1 - \int Q(\mathrm{d}y)\mathrm{acc}(x, y)\right) \delta_x(\mathsf{A})\,. \tag{18}$$

We define the mixing time of P with respect to an initial distribution μ and a precision target $\epsilon > 0$ as

$$\tau_{\mathrm{mix}}(\mu, \epsilon) = \inf\{n \in \mathbb{N} \ : \ \|\mu P^n - \pi\|_{\mathrm{TV}} \leq \epsilon\}\,, \tag{19}$$

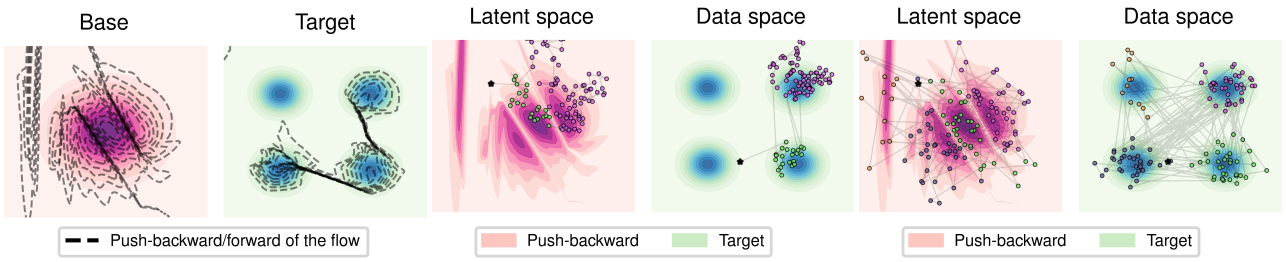

Figure 15. **An illustration of sampling with a mode-collapsed flow. (Left)** RealNVP trained with backward KL torwards a 2d target mixture of 4 Gaussians. **(Middle) neutra-MCMC** and **(Right) flow-MCMC** Samplers using the flow on the left. The 128-step MCMC chain is colored according to the closest mode in the data space (bottom row) with corresponding location in the latent space (top row). MALA's step-size was chosen to reach 75% of acceptance.

Table 6. Sampling hyper-parameters for the Gaussian mixture experiment.

Dimension	n	N (flow-MCMC)	N (neutra-flow-MCMC)	N (IS)	n_{local} (flow-MCMC)	n_{local} (neutra-flow-MCMC)
16	700	120	120	21000	5	5
32	900	140	140	31500	5	5
64	1100	160	160	44000	5	5
128	1300	180	180	58500	5	5
256	1500	200	200	75000	5	5

where the total variation distance between two distributions P_1 and P_2 is defined as

$$\|P_1 - P_2\|_{\mathrm{TV}} = \sup_{A \in \mathcal{B}(\mathbb{R}^d)} |P_1(A) - P_2(1)| \,. \tag{20}$$

The quantity $\tau_{\mathrm{mix}}(\mu, \epsilon)$ provides the minimum number of MCMC steps needed to bring the total variation distance between the chain and the target π below a precision ϵ when the chain has initial distribution μ. Now we introduce the s-conductance which will be later use to bound the mixing time. Let $s \in (0, 1/2]$, we define the s-conductance as

$$\phi_s = \inf_{\mathsf{S}, \pi(\mathsf{S}) \in (s, 1/2]} \frac{\int_{\mathsf{S}} P(x, \mathsf{S}^c) \pi(\mathrm{d}x)}{\pi(\mathsf{S}) - s} \,, \tag{21}$$

where S^c denotes the complement of S. The s-conductance quantifies the probability of transitioning from a set S charged with a probability s to its complementary. This quantity is of interest since it can be used for bounding the mixing time very useful because of Theorem E.1 and its corollary taken from (Lovász & Simonovits, 1993).

$$\tau_{mix}(\mu, \epsilon) = \inf\{n \in \mathbb{N} \,:\, \|P^n(\mu, .) - \pi\|_{\mathrm{TV}} \leq \epsilon\} \,, \text{ where } \epsilon > 0 \,. \tag{22}$$

$\tau_{mix}(\mu, \epsilon)$ provides the minimum number of MCMC steps needed to bring the total variation distance between the chain and the target π bellow an accuracy ϵ when the chain started with distribution μ.

Theorem E.1 (Corollary 1.5 (Lovász & Simonovits, 1993))**.** *A Markov chain with π as unique invariant distribution and μ a β-warm start with respect to π (i.e., for any Borel set E, $\mu(\mathsf{E}) \leq \beta\pi(\mathsf{E})$) verifies that for all $n \in \mathbb{N}$,*

$$\|P^n(\mu, .) - \pi\|_{\mathrm{TV}} \leq \beta s + \beta \left(1 - \frac{\phi_s^2}{2}\right)^n \leq \beta s + \beta \exp\left(-\frac{n}{2}\phi_s^2\right) \,. \tag{23}$$

In particular, for $0 < \epsilon < 1$, then

$$\tau_{\mathrm{mix}}(\mu, \epsilon) \leq \frac{2}{\phi_s^2} \log\left(\frac{2\beta}{\epsilon}\right) \,. \tag{24}$$

Here we follow now a common strategy to derive quantitative mixing times for Metropolis-Hastings algorithms applied to log-concave target by applying Theorem E.1 (Dwivedi et al., 2018; Mou et al., 2019; Chen et al., 2020; Chewi et al., 2021; Wu et al., 2022; Narayanan & Srivastava, 2022).

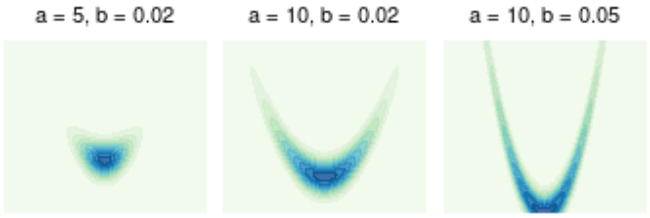

Figure 16. Banana distribution with different parameters

Table 7. RealNVP hyper-parameters for the Banana experiment.

