SOLVING BAYESIAN INVERSE PROBLEMS WITH DIFFUSION PRIORS AND OFF-POLICY RL

Luca Scimeca* Mila, Université de Montréal
Siddarth Venkatraman* Mila, Université de Montréal
Moksh Jain* Mila, Université de Montréal
Minsu Kim* Mila, Université de Montréal KAIST
Marcin Sendera* Mila, Université de Montréal Jagiellonian University
Mohsin Hasan Mila, Université de Montréal
Luke Rowe Mila, Université de Montréal
Sarthak Mittal Mila, Université de Montréal
Pablo Lemos Mila, Université de Montréal Ciela Institute Dreamfold
Emmanuel Bengio Recursion
Alexandre Adam Mila, Université de Montréal Ciela Institute
Jarrid Rector-Brooks Mila, Université de Montréal Dreamfold
Yashar Hezaveh Mila, Université de Montréal Ciela Institute
Laurence Perreault-Levasseur Mila, Université de Montréal Ciela Institute
Yoshua Bengio Mila, Université de Montréal CIFAR
Glen Berseth Mila, Université de Montréal CIFAR
Nikolay Malkin Mila, Université de Montréal University of Edinburgh

{luca.scimeca,siddarth.venkatraman,moksh.jain, minsu.kim,marcin.sendera,...,nikolay.malkin}@mila.quebec

ABSTRACT

This paper presents a practical application of Relative Trajectory Balance (RTB), a recently introduced off-policy reinforcement learning (RL) objective that can asymptotically solve Bayesian inverse problems optimally. We extend the original work by using RTB to train conditional diffusion model posteriors from pretrained unconditional priors for challenging linear and non-linear inverse problems in vision, and science. We use the objective alongside techniques such as off-policy backtracking exploration to improve training. Importantly, our results show that existing training-free diffusion posterior methods struggle to perform effective posterior inference in latent space due to inherent biases.

1 INTRODUCTION

While deep learning has seen rapid advancements, scientific discovery, particularly in high-dimensional and multimodal contexts, remains a significant challenge. Many scientific problems, such as inverse protein design and gravitational lensing, can be framed as Bayesian inverse problems (Kaipio & Somersalo, 2006; Idier, 2013; Dashti & Stuart, 2013; Latz, 2020) due to the inherent uncertainties, often introduced by imperfections in scientific instruments. Traditionally, the scientific community has approached Bayesian inference using methods like Markov chain Monte Carlo (MCMC), or more efficient alternatives like variational inference (MacKay, 2003),

Hamiltonian Monte Carlo (Duane et al., 1987; Neal, 2012), and Langevin dynamics (Besag, 1994; Roberts & Tweedie, 1996; Roberts & Rosenthal, 1998). However, these methods become impractical when applied to complex, real-world scenarios.

Recently, diffusion models (Sohl-Dickstein et al., 2015; Ho et al., 2020; Song et al., 2021b) have emerged as a promising approach for tackling Bayesian inverse problems. Diffusion models Sohl-Dickstein et al. (2015); Ho et al. (2020); Song et al. (2021b) are a powerful class of hierarchical generative models, used to model complex distributions over a varied range of objects, including images Nichol & Dhariwal (2021); Dhariwal & Nichol (2021); Rombach et al. (2022), text (Austin et al., 2021; Dieleman et al., 2022; Li et al., 2022; Han et al., 2023; Gulrajani & Hashimoto, 2023; Lou et al., 2023), actions in reinforcement learning, Janner et al. (2022); Wang et al. (2023); Kang et al. (2024), proteins and more besides. In each of these domains, downstream problems require sampling product distributions, where a pretrained diffusion model serves as a prior $p(\mathbf{x})$ that is multiplied by an auxiliary constraint $r(\mathbf{x})$. For example, if $p(\mathbf{x})$ is a prior over images defined by a diffusion model, and $r(\mathbf{x}) = p(c \mid \mathbf{x})$ is the likelihood that an image $\mathbf{x}$ belongs to class c, then class-conditional image generation requires sampling from the Bayesian posterior $p(\mathbf{x} \mid c) \propto p(\mathbf{x})p(c \mid \mathbf{x})$.

The hierarchical nature of the generative process in diffusion models, which generate samples from $p(\mathbf{x})$ by a deep chain of stochastic transformations, makes exact sampling from posteriors $p(\mathbf{x})r(\mathbf{x})$ under a black-box function $r(\mathbf{x})$ intractable. Common solutions to this problem involve inference techniques based on linear approximations Song et al. (2022); Kawar et al. (2021); Kadkhodaie & Simoncelli (2021); Chung et al. (2023) or stochastic optimization Graikos et al. (2022); Mardani et al. (2024). Others estimate the 'guidance' term – the difference in drift functions between the diffusion models sampling the prior and posterior – by training a classifier on noised data Dhariwal & Nichol (2021), but when such data is not available, one must resort to approximations or Monte Carlo estimates (Song et al., 2023; Dou & Song, 2024; Cardoso et al., 2024), which are challenging to scale to high-dimensional problems. Reinforcement learning methods that have recently been proposed for this problem Black et al. (2024); Fan et al. (2023) are biased and prone to mode collapse Venkatraman et al. (2024).

Recently, Venkatraman et al. (2024) introduced an asymptotically unbiased objective for finetuning a diffusion prior to sample from the Bayesian posterior. The objective was named *relative trajectory balance* (RTB) due to its relationship with the trajectory balance objective (Malkin et al., 2022), as they both arise from the generative flow network perspective of diffusion models (Lahlou et al., 2023; Zhang et al., 2023). RTB is an asymptotically unbiased objective for finetuning a diffusion prior to sample from the Bayesian posterior, and has been presented as an alternative to existing on-policy, policy gradient-based methods Black et al. (2024); Fan et al. (2023) due to its off-policy training capabilities. However, unlike other methods, RTB has not been thoroughly evaluated on complex, real-world challenges, raising questions about its scalability to high-dimensional problems and its feasibility in such settings.

In this paper, we demonstrate the effectiveness of RTB through its application to intractable linear and nonlinear Bayesian inverse problems in vision and the scientific application of gravitational lensing. We challenge the uncertainties over prior work by providing empirical evidence that RTB, when combined with off-policy adaptation techniques (e.g., as introduced in (Zhang & Chen, 2021; Sendera et al., 2024b)), significantly improves scalability to complex scientific discovery problems. We also extend RTB to train conditional diffusion posteriors from unconditional priors, and add experiments combining this objective with other state-of-the-art techniques (e.g. DPS Chung et al. (2023) or FPS Song et al. (2023)). Our findings demonstrate that RTB is not only a viable off-the-shelf objective for Bayesian inverse problems in scientific domains but also offer comprehensive guidance for practitioners on scaling RTB to high-dimensional settings.

To summarize, our contributions are as follows:

- We demonstrate that RTB can effectively address a wide range of complex, linear and non-linear Bayesian inverse problems in vision. W
- also provide empirical evidence that specific off-policy adaptation techniques enhance RTB, offering practical insights for real-world applications.
- We extend RTB to train conditional diffusion posteriors from unconditional priors.
- We extend RTB to integrate additional state of the art tecniques (e.g. DPS & FPS)

- We conduct a extensive benchmarks of prior methods, empirically revealing the limitations of existing training-free methods.

2 Solving Bayesian inverse problems with relative trajectory balance

Inverse problems. A typical inverse problem is the following: We are interested in recovering some quantity $\mathbf{x} \sim p(\mathbf{x})$. However, in the process of measurement, the quantity of interest is perturbed by some noise, or instrumental systematic effect. The new observation $\mathbf{y} \sim p(\mathbf{y})$ contains information about the observation of interest, but it has been distorted by the experiment. Furthermore, we assume (as it is often the case) that we have a good enough understanding of our instrumentation, to be able to compute $p(\mathbf{y}|\mathbf{x})$, *i.e.*, if we assume a true underlying $\mathbf{x}$, we know how likely it is to recover our observation. What we are interested in, however, is $p(\mathbf{x} \mid \mathbf{y})$, *i.e.*, given our observation, how likely is a given value of $\mathbf{x}$.

