STRICTLY CONSTRAINED GENERATIVE MODELING VIA SPLIT AUGMENTED LANGEVIN SAMPLING

Anonymous authors
Paper under double-blind review

ABSTRACT

Deep generative models hold great promise for representing complex physical systems, but their deployment is currently limited by the lack of guarantees on the physical plausibility of the generated outputs. Ensuring that known physical constraints are enforced is therefore critical when applying generative models to scientific and engineering problems. We address this limitation by developing a principled framework for sampling from a target distribution while rigorously satisfying physical constraints. Leveraging the variational formulation of Langevin dynamics, we propose Split Augmented Langevin (SAL), a novel primal-dual sampling algorithm that enforces constraints progressively through variable splitting, with convergence guarantees. While the method is developed theoretically for Langevin dynamics, we demonstrate its effective applicability to diffusion models. In particular, we use constrained diffusion models to generate physical fields satisfying energy and mass conservation laws. We apply our method to diffusion-based data assimilation on a complex physical system, where enforcing physical constraints substantially improves both forecast accuracy and the preservation of critical conserved quantities. We also demonstrate the potential of SAL for challenging non-convex feasibility problems in optimal control.

1 INTRODUCTION

Generative deep learning methods have recently emerged as powerful tools to model and sample from complex data distributions, with successful applications in image synthesis (Ho et al., 2020), protein and material design (Corso et al., 2023), and probabilistic weather forecasting (Price et al., 2025). By learning a stochastic process from a training dataset, these models can generate arbitrarily many plausible samples conditioned on partial information. They are particularly useful in the physical sciences, where data is often scarce and multiple states may be consistent with available observations (Epstein & Fleming, 1971; Nathaniel & Gentine, 2025). While perceptual applications mainly aim for plausibility, scientific and engineering problems require samples that obey strict physical or structural constraints, such as conservation laws or system dynamics (Kashinath et al., 2021). In such cases, approximate resemblance is not enough: generated samples must obey the governing physical principles. This requirement becomes even more critical when generative models are used out of distribution or in an autoregressive fashion, where small violations can accumulate and severely degrade long-term accuracy (Pedersen et al., 2025). Developing constrained sampling methods applicable to pre-trained generative models in a zero-shot scenario (*i.e.* without additional training) is therefore crucial.

Modern generative models, including energy-based, score-based, and diffusion models (Du & Mordatch, 2019; Song et al., 2020), typically rely on Langevin dynamics, where noisy gradient steps push the samples toward high-likelihood regions. Enforcing mathematical constraints during Langevin sampling remains a challenging problem. A natural idea is to project each iterate onto the constraint set, leading to projected Langevin dynamics (Bubeck et al., 2015; Durmus et al., 2019; Christopher et al., 2024). While these methods offer theoretical guarantees in convex settings, they tend to perform poorly when applied to non-convex constraints, which are common in physical systems. In such cases, strict projections can cause the dynamics to become trapped in limited regions of the constraint set, hindering exploration and introducing significant sampling bias. Other approaches rather use a soft constraint penalty functions such as the barrier method (Fishman et al., 2023) and diffusion guidance (Ho & Salimans, 2022; Meunier et al., 2025), requiring a

differentiable constraint model. These methods encourage but do not enforce constraints, which is insufficient when strict satisfaction is crucial. To our knowledge, no existing approach achieves both strict constraint satisfaction and unbiased exploration.

Contributions Inspired by the variational formulation of Langevin dynamics and primal-dual optimization, we propose a novel sampling algorithm that bridges the gap between complex generative modeling and constrained sampling, called Split Augmented Langevin (SAL). Our method enforces hard constraints while preserving the exploration capability of Langevin dynamics. It ensures strict constraint satisfaction and benefits from convergence guarantees via duality analysis. We show that our approach generalizes to deep generative modeling and diffusion models. We demonstrate the effectiveness of SAL on complex physically-constrained sampling tasks, including data assimilation problems where maintaining physical invariants is key to reliable forecasting, and on non-convex feasibility problems in optimal control.

2 Problem formulation of constrained Langevin sampling

In this section, we provide a mathematical formulation of constrained sampling: given a generative model and a constraint set, our goal is to generate samples from the conditional distribution supported on the constraint set. Such constrained distributions arise in many applications where samples must strictly satisfy known physical laws. We adopt the framework of the Langevin Monte Carlo algorithm (Rossky et al., 1978), a foundation of modern generative modeling frameworks. The application to deep generative models is discussed in Section 4.4.

Langevin Monte Carlo Consider a target distribution with density $p(x) = \mathrm{e}^{-f(x)}/Z$ on $\mathbb{R}^d$, where $f(x)$ is a differentiable potential. Markov chain Monte Carlo methods design iterative algorithms producing samples (x_t) whose distribution q_t converges to p. Among them, the Langevin Monte Carlo algorithm plays a central role. It requires access to the gradient of the potential $\nabla f(x)$, also called the score function (Hyvärinen & Dayan, 2005), and performs noisy gradient descent updates

$$x_{t+1} = x_t - \tau \nabla f(x_t) + \sqrt{2\tau} w_t, \qquad w_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, I_d), \tag{2.1}$$

where τ is the step size. Under standard assumptions, the chain converges to p (Durmus et al., 2019).

Constrained target distribution We now consider the case where the samples are known to satisfy hard constraints at sampling time, in the form of a bounded measurable set $\mathcal{C} \subset \mathbb{R}^d$, which models prior information such as physical conservation laws. The conditional density supported on $\mathcal{C}$ is

$$p_{\mathcal{C}}(x) := \frac{1}{Z_{\mathcal{C}}} \mathrm{e}^{-f(x)} \mathbb{1}_{\mathcal{C}}(x), \quad \forall x \in \mathbb{R}^d, \tag{2.2}$$

with $\mathbb{1}_{\mathcal{C}}$ the indicator function of $\mathcal{C}$ and $Z_{\mathcal{C}}$ is a normalizing constant. Note that the conditional distribution (2.2) can be rewritten using a modified potential: $p_{\mathcal{C}}(x) := \mathrm{e}^{-f_{\mathcal{C}}(x)}/Z_{\mathcal{C}}$, with the constrained potential $f_{\mathcal{C}}(x) := f(x) + \chi_{\mathcal{C}}(x)$, defined with the characteristic function of $\mathcal{C}$

$$\chi_{\mathcal{C}}(x) := \begin{cases} 0 & \text{if} \quad x \in \mathcal{C}, \\ +\infty & \text{otherwise.} \end{cases} \tag{2.3}$$

We do not make any assumption on the constraint set $\mathcal{C}$, except that it is bounded and that $p_{\mathcal{C}}$ is well-defined. Next, we provide examples of such constraints that may occur in physical applications.

Example 2.1 [Physical constraints] When x describes a discretized physical field, conservation of energy E can often be expressed as the non-convex set $\mathcal{C} = \{x \in \mathbb{R}^d \mid \|x\|_2^2 = E\}$, while mass conservation corresponds to $\mathcal{C} = \{x \in \mathbb{R}^d \mid \sum_i x_i = M\}$ for a prescribed mass M.

Objective Our objective is to design a sampling algorithm that produces samples distributed according to $p_{\mathcal{C}}$ for any constraint set $\mathcal{C}$. It should use only access to the score function $\nabla f(x)$ of the unconstrained density, and mathematical operations related to $\mathcal{C}$ such as constraint functions or a projection operator $P_{\mathcal{C}}$ onto $\mathcal{C}$. The method should operate in a "zero-shot" scenario, requiring no retraining or additional data.

Example 2.2 [Projected Langevin] A natural idea to enforce hard constraints is to project each unconstrained update (2.1) onto $\mathcal{C}$, leading to

$$x_{t+1} = P_{\mathcal{C}}(x_t - \tau \nabla f(x_t) + \sqrt{2\tau} w_t), \qquad w_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, I_d). \tag{2.4}$$

This projected Langevin algorithm, and its extension to diffusion models, enjoy strong theoretical guarantees when $\mathcal{C}$ is convex and p is log-concave (Bubeck et al., 2015). But with non-convex constraints, repeated projection can trap the dynamics in small feasible regions, biasing exploration (Barber & Ha, 2018; Ahn & Chewi, 2021). This motivates the need for sampling methods that enforce constraints more gradually. More details on projected Langevin and its connection to proximal methods can be found in Appendix A.

Example 2.3 [Soft penalty methods] Constraints can also be enforced softly by adding a differentiable cost $c(x) \geq 0$ to the potential, penalizing samples far from $\mathcal{C}$, with a tunable coefficient $\lambda \in \mathbb{R}$:

$$x_{t+1} = x_t - \tau(\nabla f(x_t) + \lambda \nabla c(x_t)) + \sqrt{2\tau} w_t, \qquad w_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, I_d). \tag{2.5}$$

This corresponds to guidance in diffusion models (Ho & Salimans, 2022; Huang et al., 2024b). The cost function $c(x)$ corresponds to a negative log-likelihood centered on the constraint set. Such methods encourage constraint satisfaction but do not guarantee it, as violations are only smoothly penalized.

Evaluation Assessing the performance of constrained sampling algorithms is challenging, as $p_{\mathcal{C}}$ is generally intractable. In practice, we rely on two key performance criteria: constraint violation and bias, which are both critical in physical applications. Constraint violation measures the deviation of samples from $\mathcal{C}$, via a distance function or a residual for instance. Even when samples lie within $\mathcal{C}$, they must accurately follow the conditional distribution $p_{\mathcal{C}}$ without bias. To quantify that, bias is typically estimated by comparing sample statistics to known or approximated quantities under $p_{\mathcal{C}}$.

3 Variational framework of sampling and duality

To better understand the constrained sampling problem, we formulate it as an optimization problem in the space of probability measures. In the following, we review the variational structure of Langevin Monte Carlo and Lagrangian duality introduced by (Chamon et al., 2024), which will guide the development of our strictly constrained algorithm in Section 4. Importantly, the duality framework outlined in this section enforces constraints only on average, and therefore does not directly target the strictly constrained distribution $p_{\mathcal{C}}$, which is the ultimate goal of our work.

Variational view of Langevin Monte Carlo Langevin Monte Carlo admits a variational interpretation as a gradient flow in the space of probability distributions. Let q be a density absolutely continuous with respect to p, and define the Kullback-Leibler divergence with respect to p

$$F(q) := D(q\|p) = \int_{\mathbb{R}^d} q \log(q/p), \tag{3.1}$$

which is a non-negative information-theoretic quantity measuring how q differs from p (Kullback & Leibler, 1951). Langevin updates can be viewed as a stochastic particle approximation of the gradient flow minimizing F in the Wasserstein space (Jordan et al., 1998; Villani, 2021). Each iteration drives the law q_t of the chain (x_t) closer to the minimizer $q_\infty = p$. More details are given in Appendix C.

Average-constrained sampling Building on this variational formalism, constraints can be incorporated on average using classical tools from convex optimization (Bertsekas, 2014). This framework is developed in (Chamon et al., 2024), where both equality and inequality constraints are considered. We focus here on equality constraints for clarity. Let $\mathcal{P}_2(\mathbb{R}^d)$ denote the set of probability measures with finite second moments, and $h : \mathbb{R}^d \to \mathbb{R}^m$ a constraint function. The closest distribution to p in $\mathcal{P}_2(\mathbb{R}^d)$ satisfying $h(x) = 0$ on average solves

$$\begin{aligned} \underset{q \in \mathcal{P}_2(\mathbb{R}^d)}{\text{minimize}} \quad & F(q) \\ \text{subject to} \quad & \mathbb{E}_q[h(x)] = 0. \end{aligned} \tag{3.2}$$

This convex problem admits a unique minimizer under standard assumptions (Chamon et al., 2024), but remains infinite-dimensional. To solve it, one can use the Lagrangian and its associated dual function.