Dimension	# Iteration	Patience of learning rate scheduler	Learning rate	Size of hidden layers	# RealNVP blocks
16	1000	75	10^{-2}	64	3
32	1193	82	8×10^{-3}	74	3
64	1580	96	5×10^{-3}	96	3
128	2354	125	2×10^{-3}	139	5
256	3903	183	3×10^{-4}	226	7
512	7000	300	1×10^{-5}	400	12

We denote by w the weight function $w = \pi/q$ and $\tilde{w} = \log w$ the log-weight function. We establish first conditions on the proposal Q and the log-weight function which allows us to get lower bounds on the conductance for P and therefore its mixing time. Then, we specialize our result to the case $Q = \mathcal{N}(0, \sigma^2 \mathrm{I}_d)$ and π positive and strongly log-concave, see. Assumption E.7. Without loss of generality, we assume that the unique mode of π is 0.

Assumption E.2.

- 0 is a mode for π, i.e., $\arg\max_{x \in \mathbb{R}^d} \pi(x) = 0$.
- π satisfies an isoperimetric inequality with isoperimetric constant $\psi(\pi)$ i.e., for any partition $\{\mathsf{S}_1, \mathsf{S}_2, \mathsf{S}_3\}$ of $\mathbb{R}^d$ [13], we have
$$\pi(\mathsf{S}_3) \geq \psi(\pi) \mathrm{dist}(\mathsf{S}_1, \mathsf{S}_2) \pi(\mathsf{S}_1) \pi(\mathsf{S}_2) \,, \tag{25}$$
where $\mathrm{dist}(S_1, S_2) = \inf\{\|x - y\| \ : \ x \in \mathsf{S}_1, y \in \mathsf{S}_2\}$
- The log-weight function is locally Lipschitz, i.e., for all $R \geq 0$ there exist C_R such that for all $(x, y) \in \mathrm{B}(0, R)^2$,
$$|\tilde{w}(x) - \tilde{w}(y)| \leq C_R \|x - y\| \,. \tag{26}$$

Note that if π is log-concave, then (25) holds; see (Bobkov, 1999).

Our assumption on Q is the following. Let $\alpha \in (1/2, 1)$ and $s \in (0, 1/2]$.

Assumption E.3 (α, s)**.** There exist $\Delta \in (0, 1)$, $\delta_\Delta > 0$ and $R \geq 0$ satisfying
$$\int_{\mathsf{B}_\Delta} \pi(\mathrm{d}x) Q(\mathrm{d}y) \geq 1 - \delta_\Delta \,, \tag{27}$$
where
$$\mathsf{B}_\Delta = \left\{(x, y) \in \mathrm{B}(0, R)^2 \ : \ \|x - y\| \leq R_\Delta\right\} \,, \text{ where } R_\Delta = \frac{-\log(\Delta)}{C_R} \,, \tag{28}$$
and
$$\frac{1}{1 - \alpha} \delta_\Delta \leq \min\left(\frac{s}{4}, \frac{2\alpha - 1}{64 C_R} \psi(\pi) s\right) \,. \tag{29}$$

Theorem E.4 (Conductance lower bound for IMH)**.** *Let* $1/2 < \alpha < 1$ *and* $0 < s < 1/2$*. Assume assumption E.2 and E.3*(α, s) *hold. Then, it holds*
$$\phi_s \geq \min\left(\frac{1 - \alpha}{2}, \frac{(1 - \alpha)(2\alpha - 1)}{128 C_R} \psi(\pi)\right) \,. \tag{30}$$

[13] i.e., $\mathsf{S}_1 \cup \mathsf{S}_2 \cup \mathsf{S}_3 = \mathbb{R}^d$ but for any $i \neq j, \mathsf{S}_i \cap \mathsf{S}_j = \emptyset$

Proof. Define the set $\mathsf{E}_{\Delta,\alpha}$ by

$$\mathsf{E}_{\Delta,\alpha} = \left\{ x \in \mathrm{B}(0,R) \ : \ \int 1_{(x,y)\in B_\Delta} Q(\mathrm{d}y) \geq \alpha \right\} .$$

Using (27), we have by definition that

$$\pi(\mathsf{E}_{\Delta,\alpha}) \geq 1 - \frac{1}{1-\alpha}\delta_\Delta \tag{31}$$

Indeed, it would not hold, using the law of total probability, we would have $1-\delta_\Delta < [1-\delta_\Delta/(1-\alpha)]1 + [\delta_\Delta/(1-\alpha)]\alpha = 1-\delta_\Delta$. Let S be a measurable subset of $\mathbb{R}^d$ with $s \leq \pi(\mathsf{S}) \leq 1/2$ and define

$$\begin{aligned}
\mathsf{S}_1 &= \{x \in \mathsf{S} \ : \ P(x, \mathsf{S}^c) \leq 1-\alpha\} , \\
\mathsf{S}_2 &= \{x \in \mathsf{S}^c \ : \ P(x, \mathsf{S}) \leq 1-\alpha\} , \\
\mathsf{S}_3 &= (\mathsf{S}_1 \cup \mathsf{S}_2)^c .
\end{aligned}$$

If $\pi(\mathsf{S}_1) < \pi(\mathsf{S})/2$ or $\pi(\mathsf{S}_2) < \pi(\mathsf{S}^c)/2$, then we may conclude from the reversibility of P that

$$\int_{\mathsf{S}} P(x,\mathsf{S}^c)\pi(\mathrm{d}x) = \frac{1}{2}\int_S P(x,\mathsf{S}^c)\pi(\mathrm{d}x) + \frac{1}{2}\int_{\mathsf{S}^c} P(x,S)\pi(\mathrm{d}x) > \frac{1}{2}\frac{\pi(\mathsf{S})}{2}(1-\alpha) = \frac{1-\alpha}{4}\pi(\mathsf{S}) . \tag{32}$$

