
Trust Region Constrained Measure Transport in Path Space for Stochastic Optimal Control and Inference

Anonymous Author(s)

Affiliation

Address

email

Abstract

1 Solving stochastic optimal control problems with quadratic control costs can be
2 viewed as approximating a target path space measure, e.g. via gradient-based
3 optimization. In practice, however, this optimization is challenging in particular if
4 the target measure differs substantially from the prior. In this work, we therefore
5 approach the problem by iteratively solving constrained problems incorporating
6 trust regions that aim for approaching the target measure gradually in a systematic
7 way. It turns out that this trust region based strategy can be understood as a geomet-
8 ric annealing from the prior to the target measure, where, however, the incorporated
9 trust regions lead to a principled and educated way of choosing the time steps
10 in the annealing path. We demonstrate in multiple optimal control applications
11 that our novel method can improve performance significantly, including tasks in
12 diffusion-based sampling and fine-tuning of diffusion models.

13 1 Introduction

14 Even though the theory of stochastic optimal control (SOC) dates back several decades [2, 20], it
15 has recently attracted renewed interest within the machine learning community. Building on novel
16 formulations that are well-suited for gradient-based optimization (see [14] for an overview) and
17 drawing connections to diffusion models [12, 3, 37], recent work has led to significant progress in
18 the numerical approximation of high-dimensional control problems using neural networks [34, 17].
19 Related problems are crucial in many practical applications, ranging from sampling problems (e.g., in
20 statistical physics [28, 19], Bayesian statistics [33, 22], and reinforcement learning [5]) to fine-tuning
21 of diffusion models [13, 16, 54]. In this work, we aim to further advance SOC approximation methods
22 by taking inspirations from trust region methods used in optimization [38, 1, 45, 49, 35], resulting in
23 a principled framework from the perspective of measure transport in path space.

24 **Stochastic optimal control.** SOC problems (with quadratic control costs) describe optimization
25 problems of the form

$$\min_{u \in \mathcal{U}} \mathbb{E} \left[\int_0^T \left(\frac{1}{2} \|u\|^2 + f \right) (X_s^u, s) ds + g(X_T^u) \right] \quad \text{with} \quad \begin{cases} dX_s^u = (b + \sigma u) (X_s^u, s) ds + \sigma(s) dW_s \\ X_0 \sim p_0, \end{cases} \quad (1)$$

26 where one optimizes the control u of the stochastic differential equation (SDE). Since the law of
27 the SDE solution X^u induces a so-called *path measure* \mathbb{P}^u on the space of continuous trajectories
28 (specifying how likely a certain trajectory is), finding the optimal control is equivalent to finding
29 an optimal target path space measure \mathbb{Q} . From the SOC literature it is known that the likelihood of
30 \mathbb{Q} w.r.t. \mathbb{P}^u can be expressed in closed-form (see [11] and (3) below), which allows to minimize
31 divergences¹ $D(\mathbb{P}^u, \mathbb{Q})$ via gradient-based optimization (also termed *iterative diffusion optimization*).

32 **Trust region methods.** However, if the target \mathbb{Q} is rather different from the initialization \mathbb{P}^{u_0}
33 (typically the uncontrolled process with $u_0 = \mathbf{0}$), many algorithms face challenges with high

¹Note that the cost functional (1) corresponds (up to the normalizing constant) to the reverse Kullback-Leibler (KL) divergence $D = D_{\text{KL}}$.

variances or mode discovery when directly minimizing $D(\mathbb{P}^u, \mathbb{Q})$, especially in high dimensions. To this end, we propose to approach the target measure gradually by a sequence $(\mathbb{P}^{u_i})_i$, where in the i -th step we add the constraint $D_{\text{KL}}(\mathbb{P}^u | \mathbb{P}^{u_{i-1}}) \leq \varepsilon$ to the cost functional (1), with u_{i-1} being the approximated optimal control from the previous iteration and $\varepsilon > 0$ a chosen trust region bound. We prove that the intermediate measures \mathbb{P}^{u_i} define geometric annealing between the prior \mathbb{P}^{u_0} and target measure \mathbb{Q} , where the annealing step-sizes are chosen optimally, in the sense of having an approximately constant change in Fisher-Rao distance (Props. 2.2 and 2.3). Finding an optimal annealing schedule is paramount for the convergence speed of many measure transport and sampling methods [48], and understanding physical processes [44, 8]. While the direct computation of Fisher-Rao distances can be challenging, we show that trust region methods lead to a simple way of obtaining equidistant steps in an information-geometric sense. Moreover, we show that the Lagrangian of the constraint problem can be written as another SOC problem and that the optimal Lagrangian multiplier can be obtained via a dual optimization problem without additional computational overhead (Sec. 2.1). Finally, we adapt successful approaches based on SOC matching [17, 16] and log-variance divergences [34] to the constraint SOC problem to get a practical algorithm (Sec. 2.2).

Applications. The resulting *trust region stochastic optimal control* method can be viewed as an extension of various existing algorithms, yielding significant improvements on a range of applications (Sec. 3). In particular, we consider (i) deep learning approaches to classical SOC problems (extending [34, 17]) enabling the usage of cross-entropy losses in high dimensions, (ii) diffusion-based sampling from unnormalized densities (extending [51, 40]) enabling sampling from high-dimensional, multimodal densities with 100-times fewer target evaluations, (iii) transition path sampling in molecular dynamics (extending [30, 46]) improving transition hit percentages by 21% and reducing target Kabsch-aligned root mean squared distances by 38%, and (iv) reward fine-tuning of text-to-image models (extending [16]) achieving comparable performance with 3-times fewer simulations.

Contributions. Our contributions can be summarized as follows:

- We develop a general framework for solving measure transport with trust regions and apply it to SOC problems using iterative diffusion optimization.
- We prove that our framework leads to a sequence of SOC problems whose solutions define an equispaced annealing between initialization and optimum w.r.t. the Fisher-Rao distance.
- Relying on different loss functionals, we propose two practical instantiations of our framework and demonstrate state-of-the-art performance on a series of applications, ranging from sampling from unnormalized densities to transition path sampling and reward-fine-tuning of text-to-image models.

Notation. We denote by $\mathcal{U} \subset C(\mathbb{R}^d \times [0, T]; \mathbb{R}^d)$ the set of admissible controls and by \mathcal{P} the set of all probability measures on $C([0, T], \mathbb{R}^d)$. We define the path space measure $\mathbb{P} \in \mathcal{P}$ as the law of a \mathbb{R}^d -valued stochastic process $X = (X_t)_{t \in [0, T]}$ and we denote by \mathbb{P}_s the marginal distribution at time s . We refer to App. A for further details on our notation and assumptions.

2 Trust region constrained measure transport for optimal control

The idea of *iterative diffusion optimization* in optimal control based on path space measures is to consider loss functionals of the form

$$\mathcal{L}(u) = D(\mathbb{P}^u, \mathbb{Q}) \quad (2)$$

and minimize them with gradient-descent algorithms [34]. The loss function (2) yields implementable algorithms for SOC problems since the optimal path measure \mathbb{Q} of (1) can be stated explicitly via the Radon-Nikodym derivative

$$\frac{d\mathbb{Q}}{d\mathbb{P}}(X) = \frac{e^{-\mathcal{W}(X, 0)}}{\mathcal{Z}(X_0)} \quad \text{with} \quad \mathcal{W}(X, t) = \int_t^T f(X_s, s) ds + g(X_T), \quad (3)$$

where $\mathcal{Z} := \mathbb{E}[e^{-\mathcal{W}(X, 0)} | X_0]$ and \mathbb{P} is the path measure of the uncontrolled process $X = X^0$; see App. D. In this work, we extend this attempt by using trust regions that shall make sure that the optimization is conducted in a more “regulated” fashion, where the essential idea is to divide the global problem into smaller (reasonably chosen) chunks. We quantify this in Prop. 2.3 below. To this end, we consider the iterative optimization scheme defined by

$$u_{i+1} = \arg \min_{u \in \mathcal{U}} D_{\text{KL}}(\mathbb{P}^u | \mathbb{Q}) \quad \text{s.t.} \quad D_{\text{KL}}(\mathbb{P}^u | \mathbb{P}^{u_i}) \leq \varepsilon, \quad (4)$$

for any $i \in \mathbb{N}$, where $\varepsilon > 0$ defines a trust region w.r.t. to the previous control iterate and where we often set $u_0 = \mathbf{0}$ (and thus $\mathbb{P}^{u_0} = \mathbb{P}$). This corresponds to dividing the overall optimization problem into parts according to their distance measured in the KL divergence between the respective

preceding and succeeding path measures. Due to the convexity of the KL divergence, we can show that in all but the last step we actually have an equality constraint in (4); see App. J.1. Thus, there exists an $I \in \mathbb{N}$ such that $u_I = u^*$ is the optimal control of the global control problem defined in (1).

Remark 2.1 (Controlling the variance of importance weights). The constraint $D_{\text{KL}}(\mathbb{P}^u | \mathbb{P}^{u_i}) \leq \varepsilon$ can be motivated by the goal to control the variance of importance weights $\text{Var}_{\mathbb{P}^{u_i}}(d\mathbb{P}^{u_{i+1}}/d\mathbb{P}^{u_i})$, which can be explained by the inequality $\text{Var}_{\mathbb{P}^{u_i}}(d\mathbb{P}^{u_{i+1}}/d\mathbb{P}^{u_i}) \geq e^{D_{\text{KL}}(\mathbb{P}^{u_{i+1}} | \mathbb{P}^{u_i})} - 1$, see, e.g., [24]. For small ε (which is a common choice in practice) we typically observe $\text{Var}_{\mathbb{P}^{u_i}}(d\mathbb{P}^{u_{i+1}}/d\mathbb{P}^{u_i}) \approx 2\varepsilon$ (see App. E.3), which can be explained by a Taylor expansion and assuming that $d\mathbb{P}^{u_{i+1}}/d\mathbb{P}^{u_i} \approx 1$. Low variance of importance weights is directly related to efficiency of many measure transport methods and too high variance makes it practically impossible to obtain reliable results. Note also that the reverse KL divergence allows for explicit expressions for the resulting constrained problem (see Sec. 2.1) and we leave alternative divergences for future research.