Inverse problems such as these are very common in various scientific disciplines, but can be extremely ill-posed, particularly if the noise is complex and non-linear, and if the quantities of interest are high-dimensional. Traditional methods, such as Markov-Chain Monte Carlo, quickly become unusable on complex problems, such as the ones we illustrate in Section 4 of this paper. Advances in generative modelling (Song et al., 2021b) have made diffusion models suitable for learning rich and expressive priors from data for inverse problems (Adam et al., 2022).

Summary of setting. A denoising diffusion model generates data $\mathbf{x}_1$ by a Markovian generative process:

$$\textit{(noise)} \quad \mathbf{x}_0 \to \mathbf{x}_{\Delta t} \to \mathbf{x}_{2\Delta t} \to \ldots \to \mathbf{x}_1 = \mathbf{x} \quad \textit{(data)}, \tag{1}$$

where $\Delta t = \frac{1}{T}$ and T is the number of discretization steps.[1] The initial distribution $p(\mathbf{x}_0)$ is fixed (typically to $\mathcal{N}(\mathbf{0}, \boldsymbol{I})$) and the transition from $\mathbf{x}_{t-1}$ to $\mathbf{x}_t$ is modeled as a Gaussian perturbation with time-dependent variance:

$$p(\mathbf{x}_{t+\Delta t} \mid \mathbf{x}_t) = \mathcal{N}(\mathbf{x}_{t+\Delta t} \mid \mathbf{x}_t + u_t(\mathbf{x}_t)\Delta t, \sigma_t^2 \Delta t \boldsymbol{I}). \tag{2}$$

The scaling of the mean and variance by Δt is insubstantial for fixed T, but ensures that the diffusion process is well-defined in the limit $T \to \infty$ assuming regularity conditions on u_t. The process given by (1, 2) is then identical to Euler-Maruyama integration of the stochastic differential equation (SDE) $d\mathbf{x}_t = u_t(\mathbf{x}_t)\, dt + \sigma_t\, d\mathbf{w}_t$.

The likelihood of a denoising trajectory $\mathbf{x}_0 \to \mathbf{x}_{\Delta t} \to \cdots \to \mathbf{x}_1$ factors as

$$p(\mathbf{x}_0, \mathbf{x}_{\Delta t}, \ldots, \mathbf{x}_1) = p(\mathbf{x}_0) \prod_{i=1}^{T} p(\mathbf{x}_{i\Delta t} \mid \mathbf{x}_{(i-1)\Delta t}) \tag{3}$$

and defines a marginal density over the data space:

$$p(\mathbf{x}_1) = \int p(\mathbf{x}_0, \mathbf{x}_{\Delta t}, \ldots, \mathbf{x}_1)\, d\mathbf{x}_0\, d\mathbf{x}_{\Delta t} \ldots d\mathbf{x}_{1-\Delta t}. \tag{4}$$

A reverse-time process, $\mathbf{x}_1 \to \mathbf{x}_{1-\Delta t} \to \cdots \to \mathbf{x}_0$, with densities q, can be defined analogously, and similarly defines a conditional density over trajectories:

$$q(\mathbf{x}_0, \mathbf{x}_{\Delta t}, \ldots, \mathbf{x}_{1-\Delta t} \mid \mathbf{x}_1) = \prod_{i=1}^{T} q(\mathbf{x}_{(i-1)\Delta t} \mid \mathbf{x}_{i\Delta t}). \tag{5}$$

In the training of diffusion models, as discussed below, the process q is typically fixed to a simple distribution (usually a discretized Ornstein-Uhlenbeck process), and the result of training is that p and q are close as distributions over trajectories. In this work, we consider diffusino model priors.

[1] The time indexing suggestive of an SDE discretization is used for consistency with the diffusion samplers literature Zhang & Chen (2021); Sendera et al. (2024a). The indexing $\mathbf{x}_T \to \mathbf{x}_{T-1} \to \cdots \to \mathbf{x}_0$ is often used for diffusion models trained from data.

Intractable inference under a diffusion prior. Consider a diffusion model p_θ, defining a marginal density $p_\theta(\mathbf{x}_1)$, and a positive constraint function $r : \mathbb{R}^d \to \mathbb{R}_{>0}$. We are interested in training a diffusion model p_ϕ^{post}, with drift function u_ϕ^{post}, that would sample the product distribution $p^{\text{post}}(\mathbf{x}_1) \propto p_\theta(\mathbf{x}_1) r(\mathbf{x}_1)$. If $r(\mathbf{x}_1) = p(\mathbf{y} \mid \mathbf{x}_1)$ is a conditional distribution over another variable $\mathbf{y}$, then p^{post} is the Bayesian posterior $p_\theta(\mathbf{x}_1 \mid \mathbf{y})$.

Because samples from $p^{\text{post}}(\mathbf{x}_1)$ are not assumed to be available, one cannot directly train p using the forward KL objective. Nor can one directly apply objectives for distribution-matching training, such as those that enforce the trajectory balance (TB) constraint, since the marginal $p_\theta(\mathbf{x}_1)$ is not available. However, Venkatraman et al. (2024) makes the observation that an alternate constraint relates the denoising process which samples from the posterior to the one which samples from the prior, and proposes an objective as a function of the vector ϕ that parametrizes the posterior diffusion model and the scalar Z_ϕ (parametrized via $\log Z_\phi$ for numerical stability) as follows:

$$\mathcal{L}_{\text{RTB}}(\mathbf{x}_0 \to \mathbf{x}_{\Delta t} \to \cdots \to \mathbf{x}_1; \phi) := \left(\log \frac{Z_\phi \cdot p_\phi^{\text{post}}(\mathbf{x}_0, \mathbf{x}_{\Delta t}, \ldots, \mathbf{x}_1)}{r(\mathbf{x}_1) p_\theta(\mathbf{x}_0, \mathbf{x}_{\Delta t}, \ldots, \mathbf{x}_1)} \right)^2 . \tag{6}$$

When all prior assumptions hold, optimizing this objective to 0 for all trajectories ensures that $p_\phi^{\text{post}}(\mathbf{x}_1) \propto p_\theta(\mathbf{x}_1) r(\mathbf{x}_1)$.

Notably, the gradient of this objective with respect to ϕ does not require differentiation (backpropagation) into the sampling process that produced a trajectory $\mathbf{x}_0 \to \cdots \to \mathbf{x}_1$. This offers two advantages over on-policy simulation-based methods: (1) the ability to optimize $\mathcal{L}_{\text{RTB}}$ as an off-policy objective, *i.e.*, sampling trajectories for training from a distribution different from p_ϕ^{post} itself, as discussed further in §3; (2) backpropagating only to a subset of the summands in (6), when computing and storing gradients for all steps in the trajectory is prohibitive for large diffusion models. We discuss further details about the training and parametrization in §3.

3 Training, parametrization, and conditioning

Training and exploration. The choice of which trajectories we use to take gradient steps with the RTB loss can have a large impact on sample efficiency. In *on-policy* training, we use the current policy p_ϕ^{post} to generate trajectories $\tau = (\mathbf{x}_0 \to \ldots \to \mathbf{x}_1)$, evaluate the reward $\log r(\mathbf{x}_1)$ and the likelihood of τ under p_θ, and a gradient updates on ϕ to minimize $\mathcal{L}_{\text{RTB}}(\tau; \phi)$.