Definition 1 [Lagrangian] The Lagrangian of Problem (3.2) is defined as

$$L(q, \lambda) := F(q) + \lambda^\top \mathbb{E}_q[h(x)] \qquad \forall q \in \mathcal{P}_2(\mathbb{R}^d),\ \lambda \in \mathbb{R}^m. \tag{3.3}$$

Definition 2 [Dual function] The dual function of Problem 3.2 is defined as

$$g(\lambda) := \inf_{q \in \mathcal{P}_2(\mathbb{R}^d)} L(q, \lambda), \qquad \forall \lambda \in \mathbb{R}^m. \tag{3.4}$$

The dual function g is concave, and the corresponding dual problem, consisting in maximizing $g(\lambda)$, is a finite-dimensional concave maximization problem (Boyd & Vandenberghe, 2004). It provides a lower bound on the primal value, as $g(\lambda) \leq F(q_\star)$ for all λ. A key property is that the infimum in (3.4) is achieved by $p_\lambda(x) \propto \exp(-U(x, \lambda))$, with the Lagrangian potential

$$U(x, \lambda) := f(x) + \lambda^\top h(x). \tag{3.5}$$

Strong duality refers to the case of equality, when $\sup_{\lambda \in \mathbb{R}^m} g(\lambda) = F(q_\star)$.

Proposition 1 [Attained strong duality] Suppose that strong duality holds and is attained: there exists $\lambda_\star \in \mathbb{R}^m$ such that $g(\lambda_\star) = F(q_\star)$. Then, $q_\star = p_{\lambda_\star}$.

When strong duality is attained, Proposition 1 implies that sampling from $q_\star$ amounts to finding the Lagrange multiplier $\lambda_\star$ by solving the finite-dimensional dual problem, and sampling from $p_{\lambda_\star}$. The Lagrange multiplier can be found by to the so-called dual ascent algorithm:

$$\lambda_{t+1} = \lambda_t + \eta \mathbb{E}_{q_t}[h(x)], \qquad q_t := p_{\lambda_t}, \tag{3.6}$$

where η is a step size (Ruszczyński, 2006). Dual ascent is detailed in Algorithm 5. If Proposition 1 applies, then this algorithm converges to $\lambda_\star$.

Primal-dual sampling In practice, the expectation $\mathbb{E}_{q_t}[h(x)]$ is approximated using samples obtained via Langevin dynamics under potential $U(x, \lambda_t)$. This motivates a primal-dual algorithm: alternating between Langevin sampling and stochastic dual ascent on λ. This scheme, proposed by Chamon et al. (2024), is known as primal-dual Langevin Monte Carlo and is summarized in Algorithm 1. Although the primal-dual Langevin sampling has been successfully applied to constrained sampling problems, it requires differentiable constraint functions, and it only enforces the constraint in expectation, without any control on the variance. Therefore, it does not directly target $p_\mathcal{C}$. In the next section, we address the problem of sampling while satisfying arbitrary constraints almost surely.

Algorithm 1 Primal-dual Langevin

input potential gradient ∇f, equality constraint function h, step sizes $\tau, \eta > 0$, iteration number T, initial distribution q_0
output sample $x_T \in \mathbb{R}^d$
define $U(x, \lambda) := f(x) + \lambda^\top h(x)$
initialize $x_0 \sim q_0$, $\lambda_0 \in \mathbb{R}^m$
for $0 \leq t \leq T - 1$ **do**
  draw $w_t \sim \mathcal{N}(0, I_d)$
  $x_{t+1} = x_t - \tau \nabla_x U(x_t, \lambda_t) + \sqrt{2\tau} w_t$
  $\lambda_{t+1} = \lambda_t + \eta \nabla_\lambda U(x_{t+1}, \lambda_t)$
end for

4 Split augmented Langevin for strictly constrained sampling

In this section, we introduce a novel method for the constrained sampling problem. We first derive a variational formulation of the constrained distribution $p_\mathcal{C}$ and, drawing on the duality framework of Section 3, show why standard penalty-based methods fail to enforce strict constraints. Building on this analysis, we propose Split Augmented Langevin (SAL), a constrained sampling algorithm that provably approaches $p_\mathcal{C}$ while ensuring that all samples belong to $\mathcal{C}$ through the use of a non-smooth potential.

4.1 Variational formulation of constrained sampling

Our method builds upon the following variational formulation of constrained sampling. Importantly, we observe that the constrained distribution $p_\mathcal{C}$ can be characterized as the projection of the unconstrained distribution p onto the set of distributions supported on $\mathcal{C}$.

Proposition 2 Suppose that $0 < \mathbb{P}_p(\mathcal{C}) < 1$. Then the conditional distribution $p_{\mathcal{C}}$ is the projection of p onto the set of distributions supported on $\mathcal{C}$:

$$\begin{aligned} p_{\mathcal{C}} = \underset{q \in \mathcal{P}_2(\mathbb{R}^d)}{\operatorname{argmin}} \quad & D(q\|p) \\ \text{subject to} \quad & \mathbb{P}_q(\mathcal{C}) = 1. \end{aligned} \tag{4.1}$$

This is a special case of I-projection (Csiszár, 1975). To solve it, one might try to apply the duality framework of Section 3 to this problem by casting the support constraint as an expectation constraint $\mathbb{E}_q[c(x)] = 0$, where $c(x) \geq 0$ vanishes only on $\mathcal{C}$. The resulting Lagrangian potential then exactly matches the penalty and guidance schemes of Example 2.3, thus providing a variational interpretation of these approaches. However, we show in the following result that strong duality is not attained, implying that such methods cannot ensure strict constraint satisfaction.

Proposition 3 For Problem (4.1), strong duality is only attained for an infinite Lagrange multiplier:

$$\forall \lambda \in \mathbb{R},\ g(\lambda) < F(q_\star), \quad \text{and} \qquad g(\lambda) \underset{\lambda \to +\infty}{\longrightarrow} F(q_\star). \tag{4.2}$$

Corollary 1 [Penalty methods] Penalty methods (2.5) cannot enforce $\mathbb{P}_q(\mathcal{C}) = 1$.

This singularity stems from the support set being a strict subset of $\mathbb{R}^d$. A possible relaxation is to allow a small violation probability $\mathbb{P}_q(\mathcal{C}) \geq 1 - \delta$ for small $\delta > 0$, but this allows unphysical states and leads to poor conditioning. To overcome this limitation, we introduce a different relaxation that preserves strict constraint satisfaction.

4.2 Split Augmented Langevin

To relax the problem without compromising constraint satisfaction, we propose to split the variable x into a pair $(x, z) \in \mathbb{R}^d \times \mathcal{C}$, enforcing that $z \in \mathcal{C}$ while encouraging x and z to remain close. We thus define a joint probability density $q(x, z)$, with marginals q_x and q_z.

Proposition 4 [Variable splitting] Problem (4.1) is equivalent to the following problem:

$$\begin{aligned} \underset{q \in \mathcal{P}_2(\mathbb{R}^d \times \mathcal{C})}{\text{minimize}} \quad & D(q_x\|p) \\ \text{subject to} \quad & \mathbb{P}_q\left(x = z\right) = 1. \end{aligned} \tag{4.3}$$

This formulation mirrors variable splitting techniques in optimization (Boyd et al., 2011a), and separates the roles of x and $z \in \mathcal{C}$, which are respectively maximizing likelihood and enforcing the constraint. Rather than requiring $x = z$ almost surely, we relax the condition to be satisfied in expectation, and penalize the variance. Specifically, we consider the following problem:

$$\begin{aligned} \underset{q \in \mathcal{P}_2(\mathbb{R}^d \times \mathcal{C})}{\text{minimize}} \quad & D(q\|p \otimes u_{\mathcal{C}}) + \frac{\rho}{2}\mathbb{E}_q\left[\|x - z\|^2\right] \\ \text{subject to} \quad & \mathbb{E}_q[x - z] = 0, \end{aligned} \tag{4.4}$$

where $u_{\mathcal{C}}$ denotes the uniform distribution on $\mathcal{C}$, and parameter $\rho > 0$ controls coupling strength. This relaxed formulation avoids the duality failure in Proposition 3 by softening the coupling constraint between x and z. Following Section 3, we introduce the associated non-smooth augmented Lagrangian potential

$$U_\rho(x, z, \lambda) := f(x) + \chi_{\mathcal{C}}(z) + \lambda^\top(x - z) + \frac{\rho}{2}\|x - z\|^2. \tag{4.5}$$

Stochastic proximal primal-dual updates. To sample from this non-smooth potential, we generalize the primal-dual iterations of Chamon et al. (2024) to stochastic proximal iterations. Given independent Gaussian noise vectors $w_t, w'_t \sim \mathcal{N}(0, I_d)$, the stochastic updates derived from the augmented potential (4.5) are

$$x_{t+1} = x_t - \tau\left(\nabla f(x_t) + \rho(x_t - z_t + \mu_t)\right) + \sqrt{2\tau}w_t \tag{4.6a}$$

$$z_{t+1} = P_{\mathcal{C}}(z_t - \tau\rho(z_t - x_{t+1} - \mu_t) + \sqrt{2\tau}w'_t) \tag{4.6b}$$

$$\mu_{t+1} = \mu_t + \eta(x_{t+1} - z_{t+1}), \tag{4.6c}$$

with rescaled multiplier $\mu := (1/\rho) \times \lambda$. We call this scheme Split Augmented Langevin, or SAL, detailed in Algorithm 2. The output $z_T \in \mathcal{C}$ strictly satisfies the constraint. AppendixA gives a detailed derivation.

Connection with optimization algorithms The update formulas (4.6) resemble the Alternating Direction Method of Multipliers (ADMM) (Boyd et al., 2011b), widely used in constrained optimization. Here, the variables x and z play the role of the primal variables in ADMM and λ the dual, and the stochastic augmented potential (4.5) plays the role of an augmented Lagrangian. Our sampling scheme can be seen a stochastic analog of ADMM in sample space $\mathbb{R}^d$, just like Langevin Monte Carlo parallels gradient descent. However, it differs from ADMM applied in distribution space $\mathcal{P}_2(\mathbb{R}^d)$, as our method operates directly on coupled samples.

Algorithm 2 Split Augmented Langevin (SAL)

input potential gradient ∇f, projection $P_{\mathcal{C}}$, step sizes $\tau, \eta > 0$, coupling $\rho > 0$, iteration number T, initial distribution q_0
output sample $z_T \in \mathcal{C}$
initialize $x_0 \sim q_0$, $z_0 = P_{\mathcal{C}}(x_0)$, $\mu_0 \in \mathbb{R}^d$
for $0 \leq t \leq T-1$ **do**
  draw $w_t, w_t' \sim \mathcal{N}(0, I_d)$
  $x_{t+1} = x_t - \tau\nabla f(x_t) - \tau\rho(x_t - z_t + \mu_t) + \sqrt{2\tau}w_t$
  $z_{t+1} = P_{\mathcal{C}}(z_t - \tau\rho(z_t - x_{t+1} - \mu_t) + \sqrt{2\tau}w_t')$
  $\mu_{t+1} = \mu_t + \eta(x_{t+1} - z_{t+1})$
end for

4.3 Convergence analysis

We now provide theoretical support for the proposed scheme. Proofs can be found in Appendix B. First, we prove that strong duality holds and is attained for the relaxed problem, thus ensuring the convergence of the dual ascent algorithm.

Proposition 5 [Attained duality] Strong duality holds and is attained for Problem (4.4).

Corollary 2 [Convergence guarantee] The dual ascent algorithm converges for Problem (4.4).

Corollary 2 guarantees that our relaxation leads to a well-behaved iterative algorithm. Moreover, the relaxed problem recovers the original projection in the limit of infinite coupling.