In practice, under suitable regularity assumptions, we can approach the above constrained optimization problem using a relaxed Lagrangian formalism. To this end, we consider the loss functionals

$$\mathcal{L}_{\text{TR}}^{(i)}(u, \lambda) = D_{\text{KL}}(\mathbb{P}^u | \mathbb{Q}) + \lambda (D_{\text{KL}}(\mathbb{P}^u | \mathbb{P}^{u_i}) - \varepsilon), \quad (5)$$

where $\lambda > 0$ is a Lagrange multiplier, and solve the saddle point problems

$$\max_{\lambda \geq 0} \min_{u \in \mathcal{U}} \mathcal{L}_{\text{TR}}^{(i)}(u, \lambda). \quad (6)$$

We note that $\mathcal{L}_{\text{TR}}^{(i)}$ is convex in u by convexity of the KL divergence (see App. J.1) and concave in λ since it can be expressed as the pointwise minimum $\min_u \mathcal{L}_{\text{TR}}^{(i)}(u, \lambda)$ among a family of linear functions of λ . Thus, (6) has unique optima which we denote by u_{i+1} and λ_i , respectively. We can now show the following evolution of the optimal measures.

Proposition 2.2 (Optimal change of measure as geometric annealing). *Let \mathbb{Q} be the optimal path measure defined in (3). The intermediate optimal path measures corresponding to (4) then satisfy*

$$\frac{d\mathbb{P}^{u_{i+1}}}{d\mathbb{P}^{u_i}} \propto \left(\frac{d\mathbb{Q}}{d\mathbb{P}^{u_i}} \right)^{\frac{1}{1+\lambda_i}} \quad (7)$$

and the optimal change of measure w.r.t. the base measure \mathbb{P} is given by²

$$\frac{d\mathbb{P}^{u_i}}{d\mathbb{P}}(X) \propto \left(\frac{d\mathbb{Q}}{d\mathbb{P}}(X) \right)^{\beta_i} \left(\frac{d\mathbb{P}^{u_0}}{d\mathbb{P}}(X) \right)^{1-\beta_i} \quad \text{with} \quad \beta_i = 1 - \prod_{j=0}^{i-1} \frac{\lambda_j}{1+\lambda_j}. \quad (8)$$

Proof. The first statement follows by the definition of the Lagrangian and the second follows by induction; see App. B. \square

Note that the sequence $(\beta_i)_i$ is monotonically increasing with values in $[0, 1]$, where we have $\beta_0 = 0$ and $\beta_I = 1$ (as $\lambda_{I-1} = 0$ due to optimality). Thus, the formula in (8) can be seen as a geometric annealing from the prior to the target measure. Note that when $u_0 = \mathbf{0}$, the second factor vanishes. Importantly, the step-size of the annealing is automatically chosen such that we obtain a well-behaved sequence of distributions; see also Fig. 1.

Proposition 2.3 (Equidistant steps on statistical manifold). *Up to higher-order terms in ε , the sequence of measures \mathbb{P}^{u_i} , $i \in \{0, \dots, I-1\}$, are equispaced in the Fisher-Rao distance.*

Proof. By Prop. 2.2, we obtain $\varepsilon = D_{\text{KL}}(\mathbb{P}^{u_{i+1}} | \mathbb{P}^{u_i}) = \frac{\Delta_i^2}{2} \mathcal{I}(\beta_i) + O(\Delta_i^3)$, where $\Delta_i = \beta_{i+1} - \beta_i$ and $\mathcal{I}(\beta_i)$ is the Fisher information. The Fisher-Rao distance between \mathbb{P}^{u_i} and $\mathbb{P}^{u_{i+1}}$ is then given by $\int_{\beta_i}^{\beta_{i+1}} \sqrt{\mathcal{I}(\tau)} d\tau = \sqrt{\mathcal{I}(\beta_i)} \Delta_i + O(\Delta_i^2) = \sqrt{2\varepsilon} + O(\Delta_i^{3/2})$; see App. I for details. \square

Remark 2.4 (Trust regions for general measures). The observant reader has likely noticed that so far all our arguments do not rely on the fact that we consider path space measures, but work for general probability measures. We could therefore as well write our trust region method stated in (4) as

$$\mathbb{P}_{i+1} = \arg \min_{\mathbb{P} \in \mathcal{P}} D_{\text{KL}}(\mathbb{P} | \mathbb{Q}) \quad \text{s.t.} \quad D_{\text{KL}}(\mathbb{P} | \mathbb{P}_i) \leq \varepsilon. \quad (9)$$

We refer to App. K for a treatment when the measures admit densities on \mathbb{R}^d , which can, e.g., be considered for variational inference with normalizing flows.

²As usual, the empty product is defined as 1 such that $\beta_0 = 1$.

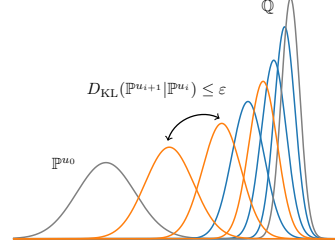


Figure 1: Illustration of a sequence of distributions $(\mathbb{P}^{u_i})_i$ resulting from our trust region method (orange) and a measure transport corresponding to non-equispaced geometric annealing (blue), leading to high variance in the importance weights for the initial steps.

2.1 Constrained stochastic optimal control

While the above formulation in principle works for arbitrary measures, in this work we focus on path space measures corresponding to optimal control problems. In this setting we can compute some of the objectives more explicitly and recover helpful relations.

Lagrangian as SOC problem. First, note that, using the Girsanov theorem (see App. A.2), it turns out that, for a fixed Lagrange multiplier λ , the Lagrangian in (5) defines another SOC problem, i.e.,

$$\mathcal{L}_{\text{TR}}^{(i)}(u, \lambda) = \mathcal{L}_{\text{TRC}}^{(i)}(u, \lambda) - \lambda\varepsilon, \quad (10)$$

where³

$$\mathcal{L}_{\text{TRC}}^{(i)}(u, \lambda) = \mathbb{E} \left[\int_0^T \left(\frac{1+\lambda}{2} \|u - \frac{\lambda}{1+\lambda} u_i\|^2 + \frac{\lambda}{2(1+\lambda)} \|u_i\|^2 + f \right) (X_s^u, s) ds + g(X_T^u) \right] \quad (11)$$

and X^u is still defined as in (1); see App. J.4 for details. Note that this cost functional is more general than then one stated in (1), which one recovers when setting $\lambda = 0$. We can show that this SOC problem satisfies the following optimality conditions.

Proposition 2.5 (Optimality for trust region SOC problems). *For fixed λ , let us define by*

$$V_{i+1}^\lambda(x, t) := \inf_{u \in \mathcal{U}} \mathbb{E} \left[\int_0^T \left(\frac{1+\lambda}{2} \|u - \frac{\lambda}{1+\lambda} u_i\|^2 + \frac{\lambda}{2(1+\lambda)} \|u_i\|^2 + f \right) (X_s^u, s) ds + g(X_T^u) \middle| X_t = x \right]$$

the value function of the SOC problem $\inf_{u \in \mathcal{U}} \mathcal{L}_{\text{TRC}}^{(i)}(u, \lambda)$ corresponding to (11) and by u_{i+1}^λ its solution. Then it holds that

$$(i) \text{ (Estimator for value function) } V_{i+1}^\lambda(x, t) = -(1 + \lambda) \log \mathbb{E} \left[e^{-\frac{1}{1+\lambda} \mathcal{W}_i(X^{u_i}, t)} \middle| X_t^{u_i} = x \right],$$

$$\text{where } \mathcal{W}_i(X^{u_i}, t) = \int_t^T \frac{1}{2} \|u_i(X_s^{u_i}, s)\|^2 ds + \int_t^T u_i(X_s^{u_i}, s) \cdot dW_s + \mathcal{W}(X^{u_i}, t).$$

$$(ii) \text{ (Connection between solution and value function) It holds } u_{i+1}^\lambda = \frac{\lambda}{1+\lambda} u_i - \frac{1}{1+\lambda} \sigma^\top \nabla V_{i+1}^\lambda.$$

Proof. The statements can be proven using the verification theorem; see App. J.4 for details. \square

We note that Prop. 2.2, the Girsanov theorem, and (3) relate the functional \mathcal{W}_i in Prop. 2.5 to the importance weights

$$\frac{d\mathbb{Q}}{d\mathbb{P}^{u_i}}(X^{u_i}) \propto e^{-\mathcal{W}_i(X^{u_i}, 0)} \quad \text{and} \quad \frac{d\mathbb{P}^{u_{i+1}}}{d\mathbb{P}^{u_i}}(X^{u_i}) \propto e^{-\frac{1}{1+\lambda} \mathcal{W}_i(X^{u_i}, 0)}. \quad (12)$$

Dual problem for Lagrange multiplier. Next, we will outline how to find the optimal Lagrange multiplier λ in (6) in the SOC setting. Plugging the optimal control u_{i+1}^λ in the Lagrangian (10) yields the dual function $\mathcal{L}_{\text{Dual}}^{(i)} \in C(\mathbb{R}, \mathbb{R})$ given by

$$\mathcal{L}_{\text{Dual}}^{(i)}(\lambda) := \mathcal{L}_{\text{TR}}^{(i)}(u_{i+1}^\lambda, \lambda) = \mathcal{L}_{\text{TRC}}^{(i)}(u_{i+1}^\lambda) - \lambda\varepsilon. \quad (13)$$

We note that evaluating the SOC problem in (11) at the optimal control can be expressed via the value function given in Prop. 2.5, which yields

$$\mathcal{L}_{\text{Dual}}^{(i)}(\lambda) = \mathbb{E} \left[V_{i+1}^\lambda(X_0^{u_i}, 0) \right] - \lambda\varepsilon = -(1 + \lambda) \log \mathbb{E} \left[e^{-\frac{1}{1+\lambda} \mathcal{W}_i(X^{u_i}, 0)} \right] - \lambda\varepsilon, \quad (14)$$

where we note that the expression in the expectation is proportional to the importance weights in (12). Note that we can obtain a Monte Carlo estimate of the dual function using only simulations X^{u_i} from the previous iterations. As it turns out, these simulations are in most cases already required when learning the control u_{i+1} and we can thus store them in a buffer. We can then obtain $\lambda_i = \arg \max_{\lambda \in \mathbb{R}^+} \mathcal{L}_{\text{Dual}}^{(i)}(\lambda)$ using any non-linear solver or dual descent method with minimal computational overhead.