However, on-policy training may be insufficient to discover the modes of the posterior distribution. In this case, we can perform *off-policy* exploration to ensure mode coverage. For instance, given samples $\mathbf{x}_1$ that have high density under the target distribution, we can sample *noising* trajectories $\mathbf{x}_1 \leftarrow \mathbf{x}_{1-\Delta t} \leftarrow \ldots \leftarrow \mathbf{x}_0$ starting from these samples and use such trajectories for training. Another effective off-policy training technique uses replay buffers. We expect the flexibility of mixing on-policy training with off-policy exploration to be a strength of RTB over on-policy RL methods, as was shown for distribution-matching training of diffusion models in Sendera et al. (2024a).

Conditional constraints and amortization. We extend the RTB objective to amortize conditional posterior inference from an unconditional diffusion prior. If the constraints depend on other variables $\mathbf{y}$ – for example, $r(\mathbf{x}_1; \mathbf{y}) = p(\mathbf{y} \mid \mathbf{x}_1)$ – then the posterior drift u_ϕ^{post} can be conditioned on $\mathbf{y}$ and the learned scalar $\log Z_\phi$ replaced by a model taking $\mathbf{y}$ as input. In this case, Z_ϕ is thus a function of the conditioning variable $Z_\phi(\mathbf{s})$. For continuous variables $\mathbf{s}$ or if the number of categories for discrete $\mathbf{s}$ are large, we can parametrize ϕ as a neural network. Such conditioning achieves amortized inference and allows generalization to new $\mathbf{y}$ not seen in training. Similarly, all of the preceding discussion easily generalizes to *priors* that are conditioned on some context variable, yielding:

$$\mathcal{L}_{\text{RTB}}(\mathbf{x}_0 \to \mathbf{x}_{\Delta t} \to \cdots \to \mathbf{x}_1; \mathbf{s}, \phi) := \left(\log \frac{Z_\phi(\mathbf{s}) \cdot p_\phi^{\text{post}}(\mathbf{x}_0, \mathbf{x}_{\Delta t}, \ldots, \mathbf{x}_1 \mid \mathbf{s})}{r(\mathbf{x}_1, \mathbf{s}) p_\theta(\mathbf{x}_0, \mathbf{x}_{\Delta t}, \ldots, \mathbf{x}_1 \mid \mathbf{s})} \right)^2 \tag{7}$$

Efficient parametrization and Langevin inductive bias. Because the deep features learned by the prior model u_θ are expected to be useful in expressing the posterior drift u_ϕ^{post}, we can choose to

initialize u_ϕ^{post} as a copy of u_θ and to fine-tune it, possibly in a parameter-efficient way (as described in each section of §4). This choice is inspired by the method of amortizing inference in large language models by fine-tuning a prior model to sample an intractable posterior Hu et al. (2024).

Furthermore, if the constraint $r(\mathbf{x}_1)$ is differentiable, we can impose an inductive bias on the posterior drift similar to the one introduced for diffusion samplers of unnormalized target densities in Zhang & Chen (2021) and shown to be useful for off-policy methods in Sendera et al. (2024a). namely, we write

$$u_\phi^{\text{post}}(\mathbf{x}_t, t) = \text{NN}_1(\mathbf{x}_t, t; \phi) + \text{NN}_2(\mathbf{x}_t, t, \phi)\nabla_{\mathbf{x}_t} \log r(\mathbf{x}_t), \tag{8}$$

where NN_1 and NN_2 are neural networks outputting a vector and a scalar, respectively. This parametrization allows the constraint to provide a signal to guide the sampler at intermediate steps.

4 Experiments

In this section, we demonstrate the wide applicability of RTB to sample from complex image posteriors with diffusion priors, and highlight important shortcomings of current methods.

Linear inverse problems

Inpainting Inpainting is a classical inverse problem where the goal is to reconstruct missing or occluded parts of an image (Chung et al., 2023). Let $\mathbf{x}$ represent the original image, and the forward operator $A(\mathbf{x}) = P\mathbf{x}$, where P is a masking matrix that zeros out the missing pixels, representing the incomplete observation. The measurement $\mathbf{y}$ is the partially observed image, and is subject to noise of scale σ, so $\mathbf{y} = P\mathbf{x} + \mathcal{N}(0, \sigma^2\mathbf{I})$. The task is to infer the posterior distribution $p(\mathbf{x} \mid \mathbf{y})$. We consider two types of inpainting tasks: random inpainting and box inpainting. In random inpainting, the mask P is applied randomly to a set of pixels, removing a random subset of the image. Box inpainting, instead, is a variant where a large rectangular region of the image is removed (Kadkhodaie & Simoncelli, 2021). We use RTB to fine-tune a score-based prior $p_\theta(\mathbf{x})$ into a posterior $p_\theta(\mathbf{x})p(\mathbf{y} \mid \mathbf{x})$ with likelihood $p(\mathbf{y} \mid \mathbf{x}) \propto \exp\left(-\frac{\|P\mathbf{x}-\mathbf{y}\|^2}{2\sigma^2}\right)$.

Non-linear inverse problems

We consider two classic non-linear inverse problems, i.e. Fourier phase retrieval, and nonlinear deblur.

Fourier phase retrieval Fourier phase retrieval is a classical inverse problem in which the objective is to recover a signal from its Fourier magnitude (Fienup & Dainty, 1987). The challenge lies in the loss of the phase information during the measurement process, making the inverse problem highly ill-posed and non-unique (Chung et al., 2023). Let $\mathbf{x}$ represent the original signal, and the forward operator $A(\mathbf{x}) = |\mathcal{F}(\mathbf{x})|$ denotes the magnitude of the Fourier transform. The measurement $\mathbf{y}$ is the observed Fourier magnitude corrupted by noise of scale σ, so $\mathbf{y} = |\mathcal{F}(\mathbf{x})| + \mathcal{N}(0, \sigma^2\mathbf{I})$. The inverse problem is to infer the posterior distribution $p(\mathbf{x} \mid \mathbf{y})$. We use RTB to fine-tune a score-based prior $p_\theta(\mathbf{x})$ into an unbiased posterior $p_\theta(\mathbf{x})p(\mathbf{y} \mid \mathbf{x})$ with likelihood $p(\mathbf{y} \mid \mathbf{x}) \propto \exp\left(-\frac{\||\mathbf{y}-\mathcal{F}(\mathbf{x})|\|^2}{2\sigma^2}\right)$, for sample $\mathbf{x}$ and reference measurement $\mathbf{y}$, and where σ controls the temperature of the likelihood.

Nonlinear deblur For nonlinear deblur, the objective is to recover a clean image from its blurry observation. The forward model generally involves complex, nonlinear, transformations, such as temporal integration of sharp images through a nonlinear camera response function. We leverage the neural network-based forward model $A(\mathbf{x})$ as described by Nah et al. (2017). We can thus describe the measurement $\mathbf{y}$ as $\mathbf{y} = A(\mathbf{x}) + \mathcal{N}(0, \sigma^2\mathbf{I})$, where $\mathbf{x}$ represent the original sharp image, and the forward operator $A(\mathbf{x})$ encapsulate the nonlinear blurring process. The use RTB to fine-tune a posterior $p(\mathbf{x} \mid \mathbf{y})$, considering the likelihood $p(\mathbf{y} \mid \mathbf{x}) \propto \exp\left(-\frac{\|A(\mathbf{x})-\mathbf{y}\|^2}{2\sigma^2}\right)$.

Table 1: Results for linear and nonlinear inverse problems on pretrained standard diffusion models. We report the mean and standard error of each metric across MNIST and CIFAR-10 datasets.