Proposition 6 [Recovery of the projection] Let q^ρ denote the solution to (4.4). Then

$$q_x^\rho, q_z^\rho \underset{\rho \to +\infty}{\longrightarrow} p_{\mathcal{C}} \quad \text{in distribution.} \tag{4.7}$$

Thus, larger values of ρ bring the x samples closer to $\mathcal{C}$, while smaller values encourage exploration.

These results support SAL as a principled method for sampling from constrained distributions.

4.4 Practical implementation and deep generative models

Implementation in diffusion models Our proposed algorithm is a constrained variant of Langevin Monte Carlo, which plays a central role in many generative frameworks (Du & Mordatch, 2019; Song & Ermon, 2019). The split-augmented update (4.6) can be used as a drop-in replacement for standard Langevin steps, without altering other sampler components, making constraint enforcement simple and modular. Leveraging the connection between Langevin dynamics and diffusion models (Ho et al., 2020), SAL provides a training-free constrained sampling algorithm for pre-trained diffusion models. This parallel has already been exploited by Christopher et al. (2024) to introduce projected diffusion models. We further extend SAL to latent diffusion and to incorporate partial signal observations, which is key in real-life applications. Details are discussed in Appendix A

Constraint satisfaction Our algorithm applies to arbitrary constraint sets, provided that a projection operator (exact or approximate) is available. Unlike primal-dual Langevin and penalty methods, it does not require a differentiable constraint model, which can be challenging to derive (Laumond, 1987). The coupling parameter ρ can be tuned or progressively increased along the diffusion process. This is detailed with ablation studies in Appendix E.

Computational cost For learning methods to accelerate large-scale physical simulations, efficiency is central. Crucially, SAL extends to latent diffusion, which is key to mitigate sampling costs. Compared to unconstrained diffusion, our method adds the cost of a projection operation at each step, as does projected diffusion. For non-convex constraints, efficient numerical methods such as augmented Lagrangian algorithms can be used to solve the projection step, and are amenable to parallelization (Boyd et al., 2011b; Liang et al., 2025). More details can be found in Appendix A.

5 APPLICATION TO PHYSICS-PRESERVING GENERATIVE MODELING

We evaluate SAL on three scientific generative modeling tasks where challenging non-convex physical constraints play a critical role. We apply SAL to diffusion models as described in Section 4.4.

Baselines Our sampling algorithm is compared with the unconstrained Langevin algorithm, the projected Langevin algorithm, constraint penalty guidance methods, primal-dual Langevin, and their diffusion analogs (Carvalho et al., 2023; Christopher et al., 2024; Zhang et al., 2025; Zampini et al., 2025). All methods share the same score function, and differ only in how constraints are incorporated. More details are given in Appendix E.

5.1 ENERGY-PRESERVING STATIONARY FIELD GENERATION

We first validate our method to constrained Monte Carlo sampling of a stationary distribution. We consider a two-dimensional field, representing for instance a fluid (see Figure 1), discretized on a 100×100 grid. The equilibrium distribution p is sampled using Langevin dynamics. Sampling from an equilibrium distribution is a critical problem in climate science and in molecular dynamics for example (Paquet & Viktor, 2015; Pedersen et al., 2025). A key macroscopic quantity is the kinetic energy, which often remains conserved and is known in advance in physical prediction tasks. The task is to sample from the conditional distribution under a fixed energy $\frac{1}{2}\|x\|_2^2 = E$, a non-convex constraint.

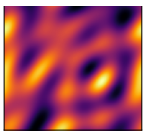

Figure 1: Sampled field snapshot.

Experimental setup The distribution p is bimodal in Fourier space, with asymmetric modes on the first Fourier coefficient: one positive and concentrated, the other negative and wider, allowing higher energy. The unconstrained distribution is sampled with the Langevin Monte Carlo algorithm, and $p_{\mathcal{C}}$ is estimated via rejection sampling. The bimodal nature of p makes the exploration challenging. We condition on a high energy level, only achievable via the negative mode. As the positive mode cannot satisfy the energy constraint, the correct conditional distribution concentrates on the negative mode, and we can easily compare it to the generated samples. For each method, 1000 independent chains are run and the last iterate is collected. We compute histograms of the first Fourier coefficient for evaluation.

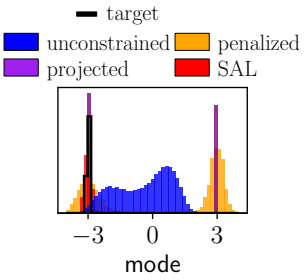

Figure 2: Empirical histograms for the first mode.

Results Figure 2 shows the results. Only SAL matches $p_{\mathcal{C}}$ closely. Projected Langevin satisfies the constraint exactly but fails to explore, yielding many samples in the wrong mode. Soft constraint penalty (Zhang et al., 2025) enforce energy conservation only on average, and therefore the produced samples do not match $p_{\mathcal{C}}$. Primal-dual Langevin yields a similar histogram. These results demonstrate that SAL enforces hard constraints while retaining enough exploration to correctly sample the conditional distribution.

5.2 PHYSICALLY-CONSTRAINED DATA ASSIMILATION

Data assimilation, a central problem in geophysics, aims to estimate the state of a dynamical system from sparse, noisy observations using prior knowledge. Recent work applies deep generative architectures to this task (Rozet & Louppe, 2023; Qu et al., 2024), but these models do not enforce physical invariants, such as energy or mass conservation, which are essential for physical plausibility in long-term forecasting. We study physically-constrained generative models for data assimilation on the Burgers equation, a reduction of the Navier-Stokes equations with conserved mass and energy that exhibits rich dynamics and complex multiscale behaviors similar to turbulence (van Gastelen et al., 2024). Appendix E.2 gives additional background.

Experimental setup We perform cyclic data assimilation on the Burgers equation discretized on a 200-point spatial grid. The ground truth trajectory evolves from a random initial condition over a time horizon $H = 8$. Observations are sparse: the system is observed at 10 equally spaced

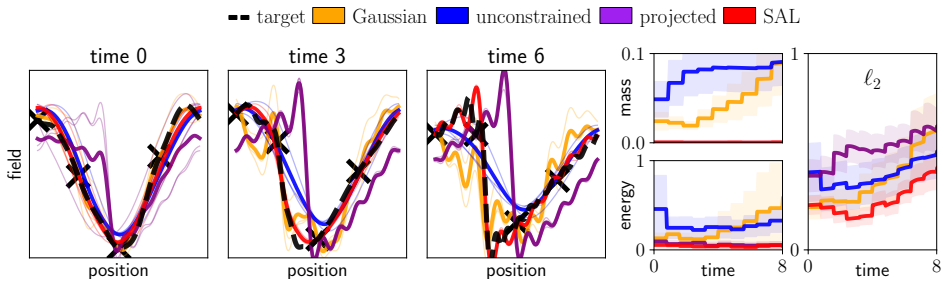

Figure 3: **Left** Data assimilation sampled states and reanalysis. The black crosses represent the observations. **Right** Averaged relative error, in terms of constraint violation and ℓ_2 norm.

times, with 4 noisy spatial measurements at fixed, evenly spaced locations. Each method runs for 5 cycles per trajectory, producing a predicted trajectory that can be compared to the ground truth. The first baseline is 3D-Var (Courtier et al., 1998), which estimates the state with a Gaussian posterior. We compare 3D-Var to deep generative models by training a latent diffusion model offline on a dataset of trajectories, without any conditioning. At sampling time, diffusion is combined with the Gaussian posterior, which conditions sampling to the available information. For each cycle, the analysis is computed as the average of 5 diffusion posterior samples. The experiment is repeated on 50 independent trajectories. We compute the average mean squared error with respect to the ground truth in the state space, in ℓ_2 norm, and in the constraint space, where the quadratic constraint violation error is reported. All methods share the same biased linear forecast model.

Results Figure 3 shows assimilated states and averaged error curves. In this under-observed setting, the diffusion prior helps to regularize the structure of complex states better than the Gaussian prior, especially for longer times, where the system shows a stiffer structure. However, unconstrained diffusion drifts away from the true trajectory, with significant deviations in both mass and energy. Projected diffusion (Christopher et al., 2024) strictly enforces constraints but introduces high-frequency artifacts, leading to physically implausible states. Our algorithm SAL achieves the best compromise: it respects conservation and guides sampling toward physically plausible states, resulting in significantly lower estimation error. These results highlight the potential of constrained generative modeling for robust data assimilation in physical systems.

5.3 Constrained priors for feasibility problems in optimal control

As a final application, we evaluate SAL on a feasibility problem in optimal control: find trajectories that satisfy both system dynamics and non-convex obstacle avoidance constraints. These problems are hard due to the non-convexity of obstacle regions. We consider a dynamical system with state $y(s)$ and control $u(s)$, with s the physical time, and define a trajectory as $x := (y(s), u(s))_s$. Dynamics are encoded via the constraint set $\mathcal{C}_\mathrm{d} := \{x \mid \dot{y} = f(y,u),\ |u| \leq u_\mathrm{max}\}$. Obstacle constraints define the potentially non-convex set $\mathcal{C}_\mathrm{o} := \{x \mid y(s) \notin O_i\ \forall s\}$, for obstacle regions O_i. The goal is to find trajectories in the intersection $\mathcal{C}_\mathrm{d} \cap \mathcal{C}_\mathrm{o}$.

For this task, ADMM (Bílková & Šorel, 2021) is a classical solver alternating projections onto $\mathcal{C}_\mathrm{d}$ and $\mathcal{C}_\mathrm{o}$, but its convergence can be compromised when $\mathcal{C}_\mathrm{o}$ is non-convex. Instead, we propose to guide ADMM with samples from a generative prior: a diffusion model trained on trajectories, with constraints enforced at sampling. This approach has seen promising results in control and robotics with diffusion penalty guidance and projected diffusion (Carvalho et al., 2023; Shaoul et al., 2025; Zampini et al., 2025), which we implement and compare with SAL.

Experimental setup We consider a planar quadrotor system, controlled in acceleration angle (Tedrake, 2009). A latent diffusion model is trained on a dataset of obstacle-free trajectories, obtained with a variety of random periodic excitations. At test time, non-convex obstacles are introduced. The corresponding constraint is imposed during sampling. In order to avoid the obstacles, the algorithm needs to find a swinging trajectory. Each sampled trajectory is then used to initialize ADMM, and we record the fraction of samples for which a feasible solution is found.

Results Figure 4 shows some sampled trajectories and success rates as the obstacle sizes r increases, computed over 10000 samples. Constraint penalty guidance favors obstacle avoidance, but some sampled trajectories penetrate the obstacles. Projected diffusion avoids obstacles but suffers projection bias, producing distorted and unphysical paths. Our algorithm balances both aspects: it produces obstacle-avoiding trajectories that remain dynamically feasible, leading to significantly higher success.

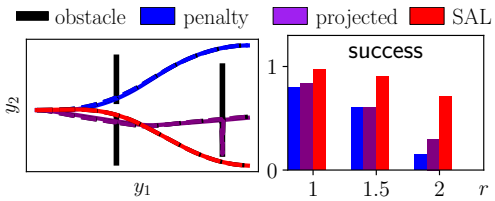

Figure 4: **Left** Dashed lines are sampled trajectories, solid lines are the projections onto the feasibility set. **Right** Feasibility success rates for different sizes of the rightmost obstacle.