In theory, we can then define $u_{i+1} = u_{i+1}^{\lambda_i}$ using the representations in Prop. 2.5 and proceed with the next iteration of our trust region method in (4). However, computing the optimal control u_{i+1} using the representations in Prop. 2.5 requires gradients and Monte Carlo estimators of the value functions. This is problematic since it relies on a large amount of samples for each state x due to the (typically) very high variance of the estimator; see App. C for details. Thus, we propose versions of iterative diffusion optimization to learn parametrized approximations to u_{i+1} in the next section.

³The SOC problem is slightly more general than (1) due to the shift in the quadratic cost.

Algorithm 1 Trust Region SOC with buffer (see App. J.2 for details)

Require: Initial path measure \mathbb{P}^{u_0} , target path measure \mathbb{Q} , divergence D , termination threshold δ
while $\lambda_i > \delta$ **do**
 Sample trajectories $X \sim \mathbb{P}^{u_i}$ by integrating the SDE in (1) with Brownian motion W and control u_i
 Compute importance weights $w = \frac{d\mathbb{Q}}{d\mathbb{P}^{u_i}}(X^{u_i}) \propto \exp(-\mathcal{W}_i(X^{u_i}, 0))$ as in (12)
 Initialize buffer $\mathcal{B} = \{W, X, w\}$
 Compute multiplier $\lambda_i = \arg \max_{\lambda \in \mathbb{R}^+} \mathcal{L}_{\text{Dual}}^{(i)}(\lambda)$ as in (14) using \mathcal{B} and a 1-dim. non-linear solver
 Compute $u^{i+1} = \arg \min_u D(\mathbb{P}^u, \mathbb{P}^{u_{i+1}})$ using \mathcal{B} and $\frac{d\mathbb{P}^{u_{i+1}}}{d\mathbb{P}^u} \propto w^{\frac{1}{1+\lambda_i}} \frac{d\mathbb{P}^{u_i}}{d\mathbb{P}^u}$ as in Sec. 2.2
return control u^i with $\mathbb{P}^{u_i} \approx \mathbb{Q}$

2.2 Learning the constrained optimal control

In this section we propose strategies to learn the optimal control for each iteration. As before, the general idea is to minimize loss functionals based on divergences between path space measures, namely $\mathcal{L}(u) = D(\mathbb{P}^u, \mathbb{P}^{u_{i+1}})$. Such divergences often rely on the Radon-Nikodym derivative

$$\begin{aligned} \frac{d\mathbb{P}^{u_{i+1}}}{d\mathbb{P}^u}(X^{u_i}) &= \frac{d\mathbb{P}^{u_{i+1}}}{d\mathbb{P}^{u_i}}(X^{u_i}) \frac{d\mathbb{P}^{u_i}}{d\mathbb{P}^u}(X^{u_i}) \\ &\propto \exp\left(\int_0^T \frac{\|u_i - u\|_2^2}{2}(X_s^{u_i}, s) ds + \int_0^T (u_i - u)(X_s^{u_i}, s) \cdot dW_s - \frac{\mathcal{W}_i(X^{u_i}, 0)}{1+\lambda_i}\right), \end{aligned} \quad (15)$$

where we used Girsanov's theorem and (12). Note that the Radon-Nikodym derivative in (15) depends only on samples of the process with the already learned u_i . Let us now suggest two concrete divergences. Those divergences are desirable for high-dimensional problems since both do not rely on computing derivatives of the stochastic process and can be optimized “off-policy” using trajectories X^{u_i} with the control u_i of the previous iteration, which can be stored in a buffer; see Algorithm 1.

Log-variance divergence. This divergence can be considered w.r.t. an arbitrary reference measure, where we choose \mathbb{P}^{u_i} for convenience [34, 41]. We can then define the loss functional

$$\mathcal{L}_{\text{LV}}(u) := \text{Var} \left[\log \left(\frac{d\mathbb{P}^{u_{i+1}}}{d\mathbb{P}^u}(X^{u_i}) \right) \right], \quad (16)$$

where the Radon-Nikodym derivative can be explicitly computed as in (15). Note that for $\lambda_i = 0$, this loss reduces to the on-policy log-variance loss typically used in the literature [40]. While this loss has beneficial theoretical properties [34], it requires to keep the full trajectory in memory for the gradient computation.

Cross-entropy divergence and SOC matching. Alternatively, we can consider the cross-entropy loss (i.e., the forward KL divergence computed using reweighting)

$$\mathcal{L}_{\text{CE}}(u) := D_{\text{KL}}(\mathbb{P}^{u_{i+1}} | \mathbb{P}^u) = \mathbb{E} \left[\left(\log \frac{d\mathbb{P}^{u_{i+1}}}{d\mathbb{P}^u}(X^{u_i}) \right) \frac{d\mathbb{P}^{u_{i+1}}}{d\mathbb{P}^{u_i}}(X^{u_i}) \right], \quad (17)$$

where the Radon-Nikodym derivative is again given by (15). Contrary to the log-variance loss, the reweighting $\frac{d\mathbb{P}^{u_{i+1}}}{d\mathbb{P}^{u_i}}$ in (12) induces exponential terms. Our trust region constrain makes sure, however, that the variance of those weights stays bounded, see Remark 2.1.

To efficiently compute this loss, we define the so-called (*lean*⁴) *adjoint state* a as in [16] via

$$\frac{d}{ds} a_{i+1}(X_s, s) = - \left[(\nabla b(X_s, s))^\top a_{i+1}(X_s, s) + \beta_{i+1} \nabla f(X_s, s) \right] \quad (18)$$

with $a_{i+1}(X_T, T) = \beta_{i+1} \nabla g(X_T)$, satisfying $a_{i+1}(X_s, s) = \nabla_{X_s} \beta_{i+1} \mathcal{W}(X_s, s)$; see [16, Lemma 5] and observe that it differs from the standard lean adjoint by the factor β_i defined in Prop. 2.2. Similar to [17], we can use the expression for the optimal control in Prop. 2.5 and the Girsanov theorem to arrive at the *SOC matching loss*⁵, a simple regression objective given by

$$\mathcal{L}_{\text{SOCM}}(u) := \mathbb{E} \left[\frac{1}{2} \int_0^T \|\sigma^\top a_{(i+1)}(X_s^{u_i}, s) - u(X_s^{u_i}, s)\|^2 ds \frac{d\mathbb{P}^{u_{i+1}}}{d\mathbb{P}^{u_i}}(X^{u_i}) \right], \quad (19)$$

see App. J.7 for details. Contrary to the log-variance divergence above, this objective does not require to keep the whole trajectory X^{u_i} in memory for backpropagation but can be computed at times $t \sim \text{Unif}([0, T])$ using a Monte Carlo approximation. We summarize our algorithm in (1) and compare the different losses against existing approaches for SOC problems in the next section.

⁴Instead of the uncontrolled process X , we could also express the adjoint state w.r.t. the process X^u ; however, this relies on more costly vector-Jacobian products; see App. J.6.

⁵The loss is similar to the SOCM-Adjoint loss in [17], which, however, involves matrix-valued functions.

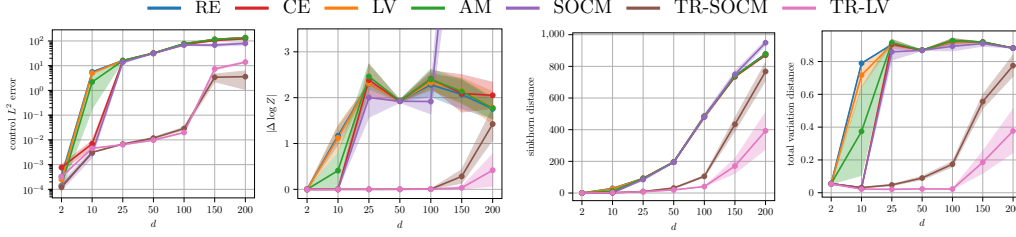


Figure 2: Performance criteria for a Gaussian mixture target density with varying dimension d , averaged across four seeds. We show the errors of estimating the optimal control, the log-normalization constant, as well as the Sinkhorn and total variation distances over different dimensions (from left to right). We observe that our trust region methods (TR-SOCM and TR-LV) are the only methods scaling to high dimensions.

3 Applications

In this section, we explore several applications of SOC, comparing our novel trust-region-based optimization algorithm against existing methods. Specifically, we consider three key tasks involving sampling from unnormalized densities, transition path sampling, and fine-tuning text-to-image models. For background information, detailed experimental setups, and additional results, we refer to Apps. E, F and H, respectively. We also include further experiments on classical SOC problems in App. G.

3.1 Diffusion-based sampling

Using (3), we can show that sampling problems can be reformulated as SOC problems. To this end, we leverage the following corollary showing that the terminal distributions \mathbb{Q}_T and \mathbb{P}_T of the optimally controlled and uncontrolled processes differ by a tilting.

Corollary 3.1 (Sampling from tilted distributions). *Let us set $f = 0$ and assume that the terminal distribution of the uncontrolled process X is independent of p_0 and admits a density denoted by \mathbb{P}_T . Then it holds that $\mathbb{Q}_T \propto \mathbb{P}_T e^{-g}$.*

Proof. Using (3) it holds that $\frac{d\mathbb{Q}}{d\mathbb{P}}(X) = \frac{e^{-g(X_T)}}{\mathcal{Z}(X_0)}$ with $\mathcal{Z}(X_0) = \mathbb{E}[e^{-g(X_T)}|X_0]$. The results follows from the independence of X_T and X_0 ; see [16] and App. E for details. \square

Cor. 3.1 shows that the optimally controlled process X^{u^*} samples from a given unnormalized density ρ_{target} when using an uncontrolled process with known terminal distribution \mathbb{P}_T and setting $g = \log \frac{\mathbb{P}_T}{\rho_{\text{target}}}$; see [10, 50, 39, 60, 52, 51, 59, 40, 53] and App. E for details. Such sampling problems are of immense practical interest, with numerous applications in the natural sciences [62], in Bayesian statistics [22], and reinforcement learning [5].