Linear Inverse Problems												
Task →	Inpainting (box)						Inpainting (random)					
Dataset →	MNIST			CIFAR-10			MNIST			CIFAR-10		
Algorithm ↓ Metric →	$\mathbb{E}[\log r(\mathbf{x})]$ (↑)	$\log Z$ (↑)	LPIPS (↓)	$\mathbb{E}[\log r(\mathbf{x})]$ (↑)	$\log Z$ (↑)	LPIPS (↓)	$\mathbb{E}[\log r(\mathbf{x})]$ (↑)	$\log Z$ (↑)	LPIPS (↓)	$\mathbb{E}[\log r(\mathbf{x})]$ (↑)	$\log Z$ (↑)	LPIPS (↓)
DPS	-0.759 ±0.054	-2067.326 ±2.15e+02	0.107 ±0.011	-0.680 ±0.083	-51810.868 ±2.07e+03	0.083 ±0.009	-1.360 ±0.017	-2945.032 ±1.67e+02	0.136 ±0.010	-0.577 ±0.047	-29942.605 ±9.83e+02	0.052 ±0.005
FPS	-1.010 ±0.140	-1200.852 ±1.28e+02	0.141 ±0.010	-0.778 ±0.035	-44381.465 ±1.44e+03	0.107 ±0.010	-1.121 ±0.072	-2089.714 ±2.75e+02	0.169 ±0.010	-1.009 ±0.021	-42189.443 ±9.54e+02	0.077 ±0.007
FPS-SMC	-0.902 ±0.153	-1205.810 ±1.28e+02	0.128 ±0.010	-0.801 ±0.053	-42424.722 ±1.52e+03	0.108 ±0.011	-1.053 ±0.067	-1891.618 ±1.87e+02	0.163 ±0.010	-1.103 ±0.024	-40025.465 ±9.39e+02	0.077 ±0.007
RTB (ours)	-3.133 ±0.205	-18.122 ±1.49e+00	0.181 ±0.013	-11.022 ±0.670	-36.567 ±3.79e+00	0.493 ±0.034	-2.940 ±0.144	-19.189 ±1.35e+00	0.172 ±0.012	-12.152 ±1.34e+00	-25.276 ±2.72e+00	0.547 ±0.039
Nonlinear Inverse Problems												
Task →	Phase Retrieval						Nonlinear Deblur					
Dataset →	MNIST			CIFAR-10			MNIST			CIFAR-10		
Algorithm ↓ Metric →	$\mathbb{E}[\log r(\mathbf{x})]$ (↑)	$\log Z$ (↑)	LPIPS (↓)	$\mathbb{E}[\log r(\mathbf{x})]$ (↑)	$\log Z$ (↑)	LPIPS (↓)	$\mathbb{E}[\log r(\mathbf{x})]$ (↑)	$\log Z$ (↑)	LPIPS (↓)	$\mathbb{E}[\log r(\mathbf{x})]$ (↑)	$\log Z$ (↑)	LPIPS (↓)
DPS	-1.722 ±0.235	-1625.894 ±1.12e+02	0.184 ±0.023	-2.782 ±0.246	-20885.408 ±2.82e+03	0.560 ±0.020	-1.698 ±0.088	-1536.865 ±5.28e+01	0.145 ±0.009	-2.672 ±0.100	-33036.342 ±1.08e+03	0.191 ±0.012
FPS	-2.123 ±0.242	-1860.426 ±2.16e+02	0.212 ±0.021	-2.910 ±0.167	-43860.793 ±6.10e+03	0.567 ±0.020	-1.760 ±0.092	-1890.115 ±1.17e+02	0.150 ±0.010	-3.190 ±0.125	-60519.596 ±2.33e+03	0.218 ±0.013
FPS-SMC	-2.058 ±0.241	-1394.984 ±7.22e+01	0.205 ±0.021	-2.865 ±0.164	-36354.029 ±5.07e+03	0.566 ±0.021	-1.585 ±0.070	-1842.228 ±8.26e+01	0.120 ±0.009	-21.893 ±1.54e+00	-49.469 ±2.51e+00	0.654 ±0.010
RTB (ours)	-3.600 ±0.209	-17.986 ±1.26e+00	0.184 ±0.021	-9.284 ±0.439	-27.573 ±2.76e+00	0.566 ±0.033	-3.286 ±0.156	-15.573 ±1.30e+00	0.181 ±0.012	-6.964 ±0.289	-36.034 ±2.17e+00	0.440 ±0.026

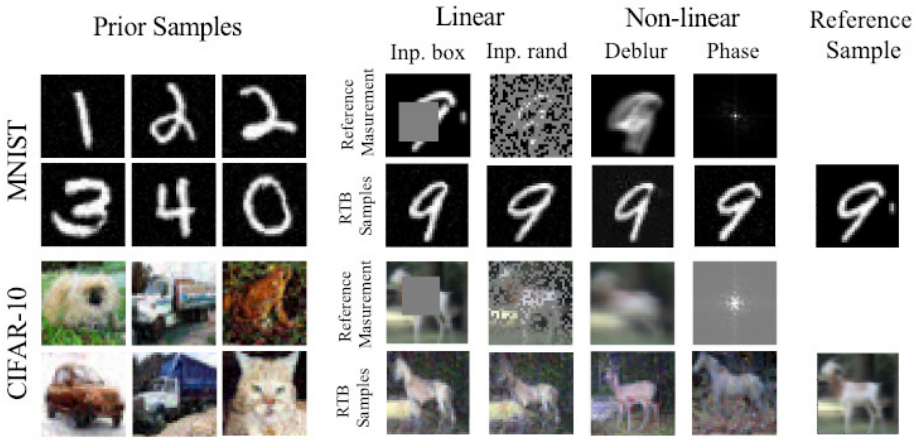

Figure 1: Samples from RTB fine-tuned diffusion posteriors.

Results

We consider diffusion score-based priors with MNIST and CIFAR-10, and re-implement several methods previously showing state-of-the art results in inverse problems in vision, including DPS Chung et al. (2023), FPS Song et al. (2023) and FPS-SMC Dou & Song (2024).

Unconditional Posteriors

In the first set of experiments we finetune unconditional diffusion priors into unconditional posteriors for a measurement $\mathbf{y}$. For each method and dataset we run experiments on inverse problems defined above, and report average performance over 10 randomly sampled measurements retrieved from the validation set of the respective datasets. We report in Table Table 1 the mean log reward ($\mathbb{E}[\log r(\mathbf{x})]$) of the generated samples, their LPIPS score and the log-partition function $\log Z$, computed with 5000 posterior samples, and averaged across 10 measurements.

We observe classifier guidance (CF)-based methods to achieve high $\mathbb{E}[\log r(\mathbf{x})]$ and LPIPS values at the expense of drifting from the true posterior to model (low $\log Z$ values). On the other hand, RTB reaches competitive rewards while maintaining significantly higher values of $\log Z$, thus getting closer to the true posterior.

Conditional Posteriors

In the second set of experiments, we fine-tune unconditional diffusion priors into conditional posteriors. We follow the formulation in Equation 7, and condition the generation of the posterior model by the corresponding measurement. For each method and dataset we run experiments on all inverse problems previously defined. We report in Table Table 1 the mean log reward ($\mathbb{E}[\log r(\mathbf{x})]$) of the generated samples, their FID score and the log-partition function $\log Z$, all computed with 10000 posterior samples and respective measurements.

Similarly to before, classifier guidance (CF)-based methods to achieve high $\mathbb{E}[\log r(\mathbf{x})]$ and low FID values at the expense of drifting from the true posterior to model (low $\log Z$ values). On the

Table 2: Conditional Diffusion results for linear and nonlinear inverse problems on pretrained standard diffusion models. We report the mean and standard error of each metric across MNIST and CIFAR-10 datasets.