6 Related work

Constrained Langevin Monte Carlo Early approaches adapted optimization methods to Langevin dynamics, including Projected Langevin Monte Carlo (Bubeck et al., 2015), proximal Monte Carlo (Salim et al., 2019), Mirrored Langevin (Hsieh et al., 2018), and penalized Langevin (Gurbuzbalaban et al., 2024). Extensions to diffusion models have also been explored (Fishman et al., 2023; Christopher et al., 2024; Liu et al., 2023). These methods offer convergence guarantees in convex settings, where constraints do not hinder exploration, but are less effective in non-convex physical problems. The variational formulation of Langevin sampling has been used by Chamon et al. (2024) to enforce constraints on average.

Variable splitting Variable splitting, inspired by ADMM, has been applied to Bayesian posterior sampling (Vono et al., 2019), plug-and-play samplers for inverse problems (Bouman & Buzzard, 2023; Wu et al., 2024; Martin et al., 2024), and guided diffusion (Zhang et al., 2025). These works apply variable splitting to a smooth maximum a posteriori optimization problem, where the auxiliary variable is updated by gradient descent. Crucially, our framework enforces exact constraint satisfaction through non-smooth constraint potential, without requiring a differentiable constraint model. Moreover, we formalize sampling as an optimization problem in density space rather than in sample space, which is key to obtain our probabilistic sampling guarantees. Our algorithm also extends to latent diffusion, enabling computational savings.

Physically constrained neural networks Physical constraints have also been imposed on deterministic neural networks (Négiar et al., 2023; Hansen et al., 2023). In related sampling approaches, Cheng et al. (2024) integrate projection into flow-matching, and Meunier et al. (2025) enforce soft constraints in diffusion models for ocean modeling. Our approach differs in targeting strict satisfaction in a sampling framework.

7 Conclusion

We introduced Split Augmented Langevin (SAL), a new principled algorithm for constrained generative modeling that enforces hard constraints while preserving complex sampling abilities. Our method formulates conditional sampling as a variational problem and applies primal-dual updates in a relaxed space where strict constraints are progressively enforced. Unlike projection methods, which can distort dynamics, or penalty methods, which may fail to enforce constraints, SAL ensures constraint satisfaction while provably maintaining fidelity to the target distribution. The algorithm is modular, training-free, and integrates seamlessly into Langevin samplers or diffusion models with minimal assumptions on the constraints. Experiments on physical systems, including data assimilation and optimal control, demonstrate improved constraint enforcement and predictive accuracy. These results highlight the potential of combining generative models with physical reasoning in scientific applications where conservation laws and feasibility constraints are essential.

Limitations include the computational cost of repeated projections, which may slow sampling but can be mitigated, the choice of the coupling parameter, and the lack of a non-asymptotic analysis. Future work will extend the framework to other generative models such as stochastic interpolants (Albergo & Vanden-Eijnden, 2023), and develop finite-time convergence guarantees in Wasserstein space (Chamon et al., 2024).

Reproducibility statement The proofs of the new theoretical results included in this paper are available in Appendix B. The code of the proposed algorithm is available online at `anonymous.4open.science/r/constrained-sampling-F7DC/`. Implementation details and comparison with other algorithms and diffusion models are discussed in Appendix A. Ablation studies, discussion about algorithm hyperparameters and experimental details are available in Appendix E.

REFERENCES

Kwangjun Ahn and Sinho Chewi. Efficient constrained sampling via the mirror-Langevin algorithm. *Advances in Neural Information Processing Systems*, 34:28405–28418, 2021.

Michael Samuel Albergo and Eric Vanden-Eijnden. Building Normalizing Flows with Stochastic Interpolants. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=li7qeBbCR1t.

Luigi Ambrosio, Nicola Gigli, and Giuseppe Savaré. *Gradient flows: in metric spaces and in the space of probability measures*. Springer Science & Business Media, 2008.

Rina Foygel Barber and Wooseok Ha. Gradient descent with non-convex constraints: local concavity determines convergence. *Information and Inference: A Journal of the IMA*, 7(4):755–806, 2018.

Dimitri P Bertsekas. *Constrained optimization and Lagrange multiplier methods*. Academic press, 2014.

Zuzana Bílková and Michal Šorel. Projection methods for finding intersection of two convex sets and their use in signal processing problems. *Electronic Imaging*, 33:1–6, 2021.

Matthieu Blanke, Ronan Fablet, and Marc Lelarge. Neural Incremental Data Assimilation. In *ICML 2024 AI for Science Workshop*, 2024.

Charles A Bouman and Gregery T Buzzard. Generative plug and play: Posterior sampling for inverse problems. In *2023 59th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pp. 1–7. IEEE, 2023.

Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers. *Foundations and Trends® in Machine Learning*, 3(1):1–122, 2011a. ISSN 1935-8237. doi: 10.1561/2200000016. URL http://dx.doi.org/10.1561/2200000016.

Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, Jonathan Eckstein, et al. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine learning*, 3(1):1–122, 2011b.

Stephen P Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

Nicolas Brosse, Alain Durmus, Éric Moulines, and Marcelo Pereyra. Sampling from a log-concave distribution with compact support with proximal Langevin Monte Carlo. In *Conference on learning theory*, pp. 319–342. PMLR, 2017.

Sebastien Bubeck, Ronen Eldan, and Joseph Lehec. Finite-time analysis of projected Langevin Monte Carlo. *Advances in Neural Information Processing Systems*, 28, 2015.

Joao Carvalho, An T Le, Mark Baierl, Dorothea Koert, and Jan Peters. Motion planning diffusion: Learning and planning of robot motions with diffusion models. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1916–1923. IEEE, 2023.

Luiz Chamon, Mohammad Reza Karimi Jaghargh, and Anna Korba. Constrained Sampling with Primal-Dual Langevin Monte Carlo. *Advances in Neural Information Processing Systems*, 37: 29285–29323, 2024.

Chaoran Cheng, Boran Han, Danielle C Maddix, Abdul Fatir Ansari, Andrew Stuart, Michael W Mahoney, and Bernie Wang. Gradient-Free Generation for Hard-Constrained Systems. In *The Thirteenth International Conference on Learning Representations*, 2024.

Jacob K Christopher, Stephen Baek, and Nando Fioretto. Constrained synthesis with projected diffusion models. *Advances in Neural Information Processing Systems*, 37:89307–89333, 2024.

Gabriele Corso, Hannes StÃ, Bowen Jing, Regina Barzilay, Tommi Jaakkola, et al. DiffDock: Diffusion Steps, Twists, and Turns for Molecular Docking. In *International Conference on Learning Representations (ICLR 2023)*, 2023.

Philippe Courtier, E Andersson, W Heckley, D Vasiljevic, M Hamrud, A Hollingsworth, F Rabier, M Fisher, and J Pailleux. The ECMWF implementation of three-dimensional variational assimilation (3D-Var). I: Formulation. *Quarterly Journal of the Royal Meteorological Society*, 124(550): 1783–1807, 1998.

Imre Csiszár. I-divergence geometry of probability distributions and minimization problems. *The annals of probability*, pp. 146–158, 1975.

Yilun Du and Igor Mordatch. Implicit generation and modeling with energy based models. *Advances in neural information processing systems*, 32, 2019.

Alain Durmus, Eric Moulines, and Marcelo Pereyra. Efficient bayesian computation by proximal Markov chain Monte Carlo: when Langevin meets Moreau. *SIAM Journal on Imaging Sciences*, 11(1):473–506, 2018.

Alain Durmus, Szymon Majewski, and Błażej Miasojedow. Analysis of Langevin Monte Carlo via convex optimization. *Journal of Machine Learning Research*, 20(73):1–46, 2019.

Edward S Epstein and Rex J Fleming. Depicting stochastic dynamic forecasts. *Journal of Atmospheric Sciences*, 28(4):500–511, 1971.

Geir Evensen. The ensemble Kalman filter: Theoretical formulation and practical implementation. *Ocean dynamics*, 53:343–367, 2003.

Nic Fishman, Leo Klarner, Valentin De Bortoli, Emile Mathieu, and Michael John Hutchinson. Diffusion Models for Constrained Domains. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856. URL https://openreview.net/forum?id=xuWTFQ4VGO. Expert Certification.

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

Neil J Gordon, David J Salmond, and Adrian FM Smith. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. In *IEE proceedings F (radar and signal processing)*, volume 140, pp. 107–113. IET, 1993.

Mert Gurbuzbalaban, Yuanhan Hu, and Lingjiong Zhu. Penalized Overdamped and Underdamped Langevin Monte Carlo Algorithms for Constrained Sampling. *Journal of Machine Learning Research*, 25(263):1–67, 2024.

Derek Hansen, Danielle C Maddix, Shima Alizadeh, Gaurav Gupta, and Michael W Mahoney. Learning physical models that can respect conservation laws. In *International Conference on Machine Learning*, pp. 12469–12510. PMLR, 2023.

Geoffrey E Hinton. Training products of experts by minimizing contrastive divergence. *Neural computation*, 14(8):1771–1800, 2002.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Ya-Ping Hsieh, Ali Kavis, Paul Rolland, and Volkan Cevher. Mirrored Langevin dynamics. *Advances in Neural Information Processing Systems*, 31, 2018.

Langwen Huang, Lukas Gianinazzi, Yuejiang Yu, Peter D. Dueben, and Torsten Hoefler. DiffDA: a diffusion model for weather-scale data assimilation. *Proceedings of the 41st International Conference on Machine Learning*, 2024a.

William Huang, Yifeng Jiang, Tom Van Wouwe, and Karen Liu. Constrained diffusion with trust sampling. *Advances in Neural Information Processing Systems*, 37:93849–93873, 2024b.

Aapo Hyvärinen and Peter Dayan. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6(4), 2005.

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

Karthik Kashinath, M Mustafa, Adrian Albert, JL Wu, C Jiang, Soheil Esmaeilzadeh, Kamyar Azizzadenesheli, R Wang, Ashesh Chattopadhyay, A Singh, et al. Physics-informed machine learning: case studies for weather and climate modelling. *Philosophical Transactions of the Royal Society A*, 379(2194):20200093, 2021.

Diederik P Kingma, Max Welling, et al. Auto-encoding Variational Bayes, 2013.

Solomon Kullback and Richard A Leibler. On information and sufficiency. *The annals of mathematical statistics*, 22(1):79–86, 1951.

Jean-Paul Laumond. Finding Collision-Free Smooth Trajectories for a Non-Holonomic Mobile Robot. In *IJCAI*, volume 87, pp. 1120–1123, 1987.

Théotime Le Hellard, Franki Nguimatsia Tiofack, Quentin Le Lidec, and Justin Carpentier. Sobolev Diffusion Policy. 2025.

Jinhao Liang, Jacob K Christopher, Sven Koenig, and Ferdinando Fioretto. Multi-agent path finding in continuous spaces with projected diffusion models. *arXiv preprint arXiv:2412.17993*, 2024.

Jinhao Liang, Jacob K Christopher, Sven Koenig, and Ferdinando Fioretto. Simultaneous Multi-Robot Motion Planning with Projected Diffusion Models. *arXiv preprint arXiv:2502.03607*, 2025.

Guan-Horng Liu, Tianrong Chen, Evangelos Theodorou, and Molei Tao. Mirror diffusion models for constrained and watermarked generation. *Advances in Neural Information Processing Systems*, 36:42898–42917, 2023.

Andrew C Lorenc. Analysis methods for numerical weather prediction. *Quarterly Journal of the Royal Meteorological Society*, 112(474):1177–1194, 1986.

Ségolène Martin, Anne Gagneux, Paul Hagemann, and Gabriele Steidl. PnP-Flow: Plug-and-play image restoration with flow matching. *arXiv preprint arXiv:2410.02423*, 2024.

Etienne Meunier, David Kamm, Guillaume Gachon, Redouane Lguensat, and Julie Deshayes. Learning to generate physical ocean states: Towards hybrid climate modeling. *arXiv preprint arXiv:2502.02499*, 2025.

Juan Nathaniel and Pierre Gentine. Generative emulation of chaotic dynamics with coherent prior. *arXiv preprint arXiv:2504.14264*, 2025.