Numerical experiments. Here, we compare existing methods for solving SOC problems with our trust-region method on challenging multimodal sampling problems. We use the *Denoising Diffusion Sampler* (DDS) [51] method, which leverages an ergodic Ornstein–Uhlenbeck process initialized at its equilibrium measure as uncontrolled process X . We consider five baselines, specifically, reverse and (importance weighted) forward KL, also known as *relative entropy* (RE) and *cross entropy* (CE) method, respectively. Additionally, we consider the *log-variance loss* [41], *adjoint matching* (AM) [16], and *stochastic optimal control matching* (SOCM) [17], for the unconstrained problem in (2); see [14] for a comprehensive overview of SOC losses. In all experiments, we deliberately avoid using gradient guidance from the target density in the diffusion process, often referred to as Langevin preconditioning (LP) [27]. Prior work has shown that LP is essential for preventing mode collapse in neural samplers [4, 27]. However, LP is computationally expensive, as it requires querying the target distribution at every discretization step, making such approaches impractical for many problems where evaluating the target gradient is costly.

First, we consider a *Gaussian Mixture Model* (GMM) comprising 10 components and randomized mixing weights. GMMs are particularly compelling as they admit an analytical solution for the optimal control, which enables direct computation of the L^2 error between the learned and optimal controls, a reliable metric for detecting mode collapse. In addition, we assess the *Sinkhorn distance* [9] between samples from the target and the model, and the absolute error in estimating the log-normalization constant, denoted $|\Delta \log \mathcal{Z}|$. Finally, we evaluate the *total variation distance* between the true mixing weights and the model’s estimated weights. The results, shown in Fig. 2, indicate that for $d = 2$, all methods closely approximate the optimal control. However, for dimensions beyond $d = 10$, most methods suffer from mode collapse, as reflected by increased control errors, except for

	$d = 2$	$d = 50$	$d = 100$	$d = 200$
RE	1.364 ± 0.002	3.443 ± 0.004	3.077 ± 0.669	2.908 ± 0.679
CE	0.001 ± 0.000	0.202 ± 0.159	0.526 ± 0.181	0.641 ± 0.527
LV	<i>diverged</i>	1.363 ± 0.325	1.809 ± 0.737	1.958 ± 0.698
AM	1.364 ± 0.002	3.432 ± 0.020	3.457 ± 0.019	3.322 ± 0.307
SOCM	0.001 ± 0.000	2.958 ± 0.831	2.971 ± 0.846	3.504 ± 0.005
TR-LV	0.000 ± 0.000	0.000 ± 0.000	0.002 ± 0.002	0.002 ± 0.001

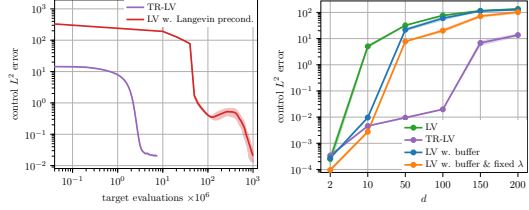


Figure 3: The left table reports $|\Delta \log \mathcal{Z}|$ values for the *Many Well* target across different dimensions d . The middle plot compares the log-variance loss of our trust-region method (TR-LV) with that of Langevin preconditioning on the GMM target with $d = 100$. The rightmost figure presents an ablation analysis of key components in our method, highlighting the importance of trust regions in preventing mode collapse and achieving low control error. All results are averaged across four seeds.

those employing trust-region updates. Trust-region methods maintain robustness across a wide range of dimensions and only begin to show signs of mode collapse at high dimensionalities ($d \geq 150$).

We additionally evaluate our method on the *Many Well* target [55] with 32 modes. For quantitative analysis, we report the log-normalization error $|\Delta \log \mathcal{Z}|$, as other ground-truth quantities are unavailable. Additionally, for the high-dimensional case $d = 200$, we visualize pairs of marginal distributions in App. E. The results, presented in Fig. 3, demonstrate that our method significantly outperforms competing approaches in normalization estimation. Furthermore, the visualizations in App. E illustrate that trust-region updates effectively prevent mode collapse, even in high dimensions. In contrast, baseline methods either suffer from mode collapse or fail to converge.

Finally, we perform an ablation study on the GMM target, analyzing key components of our proposed method. Specifically, we investigate the effects of incorporating a replay buffer and applying trust-region optimization. To this end, we compare a variant using a fixed Lagrangian multiplier λ , selected via hyperparameter tuning, with one in which λ is dynamically optimized using the trust-region approach. Additionally, we evaluate the log-variance loss both with and without using a replay buffer. Moreover, we conduct another study where we compare our method to LV with Langevin preconditioning on the GMM target with dimensionality $d = 100$. The results, shown in Figure 3, demonstrate that trust-region optimization significantly reduces control error and decreases the number of target evaluations by several orders of magnitude.

3.2 Transition path sampling

Transition path sampling is of great importance to study phase transitions and chemical reactions. The key challenge comes from the energy barrier that connects two sets A and B along the energy landscape, which makes direct sampling of transition paths extremely unlikely. These problems can also be formulated as SOC problems [47, 26, 23]. Specifically, we set $b = -\nabla U$, where $U : \mathbb{R}^{N \times 3} \rightarrow \mathbb{R}$ is the potential energy function, and $g = -\log \mathbf{1}_B$ as well as $p_0 \propto \mathbf{1}_A$, which constraints the initial and target states in the sets A and B . As in (3), it holds that $\frac{dQ}{dP} = \frac{\mathbf{1}_B(X_T)}{\mathcal{Z}(X_0)}$. Recent work has leveraged neural networks to parameterize a bias force to solve the corresponding SOC problem, employing objectives such as cross-entropy and log-variance [30, 58], as well as KL divergence [58, 18].

Numerical experiments. We evaluate the performance of the trust-region-based log-variance loss (TR-LV) on two transition path sampling problems: Alanine Dipeptide isomerization and Chignolin folding, with 22 and 138 atoms, respectively.

Our evaluation includes three metrics: *Kabsch-aligned root mean squared distance (RMSD)* between the final states of sampled paths and the target state, *transition hit percentage (THP)* measuring the proportion of final states hitting within the target region, and *energy of transition state (ETS)* identifying the highest energy values along paths that reach the target.

We compare our method to standard molecular dynamics (MD) with increased temperature (UMD), steered MD (SMD) [31] with force applied to collective variables, and PIPS [30] which uses the cross-entropy loss. We also include TPS-DPS [46] as a key baseline, which employs an (unconstrained) log-variance loss to formulate TPS as a stochastic optimal control (SOC) problem. Further experimental details are provided in App. F.

Table 1 shows that TR-LV achieves superior target state RMSD and transition hit percentage compared to the standard log-variance objective (TPS-DPS) for both molecular systems. Notably, SMD performs

Table 1: Quantitative evaluation on transition path sampling problems. † denotes that results are taken from [46]. The results for TPS-DPS and TR-LV are averaged across three seeds.

Method	RMSD (Å, ↓)	THP (% , ↑)	ETS (kJ/mol)	Method	RMSD (Å, ↓)	THP (% , ↑)	ETS (kJ/mol)
Alanine Dipeptide				Chignolin			
UMD (3600K)†	1.19 ± 0.32	6.25	812.47 ± 148.80	UMD (1200K)†	7.23 ± 0.93	1.56	388.17
SMD†	0.56 ± 0.27	54.69	78.40 ± 12.76	SMD†	0.85 ± 0.24	34.38	179.52 ± 138.87
PIPS†	0.66 ± 0.15	43.75	28.17 ± 10.86	PIPS†	4.66 ± 0.17	0.00	-
TPS-DPS	0.47 ± 0.18	39.58 ± 28.13	46.34 ± 10.16	TPS-DPS	1.06 ± 0.08	25.00 ± 10.69	-189.91 ± 23.01
TR-LV	0.29 ± 0.03	61.25 ± 4.05	49.11 ± 5.84	TR-LV	0.90 ± 0.01	43.95 ± 5.64	-303.98 ± 28.65

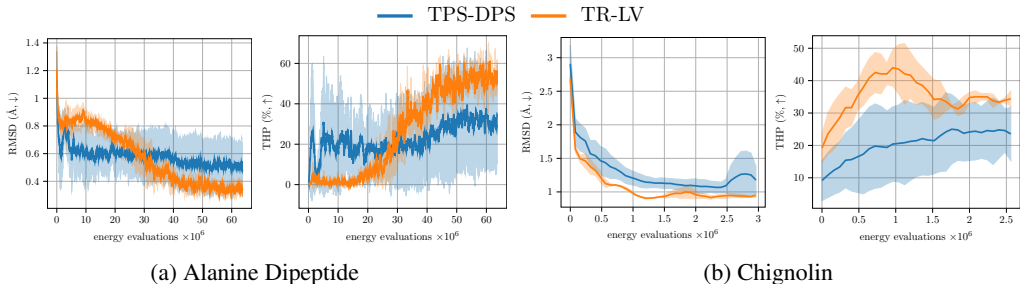


Figure 4: We compare our trust-region method (TR-LV) with Diffusion Path Sampler (TPS-DPS) [46] on Alanine Dipeptide and Chignolin. All results are averaged over three random seeds, with both the mean and standard deviation reported. Our method identifies transition paths more consistently and robustly, as evidenced by higher THP values and lower standard deviations.

well due to its use of collective variables with biased force guiding the sampling process. Figure 4 illustrates that the trust-region constraint leads to significantly more robust training compared to TPS-DPS as indicated by low standard deviations across different seeds. Moreover, on Alanine Dipeptide, the trust-region constraint initially regularizes optimization and accelerates convergence thereafter. Across both systems, the trust-region constraint significantly enhances training stability and performance.

3.3 Fine-tuning of diffusion models

Interpreting $-g$ as a *reward* and the uncontrolled process X as a pretrained diffusion model (i.e., b includes the pretrained neural network), Cor. 3.1 shows that we can perform reward fine-tuning by solving the SOC problem in (1); see also [13, 54, 16]. Reward fine-tuning has recently shown impressive results, e.g., in image [16, 6] and molecule generation [13], and SOC provides a principled framework. A special case is given by posterior sampling [13]. Setting $g = -\log p(y|x)$, where $p(y|x)$ is the likelihood and we interpret \mathbb{P}_T as a learned (*diffusion*) prior $p(x)$, Bayes' theorem shows that the optimally controlled process samples from the posterior $p(x|y)$.