In the following, we assume that $\pi(\mathsf{S}_1) \geq \pi(\mathsf{S})/2$ or $\pi(\mathsf{S}_2) \geq \pi(\mathsf{S}^c)/2$. Then it follows from the definition of the total variation distance that if $x \in \mathsf{E}_{\Delta,\alpha} \cap \mathsf{S}_1$ and $y \in \mathsf{E}_{\Delta,\alpha} \cap \mathsf{S}_2$ such that

$$\|P(x,.) - P(y,.)\|_{\mathrm{TV}} \geq 2\alpha - 1 .$$

Moreover using the fact that $x \in \mathsf{E}_{\Delta,\alpha} \cap S_1$ and $y \in \mathsf{E}_{\Delta,\alpha} \cap S_2$ and the expression of the Markov kernel, we have that

$$\begin{aligned}
\|P(x,.) - P(y,.)\|_{\mathrm{TV}} &= \int |\mathrm{acc}(x,z) - \mathrm{acc}(y,z)| \, Q(\mathrm{d}z) \\
&= \int \left| \min\left(1, \frac{\pi(z)q(x)}{\pi(x)q(z)}\right) - \min\left(1, \frac{\pi(z)q(y)}{\pi(y)q(z)}\right) \right| Q(\mathrm{d}z) \\
&= \int \left| \min\left(1, \frac{w(z)}{w(x)}\right) - \min\left(1, \frac{w(z)}{w(y)}\right) \right| Q(\mathrm{d}z) \\
&\leq \int |\tilde{w}(x) - \tilde{w}(y)| \, Q(\mathrm{d}z) \leq |\tilde{w}(x) - \tilde{w}(y)| \leq C_R \|x-y\| ,
\end{aligned}$$

where we used the Lipschitz behavior of $t \mapsto \min(1, \mathrm{e}^t)$ and property (26) in the last lines. This leads to $2\alpha - 1 \leq C_R \|x-y\|$ and more generally to

$$\mathrm{dist}(\mathsf{E}_{\Delta,\alpha} \cap \mathsf{S}_1, \mathsf{E}_{\Delta,\alpha} \cap \mathsf{S}_2) \geq \frac{2\alpha - 1}{C_R} . \tag{33}$$

Using the isoperimetric inequality (25), we have that

$$\begin{aligned}
\pi(\mathsf{E}_{\Delta,\alpha}^c \cup \mathsf{S}_3) &= \pi((\mathsf{E}_{\Delta,\alpha} \cap \mathsf{S}_1)^c \cap (\mathsf{E}_{\Delta,\alpha} \cap \mathsf{S}_2)^c) \\
&\geq \frac{2\alpha-1}{C_R}\psi(\pi)\pi(\mathsf{E}_{\Delta,\alpha} \cap \mathsf{S}_1)\pi(\mathsf{E}_{\Delta,\alpha} \cap \mathsf{S}_2) .
\end{aligned}$$

Using (29)-(31), we have with the union bound, $s \in (0, 1/2]$ and the condition $\pi(\mathsf{S}_1) \geq \pi(\mathsf{S})/2$ or $\pi(\mathsf{S}_2) \geq \pi(\mathsf{S}^c)/2 \geq$

$1/4$,

$$\begin{aligned}
\pi(\mathsf{S}_3) + \frac{1}{1-\alpha}\delta_\Delta &\geq \pi(\mathsf{S}_3) + \pi(\mathsf{E}^c_{\Delta,\alpha}) \\
&\geq \frac{2\alpha-1}{C_R}\psi(\pi)\pi(\mathsf{E}_{\Delta,\alpha}\cap\mathsf{S}_1)\pi(\mathsf{E}_{\Delta,\alpha}\cap\mathsf{S}_2) \\
&\geq \frac{2\alpha-1}{C_R}\psi(\pi)(\pi(\mathsf{S}_1)-\pi(\mathsf{E}^c_{\Delta,\alpha}))(\pi(\mathsf{S}_2)-\pi(\mathsf{E}^c_{\Delta,\alpha})) \\
&\geq \frac{2\alpha-1}{C_R}\psi(\pi)\left(\pi(\mathsf{S}_1)-\frac{1}{1-\alpha}\delta_\Delta\right)\left(\pi(\mathsf{S}_2)-\frac{1}{1-\alpha}\delta_\Delta\right) \\
&\geq \frac{2\alpha-1}{C_R}\psi(\pi)\left(\frac{\pi(\mathsf{S})}{2}-\frac{1}{1-\alpha}\delta_\Delta\right)\left(\frac{1}{4}-\frac{1}{1-\alpha}\delta_\Delta\right) \\
&\geq \frac{2\alpha-1}{C_R}\psi(\pi)\frac{\pi(\mathsf{S})}{4}(1-\alpha)\,.
\end{aligned}$$

Using (29) again, we get

$$\pi(\mathsf{S}_3) \geq \frac{2\alpha-1}{64C_R}\psi(\pi)\pi(\mathsf{S})\,.$$

Consequently, using that P is reversible, we get

$$\begin{aligned}
\int_{\mathsf{S}} P(x,\mathsf{S}^c)\pi(\mathrm{d}x) &\geq \frac{1}{2}\left(\int_{\mathsf{S}\cap\mathsf{S}_3} P(x,\mathsf{S}^c)\pi(\mathrm{d}x) + \int_{\mathsf{S}^c\cap\mathsf{S}_3} P(x,\mathsf{S})\pi(\mathrm{d}x)\right) \\
&\geq \frac{1}{2}\left(\int_{\mathsf{S}\cap\mathsf{S}_3}(1-\alpha)\pi(\mathrm{d}x) + \int_{\mathsf{S}^c\cap\mathsf{S}_3}(1-\alpha)\pi(\mathrm{d}x)\right) \\
&\geq \frac{1-\alpha}{2}\pi(\mathsf{S}_3) \\
&\geq \frac{(1-\alpha)(2\alpha-1)}{128C_R}\psi(\pi)\pi(\mathsf{S})\,,
\end{aligned}$$

along with (32) and the definition of the s-conductance, we get

$$\phi_s \geq \min\left(\frac{(1-\alpha)}{2}, \frac{(1-\alpha)(2\alpha-1)}{128C_R}\psi(\pi)\right)\,.$$

□

Corollary E.5 (Mixing time upper bound for IMH)**.** *Let $1/2 < \alpha < 1$, $0 < \epsilon < 1$ and μ a β-warm initial distribution with respect to π. Assume assumption E.2 and E.3 hold, then we have the following upper bound on the mixing time*

$$\tau_{mix}(\mu,\epsilon) \leq \frac{8}{(1-\alpha)^2}\log\left(\frac{2\beta}{\epsilon}\right)\max\left(1, \frac{64^2C_R^2}{\psi(\pi)^2(2\alpha-1)^2}\right)\,. \tag{34}$$