Linear Inverse Problems												
Task →	Inpainting (box)						Inpainting (random)					
Dataset →	MNIST			CIFAR-10			MNIST			CIFAR-10		
Algorithm ↓ Metric →	$\mathbb{E}[\log r(\mathbf{x})]$ (↑)	$\log Z$ (↑)	FID (↓)	$\mathbb{E}[\log r(\mathbf{x})]$ (↑)	$\log Z$ (↑)	FID (↓)	$\mathbb{E}[\log r(\mathbf{x})]$ (↑)	$\log Z$ (↑)	FID (↓)	$\mathbb{E}[\log r(\mathbf{x})]$ (↑)	$\log Z$ (↑)	FID (↓)
DPS	−0.509 ±0.004	−484.746	0.450	−0.279 ±0.001	−15309.502	0.065	−0.383 ±0.002	−870.558	1.317	−0.220 ±0.001	−13771.698	0.253
DPS+CLA	−0.507 ±0.004	−550.389	0.446	−0.273 ±0.001	−14988.690	0.066	−0.379 ±0.002	−800.751	1.312	−0.220 ±0.001	−13287.473	0.253
FPS	−0.784 ±0.005	−1036.824	0.941	−0.519 ±0.000	−14281.504	0.104	−0.610 ±0.003	−844.831	1.391	−0.495 ±0.000	−12541.325	0.304
FPS-SMC	−0.671 ±0.005	−1031.968	0.478	−0.289 ±0.001	−12216.635	0.073	−0.470 ±0.004	−744.221	1.356	−0.267 ±0.001	−14835.040	0.295
RTB (ours)	−0.704 ±0.001	−655.195	1.515	−1.394 ±0.001	−2445.463	0.804	−1.048 ±0.001	−628.810	1.397	−1.976 ±0.002	−2459.556	0.774
RTB+DPS (ours)	−0.579 ±0.001	−801.660	1.532	−0.491 ±0.001	−7600.880	0.191	−0.358 ±0.001	−1722.438	1.485	−0.330 ±0.001	−9852.971	0.674
RTB+DPS+CLA (ours)	−0.527 ±0.001	−807.689	1.457	−0.350 ±0.001	−7704.646	0.482	−0.340 ±0.001	−1982.161	1.539	−0.314 ±0.001	−11071.914	0.681
Nonlinear Inverse Problems												
Task →	Phase Retrieval						Nonlinear Deblur					
Dataset →	MNIST			CIFAR-10			MNIST			CIFAR-10		
Algorithm ↓ Metric →	$\mathbb{E}[\log r(\mathbf{x})]$ (↑)	$\log Z$ (↑)	FID (↓)	$\mathbb{E}[\log r(\mathbf{x})]$ (↑)	$\log Z$ (↑)	FID (↓)	$\mathbb{E}[\log r(\mathbf{x})]$ (↑)	$\log Z$ (↑)	FID (↓)	$\mathbb{E}[\log r(\mathbf{x})]$ (↑)	$\log Z$ (↑)	FID (↓)
DPS	−1.249 ±0.009	−1630.394	1.237	−2.402 ±0.011	−12885.045	0.541	−1.675 ±0.004	−986.008	1.402	−2.214 ±0.005	−10682.691	0.442
DPS+CLA	−1.272 ±0.009	−1372.473	1.241	−2.414 ±0.011	−14362.470	0.531	−1.667 ±0.004	−780.577	1.413	−2.211 ±0.005	−11472.098	0.441
FPS	−1.715 ±0.011	−1029.060	1.424	−2.842 ±0.011	−12805.804	0.682	−1.754 ±0.005	−815.970	1.386	−2.353 ±0.006	−9779.317	0.495
FPS-SMC	−1.668 ±0.011	−1117.023	1.374	−2.796 ±0.011	−11583.854	0.657	−1.948 ±0.006	−324.646	1.479	−6.934 ±0.013	−276.732	0.849
RTB (ours)	−2.860 ±0.010	8960.105	1.440	−4.551 ±0.010	−968.128	1.642	−2.406 ±0.004	−204.788	1.464	−2.827 ±0.005	−1912.678	1.018
RTB+DPS (ours)	−6.705 ±0.020	−58.640	1.548	−2.622 ±0.010	−6290.827	1.305	−1.896 ±0.004	−638.202	1.413	−2.253 ±0.005	−4747.092	0.873
RTB+DPS+CLA (ours)	−2.906 ±0.009	−1321.135	1.382	−2.354 ±0.009	−11500.541	1.087	−1.849 ±0.003	−791.639	1.305	−2.701 ±0.004	−3525.784	1.635

Table 3: Comparison between RTB and CLA for the lensing problem. We compare mean likelihood $\log p(\mathbf{y} \mid \mathbf{x})$, and lower bound on the log-partition function $\log Z$. Metrics are computed with 50 posterior samples, and averaged across 3 runs.

Algorithm	$\log p(\mathbf{y} \mid \mathbf{x})(\uparrow)$	$\log Z(\uparrow)$
CLA	−8216.02	−12514.67
RTB	−8367.9	−8676.85

other hand, RTB reaches competitive rewards while maintaining significantly higher values of $\log Z$. Importantly, perform experiments whereby DPS and DPS + CLA drifts are added to the RTB objective following a similar formulation to Equation 8. We obverse increased rewards when blending these methods, mitigating the original pitifals of exceptionally low $\log Z$ values.

4.1 Gravitational Lensing

In general relativity, light travels along the shortest paths in a spacetime curved by the mass of objects (Einstein, 1916), with greater masses inducing larger curvature. An interesting inverse problem involves the inference of the undistorted images of distant astronomical sources whose images have been gravitationally lensed by the gravity of intervening structures (Einstein, 1911). In the case of strong lensing, for example when the background source and the foreground lens are both almost perfectly aligned galaxies, multiple images of the background source are formed and heavy distortions such as rings or arcs are induced. In this problem, the parameters of interest are the undistorted pixel values of the background source x, given an observed distorted image y. This problem is then linear, since the distortions can be encoded in a lensing matrix A (which we assume to be known): $\mathbf{y} = A\mathbf{x} + \epsilon$, with $\epsilon \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I})$ a small Gaussian observational noise. The Bayesian inverse problem of interest is the inference of the posterior distribution over source images given the lensed observation, that is $p(\mathbf{x} \mid \mathbf{y})$. We use the Probes dataset Stone et al. (2022), containing telescope images of undistorted galaxies in the local Universe, to train a score based prior over source images $p_\theta(\mathbf{x})$. Drawing unbiased samples from the posterior $p(\mathbf{x} \mid \mathbf{y}) \propto p_\theta(\mathbf{x})\mathcal{N}(\mathbf{y}; A\mathbf{x}, \sigma^2\mathbf{I})$ is quite difficult, especially if the distribution is very peaky with small σ. RTB allows us to train an asymptotically unbiased posterior sampler.

We use RTB to finetune the prior model to this posterior, and compare against a biased training-free diffusion posterior inference baseline (Adam et al., 2022) that previous work has used for this gravitational lensing inverse problem. This method uses a convolved likelihood approximation (CLA) $p_t(\mathbf{y} \mid \mathbf{x}) \approx \mathcal{N}(\mathbf{y} \mid A\mathbf{x}, (\sigma^2 + \sigma^2(t))\mathbf{I})$. For RTB we use 300 diffusion steps for sampling, but for CLA we require 2000 steps to obtain reasonable samples. We fix $\sigma = 0.05$ for our experiments. We report metrics comparing these approaches in Table 3, and illustrative samples in Fig. 2. We found RTB to be a bit unstable while training, likely because of the peaky reward function. About 30% of runs, the policy diverged irrecoverably. For the sake of highlighting the advantages of unbiased posterior sampling, the metrics computed in Table 3 excluded diverged runs. For this problem, we only used on-policy samples, and we expect off-policy tricks to help stabilize training.