Numerical experiments. We perform reward fine-tuning on Stable Diffusion 1.5 [42], using ImageReward [57], which is a reward model designed to capture prompt alignment and image quality according to human preferences. We take the adjoint matching (AM) method as baseline, and compare it against our TR-SOCM loss (19), keeping all other hyperparameters fixed. Our TR-SOCM allows the principled use of buffers, and we perform three passes on each buffer of size 500, leading to three times fewer rollouts for a fixed number of model updates. For faster convergence, we use a modified version of TR-SOCM with annealing factor $\beta_i = 1$. For each algorithm, we evaluate 5 checkpoints during fine-tuning (with ODE and SDE inference) on ImageReward and three additional metrics: CLIP-Score [29], which measures prompt alignment, Human Preference Score [56], which measures human-perceived image quality, and Dreamsim diversity [21], which measures per-prompt diversity. We observe that TR-SOCM achieves similar performance metrics to AM at a fraction of the cost, as sampling the rollouts and solving the lean adjoint ODE, which dominates the compute usage, is amortized over buffer passes; see Figs. 5 and 6, and App. H for more details.

4 Related works

In this section, we discuss the most related works. We compare our approach to existing methods for solving SOC problems and provide a more extensive comparison in App. C.

Iterative diffusion optimization. Recently developed methods approach SOC problems by simulating the (diffusion) process X^u , computing a suitable cost function, and optimizing the parameters of the control function u using variants of stochastic gradient methods. These techniques are col-

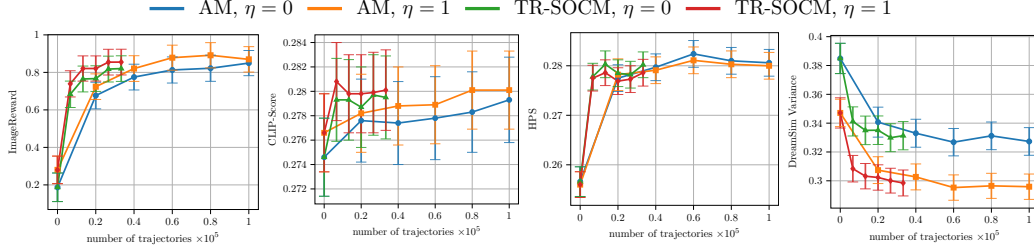


Figure 5: Comparison of Adjoint Matching against Trust Region SOCM for Stable Diffusion 1.5 fine-tuning w.r.t. four quality metrics, where $\eta = 0$ and $\eta = 1$ refer to ODE (DDIM) and SDE (DDPM) inference, respectively.

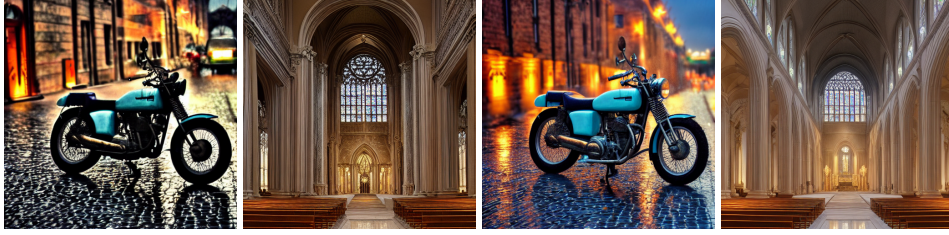


Figure 6: Comparison between images generated by the base Stable Diffusion 1.5 model (left) and its version fine-tuned with TR-SOCM (right), using the same prompts and noise seeds. See prompts in App. H.

lectively referred to as *iterative diffusion optimization* (IDO) methods. While the theory underlying these objectives dates back to [36, 10], combinations with deep learning in the context of SOC have been explored by [60, 39, 63, 51, 3, 40, 53]. One can derive most of these objectives starting from the Radon-Nikodym derivative $\frac{d\mathbb{P}^u}{d\mathbb{Q}}(X^u)$ as in (12) (with $u = u_i$). One can then minimize a loss based on a suitable divergences as in (2). Previous works have, e.g., proposed the log-variance divergence [40, 46] or the forward KL divergence (corresponding to the cross-entropy loss [25, 32, 43, 61, 30]), and we develop the trust region versions in (16) and (17). The SOC matching loss [17], which we extended to trust regions in (19), is equal to the cross entropy loss in expectation but exhibits lower variance empirically. We refer to [15] for more IDO losses. However, all existing methods have either directly tackled the target measure \mathbb{Q} or relied on a form of hand-tuned annealing.

Trust region methods. We show how IDO methods can generally be extended to trust region methods, enabling (1) automatic control on the variance of the importance weights and (2) principled usage of buffers, leading to faster and more stable convergence, in particular avoiding mode collapse in high dimensions. Trust region methods have a long history as robust optimization algorithms that iteratively minimize an objective within an adaptively sized “trust region”; see [7] for an overview. These methods have also been extended to optimize over spaces of probability distributions, particularly in reinforcement learning (e.g., *Natural Policy Gradients* and *Trust Region Policy Optimization*) and stochastic search problems [38, 1, 45, 49, 35]. However, to the best of our knowledge, they have not yet been extended to path measures.

5 Conclusion

In this work, we develop a novel framework for solving stochastic optimal control (SOC) problems using deep learning. Such problems can be reformulated as finding an optimal path measure induced by a controlled stochastic differential equation. We divide the unconstrained problem into a sequence of constrained optimization problems by bounding the KL divergence between adjacent measures. We show that this defines a well-behaved geometric annealing between the initialization and the target measure resulting in equidistant steps on the information manifold. Each intermediate problem turns out to be an adapted SOC problem that can be efficiently solved without simulations by using a buffer of trajectories with the control of the previous iteration. In our experiments, we show that this leads to significantly reduced number of target evaluations for sampling high-dimensional multimodal unnormalized targets. In particular, we stabilize and improve existing methods for transition path sampling in molecular dynamics. Finally, we show that our method can be scaled to improve the efficiency of reward fine-tuning for text-to-image diffusion models. We expect our framework to also improve other applications of SOC. Moreover, our framework can in theory be extended to other divergences for the constraints. Finally, our results for general measures motivate to use trust region methods for other learned measure transports, e.g., normalizing flows.

References

- [1] A. Abdolmaleki, R. Lioutikov, J. R. Peters, N. Lau, L. Pualo Reis, and G. Neumann. Model-based relative entropy stochastic search. *Advances in Neural Information Processing Systems*, 28, 2015.
- [2] R. Bellman. *Dynamic programming*. Princeton University Press, 1957.
- [3] J. Berner, L. Richter, and K. Ullrich. An optimal control perspective on diffusion-based generative modeling. *arXiv preprint arXiv:2211.01364*, 2022.
- [4] D. Blessing, X. Jia, J. Esslinger, F. Vargas, and G. Neumann. Beyond elbos: A large-scale evaluation of variational methods for sampling. *arXiv preprint arXiv:2406.07423*, 2024.
- [5] O. Celik, Z. Li, D. Blessing, G. Li, D. Palanicek, J. Peters, G. Chalvatzaki, and G. Neumann. DIME: Diffusion-based maximum entropy reinforcement learning. *arXiv preprint arXiv:2502.02316*, 2025.
- [6] K. Clark, P. Vicol, K. Swersky, and D. J. Fleet. Directly fine-tuning diffusion models on differentiable rewards. In *The Twelfth International Conference on Learning Representations*, 2024.
- [7] A. R. Conn, N. I. Gould, and P. L. Toint. *Trust region methods*. SIAM, 2000.
- [8] G. E. Crooks. Measuring thermodynamic length. *Physical Review Letters*, 99(10):100602, 2007.
- [9] M. Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 26, 2013.
- [10] P. Dai Pra. A stochastic control approach to reciprocal diffusion processes. *Applied mathematics and Optimization*, 23(1):313–329, 1991.
- [11] P. Dai Pra, L. Meneghini, and W. J. Runggaldier. Connections between stochastic control and dynamic games. *Mathematics of Control, Signals and Systems*, 9:303–326, 1996.
- [12] V. De Bortoli, J. Thornton, J. Heng, and A. Doucet. Diffusion Schrödinger bridge with applications to score-based generative modeling. *Advances in Neural Information Processing Systems*, 34:17695–17709, 2021.
- [13] K. Didi, F. Vargas, S. V. Mathis, V. Dutordoir, E. Mathieu, U. J. Komorowska, and P. Lio. A framework for conditional diffusion modelling with applications in motif scaffolding for protein design. *arXiv preprint arXiv:2312.09236*, 2023.
- [14] C. Domingo-Enrich. A taxonomy of loss functions for stochastic optimal control. *arXiv preprint arXiv:2410.00345*, 2024.
- [15] C. Domingo-Enrich, M. Drozdal, B. Karrer, and R. T. Chen. Adjoint matching: Fine-tuning flow and diffusion generative models with memoryless stochastic optimal control. *arXiv preprint arXiv:2409.08861*, 2024.
- [16] C. Domingo-Enrich, M. Drozdal, B. Karrer, and R. T. Q. Chen. Adjoint matching: Fine-tuning flow and diffusion generative models with memoryless stochastic optimal control. In *The Thirteenth International Conference on Learning Representations*, 2025.
- [17] C. Domingo-Enrich, J. Han, B. Amos, J. Bruna, and R. T. Q. Chen. Stochastic optimal control matching. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.
- [18] Y. Du, M. Plainer, R. Brekelmans, C. Duan, F. Noe, C. P. Gomes, A. Aspuru-Guzik, and K. Neklyudov. Doob’s lagrangian: A sample-efficient variational approach to transition path sampling. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.
- [19] M. F. Faulkner and S. Livingstone. Sampling algorithms in statistical physics: a guide for statistics and machine learning. *Statistical Science*, 39(1):137–164, 2024.