Geoffrey Négiar, Michael W. Mahoney, and Aditi Krishnapriyan. Learning differentiable solvers for systems with hard constraints. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=vdv6CmGksr0.

Eric Paquet and Herna L Viktor. Molecular dynamics, monte carlo simulations, and langevin dynamics: a computational review. *BioMed research international*, 2015(1):183918, 2015.

Chris Pedersen, Laure Zanna, and Joan Bruna. Thermalizer: Stable autoregressive neural emulation of spatiotemporal chaos. *arXiv preprint arXiv:2503.18731*, 2025.

Ilan Price, Alvaro Sanchez-Gonzalez, Ferran Alet, Tom R Andersson, Andrew El-Kadi, Dominic Masters, Timo Ewalds, Jacklynn Stott, Shakir Mohamed, Peter Battaglia, et al. Probabilistic weather forecasting with machine learning. *Nature*, 637(8044):84–90, 2025.

Yongquan Qu, Juan Nathaniel, Shuolin Li, and Pierre Gentine. Deep generative data assimilation in multimodal setting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 449–459, 2024.

P. J. Rossky, J. D. Doll, and H. L. Friedman. Brownian dynamics as smart Monte Carlo simulation. *The Journal of Chemical Physics*, 69(10):4628–4633, 11 1978. ISSN 0021-9606. doi: 10.1063/1.436415. URL https://doi.org/10.1063/1.436415.

Litu Rout, Negin Raoof, Giannis Daras, Constantine Caramanis, Alex Dimakis, and Sanjay Shakkottai. Solving linear inverse problems provably via posterior sampling with latent diffusion models. *Advances in Neural Information Processing Systems*, 36:49960–49990, 2023.

François Rozet and Gilles Louppe. Score-based data assimilation. *Advances in Neural Information Processing Systems*, 36:40521–40541, 2023.

Andrzej Ruszczyński. *Constrained Optimization of Differentiable Functions*, pp. 286–342. Princeton University Press, 2006. ISBN 9780691119151. URL http://www.jstor.org/stable/j.ctvcm4hcj.9.

Adil Salim and Peter Richtarik. Primal dual interpretation of the proximal stochastic gradient Langevin algorithm. *Advances in Neural Information Processing Systems*, 33:3786–3796, 2020.

Adil Salim, Dmitry Kovalev, and Peter Richtárik. Stochastic proximal Langevin algorithm: Potential splitting and nonasymptotic rates. *Advances in Neural Information Processing Systems*, 32, 2019.

Yoshikazu Sasaki. Some basic formalisms in numerical variational analysis. *Monthly Weather Review*, 98(12):875–883, 1970.

Yorai Shaoul, Itamar Mishani, Shivam Vats, Jiaoyang Li, and Maxim Likhachev. Multi-Robot Motion Planning with Diffusion Models. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=AUCYptvAf3.

Louis Sharrock, Lester Mackey, and Christopher Nemeth. Learning rate free Bayesian inference in constrained domains. In *NeurIPS*, 2023.

Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019.

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*, 2020.

Russ Tedrake. Underactuated robotics: Learning, planning, and control for efficient and agile machines course notes for mit 6.832. *Working draft edition*, 3(4):2, 2009.

T. van Gastelen, W. Edeling, and B. Sanderse. Energy-conserving neural network for turbulence closure modeling. *Journal of Computational Physics*, 508:113003, 2024. ISSN 0021-9991. doi: https://doi.org/10.1016/j.jcp.2024.113003. URL https://www.sciencedirect.com/science/article/pii/S0021999124002523.

Cédric Villani. *Topics in optimal transportation*, volume 58. American Mathematical Soc., 2021.

Maxime Vono, Nicolas Dobigeon, and Pierre Chainais. Split-and-Augmented Gibbs Sampler—Application to Large-Scale Inference Problems. *IEEE Transactions on Signal Processing*, 67(6):1648–1661, 2019. doi: 10.1109/TSP.2019.2894825.

Max Welling and Yee Whye Teh. Bayesian learning via stochastic gradient Langevin dynamics. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, ICML'11, pp. 681–688, Madison, WI, USA, 2011. Omnipress. ISBN 9781450306195.

Zihui Wu, Yu Sun, Yifan Chen, Bingliang Zhang, Yisong Yue, and Katherine Bouman. Principled probabilistic imaging using diffusion models as plug-and-play priors. *Advances in Neural Information Processing Systems*, 37:118389–118427, 2024.

Stefano Zampini, Jacob K Christopher, Luca Oneto, Davide Anguita, and Ferdinando Fioretto. Training-free constrained generation with stable diffusion models. *arXiv preprint arXiv:2502.05625*, 2025.

Youyuan Zhang, Zehua Liu, Zenan Li, Zhaoyu Li, James J Clark, and Xujie Si. Decoupling Training-Free Guided Diffusion by ADMM. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 23292–23302, 2025.

A ALGORITHMS

A.1 DETAILED ALGORITHMS

Algorithm 3 Langevin Monte Carlo

input potential gradient ∇f, step size τ, iteration number T
output sample x_T
initialize $x_0 \sim q_0$
for $0 \leq t \leq T-1$ **do**
  $w_t \sim \mathcal{N}(0, I_d)$
  $x_{t+1} = x_t - \tau \nabla f(x_t) + \sqrt{2\tau} w_t$
end for

Algorithm 4 Projected Langevin Monte Carlo

input potential $f(x)$, projection $P_{\mathcal{C}}$, step size τ, iteration number T
output sample $x_T \in \mathcal{C}$
initialize $x_0 \sim q_0$
for $0 \leq t \leq T-1$ **do**
  $w_t \sim \mathcal{N}(0, I_d)$
  $x_{t+1} = z_t - \tau \nabla f(z_t) + \sqrt{2\tau} w_t$
  $z_{t+1} = P_{\mathcal{C}}(x_{t+1})$
end for

Algorithm 5 Dual ascent

input constraint function h, dual step size $\eta > 0$, iteration number T
output sample x_T
initialize $x_0 \in \mathbb{R}^d, \lambda_0 \in \mathbb{R}^m$
for $0 \leq t \leq T-1$ **do**
  $q_t = \underset{q \in \mathcal{P}_2(\mathbb{R}^d)}{\operatorname{argmin}} \; L(q, \lambda_t).$
  $\lambda_{t+1} = \lambda_t + \eta \mathbb{E}_{q_t}[h(x)]$
end for

Algorithm 6 Time-dependent SAL

input time dependent potential gradient $f(x,t)$, iteration number T,time-dependent step sizes τ_t, projection $P_{\mathcal{C}}$, step size $\eta > 0$, regularization $\rho > 0$, intial distribution q_0
output sample $z_T \in \mathcal{C}$
initialize $x_0 \sim q_0\,, z_0 = P_{\mathcal{C}}(x_0)\,, \mu_0 = 0 \in \mathbb{R}^d$
for $0 \leq t \leq T-1$ **do**
  draw $w_t, w_t' \sim \mathcal{N}(0, I_d)$
  $x_{t+1} = x_t - \tau_t \nabla f(x_t, t) - \tau_t \rho (x_t - z_t + \mu_t) + \sqrt{2\tau_t} w_t$
  $z_{t+1} = P_{\mathcal{C}}(z_t - \tau_t \rho (z_t - x_{t+1} - \mu_t) + \sqrt{2\tau} w_t')$
  $\mu_{t+1} = \mu_t + \eta (x_{t+1} - z_{t+1})$
end for

A.2 PROJECTED LANGEVIN

Projected Langevin consists in applying Langevin dynamics to the constrained potential $f_{\mathcal{C}}$, However, since $f_{\mathcal{C}}$ is non-smooth, its gradient is not defined. This issue can be addressed using the proximal operator:

$$\operatorname{prox}_{\varphi}(x) := \underset{z \in \mathbb{R}^d}{\operatorname{argmin}} \frac{1}{2}\|z - x\|^2 + \varphi(z). \tag{A.1}$$

An important case for non-smooth functions is the proximal operator of the characteristic function $\chi_{\mathcal{C}}$, which is the projection onto $\mathcal{C}$:

$$P_{\mathcal{C}}(x) := \operatorname{prox}_{\chi_{\mathcal{C}}}(x). \tag{A.2}$$

When well-defined, the proximal operator generalizes the gradient step of a smooth function φ in the sense that $\operatorname{prox}_{\tau\varphi}(x) = x - \tau \nabla \varphi(x)$. Applying the proximal step associated with $\tau f_{\mathcal{C}}$ to the noisy iterate $x_t + \sqrt{2\tau} w_t$ yields the so-called Projected Langevin iteration $x_{t+1} = P_{\mathcal{C}}(x_t - \tau \nabla f(x_t) + \sqrt{2\tau} w_t)$. The corresponding constrained sampling algorithm is the Projected Langevin Algorithm (Brosse et al., 2017), which we detail in Algorithm 4.

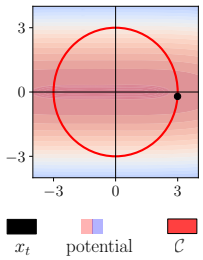

Figure 5: Projected Langevin Algorithm.

Limited exploration Figure 5 shows the exploration issue arising with Projected Langevin Monte Carlo in the case of non-convex constraints and a bi-modal distribution. Here, projecting on the constraint set $\mathcal{C} = \{x \mid \frac{1}{2}\|x\|^2 = E\}$ leads to poor exploration, as the samples are suck on the positive side of the likelihood landscape, while the only high-likelihood zone compatible with the constraint is on the other side.

A.3 Derivation of the Split-augmented sampling formulas

Recall the augmented Lagrangian potential

$$U_\rho(x, z, \lambda) := f(x) + \chi_{\mathcal{C}}(z) + \lambda^\top (x - z) + \frac{\rho}{2}\|x - z\|^2, \tag{A.3}$$

and let $\mu = (1/\rho)\lambda$. Taking a stoachastic gradient step with respect to x yields

$$x_{t+1} = x_t - \tau\left(\nabla f(x_t) + \rho(x_t - z_t + \mu_t)\right) + \sqrt{2\tau} w_t \tag{A.4}$$

Taking a stoachastic proximal step with respect to z yields

$$z_{t+1} = P_{\mathcal{C}}(z_t - \tau\rho(z_t - x_{t+1} - \mu_t) + \sqrt{2\tau} w_t') \tag{A.5}$$

Taking a stoachastic gradient step with respect to λ yields

$$\mu_{t+1} = \mu_t + \eta(x_{t+1} - z_{t+1}). \tag{A.6}$$

A.4 Extension of SAL to deep generative models

Many modern generative frameworks—from energy-based models to state-of-the-art diffusion models rely on Langevin dynamics for sampling (Hinton, 2002; Du & Mordatch, 2019; Song & Ermon, 2019; Song et al., 2020). In Appendix D, we review how key classes of generative models relate to Langevin updates.

For these generative models, sampling takes the form

$$x_{t+1} = x_t - \tau_t \nabla f(x_t, t) + \sqrt{2\tau_t}\, w_t, \quad w_t \sim \mathcal{N}(0, I). \tag{A.7}$$

We interpret these steps as the discretization of a Wasserstein flow for a time-dependent functional $F(q, t)$, which is summarized in Appendix C. We can then identically apply our constrained sampling algorithm, as a time-dependent variation of Algorithm 2, detailed in Algorithm 6. From a variational point of view, this results in framing the constrained sampling as a time-varying constrained optimization problem.