Proof. Apply Theorem E.1 taking $s = \epsilon/(2\beta)$. □

Theorem E.6. *Take $Q = \mathcal{N}(0,\sigma^2 I_d)$ where $\sigma > 0$ and assume E.2. Let $0 < \epsilon < 1$ and μ a β-warm distribution with respect to π. Assume that there exist $R_1 > 0$ and $0 < c_1 < 1/16$ such that*

$$\pi(\mathrm{B}(0,R_1)) \geq 1 - \frac{c_1\epsilon}{2\beta}\,,$$

then if

$$C_{R_{\epsilon,\beta}} \leq \frac{\psi(\pi)}{32}\,,$$

where

$$R_{\epsilon,\beta} = \max\left(2\sigma\sqrt{d}\left(1+\max\left(\left(\frac{-\log(\frac{c_2\epsilon}{2\beta})}{d}\right)^{1/4}, \sqrt{\frac{-\log(\frac{c_2\epsilon}{2\beta})}{d}}\right)\right), R_1\right),$$

with $c_2 = 1/16 - c_1$, *then*

$$\tau_{mix}(\mu,\epsilon) \leq 128\log\left(\frac{2\beta}{\epsilon}\right)\max\left(1, \frac{128^2 C^2_{R_{\epsilon,\beta}}}{\psi(\pi)^2}\right).$$

Proof. Let $s = \epsilon/(2\beta)$. (Dwivedi et al., 2018, Lemma 1) states that for any $c > 0$, there exists R_s such that $\phi(\mathrm{B}(0, R_s)) \geq 1 - cs$ where ϕ is a m-strongly log-concave distribution and

$$R_s = \sqrt{\frac{d}{m}}\left(2 + 2\max\left(\left(\frac{-\log(cs)}{d}\right)^{1/4}, \sqrt{\frac{-\log(cs)}{d}}\right)\right).$$

Applying this result to Q which is $1/\sigma^2$-strongly log-concave leads to $Q(\mathrm{B}(0, R_{2,s})) \geq 1 - c_2 s$ where $c_2 > 0$ and

$$R_{2,s} = \sigma\sqrt{d}\left(2 + 2\max\left(\left(\frac{-\log(c_2 s)}{d}\right)^{1/4}, \sqrt{\frac{-\log(c_2 s)}{d}}\right)\right).$$

Let $R_\Delta = R_1 + R_{2,s}$ and $R = \max(R_1, R_{2,s})$ in assumption E.3, this leads to $\mathrm{B}(0, R_1) \times \mathrm{B}(0, R_{2,s}) \subset B_\Delta$ so

$$\int_{\mathsf{B}_\Delta} \pi(\mathrm{d}x)Q(\mathrm{d}y) \geq \int_{\mathrm{B}(0,R_1)\times\mathrm{B}(0,R_{2,s})} \pi(\mathrm{d}x)Q(\mathrm{d}y) \geq (1 - c_1 s)(1 - c_2 s).$$

Because $c_1 > 0$ and $c_2 > 0$, we have that $\int_{\mathsf{B}_\Delta} \pi(\mathrm{d}x)Q(\mathrm{d}y) \geq 1 - (c_1 + c_2)s = 1 - \delta_\Delta$ thus checking condition (27) of assumption E.3. Checking (29) of assumption E.3 amounts to check

$$\begin{aligned}\frac{1}{1-\alpha}\delta_\Delta \leq \min\left(\frac{1}{4}, \frac{2\alpha-1}{64 C_R}\psi(\pi)\right) &\iff \begin{cases}\delta_\Delta & \leq \frac{1-\alpha}{4}s \\ \delta_\Delta & \leq \frac{(1-\alpha)(2\alpha-1)}{64 C_R}\psi(\pi)s\end{cases} \\ &\iff \begin{cases}(c_1+c_2) & \leq \frac{1-\alpha}{4} \\ C_R & \leq \frac{(1-\alpha)(2\alpha-1)}{64(c_1+c_2)}\psi(\pi)\end{cases}\end{aligned}$$

Maximising $(1-\alpha)(2\alpha-1)$ with $\alpha = 3/4$ leads to

$$\begin{cases}(c_1+c_2) & \leq \frac{1}{16} \\ C_R & \leq \frac{1}{512(c_1+c_2)}\psi(\pi)\end{cases},$$

and taking $c_2 = 1/16 - c_1$, we get $C_R \leq \psi(\pi)/32$. □

Assumption E.7. The target π is positive and $-\log\pi$ is m-strongly convex on $\mathbb{R}^d$: for any $x, y \in \mathbb{R}^d$ and $t \in [0,1]$,

$$-\log\pi(tx + (1-t)y) \leq -t\log\pi(x) - (1-t)\log\pi(y) - \frac{mt(1-t)}{2}|x-y|.$$

Corollary E.8. *Let π be a m-strongly log-concave distribution (Assumption E.7) with a mode at 0 and $Q = \mathcal{N}(0, \sigma^2 I_d)$ with $\sigma > 0$. Let $0 < \epsilon < 1$ and μ a β-warm distribution with respect to π. Assume that the log-weight function checks the local Lipschitz condition (26) from assumption E.2 then provided that*

$$C_{R_{\epsilon,\beta}} \leq \frac{\log 2\sqrt{m}}{32},$$

with

$$R_{\epsilon,\beta} = \max\left(\sigma\sqrt{d}\,\mathrm{r}\left(\frac{\epsilon}{272}, \beta, d\right), \sqrt{\frac{d}{m}}\,\mathrm{r}\left(\frac{\epsilon}{17}, \beta, d\right)\right),$$

where

$$\mathrm{r}(\epsilon,\beta,d) = 2\left(1+\max\left(\left(\frac{-\log(\frac{\epsilon}{2\beta})}{d}\right)^{1/4}, \sqrt{\frac{-\log(\frac{\epsilon}{2\beta})}{d}}\right)\right),$$

then

$$\tau_{mix}(\mu,\epsilon) \leq 128\log\left(\frac{2\beta}{\epsilon}\right)\max\left(1, \frac{128^2 C^2_{R_{\epsilon,\beta}}}{\log(2)^2 m}\right).$$

Proof. We use the fact that π is a m-strongly log-concave distribution to deduce that $\psi(\pi) = \log 2\sqrt{m}$ using (Cousins & Vempala, Theorem 4.4). Moreover, as the mode of π is 0, we apply (Dwivedi et al., 2018, Lemma 1) again to obtain R_1 and c_1 for Theorem E.6 with $c_1 = 1/17 < 1/16$. □

Proposition 4.3 in the main text corresponds to the asymptotic version of corollary E.8.