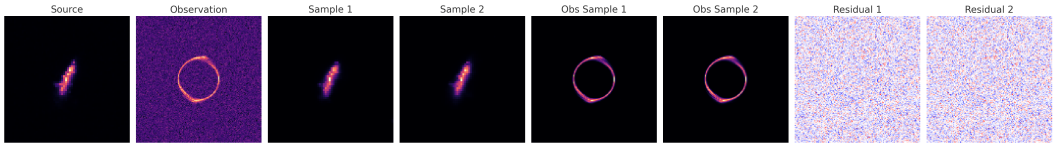

Figure 2: Lensing problem RTB samples. Plotted are ground truth source, observation, samples from RTB posterior, their mean observation after forward model (without observation white noise), and the residual between posterior mean observations and ground truth observation.

5 Discussion

Our study has demonstrated the potential of relative trajectory balance (RTB) as an effective framework for solving Bayesian inverse problems with diffusion models. By systematically evaluating RTB across a variety of inverse problems, including vision-based reconstructions and gravitational lensing, we provide comprehensive empirical evidence supporting its scalability and flexibility.

5.1 Comparison with Existing Methods

RTB exhibits several advantages over existing approaches such as diffusion posterior sampling (DPS) (Chung et al., 2023) and function-space posterior sampling (FPS) (Song et al., 2023). While these methods provide practical solutions for Bayesian inverse problems, they often tend to shift the distribution of samples away from the true posterior, leading to artificially high likelihood values at the cost of reduced diversity. As an asymptotically unbiased objective, RTB mitigates these issues while enabling flexible off-policy training strategies. Our results indicate that RTB achieves competitive likelihood scores while preserving a closer match to the true posterior, as evidenced by higher log-partition function ($\log Z$) values.

5.2 Extensions and Future Directions

Despite its strong empirical performance, RTB presents certain challenges that warrant further exploration.

Off-Policy Stabilization: While we leveraged replay buffers to prevent mode collapse, stability during training remains a challenge in some settings. Future work could explore improved adaptive strategies for selecting trajectories and incorporating uncertainty estimation during training.

Integration with Other State-of-the-Art Methods: Our experiments have shown that RTB can benefit from hybrid approaches, such as combining DPS-based constraints with RTB optimization. Further research could explore novel ways to blend RTB with other advanced generative modeling techniques to enhance robustness and scalability.

Application to Broader Scientific Domains: We have primarily focused on inverse problems in vision and gravitational lensing. However, RTB has the potential to generalize to other scientific fields, such as climate modeling, computational biology, and medical imaging. Future studies should investigate its applicability in these domains, particularly in handling multimodal constraints and heterogeneous data.

6 Conclusion

In this work, we demonstrated the effectiveness of off-policy RL fine-tuning via the RTB objective for asymptotically unbiased posterior inference for diffusion models. We applied RTB to challenging linear and non-linear Bayesian inverse problems, demonstrating its effectiveness in inverse imaging and gravitational lensing. The ability to seamlessly integrate off-policy training and other classifier guidance techniques, as well as the extension of the objective for conditional posteriors, allows for RTB to be leveraged in more complex and critical domains at scale. Extending RTB to other important scientific applications, such as inverse protein design would be a promising direction for future research.

ACKNOWLEDGMENTS

The authors acknowledge funding from CIFAR, NSERC, IVADO, UNIQUE, FACS Acuité, NRC AI4Discovery, Samsung, and Recursion.

The research was enabled in part by computational resources provided by the Digital Research Alliance of Canada (`https://alliancecan.ca`), Mila (`https://mila.quebec`), and NVIDIA.

REFERENCES

Alexandre Adam, Adam Coogan, Nikolay Malkin, Ronan Legin, Laurence Perreault-Levasseur, Yashar Hezaveh, and Yoshua Bengio. Posterior samples of source galaxies in strong gravitational lenses with score-based priors. *arXiv preprint arXiv:2211.03812*, 2022.

Jacob Austin, Daniel D Johnson, Jonathan Ho, Daniel Tarlow, and Rianne Van Den Berg. Structured denoising diffusion models in discrete state-spaces. *Neural Information Processing Systems (NeurIPS)*, 2021.

Julian Besag. Comments on "representations of knowledge in complex systems" by u. grenander and mi miller. *J. Roy. Statist. Soc. Ser. B*, 56(591-592):4, 1994.

Kevin Black, Michael Janner, Yilun Du, Ilya Kostrikov, and Sergey Levine. Training diffusion models with reinforcement learning. *International Conference on Learning Representations (ICLR)*, 2024.

Gabriel Cardoso, Yazid Janati el idrissi, Sylvain Le Corff, and Eric Moulines. Monte carlo guided denoising diffusion models for bayesian linear inverse problems. *International Conference on Learning Representations (ICLR)*, 2024.

Hyungjin Chung, Jeongsol Kim, Michael Thompson Mccann, Marc Louis Klasky, and Jong Chul Ye. Diffusion posterior sampling for general noisy inverse problems. *International Conference on Learning Representations (ICLR)*, 2023.

Masoumeh Dashti and Andrew M Stuart. The bayesian approach to inverse problems. *arXiv preprint arXiv:1302.6989*, 2013.

Prafulla Dhariwal and Alexander Nichol. Diffusion models beat GANs on image synthesis. *Neural Information Processing Systems (NeurIPS)*, 2021.

Sander Dieleman, Laurent Sartran, Arman Roshannai, Nikolay Savinov, Yaroslav Ganin, Pierre H Richemond, Arnaud Doucet, Robin Strudel, Chris Dyer, Conor Durkan, et al. Continuous diffusion for categorical data. *arXiv preprint arXiv:2211.15089*, 2022.

Zehao Dou and Yang Song. Diffusion posterior sampling for linear inverse problem solving: A filtering perspective. *International Conference on Learning Representations (ICLR)*, 2024.

Simon Duane, Anthony D Kennedy, Brian J Pendleton, and Duncan Roweth. Hybrid monte carlo. *Physics letters B*, 195(2):216–222, 1987.

A Einstein. Graviton mass and inertia mass. *Ann Physik*, 35:898, 1911.

A. Einstein. The foundation of the general theory of relativity. 1916.

Ying Fan, Olivia Watkins, Yuqing Du, Hao Liu, Moonkyung Ryu, Craig Boutilier, Pieter Abbeel, Mohammad Ghavamzadeh, Kangwook Lee, and Kimin Lee. Reinforcement learning for fine-tuning text-to-image diffusion models. *Neural Information Processing Systems (NeurIPS)*, 2023.

C Fienup and J Dainty. Phase retrieval and image reconstruction for astronomy. *Image recovery: theory and application*, 231:275, 1987.

Alexandros Graikos, Nikolay Malkin, Nebojsa Jojic, and Dimitris Samaras. Diffusion models as plug-and-play priors. *Neural Information Processing Systems (NeurIPS)*, 2022.

Ishaan Gulrajani and Tatsunori B Hashimoto. Likelihood-based diffusion language models. *Neural Information Processing Systems (NeurIPS)*, 2023.

Xiaochuang Han, Sachin Kumar, and Yulia Tsvetkov. SSD-LM: Semi-autoregressive simplex-based diffusion language model for text generation and modular control. *Association for Computational Linguistics (ACL)*, 2023.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Neural Information Processing Systems (NeurIPS)*, 2020.

Edward J. Hu, Moksh Jain, Eric Elmoznino, Younesse Kaddar, Guillaume Lajoie, Yoshua Bengio, and Nikolay Malkin. Amortizing intractable inference in large language models. *International Conference on Learning Representations (ICLR)*, 2024.

Jérôme Idier. *Bayesian approach to inverse problems*. John Wiley & Sons, 2013.

Michael Janner, Yilun Du, Joshua Tenenbaum, and Sergey Levine. Planning with diffusion for flexible behavior synthesis. *International Conference on Machine Learning (ICML)*, 2022.

Zahra Kadkhodaie and Eero P. Simoncelli. Solving linear inverse problems using the prior implicit in a denoiser. *Neural Information Processing Systems (NeurIPS)*, 2021.

Jari Kaipio and Erkki Somersalo. *Statistical and computational inverse problems*, volume 160. Springer Science & Business Media, 2006.