A.5 Extension to latent diffusion and partial observations

Our framework naturally extends to the case where the Langevin steps, or the diffusion model, are operated in a latent space R^k, which is mapped to the physical space through a decoder

$$\varphi : \mathbb{R}^k \to \mathbb{R}^d. \tag{A.8}$$

The augmented potential takes the form

$$U_\rho(x, z, \lambda) := f(x) + \chi_{\mathcal{C}}(z) + \lambda^\top (\varphi(x) - z) + \frac{\rho}{2}\|\varphi(x) - z\|^2, \tag{A.9}$$

and the stochastic updates (4.6) become

$$x_{t+1} = x_t - \tau\left(\nabla f(x_t) + \rho J_\varphi^\top(x_t)(\varphi(x_t) - z_t + \mu_t)\right) + \sqrt{2\tau} w_t \tag{A.10a}$$

$$z_{t+1} = P_{\mathcal{C}}(z_t - \tau\rho(z_t - \varphi(x_{t+1}) - \mu_t) + \sqrt{2\tau} w_t') \tag{A.10b}$$

$$\mu_{t+1} = \mu_t + \eta(\varphi(x_{t+1}) - z_{t+1}), \tag{A.10c}$$

where J_φ is the decoder Jacobian.

When linear observations are available, in the form of a likelihood

$$y \sim Hz + w, \quad w \sim \mathcal{N}(0, R), \tag{A.11}$$

this additional source of information can naturally be accounted by our framework, by modifying the augmented potential as

$$U_\rho(x, z, \lambda) := f(x) + \chi_{\mathcal{C}}(z) + \lambda^\top(\varphi(x) - z) + \frac{\rho}{2}\|\varphi(x) - z\|^2 + \frac{1}{2}\|Hz - y\|^2_{R^{-1}}, \quad \text{(A.12)}$$

and sampling from the corresponding distribution accordingly.

A.6 Choice of algorithm hyperparameters

The hyperparameters of our algorithm are the primal initial distribution q_0, step size τ, the initial dual variable λ_0, the dual step size η, and the coupling parameter ρ.

In practice, in diffusion models, q_0 is a normal distribution, and τ follows a prescribed schedule. We choose $\lambda_0 = \mu_0 = 0$ and we tune $\eta \in [10^{-3}, 10^{-1}]$ so that we observe convergence in expectation of the two variables x and z. The coupling parameter can be fixed or tuned during sampling. Note that, in diffusion models, the time-varying step size directly influence the effective coupling parameter.

In Appendix E, we detail the hyerparameters for each experiment. We also provide ablation studies are provided to study the influence of λ_0 and ρ quantitatively.

A.7 Comparison of constrained sampling algorithm

Connection with constrained Langevin Monte Carlo Constrained sampling adapted methods from classical optimization include Projected Langevin Monte Carlo (Bubeck et al., 2015; Durmus et al., 2019), proximal Monte Carlo (Salim et al., 2019; Salim & Richtarik, 2020; Brosse et al., 2017; Durmus et al., 2018), Mirrored Langevin (Hsieh et al., 2018; Ahn & Chewi, 2021; Sharrock et al., 2023), and penalized Langevin (Gurbuzbalaban et al., 2024). These approaches are designed for convex constraint sets, for which they enjoy theoretical guarantees. Those guarantees do not hold for non-convex settings, such as those encountered in our applications to physical systems, where exploration of sample space is key.

Connection with Split-and-Augmented Gibbs samplers The constrained sampling formulas of SAL are related to the Split-and-Augmented Gibbs samplers of Vono et al. (2019), which themselves are inspired by ADMM. The main difference is that, z represents a smooth, prior distribution in their case, while it represents hard constraints in our case. Therefore, the framework developed in (Vono et al., 2019) is different from the constrained sampling approach developed in our present work, and Split-and-Augmented Gibbs samplers cannot be applied to enforce strict constraints in deep generative models for example.

Comparison with diffusion guidance Enforcing constraints in diffusion models using penalty and guidance methods has been proposed by Huang et al. (2024b) and Carvalho et al. (2023). Crucially, unlike SAL, constraint penalty and guidance methods rely on a differentiable constraint penalty function. Therefore, their sampling objective is inherently different from (4.1), as the almost-sure constraint defines a non-smooth potential (4.5) that is not differentiable. Our approach tackles this non-smoothness using a proximal operator and projections, thereby ensuring strict constraint satisfaction, rather gradient steps on a smooth approached loss. In the experiments of Section 5.1, we implement the algorithm of Zhang et al. (2025) applying guidance with variable splitting. In the other experiments, we tried to generalize it to latent space diffusion but found the sampling algorithm to fail in sampling with constraints. This is due to the difficulty of scaling the augmentation term and the constraint penalty term to simultaneously ensure constraint satisfaction and coupling with the diffusion model. Instead, we propagated the penalty function through the decoder.

Comparison with projected diffusion Projected diffusion can take different forms (Liang et al., 2024; Christopher et al., 2024). Projected diffusion can be applied in a latent space Zampini et al. (2025), which we implement in 5.3. The constraints are enforced in the latent space by solving an optimization problem through the decoder. Our algorithm allows us to decouple these the two

problems of sampling and of projection by relaxing the physical and the latent variable to be equal only on average. Specifically, the iterations of Zampini et al. (2025) can be obtained as a special case of (A.10) by setting $\lambda_0 = 0$, $\eta = 0$, and by projecting the maximum likelihood iterate $\varphi(x_{t+1})$. In our experiments, we tried to extend this algorithm to account for linear observations, but found no simple way of propagating both a hard constraint projection and an observation likelihood to the latent state.

A.8 Computational cost

To assess the computational cost of SAL, we summarize below the costs for the constraint sets used in our experiments. We provide below a breakdown of the projection cost for different constraint types, and compare them to the overall runtime.

In our experiments, we used three types of projections:

- projection onto a sphere for energy conservation,
- projection onto intervals for obstacle avoidance,
- projection onto a linear subspace for mass conservation,

All of these admit efficient implementations.

For the sphere and the interval, cost is $\mathcal{O}(d)$ with closed-form formulas.

For a linear subspace,

$$\mathcal{C} = \{x \mid Ax + b = 0\} \tag{A.13}$$

with $A \in \mathbb{R}^{m \times d}$. The projection is given by $P_{\mathcal{C}}(x) = x + A^\top (AA^\top)^{-1}(b - Ax)$, which requires a precomputed pseudoinverse at cost $\mathcal{O}(d^2 m)$ and a matrix-vector product at cost $\mathcal{O}(m)$, which remains small compared to neural network evaluations.

In all tested settings, the runtime overhead from projections was small compared to the cost of score evaluations in diffusion models. Furthermore, SAL is compatible with approximate projections, allowing further savings. For more complex constraint sets, iterative solvers such as ADMM can be employed with a limited number of steps, trading accuracy for speed in early iterations, where perfect constraint enforcement is not yet required.

To validate this point, we conducted the following additional runtime experiment. We measure the average wall-clock time for the different sampling algorithms in the data assimilation problem, where each sampled state is projected on the intersection of 2 constraint sets: one for mass and one for energy. All times are in seconds per 10^6 sampling steps, measured on an Apple M1 setup.

Experiment	unconstrained Langevin	primal-dual Langevin	projected Langevin	SAL
Fluid generation	0.37	0.41	0.43	0.45
Data assimilation	1.27	N/A	1.34	1.36

Table 1: Comparison of computational times.

We also note that diffusion guidance can be computationally costly as it can require multiple penalty gradient steps per sampling step to enforce constraints. In the experiment 5.3, we found this method to be of the order of 3 times slower than SAL and projected diffusion.

In summary, SAL has comparable runtime to projected Langevin. For a number of usecases, its additional cost is modest, especially in the context of deep generative models where the computational budget is dominated by score evaluations. This cost can also be adjusted in practice, by computing approximate projections in the early steps of sampling.

B PROOFS

B.1 PROOF OF PROPOSITION 1

The proof can be found in the work of Chamon et al. (2024).

B.2 PROOF OF PROPOSITION 2

Proposition Suppose that $\mathbb{P}_p(\mathcal{C}) > 0$. Then the conditional distribution $p_{\mathcal{C}}$ is the projection of p onto the set of distributions supported on $\mathcal{C}$:

$$\begin{aligned} p_{\mathcal{C}} = \underset{q \in \mathcal{P}_2(\mathbb{R}^d)}{\operatorname{argmin}} \quad & D(q\|p) \\ \text{subject to} \quad & \mathbb{P}_q(\mathcal{C}) = 1. \end{aligned} \tag{B.1}$$

Proof. Let $q \in \mathcal{P}_2(\mathbb{R}^d)$ such that $\mathbb{P}_q(\mathcal{C}) = 1$. Then q vanishes almost everywhere out of $\mathcal{C}$. Hence,

$$\begin{aligned} D(q\|p) &= \int_{\mathcal{C}} q(x) \log \frac{q(x)}{p(x)} \mathrm{d}x \\ &= \int_{\mathcal{C}} q(x) \log \left(\frac{q(x)}{p_{\mathcal{C}}(x)} \frac{Z_{\mathcal{C}}}{Z} \right) \mathrm{d}x \\ &= D(q\|p_{\mathcal{C}}) + \frac{Z_{\mathcal{C}}}{Z} \end{aligned} \tag{B.2}$$

where $Z_{\mathcal{C}}$ satisfies

$$\begin{aligned} 1 &= \int_{\mathbb{R}^d} p_{\mathcal{C}} \\ &= \frac{Z}{Z_{\mathcal{C}}} \int_{\mathcal{C}} p(x) \mathrm{d}x \\ &= \frac{Z}{Z_{\mathcal{C}}} \mathbb{P}_p(\mathcal{C}). \end{aligned} \tag{B.3}$$

Therefore,

$$D(q\|p) = D(q\|p_{\mathcal{C}}) + \mathbb{P}_p(\mathcal{C}). \tag{B.4}$$

This quantity is minimized for $q = p_{\mathcal{C}}$, and the minimal value is $\mathbb{P}_p(\mathcal{C})$. □

B.3 PROOF OF PROPOSITION 3

Proposition Consider the following problem:

$$\begin{aligned} p_{\mathcal{C}} = \underset{q \in \mathcal{P}_2(\mathbb{R}^2)}{\operatorname{argmin}} \quad & D(q||p) \\ \text{subject to} \quad & \mathbb{P}_q(\mathcal{C}) = 1, \end{aligned} \tag{B.5}$$

with the constrained expressed as $\mathbb{E}[c(x)] = 0$ for a penalty function $c(x) \geq 0$ such that $c(x) = 0$ only on $\mathcal{C}$. For example, $c(x) = 1 - \mathbb{1}_{\mathcal{C}}(x)$. Recall that $F(q) = D(q||p)$ and

$$g(\lambda) := \inf_{q \in \mathcal{P}_2(\mathbb{R}^d)} L(q, \lambda). \tag{B.6}$$

Strong duality holds, but is attained only for an infinite Lagrange multiplier:

$$\forall \lambda \in \mathbb{R},\ g(\lambda) < F(q_\star), \quad \text{and} \qquad g(\lambda) \underset{\lambda \to +\infty}{\longrightarrow} F(q_\star). \tag{B.7}$$

Proof.

$$\begin{aligned} L(q, \lambda) &= D(q||p) + \lambda \left(1 - \mathbb{P}_q(x \in \mathcal{C})\right) \\ &= D(q||p) + \lambda \mathbb{E}_q[c(x)]. \end{aligned} \tag{B.8}$$