Illustration with a Gaussian target Take $\pi = \mathcal{N}(0, I_d)$ and $Q = \mathcal{N}(0, (1+\lambda)^2 I_d)$ with $\lambda > 0$. λ is the error term of the proposal. π is m-strongly log-concave with $m = 1$. The importance weight function w can be computed

$$w(x) = \frac{\pi(x)}{q(x)} = \sigma^{d/2}\exp\left(-\frac{1}{2}x^T\left(I_d - \frac{1}{(1+\lambda)^2}I_d\right)x\right),$$

so the log-weight function is

$$\tilde{w}(x) = -\frac{1}{2}x^T\left(I_d - \frac{1}{(1+\lambda)^2}I_d\right)x + \frac{d}{2}\log(\sigma).$$

Let $R > 0$ and $(x, y) \in \mathrm{B}(0, R)^2$,

$$\begin{aligned}
|\tilde{w}(x) - \tilde{w}(y)| &= \frac{1}{2}\left|1 - \frac{1}{(1+\lambda)^2}\right|\left|\sum_{k=1}^{d}(x_k^2 - y_k^2)\right| \\
&\leq \frac{1}{2}\left|1 - \frac{1}{(1+\lambda)^2}\right|\sum_{k=1}^{d}|x_k^2 - y_k^2| \\
&\leq R\left|1 - \frac{1}{(1+\lambda)^2}\right|\sum_{k=1}^{d}|x_k - y_k| \\
&\leq R\left|1 - \frac{1}{(1+\lambda)^2}\right|\sqrt{d}\,\|x - y\|,
\end{aligned}$$

where in the last line we used the Cauchy-Schwartz inequality $\sum_{k=1}^{d}|x_k| = \langle |x|, 1\rangle \leq |\langle |x|, 1\rangle| \leq \|x\|\sqrt{d}$. This gives us $C_R = R\sqrt{d}\left|1 - 1/(\lambda+1)^2\right|$. Using corollary (E.8), we find that asymptotically $R_{\epsilon,\beta} \underset{d\to\infty}{\sim} 2\sqrt{d}(\lambda+1)$ which leads to

$$C_{R_{\epsilon,\beta}} \underset{d\to\infty}{\sim} 2d(\lambda+1)\left|1 - \frac{1}{(1+\lambda)^2}\right|.$$

While checking the hypothesis of corollary (E.8), the error of Q need to scale as an inverse law of the dimension d

$$\lambda \underset{d\to\infty}{\sim} \sqrt{\frac{K}{2d} + 1} - 1 \underset{d\to\infty}{\sim} \frac{1}{d} \text{ where } K = \frac{\log 2}{32},$$

in order to keep the mixing time bellow a constant despite the increase in dimension.

Illustration with a non-isotropic Gaussian target Take $\pi = \mathcal{N}(0, \Sigma)$ with $\Sigma = \mathrm{diag}(c_1^2, \dots, c_d^2)$ with $c_i > 0$ for all $i \in \{1, \dots, d\}$ such $m \leq c_1 \leq \dots \leq L$ with $m, L > 0$. Moreover, we set $Q = \mathcal{N}(0, \sigma^2 I_d)$ with $\sigma > 0$. For sake of simplicity, we assume that $L = m^{-1}$ and $m < 1$. Note that m is the strong log-concave constant of π. With the same reasoning as the previous example, we have that

$$w(x) = \frac{\pi(x)}{q(x)} = \sigma^{d/2} \exp\left(-\frac{1}{2} x^T \left(\Sigma^{-1} - \frac{1}{\sigma^2} I_d\right) x\right) ,$$

leading to

$$\tilde{w}(x) = -\frac{1}{2} x^T \left(\Sigma^{-1} - \frac{1}{\sigma^2} I_d\right) x + \frac{d}{2} \log(\sigma) .$$

Let $R > 0$ and $(x, y) \in \mathrm{B}(0, R)^2$,

$$\begin{aligned} |\tilde{w}(x) - \tilde{w}(y)| &= \frac{1}{2} \left| \sum_{k=1}^{d} \left(\frac{1}{c_i^2} - \frac{1}{\sigma^2}\right) (x_k^2 - y_k^2) \right| \\ &\leq R \max_{i \in \{1, \dots, d\}} \left| \frac{1}{c_i^2} - \frac{1}{\sigma^2} \right| \sum_{k=1}^{d} |x_k - y_k| \\ &\leq R \tilde{c}_\sigma \sqrt{d} \, \|x - y\| , \end{aligned}$$

where $\tilde{c}_\sigma = \max_{i \in \{1, \dots, d\}} \left| \frac{1}{c_i^2} - \frac{1}{\sigma^2} \right|$. This leads to

$$C_R = \sqrt{d} R \tilde{c}_\sigma .$$

Using, corollary E.8 we find that asymptotically

$$C_R \underset{d \to \infty}{\sim} 2d \max\left(\sigma, \frac{1}{\sqrt{m}}\right) \tilde{c}_\sigma .$$

We now choose σ to minimize either the forward KL or the backward KL between $\mathcal{N}(0, \Sigma)$ and $\mathcal{N}(0, \sigma^2 I_d)$.

$$\begin{aligned} D_{KL}(\mathcal{N}(0, \Sigma), \mathcal{N}(0, \sigma^2 I_d)) &= \frac{1}{2} \left(\frac{1}{\sigma^2} \sum_{i=1}^{d} c_i^2 - d + d \log \sigma^2 - \sum_{i=1}^{d} \log c_i^2 \right) , \\ \frac{\mathrm{d} D_{KL}(\mathcal{N}(0, \Sigma), \mathcal{N}(0, \sigma^2 I_d))}{\mathrm{d}\sigma^2}(\sigma^2) &= \frac{1}{2\sigma^2} \left(d - \frac{1}{\sigma^2} \sum_{i=1}^{d} c_i^2 \right) , \\ D_{KL}(\mathcal{N}(0, \sigma^2 I_d), \mathcal{N}(0, \Sigma)) &= \frac{1}{2} \left(\sigma^2 \sum_{i=1}^{d} \frac{1}{c_i^2} - d + \sum_{i=1}^{d} c_i^2 - d \log \sigma^2 \right) , \\ \frac{\mathrm{d} D_{KL}(\mathcal{N}(0, I_d), \mathcal{N}(0, \Sigma))}{\mathrm{d}\sigma^2}(\sigma^2) &= \frac{1}{2} \left(\sum_{i=1}^{d} \frac{1}{c_i^2} - \frac{d}{\sigma^2} \right) , \end{aligned}$$