Bingyi Kang, Xiao Ma, Chao Du, Tianyu Pang, and Shuicheng Yan. Efficient diffusion policies for offline reinforcement learning. *Neural Information Processing Systems (NeurIPS)*, 2024.

Bahjat Kawar, Gregory Vaksman, and Michael Elad. SNIPS: Solving noisy inverse problems stochastically. *Neural Information Processing Systems (NeurIPS)*, 2021.

Salem Lahlou, Tristan Deleu, Pablo Lemos, Dinghuai Zhang, Alexandra Volokhova, Alex Hernández-Garcıa, Léna Néhale Ezzine, Yoshua Bengio, and Nikolay Malkin. A theory of continuous generative flow networks. *International Conference on Machine Learning (ICML)*, 2023.

Jonas Latz. On the well-posedness of bayesian inverse problems. *SIAM/ASA Journal on Uncertainty Quantification*, 8(1):451–482, 2020.

Xiang Li, John Thickstun, Ishaan Gulrajani, Percy S Liang, and Tatsunori B Hashimoto. Diffusion-LM improves controllable text generation. *Neural Information Processing Systems (NeurIPS)*, 2022.

Aaron Lou, Chenlin Meng, and Stefano Ermon. Discrete diffusion language modeling by estimating the ratios of the data distribution. *arXiv preprint arXiv:2310.16834*, 2023.

David JC MacKay. *Information theory, inference and learning algorithms*. Cambridge university press, 2003.

Nikolay Malkin, Moksh Jain, Emmanuel Bengio, Chen Sun, and Yoshua Bengio. Trajectory balance: Improved credit assignment in GFlowNets. *Neural Information Processing Systems (NeurIPS)*, 2022.

Morteza Mardani, Jiaming Song, Jan Kautz, and Arash Vahdat. A variational perspective on solving inverse problems with diffusion models. *International Conference on Learning Representations (ICLR)*, 2024.

Seungjun Nah, Tae Hyun Kim, and Kyoung Mu Lee. Deep multi-scale convolutional neural network for dynamic scene deblurring. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3883–3891, 2017.

Radford M Neal. Mcmc using hamiltonian dynamics. *arXiv preprint arXiv:1206.1901*, 2012.

Alex Nichol and Prafulla Dhariwal. Improved denoising diffusion probabili1stic models. *International Conference on Machine Learning (ICML)*, 2021.

Lorenz Richter, Ayman Boustati, Nikolas Nüsken, Francisco J. R. Ruiz, and Ömer Deniz Akyildiz. VarGrad: A low-variance gradient estimator for variational inference. *Neural Information Processing Systems (NeurIPS)*, 2020.

Gareth O Roberts and Jeffrey S Rosenthal. Optimal scaling of discrete approximations to langevin diffusions. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 60(1): 255–268, 1998.

Gareth O Roberts and Richard L Tweedie. Exponential convergence of Langevin distributions and their discrete approximations. *Bernoulli*, pp. 341–363, 1996.

Robin Rombach, A. Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. *Computer Vision and Pattern Recognition (CVPR)*, 2022.

Marcin Sendera, Minsu Kim, Sarthak Mittal, Pablo Lemos, Luca Scimeca, Jarrid Rector-Brooks, Alexandre Adam, Yoshua Bengio, and Nikolay Malkin. On diffusion models for amortized inference: Benchmarking and improving stochastic control and sampling. *arXiv preprint arXiv:2402.05098*, 2024a.

Marcin Sendera, Minsu Kim, Sarthak Mittal, Pablo Lemos, Luca Scimeca, Jarrid Rector-Brooks, Alexandre Adam, Yoshua Bengio, and Nikolay Malkin. Improved off-policy training of diffusion samplers, 2024b. URL `https://arxiv.org/abs/2402.05098`.

Jascha Sohl-Dickstein, Eric A. Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. *International Conference on Machine Learning (ICML)*, 2015.

Jiaming Song, Qinsheng Zhang, Hongxu Yin, Morteza Mardani, Ming-Yu Liu, Jan Kautz, Yongxin Chen, and Arash Vahdat. Loss-guided diffusion models for plug-and-play controllable generation. *International Conference on Maching Learning (ICML)*, 2023.

Yang Song, Conor Durkan, Iain Murray, and Stefano Ermon. Maximum likelihood training of score-based diffusion models. *Neural Information Processing Systems (NeurIPS)*, 2021a.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *International Conference on Learning Representations (ICLR)*, 2021b.

Yang Song, Liyue Shen, Lei Xing, and Stefano Ermon. Solving inverse problems in medical imaging with score-based generative models. *International Conference on Learning Representations (ICLR)*, 2022.

Connor Stone, Stéphane Courteau, Nikhil Arora, Matthew Frosst, and Thomas H. Jarrett. PROBES. I. A Compendium of Deep Rotation Curves and Matched Multiband Photometry. *apjs*, 262(1):33, September 2022. doi: 10.3847/1538-4365/ac83ad.

Siddarth Venkatraman, Moksh Jain, Luca Scimeca, Minsu Kim, Marcin Sendera, Mohsin Hasan, Luke Rowe, Sarthak Mittal, Pablo Lemos, Emmanuel Bengio, et al. Amortizing intractable inference in diffusion models for vision, language, and control. *arXiv preprint arXiv:2405.20971*, 2024.

Pascal Vincent. A connection between score matching and denoising autoencoders. *Neural computation*, 23(7):1661–1674, 2011.

Zhendong Wang, Jonathan J Hunt, and Mingyuan Zhou. Diffusion policies as an expressive policy class for offline reinforcement learning. *International Conference on Learning Representations (ICLR)*, 2023.

Dinghuai Zhang, Ricky T. Q. Chen, Nikolay Malkin, and Yoshua Bengio. Unifying generative models with GFlowNets and beyond. *arXiv preprint arXiv:2209.02606*, 2023.

Qinsheng Zhang and Yongxin Chen. Path integral sampler: a stochastic control approach for sampling. *arXiv preprint arXiv:2111.15141*, 2021.

A BACKGROUND AND SETUP

Diffusion model training as divergence minimization. Diffusion models parametrize the drift $u_t(\mathbf{x}_t)$ in (Equation 2) as a neural network $u(\mathbf{x}_t, t; \theta)$ with parameters θ and taking $\mathbf{x}_t$ and t as input. We denote the distributions over trajectories induced by (Equation 3, Equation 4) by p_θ to show their dependence on the parameter.

In the most common setting, diffusion models are trained to maximize the likelihood of a dataset. In the notation above, this corresponds to assuming $q(\mathbf{x}_1)$ is fixed to an empirical measure (with the points of a training dataset $\mathcal{D}$ assumed to be i.i.d. samples from $q(\mathbf{x}_1)$). Training minimizes with respect to θ the divergence between the processes q and p_θ:

$$\begin{aligned} & D_{\mathrm{KL}}(q(\mathbf{x}_0, \mathbf{x}_{\Delta t}, \ldots, \mathbf{x}_1) \,\|\, p_\theta(\mathbf{x}_0, \mathbf{x}_{\Delta t}, \ldots, \mathbf{x}_1)) \\ & = D_{\mathrm{KL}}(q(\mathbf{x}_1) \,\|\, p_\theta(\mathbf{x}_1)) +_{\mathbf{x}_1 \sim q(\mathbf{x}_1)} D_{\mathrm{KL}}(q(\mathbf{x}_0, \mathbf{x}_{\Delta t}, \ldots, \mathbf{x}_{1-\Delta t} \mid \mathbf{x}_1) \,\|\, p_\theta(\mathbf{x}_0, \mathbf{x}_{\Delta t}, \ldots, \mathbf{x}_{1-\Delta t} \mid \mathbf{x}_1)) \\ & \geq D_{\mathrm{KL}}(q(\mathbf{x}_1) \,\|\, p_\theta(\mathbf{x}_1)) =_{\mathbf{x}_1 \sim q(\mathbf{x}_1)} [-\log p_\theta(\mathbf{x}_1)] + \text{const.} \end{aligned} \tag{9}$$

where the inequality – an instance of the data processing inequality for the KL divergence – shows that minimizing the divergence between distributions over trajectories is equivalent to maximizing a lower bound on the data log-likelihood under the model p_θ.