- [20] W. Fleming and R. Rishel. *Deterministic and Stochastic Optimal Control*. Applications of mathematics. Springer, 1975.
- [21] S. Fu, N. Tamir, S. Sundaram, L. Chai, R. Zhang, T. Dekel, and P. Isola. Dreamsime: Learning new dimensions of human visual similarity using synthetic data. *arXiv preprint arXiv:2306.09344*, 2023.
- [22] A. Gelman, J. Carlin, H. Stern, D. Dunson, A. Vehtari, and D. Rubin. *Bayesian Data Analysis, Third Edition*. Chapman & Hall/CRC Texts in Statistical Science. Taylor & Francis, 2013.
- [23] C. Hartmann, O. Kebiri, L. Neureither, and L. Richter. Variational approach to rare event simulation using least-squares regression. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 29(6), 2019.
- [24] C. Hartmann and L. Richter. Nonasymptotic bounds for suboptimal importance sampling. *SIAM/ASA Journal on Uncertainty Quantification*, 12(2):309–346, 2024.
- [25] C. Hartmann, L. Richter, C. Schütte, and W. Zhang. Variational characterization of free energy: Theory and algorithms. *Entropy*, 19(11), 2017.
- [26] C. Hartmann and C. Schütte. Efficient rare event simulation by optimal nonequilibrium forcing. *Journal of Statistical Mechanics: Theory and Experiment*, 2012(11):P11004, 2012.
- [27] J. He, Y. Du, F. Vargas, D. Zhang, S. Padhy, R. OuYang, C. Gomes, and J. M. Hernández-Lobato. No trick, no treat: Pursuits and challenges towards simulation-free training of neural samplers. *arXiv preprint arXiv:2502.06685*, 2025.
- [28] J. Hénin, T. Lelièvre, M. R. Shirts, O. Valsson, and L. Delemotte. Enhanced sampling methods for molecular dynamics simulations. *arXiv preprint arXiv:2202.04164*, 2022.
- [29] J. Hessel, A. Holtzman, M. Forbes, R. L. Bras, and Y. Choi. Clipscore: A reference-free evaluation metric for image captioning. *arXiv preprint arXiv:2104.08718*, 2021.
- [30] L. Holdijk, Y. Du, P. Jaini, F. Hooft, B. Ensing, and M. Welling. Path integral stochastic optimal control for sampling transition paths. In *ICML 2022 2nd AI for Science Workshop*, 2022.
- [31] S. Izrailev, S. Stepaniants, B. Isralewitz, D. Kosztin, H. Lu, F. Molnar, W. Wriggers, and K. Schulten. Steered molecular dynamics. In *Computational Molecular Dynamics: Challenges, Methods, Ideas: Proceedings of the 2nd International Symposium on Algorithms for Macromolecular Modelling, Berlin, May 21–24, 1997*, pages 39–65. Springer, 1999.
- [32] H. J. Kappen and H. C. Ruiz. Adaptive importance sampling for control and inference. *Journal of Statistical Physics*, 162(5):1244–1266, 2016.
- [33] R. M. Neal. Probabilistic inference using markov chain monte carlo methods. 1993.
- [34] N. Nüsken and L. Richter. Solving high-dimensional Hamilton–Jacobi–Bellman pdes using neural networks: perspectives from the theory of controlled diffusions and measures on path space. *Partial differential equations and applications*, 2:1–48, 2021.
- [35] F. Otto, P. Becker, N. A. Vien, H. C. Ziesche, and G. Neumann. Differentiable trust region layers for deep reinforcement learning. *arXiv preprint arXiv:2101.09207*, 2021.
- [36] M. Pavon. Stochastic control and nonequilibrium thermodynamical systems. *Applied Mathematics and Optimization*, 19(1):187–202, 1989.
- [37] M. Pavon. On local entropy, stochastic control and deep neural networks. *arXiv preprint arXiv:2204.13049*, 2022.
- [38] J. Peters, K. Mulling, and Y. Altun. Relative entropy policy search. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 24, pages 1607–1612, 2010.
- [39] L. Richter. *Solving high-dimensional PDEs, approximation of path space measures and importance sampling of diffusions*. PhD thesis, BTU Cottbus-Senftenberg, 2021.

- [40] L. Richter, J. Berner, and G.-H. Liu. Improved sampling via learned diffusions. *arXiv preprint arXiv:2307.01198*, 2023.
- [41] L. Richter, A. Boustati, N. Nüsken, F. Ruiz, and O. D. Akyildiz. Vargrad: a low-variance gradient estimator for variational inference. *Advances in Neural Information Processing Systems*, 33:13481–13492, 2020.
- [42] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022.
- [43] R. Y. Rubinstein and D. P. Kroese. *The cross-entropy method: a unified approach to combinatorial optimization, Monte-Carlo simulation and machine learning*. Springer Science & Business Media, 2013.
- [44] P. Salamon and R. S. Berry. Thermodynamic length and dissipated availability. *Physical Review Letters*, 51(13):1127, 1983.
- [45] J. Schulman, S. Levine, P. Abbeel, M. Jordan, and P. Moritz. Trust region policy optimization. In *International conference on machine learning*, pages 1889–1897. PMLR, 2015.
- [46] K. Seong, S. Park, S. Kim, W. Y. Kim, and S. Ahn. Transition path sampling with improved off-policy training of diffusion path samplers. *arXiv preprint arXiv:2405.19961*, 2024.
- [47] A. N. Singh, A. Das, and D. T. Limmer. Variational path sampling of rare dynamical events. *Annual Review of Physical Chemistry*, 76, 2025.
- [48] S. Syed, A. Bouchard-Côté, K. Chern, and A. Doucet. Optimised annealed sequential monte carlo samplers. *arXiv preprint arXiv:2408.12057*, 2024.
- [49] D. Thalmeier, H. J. Kappen, S. Totaro, and V. Gómez. Adaptive smoothing for path integral control. *Journal of Machine Learning Research*, 21(191):1–37, 2020.
- [50] B. Tzen and M. Raginsky. Theoretical guarantees for sampling and inference in generative models with latent diffusions. In *Conference on Learning Theory*, pages 3084–3114. PMLR, 2019.
- [51] F. Vargas, W. Grathwohl, and A. Doucet. Denoising diffusion samplers. *arXiv preprint arXiv:2302.13834*, 2023.
- [52] F. Vargas, A. Ovsianas, D. Fernandes, M. Girolami, N. D. Lawrence, and N. Nüsken. Bayesian learning via neural schrödinger–föllmer flows. *Statistics and Computing*, 33(1):3, 2023.
- [53] F. Vargas, S. Padhy, D. Blessing, and N. Nüsken. Transport meets variational inference: Controlled Monte Carlo diffusions. In *The Twelfth International Conference on Learning Representations*, 2024.
- [54] S. Venkatraman, M. Jain, L. Scimeca, M. Kim, M. Sendera, M. Hasan, L. Rowe, S. Mittal, P. Lemos, E. Bengio, et al. Amortizing intractable inference in diffusion models for vision, language, and control. *arXiv preprint arXiv:2405.20971*, 2024.
- [55] H. Wu, J. Köhler, and F. Noé. Stochastic normalizing flows. *Advances in neural information processing systems*, 33:5933–5944, 2020.
- [56] X. Wu, Y. Hao, K. Sun, Y. Chen, F. Zhu, R. Zhao, and H. Li. Human preference score v2: A solid benchmark for evaluating human preferences of text-to-image synthesis. *arXiv preprint arXiv:2306.09341*, 2023.
- [57] J. Xu, X. Liu, Y. Wu, Y. Tong, Q. Li, M. Ding, J. Tang, and Y. Dong. Imagereward: Learning and evaluating human preferences for text-to-image generation. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.
- [58] J. Yan, H. Touchette, and G. M. Rotskoff. Learning nonequilibrium control forces to characterize dynamical phase transitions. *Physical Review E*, 105(2):024115, 2022.

- 481 [59] D. Zhang, R. T. Chen, C.-H. Liu, A. Courville, and Y. Bengio. Diffusion generative flow
482 samplers: Improving learning signals through partial trajectory optimization. In *The Twelfth*
483 *International Conference on Learning Representations*, 2024.
- 484 [60] Q. Zhang and Y. Chen. Path Integral Sampler: a stochastic control approach for sampling. In
485 *International Conference on Learning Representations*, 2022.
- 486 [61] W. Zhang, H. Wang, C. Hartmann, M. Weber, and C. Schütte. Applications of the cross-entropy
487 method to importance sampling and optimal control of diffusions. *SIAM Journal on Scientific*
488 *Computing*, 36(6):A2654–A2672, 2014.
- 489 [62] X. Zhang, L. Wang, J. Helwig, Y. Luo, C. Fu, Y. Xie, M. Liu, Y. Lin, Z. Xu, K. Yan, et al.
490 Artificial intelligence for science in quantum, atomistic, and continuum systems. *arXiv preprint*
491 *arXiv:2307.08423*, 2023.
- 492 [63] M. Zhou, J. Han, and J. Lu. Actor-critic method for high dimensional static Hamilton–Jacobi–
493 Bellman partial differential equations based on neural networks. *SIAM Journal on Scientific*
494 *Computing*, 43(6):A4043–A4066, 2021.

NeurIPS Paper Checklist

1. Claims

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

Answer: [\[Yes\]](#)

Justification: The abstract and introduction clearly state and reference our claims, which match theoretical and experimental results.

Guidelines:

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

2. Limitations

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

Answer: [\[Yes\]](#)

Justification: We discuss all limitations that we are aware of and reflect on the scope of our claims in App. C.

Guidelines:

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

3. Theory assumptions and proofs

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

Answer: [\[Yes\]](#)

Justification: Our assumptions are clearly stated in the theoretical results and general assumptions can be found in App. A.2. The proofs of our theoretical results can be found in the appendix and are referenced after each result.

Guidelines:

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

4. Experimental result reproducibility

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

Answer: [Yes]

Justification: We provide detailed instructions for how to replicate our results in the respective repositories.

Guidelines:

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

5. Open access to data and code

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

Answer: [No]

Justification: We will make our code publicly available upon acceptance.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.

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

6. Experimental setting/details

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

Answer: [Yes]

Justification: We reference the repositories of our baselines and specify all training and test details for our methods in App. J.2.

Guidelines:

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

7. Experiment statistical significance

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

Answer: [Yes]

Justification: We provide error bars and corresponding explanations for all experiments that support the main claims of the paper.

Guidelines:

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

8. Experiments compute resources

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

661 Answer: [Yes]

662 Justification: We provide sufficient information on the computer resources for each experiments

663 in App. J.2.

664 Guidelines:

665 • The answer NA means that the paper does not include experiments.

666 • The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud

667 provider, including relevant memory and storage.

668 • The paper should provide the amount of compute required for each of the individual experimental

669 runs as well as estimate the total compute.

670 • The paper should disclose whether the full research project required more compute than the

671 experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it

672 into the paper).

673 **9. Code of ethics**

674 Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS

675 Code of Ethics <https://neurips.cc/public/EthicsGuidelines?>

676 Answer: [Yes]

677 Justification: The research conducted in the paper conforms, in every respect, with the NeurIPS

678 Code of Ethics.

679 Guidelines:

680 • The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.