leading to $\sigma_f^2 = \sum_{i=1}^{d} c_i^2 / d$ being the minimizer of the forward KL and $\sigma_b^2 = d / \sum_{i=1}^{d} (1/c_i^2)$ being the minimizer of the backward KL. As mentionned in Sec. 3, σ_f^2 corresponds to an over-spread Gaussian while σ_b^2 corresponds to an over-concentrated Gaussian. Using that $m \leq c_i^2 \leq L$ for all $i \in \{1, \dots, d\}$, we can compute $\max(\sigma_f, 1/\sqrt{m}) = 1/\sqrt{m}$ and $\max(\sigma_b, 1/\sqrt{m}) = 1/\sqrt{m}$. Using the same property, we have that $\tilde{c}_{\sigma_f} \leq 1/m - m$ and $\tilde{c}_{\sigma_b} \leq 1/m - m$. Using the upper bound for the mixing time from corollary E.8 and ignoring all the constants, we get an upper bound in $\mathcal{O}\left(d^2 (1-m)^2 / m^4\right)$ for both σ_f and σ_b which is much more dimension sensitive than MALA's bound $\mathcal{O}(\sqrt{d}/m^2)$ (Dwivedi et al., 2018).

F. Additional details on real world examples

F.1. Computational considerations

neutra-MCMC methods should have a much higher computational cost compared to other methods. Figure 17 shows that in two of our real world experiments, neutra-MCMC methods were significantly more expensive than flow-MCMC methods.

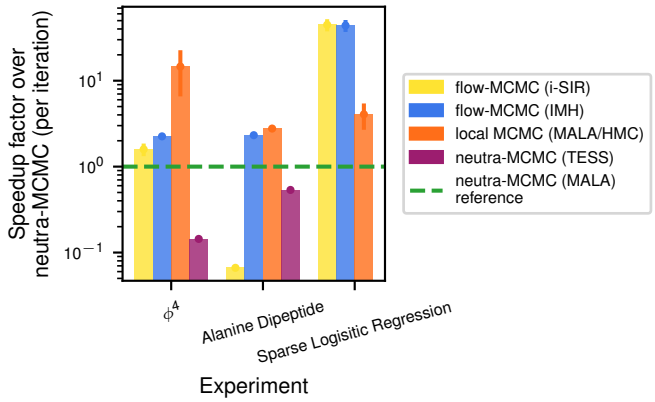

Figure 17. Speedup factor compared to neutra-MCMC on real world experiments

However, in the molecular system experiment (Sec. 6.1) flow-MCMC methods were more costly : that's because the distribution of this molecular system is very expensive to evaluate and breaks the parallel capabilities of flow-MCMC methods such as i-SIR. Note that using ESS always led to high computational costs.

F.2. Experimental details on logistic regression

Logistic regression Consider a training dataset $\mathcal{D} = \{(x_k, y_k)\}_{j=1}^M$ where $x_k \in \mathbb{R}^d$ and $y_k \in \{-1, 1\}$. In our case $\mathcal{D}$ is the german credit dataset ($M = 750$) where each x_k represent 24 details about a person who takes a credit (age, sex, job, ...) and y_k is -1 if this person is considered having bad credits risks and 1 otherwise. We standardised this dataset so that each component of x_k is between -1 and 1 and also added a constant 1 component.

We consider that the likelihood of a pair (x, y) is given by $p(y|x, w) = \text{Bernoulli}(y; \sigma(x^T w))$ where $w \in \mathbb{R}^d$ is a weight vector and σ is the sigmoid function. Given a prior distribution $p(w)$, we sample the posterior distribution $p(w|\mathcal{D})$ and compute the posterior predictive distribution $p(y|x, \mathcal{D}) = \int p(y|x, w, \mathcal{D})p(w|\mathcal{D})dw \simeq 1/n \sum_{i=1}^n p(y|x, w_i, \mathcal{D})p(w_i|\mathcal{D})$ for $(x, y) \in \mathcal{D}_{test}$.

Here, we consider a sparse prior suggested by (Carvalho et al., 2009) which can be written as follows, $w = \tau\beta \circ \lambda$ and

$$p(w) = p(\tau, \beta, \lambda) = \text{Gamma}(\tau; \alpha = 0.5, \beta = 0.5) \times \prod_{i=1}^{d} \text{Gamma}(\lambda_i; \alpha = 0.5, \beta = 0.5) \times \text{Normal}(\beta_i; 0, 1)\,.$$

We log-transform the Gamma distributions by replacing Gamma with Gamma_{log} to ensure the positivity of the global scale τ and the local scale λ.

$$\text{Gamma}_{log}(y; \alpha, \beta) = \text{Gamma}(\exp y; \alpha, \beta) \times \exp y\,.$$

Flow and training The normalizing flow at stake is an Inverse Autoregressive Flow (IAF) (Papamakarios et al., 2017) trained with the procedure described in (Hoffman et al., 2019) on a train dataset : it is a 3 layers deep flow using a residual neural networks wide of 51 neurons and deep of 2 layers with elu activation function. The flow was trained by optimizing the backward Kullback-Leiber with a learning rate of 10^{-2} (scaled by 10% every 1000 optimization steps) using Adam optimizer for 5000 steps with a batch size of size 4096.

Sampling details We used Pyro's (Bingham et al., 2019) implementation of Hamilotonian Monte Carlo (HMC) to sample the previously described model. The warmup phase [14] lasted 64 steps and the MCMC chains were 256 steps long - this number was chosen as it guarantees a $\hat{R}$ close to 1. We automatically adapted both the step size and the mass matrix while the length of the trajectory (for the Leapfrog integrator) was frozen to 8 . Samplers involving i-SIR used $N = 100$ particles and importance sampling leveraged $N \times 256 = 25600$ particles. The global/local samplers interleaved 20 local steps between global steps.