For all $\lambda \in \mathbb{R}$, the infimum in the dual function definition is attained by

$$\begin{aligned} p_\lambda(x) &= \frac{1}{Z_\lambda} \mathrm{e}^{-f(x)-\lambda c(x)} \\ &= \frac{Z}{Z_\lambda} p(x) \mathrm{e}^{-\lambda c(x)} \end{aligned} \tag{B.9}$$

and the Lagrangian evaluated at p_λ equals

$$g(\lambda) = \log \frac{Z}{Z_\lambda}. \tag{B.10}$$

To compute Z_λ, we note that

$$\begin{aligned} 1 &= \int_{\mathbb{R}^d} p_\lambda \\ &= \frac{Z}{Z_\lambda} \int_{\mathcal{C}} p(x) \mathrm{d}x + \frac{Z}{Z_\lambda} \int_{\bar{\mathcal{C}}} \mathrm{e}^{-\lambda c(x)} p(x) \mathrm{d}x \end{aligned} \tag{B.11}$$

Let

$$\varepsilon(\lambda) := \int_{\bar{\mathcal{C}}} \mathrm{e}^{-\lambda c(x)} p(x) \mathrm{d}x. \tag{B.12}$$

Then,

$$1 = \frac{Z}{Z_\lambda} \left[\mathbb{P}_p(\mathcal{C}) + \varepsilon(\lambda) \right] \tag{B.13}$$

By assumption, for all $\lambda \in \mathbb{R}^d$, $0 < \varepsilon(\lambda) < 1$. Furthermore, we obtain by combining (B.10) and (B.13), that

$$g(\lambda) = \log \frac{1}{\mathbb{P}_p(\mathcal{C}) + \varepsilon(\lambda)}. \tag{B.14}$$

This value is always strictly lower than its limit:

$$\forall \lambda,\ g(\lambda) < \log \frac{1}{\mathbb{P}_p(\mathcal{C})} = \lim_{\lambda \to +\infty} g(\lambda), \tag{B.15}$$

which is precisely the optimal value of Problem (4.1), attained by $q = p_{\mathcal{C}}$. Indeed,

$$\begin{aligned} D(p_{\mathcal{C}} || p) &= \int_{\mathcal{C}} \frac{Z}{Z_{\mathcal{C}}} p(x) \log \frac{Z}{Z_{\mathcal{C}}} \mathrm{d}x \\ &= \mathbb{P}_p(\mathcal{C}) \frac{Z}{Z_{\mathcal{C}}} \log \frac{Z}{Z_{\mathcal{C}}}, \end{aligned} \tag{B.16}$$

where $Z_{\mathcal{C}}$ satisfies

$$\begin{aligned} 1 &= \int_{\mathbb{R}^d} p_{\mathcal{C}} \\ &= \frac{Z}{Z_{\mathcal{C}}} \int_{\mathcal{C}} p(x) \mathrm{d}x \\ &= \frac{Z}{Z_{\mathcal{C}}} \mathbb{P}_p(\mathcal{C}). \end{aligned} \tag{B.17}$$

It follows that

$$D(p_{\mathcal{C}} || p) = \log \frac{1}{\mathbb{P}_p(\mathcal{C})}. \tag{B.18}$$

This value is found to be the minimizer of Problem (4.1) using Gibbs' inequality. □

B.4 Proof of Corollary 1

Corollary [Penalty methods] Penalty methods (2.5) cannot enforce $\mathbb{P}_q(\mathcal{C}) = 1$.

Proof. Penalty methods sample from p_λ, with finite λ. For all densities $q \in \mathcal{P}_2(\mathbb{R}^d)$ satisfying the constraint $\mathbb{P}_q(\mathcal{C}) = 1$, the duality gap implies

$$L(q_\lambda, \lambda) = g(\lambda) < F(p_{\mathcal{C}}) \leq F(q) = L(q, \lambda). \tag{B.19}$$

Therefore, p_λ does not satisfy $\mathbb{P}_q(\mathcal{C}) = 1$. □

B.5 Proof of Proposition 4

Proposition 7 [Variable splitting] Problem (4.1) is equivalent to the following problem:

$$\begin{aligned} \underset{q \in \mathcal{P}_2(\mathbb{R}^d \times \mathcal{C})}{\text{minimize}} \quad & D(q_x \| p) \\ \text{subject to} \quad & \mathbb{P}_q\left(x = z\right) = 1. \end{aligned} \tag{B.20}$$

Proof. Given $q(x, z)$ the solution of Problem (4.3), the marginal q_x gives the solution of Problem (4.1). Given $q(x)$ the solution of Problem (4.1), the solution of Problem (4.3) can be obtained by defining z as a copy of x. □

B.6 Proof of Proposition 5

Proposition 8 [Attained duality] Strong duality holds and is attained for Problem (4.4).

Proof. In order to apply Proposition 2.2 from Chamon et al. (2024), we verify the required assumption: there exists $q > 0$, such that $\mathbb{E}_q[x - z] = 0$ (positivity ensures constraint qualification). Such distribution can be obtained by defining $q(x, z) := q(x)q(z|x)$, with for example $q(x)$ a Gaussian normal density and $q(z|x)$ a Gaussian density centered on x. Then, the aforementioned proposition can be applied and Proposition 5 follows. This result cannot be applied to Problem (4.1) because the feasibility set for q imposes that the density has zeros measure out of $\mathcal{C}$, making the non-negativity constraint of the density not qualified. □

B.7 Proof of Proposition 6

Recall the relaxed problem

$$\begin{aligned} \underset{q \in \mathcal{P}(\mathbb{R}^d \times \mathcal{C})}{\text{minimize}} \quad & D(q||p \otimes u_{\mathcal{C}}) + \rho \mathbb{E}_q\left[\|x - z\|^2\right] \\ \text{subject to} \quad & \mathbb{E}_q[x - z] = 0. \end{aligned} \tag{B.21}$$

Proposition [Problem approximation] The ρ-approximation converges to the strictly constrained problem, as

$$q_\rho \xrightarrow[\rho \to +\infty]{\text{law}} p_{\mathcal{C}}.$$

Proof of Proposition 6. Recall that, because strong duality is attained, the solution of (4.4) is attained by a distribution of the form

$$q_\rho(x, z) = \frac{1}{Z_\lambda} \mathrm{e}^{-f(x)} \mathrm{e}^{-\chi_{\mathcal{C}}(z)} \mathrm{e}^{-\frac{\rho}{2}\|x-z\|^2} \mathrm{e}^{-\lambda^\top (x-z)} \tag{B.22}$$

Let $z \in \mathcal{C}$ and $x \neq z$ in $\mathbb{R}^d$. Then, $q_\rho(x, z) \underset{\rho \to +\infty}{\longrightarrow} 0 = p_{\mathcal{C}}^2(x, z)$.

Additionally,

$$q_\rho(z, z) = \frac{1}{Z_\lambda} \mathrm{e}^{-f(z)} \tag{B.23}$$

□

C VARIATIONAL FRAMEWORK FOR LANGEVIN MONTE CARLO

Consider the functional

$$F(q) = D(q\|p) = \int q \log(q/p). \tag{C.1}$$

The Wasserstein gradient flow is defined as the following differential system

$$\frac{\partial q}{\partial t} = \nabla \cdot \left(q \nabla \frac{\partial F}{\partial q} \right), \tag{C.2}$$

For functional (C.1), the differential system is found to be

$$\frac{\partial q}{\partial t} = \nabla \cdot (q \nabla f(x)) + \Delta q(x, t), \tag{C.3}$$

which is found to be the Fokker-Planck equation for the Langevin dynamics

$$\mathrm{d}x = -\nabla f(x)\mathrm{d}t + \mathrm{d}B. \tag{C.4}$$

More details can be found in (Jordan et al., 1998; Ambrosio et al., 2008; Villani, 2021; Chamon et al., 2024).

D Connection between Generative Models and Langevin Sampling

D.1 Energy-Based Models (EBMs)

An EBM defines a density

$$p(x) \;=\; \frac{1}{Z}\exp\bigl(-f_\theta(x)\bigr), \tag{D.1}$$

where f_θ is a learned energy function. Sampling from p typically relies on Langevin dynamics (2.1) or stochastic gradient Langevin dynamics (SGLD) (Welling & Teh, 2011). EBMs with Langevin sampling have demonstrated strong performance across a range of tasks(Du & Mordatch, 2019), and offer distinct advantages over methods such as Variational Autoencoders (VAEs)(Kingma et al., 2013) and Generative Adversarial Networks (GANs)(Goodfellow et al., 2014). A particularly valuable property of EBMs is their flexibility in incorporating constraints via summing up the corresponding energies. From this perspective, our algorithm, when applied to EBMs, can be interpreted as providing stronger constraint enforcement through an augmented Lagrangian potential and corresponding proximal Langevin updates—going beyond the simple addition of constraint energies.

D.2 Score-Based Generative Models

Score-based generative models aim to learn the score function $\nabla \log p_t(x)$ of a family of progressively noised data distributions $\{p_t\}_{t\in[0,T]}$, rather than modeling the data density directly. Once the score is learned—typically via denoising score matching—samples can be generated by Langevin-type updates.

Annealed Langevin Dynamics Proposed by Song & Ermon (2019), this method generates samples by applying Langevin dynamics at a sequence of decreasing noise levels $\sigma_T > \cdots > \sigma_1$. A score model $s_\theta(x,\sigma)$ is trained to approximate the noise-dependent score $\nabla_x \log q(x;\sigma)$ of the perturbed data distribution $p(x;\sigma)$, which is obtained by convolving $p(x)$ with a Gaussian of various noise level σ_t. Then update step takes the form

$$x_{t+1} = x_t + \tau_t\, s_\theta(x,\sigma_t) + \sqrt{2\tau_t}\, w_t, \quad w_t \sim \mathcal{N}(0,I), \tag{D.2}$$

where $\tau_t \;\propto\; \sigma_t^2$ are time-varying step sizes. The update takes the form of (A.7) with $\nabla f(x,t) = -s_\theta(x,\sigma_t)$. This can be seen as an unadjusted Langevin algorithm with temperature σ_t, gradually refining the sample as noise decreases. In this case our algorithm can be directly applied at each noise level to impose constraints. It is worth noting that the projected diffusion model (Christopher et al., 2024) also falls into this category – a hard projection following each Langevin update within the annealed Langevin dynamics framework. Note that this covers the case where several Langevin steps are taken at fixed noise level, as in the work of Song & Ermon (2019), by choosing τ_t to be constant for a number of steps t.

D.3 Diffusion Models

Denoising Diffusion Probabilistic Models (DDPM) Denoising diffusion probabilistic models (DDPMs), introduced by Ho et al. (2020), define a forward process that gradually corrupts a data point y_0 by adding Gaussian noise through a fixed Markov chain:

$$q(y_t \mid y_{t-1}) = \mathcal{N}(y_t; \sqrt{1-\beta_t}\, y_{t-1}, \beta_t I), \tag{D.3}$$

where $\beta_t \in (0,1)$ is a small noise schedule. This leads to a closed-form expression for $q(x_t \mid x_0)$, with the following definitions:

$$\alpha_t = 1-\beta_t, \quad \bar\alpha_t = \prod_{s=1}^{t} \alpha_s. \tag{D.4}$$

The reverse process is parameterized by a neural network $\epsilon_\theta(x_t,t)$, which predicts the noise component. The sampling procedure follows:

$$x_{t+1} = \frac{1}{\sqrt{\alpha_t}}\left(x_t - \frac{1-\alpha_t}{\sqrt{1-\bar\alpha_t}}\,\epsilon_\theta(x_t,t)\right) + \sigma_t w, \quad w_t \sim \mathcal{N}(0,I), \tag{D.5}$$

where σ_t is typically set to match the forward variance β_t. As noted by Ho et al. (2020), this step corresponds to an Euler-Maruyama discretization of a variant of Langevin dynamics, and the learned noise predictor ϵ_θ implicitly estimates the score $\nabla \log p_t(x)$ up to a scaling factor. Hence, the sampling formula (D.5) takes the form (A.7) with $\tau_t = \sigma_t^2/2$ and

$$\nabla \log p_t(x_t) \approx s_\theta(x_t, t) = -\frac{1}{\sqrt{1-\bar{\alpha}_t}} \epsilon_\theta(x_t, t). \tag{D.6}$$

The DDPM can be regarded as a discrete score-based model under the variance preserving stochastic differential equation (VP-SDE) interpretation (Song et al., 2020), and thus our SAL sampling is valid for DDPM sampling.