681 • If the authors answer No, they should explain the special circumstances that require a deviation

682 from the Code of Ethics.

683 • The authors should make sure to preserve anonymity (e.g., if there is a special consideration due

684 to laws or regulations in their jurisdiction).

685 **10. Broader impacts**

686 Question: Does the paper discuss both potential positive societal impacts and negative societal

687 impacts of the work performed?

688 Answer: [Yes]

689 Justification: We discuss both potential positive societal impacts and negative societal impacts of

690 our work in App. C.

691 Guidelines:

692 • The answer NA means that there is no societal impact of the work performed.

693 • If the authors answer NA or No, they should explain why their work has no societal impact or

694 why the paper does not address societal impact.

695 • Examples of negative societal impacts include potential malicious or unintended uses (e.g.,

696 disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deploy-

697 ment of technologies that could make decisions that unfairly impact specific groups), privacy

698 considerations, and security considerations.

699 • The conference expects that many papers will be foundational research and not tied to par-

700 ticular applications, let alone deployments. However, if there is a direct path to any negative

701 applications, the authors should point it out. For example, it is legitimate to point out that

702 an improvement in the quality of generative models could be used to generate deepfakes for

703 disinformation. On the other hand, it is not needed to point out that a generic algorithm for

704 optimizing neural networks could enable people to train models that generate Deepfakes faster.

705 • The authors should consider possible harms that could arise when the technology is being used

706 as intended and functioning correctly, harms that could arise when the technology is being used

707 as intended but gives incorrect results, and harms following from (intentional or unintentional)

708 misuse of the technology.

709 • If there are negative societal impacts, the authors could also discuss possible mitigation strategies

710 (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for

711 monitoring misuse, mechanisms to monitor how a system learns from feedback over time,

712 improving the efficiency and accessibility of ML).

713 **11. Safeguards**

714 Question: Does the paper describe safeguards that have been put in place for responsible release of

715 data or models that have a high risk for misuse (e.g., pretrained language models, image generators,

716 or scraped datasets)?

717 Answer: [NA]
 718 Justification: Our paper poses no such risks since we do not release new data or models.
 719 Guidelines:
 720 • The answer NA means that the paper poses no such risks.
 721 • Released models that have a high risk for misuse or dual-use should be released with necessary
 722 safeguards to allow for controlled use of the model, for example by requiring that users adhere
 723 to usage guidelines or restrictions to access the model or implementing safety filters.
 724 • Datasets that have been scraped from the Internet could pose safety risks. The authors should
 725 describe how they avoided releasing unsafe images.
 726 • We recognize that providing effective safeguards is challenging, and many papers do not require
 727 this, but we encourage authors to take this into account and make a best faith effort.

728 **12. Licenses for existing assets**
 729 Question: Are the creators or original owners of assets (e.g., code, data, models), used in the
 730 paper, properly credited and are the license and terms of use explicitly mentioned and properly
 731 respected?
 732 Answer: [Yes]
 733 Justification: For each asset that we use, we cite the paper and provide the URL (including the
 734 license).
 735 Guidelines:
 736 • The answer NA means that the paper does not use existing assets.
 737 • The authors should cite the original paper that produced the code package or dataset.
 738 • The authors should state which version of the asset is used and, if possible, include a URL.
 739 • The name of the license (e.g., CC-BY 4.0) should be included for each asset.
 740 • For scraped data from a particular source (e.g., website), the copyright and terms of service of
 741 that source should be provided.
 742 • If assets are released, the license, copyright information, and terms of use in the package should
 743 be provided. For popular datasets, paperswithcode.com/datasets has curated licenses for
 744 some datasets. Their licensing guide can help determine the license of a dataset.
 745 • For existing datasets that are re-packaged, both the original license and the license of the derived
 746 asset (if it has changed) should be provided.
 747 • If this information is not available online, the authors are encouraged to reach out to the asset's
 748 creators.

749 **13. New assets**
 750 Question: Are new assets introduced in the paper well documented and is the documentation
 751 provided alongside the assets?
 752 Answer: [NA]
 753 Justification: Our paper does not release new assets.
 754 Guidelines:
 755 • The answer NA means that the paper does not release new assets.
 756 • Researchers should communicate the details of the dataset/code/model as part of their sub-
 757 missions via structured templates. This includes details about training, license, limitations,
 758 etc.
 759 • The paper should discuss whether and how consent was obtained from people whose asset is
 760 used.
 761 • At submission time, remember to anonymize your assets (if applicable). You can either create
 762 an anonymized URL or include an anonymized zip file.

763 **14. Crowdsourcing and research with human subjects**
 764 Question: For crowdsourcing experiments and research with human subjects, does the paper
 765 include the full text of instructions given to participants and screenshots, if applicable, as well as
 766 details about compensation (if any)?
 767 Answer: [NA]
 768 Justification: Our paper does not involve crowdsourcing nor research with human subjects.
 769 Guidelines:
 770 • The answer NA means that the paper does not involve crowdsourcing nor research with human
 771 subjects.

772 • Including this information in the supplemental material is fine, but if the main contribution of
773 the paper involves human subjects, then as much detail as possible should be included in the
774 main paper.

775 • According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other
776 labor should be paid at least the minimum wage in the country of the data collector.

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

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

782 Answer: [NA]

783 Justification: Our paper does not involve crowdsourcing nor research with human subject.

784 Guidelines:

785 • The answer NA means that the paper does not involve crowdsourcing nor research with human
786 subjects.

787 • Depending on the country in which research is conducted, IRB approval (or equivalent) may be
788 required for any human subjects research. If you obtained IRB approval, you should clearly
789 state this in the paper.

790 • We recognize that the procedures for this may vary significantly between institutions and
791 locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for
792 their institution.

793 • For initial submissions, do not include any information that would break anonymity (if applica-
794 ble), such as the institution conducting the review.

795 **16. Declaration of LLM usage**

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

800 Answer: [NA]

801 Justification: The core method development in this research does not involve LLMs as any
802 important, original, or non-standard components.

803 Guidelines:

804 • The answer NA means that the core method development in this research does not involve LLMs
805 as any important, original, or non-standard components.

806 • Please refer to our LLM policy (<https://neurips.cc/Conferences/2025/LLM>) for what
807 should or should not be described.

808 Appendix

809	A Assumptions and auxiliary results	22
810	A.1 Additional notation	22
811	A.2 Technical assumptions	22
812	A.3 Useful identities	22
813	B Proofs	23
814	C Further related works, broader impact, and limitations	23
815	C.1 Further related works	23
816	C.2 Limitations	24
817	C.3 Broader impact	24
818	D Background on SOC	24
819	D.1 Stochastic optimal control	24
820	D.2 Iterative diffusion optimization	25
821	E Background on diffusion-based sampling	25
822	E.1 Experimental setup	26
823	E.2 Evaluation criteria	27
824	E.3 Additional experiments	28
825	F Transition path sampling	29
826	F.1 Experimental details	29
827	F.2 Additional Experimental Result Discussion	29
828	G Classical SOC problems	31
829	G.1 Experimental Setup	31
830	G.2 Benchmark problem details	31
831	G.3 Results	32
832	H Fine-tuning of diffusion models	32
833	H.1 Fine-tuning experimental details	32
834	I Trust region SOC sequences are equispaced in the Fisher-Rao distance	33
835	I.1 Basics on information geometry	33
836	I.1.1 A first example: the manifold of smooth densities	33
837	I.2 A second example: exponential families	34
838	I.3 Local expansion of the Kullback–Leibler divergence	35
839	I.4 Information geometry on the exponential family of path measures	35
840	J Details on trust region SOC algorithms	36
841	J.1 Characterizing the solutions of the trust region optimization problem	36
842	J.2 Implementation	37

843	J.3	Variance of the importance weights and trust region bounds	37
844	J.4	Lagrangian formulation	38
845	J.5	Log-variance with buffer and trust regions	39
846	J.6	Trust-region stochastic optimal control matching via adjoint method	40
847	J.7	Trust-region stochastic optimal control matching via lean adjoint method	41
848	K	Trust regions for probability measures	41

A Assumptions and auxiliary results

A.1 Additional notation

For vectors $v_1, v_2 \in \mathbb{R}^d$, we denote by $\|v\|$ the Euclidean norm and by $v_1 \cdot v_2$ the Euclidean inner product. For a real-valued matrix A , we denote by $\text{Tr}(A)$ and A^\top its trace and transpose.

For a sufficiently smooth function $f: \mathbb{R}^d \times [0, T] \rightarrow \mathbb{R}$, we denote by $\nabla f = \nabla_x f$ its gradient w.r.t. the spatial variables x and by $\partial_t f$ and $\partial_{x_i} f$ its partial derivatives w.r.t. the time coordinate t and the spatial coordinate x_i , respectively.

We denote by $\mathcal{N}(\mu, \Sigma)$ a multivariate normal distribution with mean $\mu \in \mathbb{R}^d$ and covariance matrix $\Sigma \in \mathbb{R}^{d \times d}$. Moreover, we denote by $\text{Unif}([0, T])$ the uniform distribution on $[0, T]$. For random variables X_1, X_2 , we denote by $\mathbb{E}[X_1]$ and $\text{Var}[X_1]$ the expectation and variance of X_1 and by $\mathbb{E}[X_1|X_2]$ the conditional expectation of X_1 given X_2 .

A.2 Technical assumptions

Throughout our work, we make the same assumptions as in [101, 85], which are needed for all the objects considered to be well-defined. Namely, we assume that:

(i) The set \mathcal{U} of *admissible controls* is given by

$$\mathcal{U} = \{u \in C^1(\mathbb{R}^d \times [0, T]; \mathbb{R}^d) \mid \exists C > 0, \forall (x, s) \in \mathbb{R}^d \times [0, T], u(x, s) \leq C(1 + \|x\|)\}. \quad (20)$$

(ii) The coefficients b and σ are continuously differentiable, σ has bounded first-order spatial derivatives, and $(\sigma\sigma^\top)(x, s)$ is positive definite for all $(x, s) \in \mathbb{R}^d \times [0, T]$. Furthermore, there exist constants $C, c_1, c_2 > 0$ such that

$$\begin{aligned} \|b(x, s)\| &\leq C(1 + \|x\|), & \text{(linear growth)} \\ c_1 \|\beta\|^2 &\leq \beta^\top (\sigma\sigma^\top)(x, s) \beta \leq c_2 \|\beta\|^2, & \text{(ellipticity)} \end{aligned} \quad (21)$$

for all $(x, s) \in \mathbb{R}^d \times [0, T]$ and $\beta \in \mathbb{R}^d$.