[14] In the following, warmup samples will be always discarded

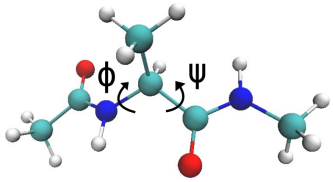

Figure 18. Visualization of alanine dipeptide and the dihedral angles ϕ and ψ for the Ramachandran plot (taken from (Midgley et al., 2022))

F.3. Experimental details on alanine dipeptide

The alanine dipeptide experiment aimed at sampling from the Boltzmann distribution on the 3D coordinates of the 22 atoms alanine dipeptide molecule (shown in Fig. 18). The details of the distribution can be found in (Midgley et al., 2022), we used their neural spline normalizing flow trained using FAB as the common flow in all flow-based samplers. The ground truth was obtained through parallel tempering MD simulations from the same paper. We used 512 chains of length 600 using the same hyperparameters as the mixture of Gaussians : $N = 128$, $n_{local} = 5$ and a target acceptance of 75%.

F.4. Experimental details on ϕ^4

Target distribution Following (Gabrié et al., 2022) we consider a 1d ϕ^4 on a grid of d points with Dirichlet boundary conditions at both extremeties of the field; that is $\phi_0 = \phi_{d+1} = 0$ where one configuration is the d-dimensional vector $(\phi_i)_{i=1}^d$. The negative logarithm of the density is given by

$$-\ln \pi(\phi) = \beta \left(\frac{ad}{2} \sum_{i=1}^{d+1} (\phi_i - \phi_{i-1})^2 + \frac{1}{4ad} \sum_{i=1}^{d} (1 - \phi_i^2)^2 \right) \tag{35}$$

with a flow a colored normal distribution as base built by keeping the quadratic terms from above:

$$-\ln \rho(\phi) = \beta \left(\frac{ad}{2} \sum_{i=1}^{d+1} (\phi_i - \phi_{i-1})^2 + \frac{1}{2ad} \sum_{i=1}^{d} \phi_i^2 \right), \tag{36}$$

also with Dirichlet boundary conditions at 0. We chose parameter values for which the system is bimodal: $a = 0.1$ and inverse temperature $\beta = 20$ (see Fig. 4 in the main paper).

Training of the normalizing flow The flows at stake are again RealNVPs and their hyper-parameters are given in table 8. We used Adam optimizer to minimize the forward KL. This forward KL was approximated by taken exactly 50% of samples in each mode by running MALA locally. The size of those chains is $\frac{\text{batch size}}{\text{\# of parallel MCMC chains}}$.

Sampling parameters The sampling parameters were selected using a grid-search which optimized the quality of the mode weight estimation. The starting sample of each chain was taken to be in a single mode [15]. We used 256 MCMC chains of length 512 where the first half of the chain was discarded for warmup purposes. The sampling hyperparameters are detailed in table 9.

Table 8. RealNVP hyperparameters for the ϕ^4 experiment.

Dimension	Depth of the flow	# of layers in NN	# of neuron per layer	# of training steps	Learning rate	Learning rate decay factor	Batch size	# MCMC chains
64	5	3	128	10^4	10^{-2}	0.98	4096	64
128	6	256	3	2×10^4	10^{-2}	0.98	4096	64
256	10	512	4	3.5×10^4	5×10^{-3}	0.98	8192	64
512	15	512	5	6×10^4	5×10^{-4}	0.99	8192	32

[15] This is achieved by running a gradient descent starting from a constant unit configuration

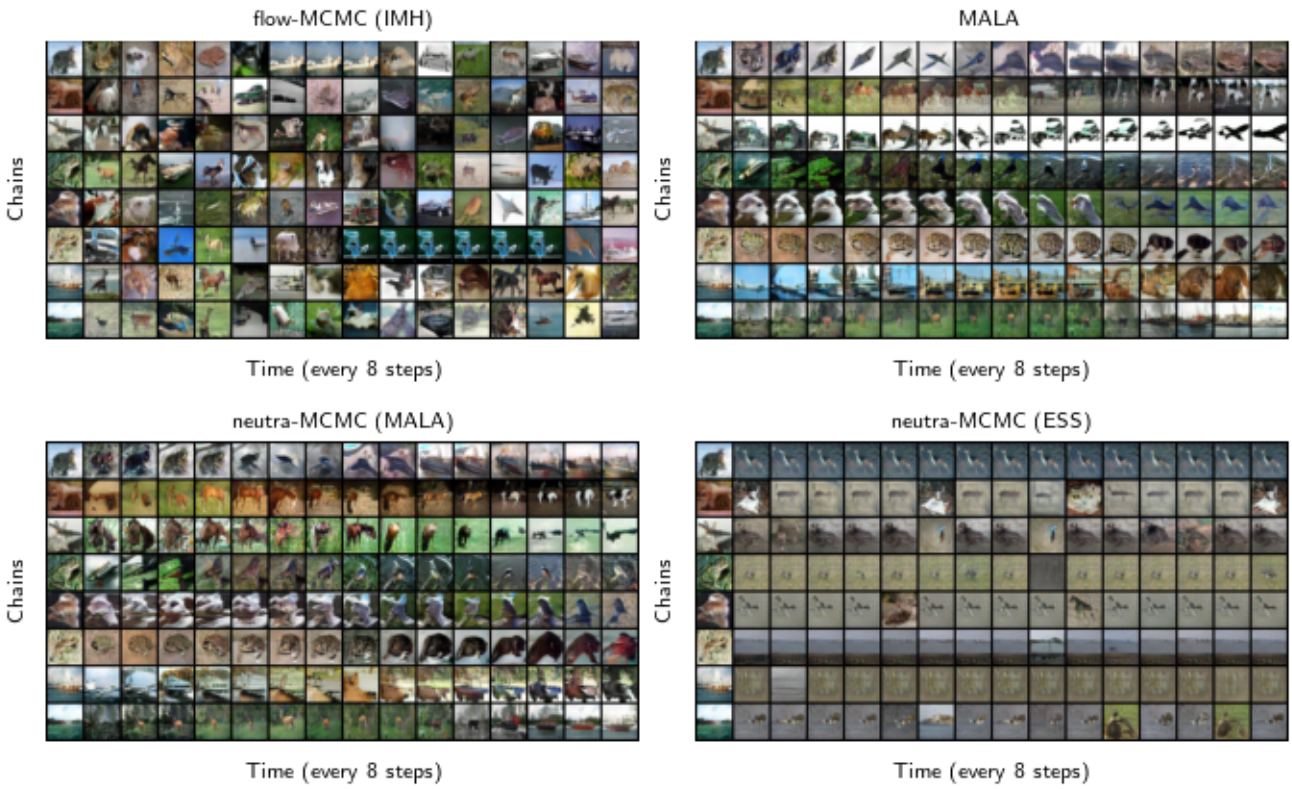

Figure 19. MCMC chains sampled from a SN-GAN as an energy-based model

Table 9. Sampling hyperparameters for the ϕ^4 experiment.