D.4 Score-based Diffusion Models

Score-based diffusion models (Song et al., 2020) directly learn the score function of perturbed data distributions and generate samples by simulating the reverse-time stochastic dynamics.

Forward SDE. Define a forward Itô SDE that gradually adds noise to data $x_0 \sim p_{\text{data}}$:

$$\mathrm{d}x = a(x,t)\,\mathrm{d}t \;+\; b(t)\,\mathrm{d}W_t, \tag{D.7}$$

where for the variance-preserving (VP) choice,

$$a(x,t) = -\tfrac{1}{2}\,\beta(t)\,x, \quad b(t) = \sqrt{\beta(t)}. \tag{D.8}$$

This yields marginal distributions $p_t(x)$ that interpolate between the data and near-Gaussian noise as t increases.

Reverse SDE. The time-reversed process follows

$$\mathrm{d}x = \big[a(x,t) - b(t)^2\,\nabla_x \log p_t(x)\big]\mathrm{d}t \;+\; b(t)\,\mathrm{d}W'_t, \tag{D.9}$$

where W'_t is a reverse-time Wiener process. A neural network $s_\theta(x,t)$ is trained by score matching to approximate $\nabla_x \log p(x,t)$.

Predictor–Corrector sampling. Once the score network is trained, our SAL sampling is applicable. SAL can also be integrated seamlessly into the predictor-corrector sampling scheme proposed by Song et al. (2020). The predictor-corrector sampler interleaves:

- *Predictor*: an Euler–Maruyama step of the reverse SDE,

$$x_{t+1} = x_t - \tau_t\big[a(x_t,t) - b(t)^2\,s_\theta(x_t,t)\big] + b(t)\,\sqrt{2\tau_t}w_t \quad w_t \sim \mathcal{N}(0, I_d). \tag{D.10}$$

- *Corrector*: a few steps of Langevin MCMC to refine samples,

$$x_{t+1} = x_t + \tau_t\,s_\theta(x_t,t) + \sqrt{2\tau_t}w_t, \quad w_t \sim \mathcal{N}(0, I_d). \tag{D.11}$$

Similar to the previous sections, these formulas take the form of (A.7), with different time-varying potential gradients $\nabla f(x,t)$.

Summary Across EBMs, diffusion models, and hybrid schemes, the core sampling formula is an overdamped Langevin update, possibly annealed through noise scales. This makes our constrained sampling algorithm SAL compatible with all these approaches as a zero-shot plug-in.

E EXPERIMENTAL DETAILS

E.1 FIELD GENERATION

Baselines Our algorithm is compared to projected Langevin Monte Carlo, primal-dual Monte Carlo and constraint-penalized Langevin Monte Carlo. For the latter, we implement the variable-splitting algorithm of Zhang et al. (2025), and the penalty parameter is a dual variable that is adapted and updated following the same scheme as SAL.

Sampling Langevin Monte Carlo is iterated over 1000 steps, and we set ρ to follow a linear interpolation schedule between 2 and 20.

Constraints The field is subject to energy conservation (Example 2.1). The projection in closed forms. For the penalty method, the penalty cost is $c(x) = (\sum x_i - M)^2$. The primal-dual Langevin Monte Carlo algorithm enforces the constraint function $h(x) = \sum x_i - M$ on average.

E.2 DATA ASSIMILATION

Context In many geophysical and engineering applications, one relies on numerical simulation to predict the time-dependent evolution of a complex system, whose state at physical time t is denoted by $x \in \mathbb{R}^d$. But these models are inherently imperfect—either because of computational constraints or incomplete knowledge of the true dynamics. When real-world observations $y \in \mathbb{R}^m$ become available (for example in digital-twin settings), we assume a statistical model of the form

$$y = h(x) + \varepsilon, \tag{E.1}$$

where $h : \mathbb{R}^d \to \mathbb{R}^m$ is an observation operator and ε is the measurement error. The imperfect simulation yields a prior forecast $b \in \mathbb{R}^d$, the background estimate, which must be adjusted using y to produce a more accurate estimate of the true state, usually referred to as the analysis, as the initial condition for the next simulation. Equivalently, one seeks samples from the posterior

$$p(x \mid b, y) \propto p(y \mid x)\, p(x \mid b). \tag{E.2}$$

This estimation problem is formulated sequentially for each new observation, by propagating the obtained posterior analysis with a forecast model, and repeating the process. Classically, this is achieved by one of three approaches: sequential Monte Carlo methods (e.g. particle filters (Gordon et al., 1993)), ensemble-based filters (e.g. the Ensemble Kalman Filter (Evensen, 2003)), or variational methods that solve for the MAP estimate (e.g. 3D-Var/4D-Var (Sasaki, 1970; Lorenc, 1986)). The 3D-Var algorithm assumes that the background error distribution and observation error distribution are Gaussian,

$$x \mid b \sim \mathcal{N}(b,\, B), \quad \varepsilon \sim \mathcal{N}(0,\, P), \tag{E.3}$$

then taking negative logarithm of (E.2) yields the following optimization target:

$$J(x) = \tfrac{1}{2}\big\|y - h(x)\big\|^2_{P^{-1}} + \tfrac{1}{2}\big\|x - b\big\|^2_{B^{-1}}. \tag{E.4}$$

Deep learning represents a promising tool to learn more complex priors for data assimilation (Huang et al., 2024a; Rozet & Louppe, 2023; Qu et al., 2024; Blanke et al., 2024)

Data For simulating the Burgers equation, we implemented the same method as van Gastelen et al. (2024), but we added an extra constant linear advection term. We work in Fourier space with the first 20 Fourier modes. The field evolves according to the Burgers equation for 4 time units. We generate 1,000 trajectories, with the field recorded at 10 timesteps for each trajectory, with the initial state drawn at random in Fourier space with a power-law decay of the coefficient magnitude.

Learning architecture We implemented a DDPM diffusion model, using the formalism detailed in Appendix D. Diffusion is learned in a latent space, defined as the first 10 Fourier modes. The neural network involved is a fully connected network with depth 3 and width 128, using a cosine time embedding. It is trained for 200 epochs. At sampling time, 1000 diffusion steps are used with $\rho = 10$.

Baselines Our algorithm is compared to unconstrained latent diffusion and Projected diffusion, which incorporate observations using diffusion posterior sampling Rout et al. (2023). We tried to apply penalty guidance diffusion in latent space but did not find a suitable method of incorporating observations in latent space.

Constraints The field is sampled subject to energy and mass conservation constraints (Example 2.1). The projection is computed by alternating projections on the two constraint set, which have closed forms.

The initial conditions are drawn at random following the same distribution of the training data.

Additional results Figure 6 shows the evolution of key metrics for a data assimilation trajectory, for the various methods compared.

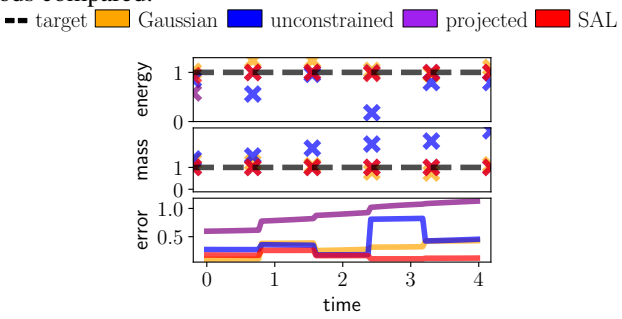

Figure 6: Mass conservation, energy conservation and ℓ_2 error.

E.3 Feasibility problem

Trajectory planning for quadrotor obstacle avoidance is an imoprtant problem Le Hellard et al. (2025). We implement a linearized version of the planar quadrotor dynamics described in (Tedrake, 2009).

Data The trajectories are discretized in time as $(y_1, \dots, y_S, u_1, \dots u_S) \in \mathbb{R}^{2S}$, with $S = 200$ and a time interval $\Delta s = 0.01$. The dynamics constraint

$$\mathcal{C}_{\mathrm{d}} := \{x \mid y(0) = 0, \dot{y}(s) = Ay(s) + Bu(s), |u(s)| \leq u_{\max}\} \tag{E.5}$$

is described by a linear equality constraint, discretized into a linear system, and an inequality constraint on the control inputs. The projection on this convex constraint set is obtained by Dykstra's double projection algorithm (Bílková & Šorel, 2021), and is used within the ADMM solver.

Learning architecture We implemented a DDPM diffusion model, using the formalism detailed in Appendix D. The trajectories signals are learned in a latent Fourier space encoding the first 10 modes of the input signal. The neural network involved is a fully connected network with depth 3 and width 128, using a cosine time embedding. It is trained for 200 epochs. At sampling time, 1000 diffusion steps are used with $\rho = 100$.

Baselines We implement the latent projected diffusion algorithm Zampini et al. (2025), and diffusion guidance with constraint penalties (Carvalho et al., 2023), and propagate the penalty function through the decoder.

Constraint The obstacles are segments, and projecting onto the feasible region is performed by moving the penetrating trajectory portions trajectory either directly above or directly underneath the obstacle. For the penalty method, the constraint penalty is the quadratic distance between the trajectory and the obstacle, which is simple enough to differentiate through in this case.

E.4 Ablation studies for ρ and λ

For the field generation task of Section 5.1, we generate 1000 samples of energy-constrained fields, using different schedules for ρ. For each schedule, we evaluate the samples with the following measure of error: we report the proportion of samples that fall near the unlikely positive mode of the bimodal potential, implying that the sampled distribution deviates from the target

We experiment with 4 different schedules: two schedules use constant ρ throughout the iterations. The two other schedules are linear and logarithmic interpolation between these two values. We run the experiment for different numbers of Langevin iterations. The results are reported in the following table.

number of steps / schedule	constant $\rho = \rho_{\min}$	constant $\rho = \rho_{\max}$	linear	logarithmic
1000	42.4%	22%	0.04%	3.8%
5000	44.7%	2.8%	0.001%	0.0002%

Table 2: Proportion of samples in the wrong mode for different schedules of ρ

We observe that allowing ρ to vary across iterations substantially improves sample quality. With too small ρ, the deviation between x and z is too large. With too largeρ, the chain fails to explore the energy landscape. When the number of steps is limited, only annealed schedules manage to recover the correct mode. This experiment highlights the importance of adaptive schedules in practice.

In the other experiments, we found that the time-varying step size induce by diffusion models, which also scales ρ, was sufficient to balance exploration and constraint satisfaction

We conduct an ablation study on both the field generation experiment (Section 5.1) and the Burgers data assimilation task (Section 5.2) to investigate the influence of the initial value of the dual variables.

In the first experiment, the fields are sampled with the Langevin Monte Carlo algorithm, with fixed potential. In the second experiment, a diffusion model is used, so the score function is time-varying function throughout iterations.

We run our sampling algorithm with Gaussian initialization of λ, with different sizes σ. For each value of σ, we run 100 independent chains. For the flow sampling experiment, we report the maximum number of sampling steps required to converge. For the data assimilation experiment, because the dual problem changes over time, we do not evaluate the convergence of the dual variables. Instead, we report the average reconstruction accuracy.

σ	Steps to convergence	Reconstruction error
0	10	0.56
1	200	0.56
5	800	0.56
10	1000	0.57
20	1600	0.58
50	2500	0.79
100	N/A	1.34

Table 3: Influence of λ

Usage of Large Language Models We used large language models at the sentence level to correct English writing and avoid word repetition