As shown in Song et al. (2021a), minimization of the KL in (Equation 9) is essentially equivalent to the traditional approach to training diffusion models via denoising score matching Vincent (2011); Sohl-Dickstein et al. (2015); Ho et al. (2020). Such training exploits that for typical choices of the noising process q, the optimal $u_t(\mathbf{x}_t)$ can be expressed in terms of the Stein score of $q(\mathbf{x}_1)$ convolved with a Gaussian, allowing an efficient stochastic regression objective for u_t. For full generality of our exposition for arbitrary iterative generative processes, we prefer to think of (Equation 9) as the primal objective and denoising score matching as an efficient means of minimizing it.

A.1 DIFFUSION GFLOWNETS

Generative Flow Networks (GFlowNets) aim to sample from a distribution proportional to an unnormalized density, $p(x) \propto r(x)$, through a sequential decision-making process. Diffusion GFlowNets are a family of GFlowNets that model discretized reverse stochastic differential equation (SDE) trajectories,

$$\tau = (x_0 \to x_{\Delta t} \to x_{2\Delta t} \to \ldots \to x_1)\,, \tag{10}$$

where $x_0 = (\mathbf{0}, t = 0)$ is the initial state. Here, $\Delta t = \frac{1}{T}$, where T is the number of discrete time steps.

The forward policy $P_F(x_{t+\Delta t} \mid x_t; \theta)$ is defined to model the mean of a Gaussian kernel, expressed as:

$$P_F(x_{t+\Delta t} \mid x_t; \theta) = \mathcal{N}\left(x_{t+\Delta t};\, x_t + u(x_t, t; \theta)\Delta t,\, \sigma(t)^2 \Delta t\, \mathbb{I}\right). \tag{11}$$

Here, $u(x_t, t; \theta)$ is the learnable score with parameter θ, and $\sigma(t)$ represents the standard deviation at time t. The term $\mathbb{I}$ denotes the identity matrix, ensuring that the covariance matrix is isotropic.

The backward policy $P_B(x_{t-\Delta t} \mid x_t)$ is defined as a discretized Brownian bridge with a noise rate σ:

$$P_B(x_{t-\Delta t} \mid x_t) = \mathcal{N}\left(x_{t-\Delta t};\, \frac{t-\Delta t}{t} x_t,\, \frac{t-\Delta t}{t} \sigma^2 \Delta t\, \mathbb{I}\right). \tag{12}$$

The forward and backward policies defined over complete trajectories are expressed as:

$$P_F(\tau; \theta) = \prod_{i=0}^{T-1} P_F\left(x_{(i+1)\Delta t} \mid x_{i\Delta t}; \theta\right), \quad P_B(\tau; x_1) = \prod_{i=0}^{T-1} P_B\left(x_{i\Delta t} \mid x_{(i+1)\Delta t}\right). \tag{13}$$

$$\mathcal{L}_{\mathrm{TB}}(\tau; \theta, f(x_1)) = \log\left(\frac{Z_\theta\, P_F(\tau; \theta)}{f(x_1)\, r_{\mathrm{target}}(x_1)\, P_B(\tau; x_1)}\right), \tag{14}$$

where Z_θ is a learnable constant representing the partition function, $f(x_1)$ is a weighting function for the density, and $r_{\mathrm{target}}(x_1)$ is the accessible unnormalized true density.

By ensuring that $\mathcal{L}_{\mathrm{TB}}(\tau; \theta, f(x_1)) = 0$ for all trajectories τ, we guarantee an optimal amortized sampler over the weighted distribution. This condition ensures that the distribution satisfies $p(x_1) \propto f(x_1)\, r(x_1)$. Specifically, when the weighting function is set to $f(x_1) = 1$, the target distribution simplifies to $p(x_1) \propto r(x_1)$.

B Relative trajectory balance (RTB)

B.1 Method

In diffusion GFlowNets, the Trajectory Balance (TB) objective is employed to perform amortized inference, aiming to approximate the distribution $p(x_1) \propto r(x_1)$. In contrast, Relative Trajectory Balance (RTB) focuses on amortized posterior inference over the prior distribution. Specifically, RTB defines the posterior as

$$p^{\text{post}}(x_1; \theta) \propto p^{\text{prior}}(x_1)\, r(x_1),$$

where $p^{\text{prior}}(x_1)$ is the prior distribution (e.g., a learned diffusion model) and $r(x_1)$ represents the likelihood. The product $p^{\text{prior}}(x_1)\, r(x_1)$ serves as the target unnormalized density for amortized inference of $p^{\text{post}}(x_1; \theta)$.

RTB is TB that has weighted reward $p^{\text{prior}}(x_1)\, r(x_1)$, where the weight is prior distribution $p^{\text{prior}}(x_1)$:

$$\begin{aligned}
\mathcal{L}_{\text{TB}}(\tau; \theta, p^{\text{prior}}(x_1)) &= \log \left(\frac{Z_\theta\, P_F^{\text{post}}(\tau; \theta)}{p^{\text{prior}}(x_1)\, r(x_1)\, P_B^{\text{post}}(\tau; x_1)} \right) && (15)\\
&= \log \left(\frac{Z_\theta\, P_F^{\text{post}}(\tau; \theta) \cancel{P_B^{\text{prior}}(\tau; x_1)}}{r(x_1)\, P_F^{\text{prior}}(\tau)\, \cancel{P_B^{\text{post}}(\tau; x_1)}} \right) && (16)\\
&= \log \left(\frac{Z_\theta\, P_F^{\text{post}}(\tau; \theta)}{r(x_1)\, P_F^{\text{prior}}(\tau)} \right) && (17)\\
&= \mathcal{L}_{\text{RTB}}(\tau; \theta). && (18)
\end{aligned}$$

The cancellation arises from the fact that $P_B^{\text{post}}(\tau; x_1) = P_B^{\text{prior}}(\tau; x_1)$, since we assumed the backward policy to be a fixed Brownian bridge with σ noise.

C Training, parametrization, and conditioning

Stabilizing the loss. We propose two simple design choices for stabilizing RTB training. First, the loss in (6) can be replaced by the empirical *variance* over a minibatch of the quantity inside the square, which removes dependence on $\log Z_\phi$ and is especially useful in conditional settings, consistent with the findings of Sendera et al. (2024a). This amounts to a relative variant of the VarGrad objective (Richter et al., 2020). Second, we employ loss clipping: to reduce sensitivity to an imperfectly fit prior model, we do not perform updates on trajectories where the loss is close to 0.

C.1 Off-policy adaptation techniques

Similar to methods used in GFlowNets, RTB benefits from the powerful advantages of off-policy learning. This flexibility allows us to choose any behavior policy over trajectories τ, denoted as $P(\tau)$, independent of the current diffusion sampling process. To leverage this advantage, we employ three off-policy techniques to enhance the performance of RTB for posterior inference.

Backtracking exploration with replay buffer

Using a replay buffer $\mathcal{D} = \{(x_1, r(x_1))\}$ with a prioritizing distribution $P(x_1; \mathcal{D})$ can prevent catastrophic forgetting of the sampler, such as mode dropping. By leveraging the off-policy property of RTB, we utilize a behavior policy defined as

$$P_\beta(\tau) = P_B(\tau \mid x_1)\, P(x_1; \mathcal{D}),$$

where we train the RTB objective over $\tau \sim P_\beta(\tau)$.