Dimension	n	N (flow-MCMC)	N (neutra-flow-MCMC)	N (IS)	n_{local} (flow-MCMC)	n_{local} (neutra-flow-MCMC)
64	256 ($\times$2)	60	60	3840	25	25
128	256 ($\times$2)	80	80	5120	25	25
256	256 ($\times$2)	100	100	6400	25	25
512	256 ($\times$2)	120	120	7680	25	25

G. Sampling image distributions

Design of the target distribution We trained a SN-GAN (Miyato et al., 2018) on the CIFAR10 dataset for 100.000 epochs (implementation from https://github.com/kwotsin/mimicry). The latent space of the SNGAN is of dimension 128. We use a multivariate centered standard Gaussian as a base distribution. Following (Che et al., 2020), we define a probability measure as an energy-based model on the latent space of the GAN

$$p(z) = \exp(-E_{GAN}(z))/Z_\theta \text{ with } E_{GAN}(z) = -\log p_0(z) - \text{logit}(D(G(z))),$$

where p_0 is the base distribution, G is the generator, D is the discriminator and $\text{logit}(y) = \log(y/(1-y))$ is the inverse sigmoid function. Slices of E_{GAN} can be found on Fig. 20 : $p(z)$ is multimodal and has many bad geometries. The samples of $p(z)$ can be transformed in images (which belong to $\mathbb{R}^{3\times32\times32}$) by pushing them through G.

Sampling procedure We sampled $p(z)$ using 4 different MCMC algorithms. Fig. 19 depicts 8 chains for each sampler started in the same state. The flow based samplers use a RealNVP normalizing flow which has 8 layers each one having a MLP wide of 128 neurons and deep of 3 layers. This normalizing flow was trained using the backward KL loss for 512

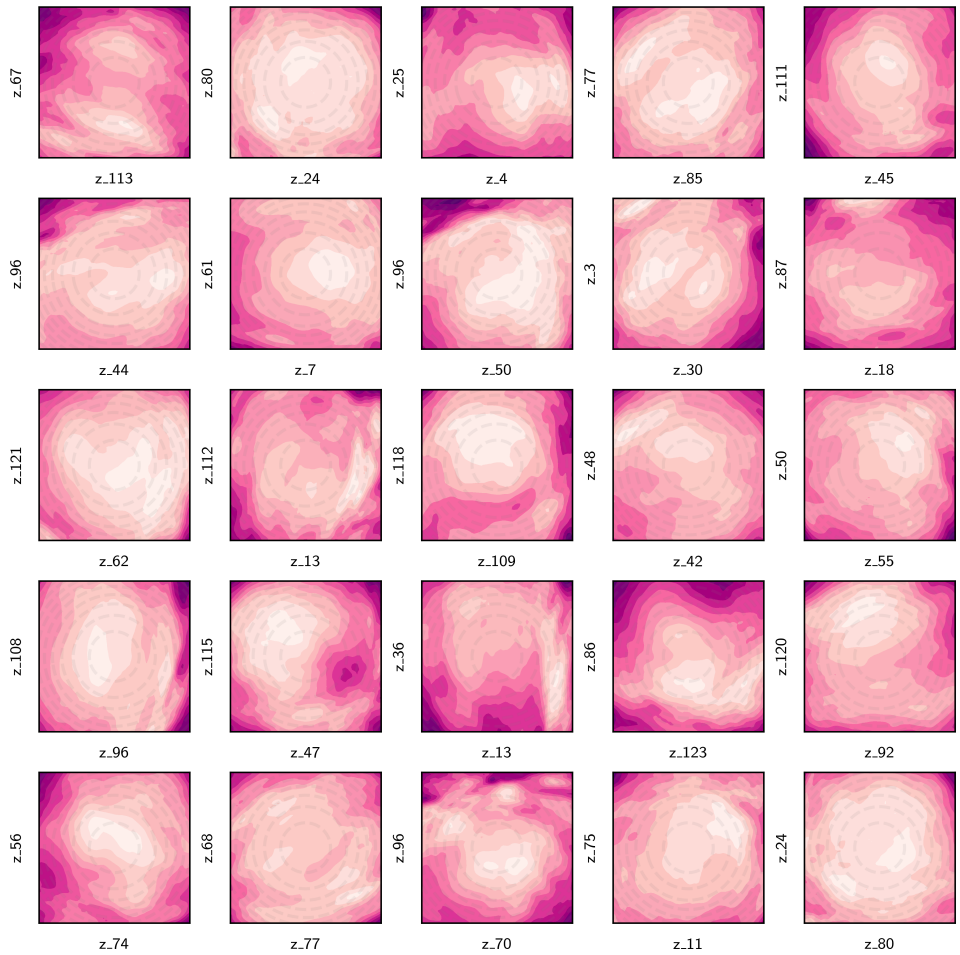

Figure 20. Random 2D slices of E_{GAN}. The purple colors reperensent the level lines of $E_{GAN}(z)$ and the grey lines are the level lines of $z \mapsto -\log p_0(z)$

iterations using batches of size 1024. The chains are 128 samples long (on Fig. 19, the chains are subsampled every 8 steps) and have the same hyper-parameters as in the four Gaussians experiment in dimension 128.

Fig. 19 shows that flow-MCMC algorithm mixes between the modes of this moderately-high dimensional distribution ($d = 128$) more often than its neutra-MCMC counterparts. This observation is consistent with the results of Sec. 3 and Sec. 6.

H. Amount of computation and type of resources

We used a single type of GPU, the Nvidia A100. Each experiment of Section 3 took about 2 hours to run when distributed on 4 GPUs. Moreover, the training of the 5 RealNVPs for Section 3.2 took 6 hours on 4 GPUs. Regarding the real world experiments (see Section 6), training the flow for the molecular experiment took 24 hours on a single GPU and 6 hours to perform the sampling part on 4 GPUs. Training the flows for the ϕ^4 experiments took 12 hours on 2 GPUs and 1 hours on 4 GPUs for sampling. Finally, training the flow for the bayesian logistic regression took less than a minute but sampling with HMC lasted 6 hours on 4 GPUs. In total, this work required 55 hours of parallel GPU run-time.