A.3 Useful identities

Girsanov. For a generic $v \in \mathcal{U}$, consider the two SDEs

$$dX_s^v = (b(X_s^v, s) + \sigma v(X_s^v, s)) ds + \sigma dW_s, \quad X_0^v \sim p_0 \quad (22)$$

$$dX_s = b(X_s, s) ds + \sigma dW_s, \quad X_0 \sim p_0. \quad (23)$$

By Girsanov's theorem, we have that for any $u, w \in \mathcal{U}$

$$\log \frac{d\mathbb{P}^u}{d\mathbb{P}}(X^w) = \int_0^T \sigma^{-1} u(X_s^w, s) \cdot dX_s^w - \int_0^T (\sigma^{-1} b \cdot u)(X_s^w, s) ds - \frac{1}{2} \int_0^T \|u(X_s^w, s)\|^2 ds. \quad (24)$$

It follows that

$$\log \frac{d\mathbb{P}^u}{d\mathbb{P}}(X^u) = \int_0^T u(X_s^u, s) \cdot dW_s + \frac{1}{2} \int_0^T \|u(X_s^u, s)\|^2 ds, \quad (25)$$

and

$$\log \frac{d\mathbb{P}^u}{d\mathbb{P}}(X) = \int_0^T u(X_s, s) \cdot dW_s - \frac{1}{2} \int_0^T \|u(X_s, s)\|^2 ds. \quad (26)$$

Moreover, it holds that (see [123] Appendix E)

$$\log \frac{d\mathbb{P}^u}{d\mathbb{P}^v}(X^w) = \int_0^T \sigma^{-1} (u - v)(X_s^w, s) \cdot dX_s^w - \frac{1}{2} \int_0^T (\|\sigma^{-1} b + u\|^2 - \|\sigma^{-1} b + v\|^2)(X_s^w, s) ds, \quad (27)$$

from which follows that

$$\log \frac{d\mathbb{P}^u}{d\mathbb{P}^v}(X^u) = \int_0^T (u - v)(X_s^u, s) \cdot dW_s + \frac{1}{2} \int_0^T \|u - v\|^2(X_s^u, s) ds, \quad (28)$$

and

$$\log \frac{d\mathbb{P}^u}{d\mathbb{P}^v}(X^v) = \int_0^T (u - v)(X_s^v, s) \cdot dW_s - \frac{1}{2} \int_0^T \|u - v\|^2(X_s^v, s) ds, \quad (29)$$

Itô formula. Consider the stochastic process X defined by the SDE

$$dX_s = b(X_s, s) ds + \sigma dW_s \quad (30)$$

with infinitesimal generator $L := \frac{1}{2} \sum_{i,j} (\sigma\sigma^\top)_{ij} \partial_{x_i} \partial_{x_j} + \sum_i b_i(x, t) \partial_{x_i}$. The dynamics of $f(X_s, s)$ is given by

$$df(X_s, s) = (\partial_s + L)f(X_s, s) ds + \sigma^\top \nabla f(X_s, s) \cdot dW_s. \quad (31)$$

B Proofs

Proof of Prop. 2.2. Let $\tilde{\mathbb{P}}$ be the measure defined by $\frac{d\tilde{\mathbb{P}}}{d\mathbb{P}^{u_i}} = \left(\frac{d\mathbb{Q}}{d\mathbb{P}^{u_i}}\right)^{\frac{1}{1+\lambda_i}} / \tilde{\mathcal{Z}}$, where $\tilde{\mathcal{Z}}$ is the normalizing constant. Then we have that

$$(1 + \lambda_i) \log \frac{d\mathbb{P}^u}{d\tilde{\mathbb{P}}} = (1 + \lambda_i) \log \left(\frac{d\mathbb{P}^u}{d\mathbb{P}^{u_i}} \frac{d\mathbb{P}^{u_i}}{d\tilde{\mathbb{P}}} \right) = (1 + \lambda_i) \log \frac{d\mathbb{P}^u}{d\mathbb{P}^{u_i}} + \log \frac{d\mathbb{P}^{u_i}}{d\mathbb{Q}} + (1 + \lambda_i) \log \tilde{\mathcal{Z}} \quad (32)$$

$$= \lambda_i \log \frac{d\mathbb{P}^u}{d\mathbb{P}^{u_i}} + \log \frac{d\mathbb{P}^u}{d\mathbb{Q}} + (1 + \lambda_i) \log \tilde{\mathcal{Z}}. \quad (33)$$

Using the definition of the Lagrangian in (5), this implies that

$$(1 + \lambda_i) D_{\text{KL}}(\mathbb{P}^u | \tilde{\mathbb{P}}) = \lambda_i D_{\text{KL}}(\mathbb{P}^u | \mathbb{P}^{u_i}) + D_{\text{KL}}(\mathbb{P}^u | \mathbb{Q}) + (1 + \lambda_i) \log \tilde{\mathcal{Z}} \quad (34)$$

$$= \mathcal{L}_{\text{Tr}}^{(i)}(u, \lambda_i) + (1 + \lambda_i) \log \tilde{\mathcal{Z}} + \lambda_i \varepsilon, \quad (35)$$

Since we defined the minimizer of the Lagrangian (with optimal multiplier λ_i) in the last expression as u_{i+1} , we have that $u_{i+1} = \arg \min_{u \in \mathcal{U}} D_{\text{KL}}(\mathbb{P}^u | \tilde{\mathbb{P}})$. This shows that $\tilde{\mathbb{P}} = \mathbb{P}^{u_{i+1}}$ by the uniqueness of the Radon-Nikodym derivative. The second statement then follows by induction. \square

C Further related works, broader impact, and limitations

C.1 Further related works

Monte Carlo estimator. In theory, one could directly compute the optimal control using the representations in Prop. 2.5 (for $\lambda = 0$ and $i = 0$; see Item 1 in Thm. D.1) combined with Monte Carlo estimates⁶ of the value function in Item 4 [94, 122, 95, 80, 116]. However, in practice this can be problematic since it requires a large amount of samples *for each state* x due to the (typically) very high variance of the estimator for V [122]. In particular, we note that the variance translates to bias in the control due to the logarithmic transform. Moreover, for nonzero f or general b (e.g., in the fine-tuning setting), one needs to *simulate* the uncontrolled process to obtain samples.

PDE solver. One can also leverage the representation of the value function as the solution of a HJB equation (see Item 3 in Thm. D.1). While solving PDEs in high dimensions is very challenging, there exist scalable approaches based on tensor trains and neural networks⁷ that leverage backward stochastic differential equations or the Hopf-Cole transform in combination with the Feynman-Kac formula [90, 67, 104, 66, 105, 108, 107, 64]. However, in practice, we only need the value function in the domain where the optimal path measure has sufficiently large values, which is typically not considered for PDE solvers.

Iterative diffusion optimization. To focus more on promising regions of the path space, methods for iterative diffusion optimization simulate (sub-)trajectories of the controlled SDE to compute a suitable loss and update the control. Typically, the control is parametrized as a neural network and optimized using variants of stochastic gradient descent. While such methods have been explored for general SOC problems with quadratic control costs [101, 104, 84, 81], many recent works have focus on the special case of sampling from unnormalized densities as described in Sec. 3.1; see, e.g., [129, 68, 122, 127, 123, 111, 100, 65]. From the perspective of path measures, all these works propose to minimize suitable divergences between measures induced by controlled SDEs. While we demonstrate the benefits of leveraging trust region methods for the *Denoising Diffusion Sampler* (DDS) [121], our method could also be extended to other samplers.

Transition path sampling. Transition path sampling has been a longstanding problem in physics and chemistry to understand phase transition and chemical reaction, with applications in energy, catalysis, and drug discovery [71, 120]. Computationally, MCMC-based approaches have been extended to path space to mix the transition path distribution, pioneered by [79]. As discussed in Sec. 3.2, transition path sampling can be formulated as a stochastic optimal control problem and has been numerically solved using reverse KL divergence [126], cross-entropy divergence [93], and

⁶One can obtain derivative estimates using adjoint states (as defined in Sec. 2.2) or using reparametrization tricks if the uncontrolled process has suitable, known marginals. For Gaussian marginals, one can also use Stein’s lemma [94]. We also note that control variates for such estimators have been analyzed in [110, 105].

⁷Note that some of these approaches correspond to regressions of the Monte Carlo estimators mentioned above [122] or to the IDO methods mentioned below [115].

log-variance divergence [112], the optimal control is known to be the Doob’s h-function [73, 114, 86] (for a review, we refer to [113]). To solve the Doob’s h-function, [114] proposes a shooting-based method which requires MD simulation to reach the target state, while [86] proposes a Gaussian approximation conditioned on both the initial and target state which satisfies boundary conditions by design and provides a simulation-free optimization algorithm. Similarly to SOC, transition path sampling can also naturally be formulated as a reinforcement learning problem, as in [78, 109].

Diffusion and flow matching reward fine-tuning. Several of the early works on diffusion fine-tuning focused on directly optimizing the reward model making use of its differentiability [125, 74], without any KL regularization, which can lead to reward hacking. Some other works [69, 87] framed reward fine-tuning as an RL problem, but did not make the probabilistic connection to tilted distributions. [118] provides a probabilistic view of the problem, but proposes an algorithm that is hard to scale. [82] give a comprehensive view of flow matching reward fine-tuning, introducing memoryless noise schedules as the right ones, as well as a new scalable SOC algorithm that we use and adapt: adjoint matching. Using the memoryless noise schedule, a recent work [97] considers GRPO for flow matching fine-tuning. [128, 98] consider alternative algorithms which learn the value functions.

C.2 Limitations

While our method for solving stochastic optimal control problems exhibits strong sample efficiency, it relies on storing entire trajectories in the replay buffer during training. In large-scale settings—such as fine-tuning text-to-image models—this necessitates keeping the replay buffer in CPU memory while training occurs on the GPU. This separation introduces additional computational overhead due to data transfers between CPU and GPU; however, the buffer still significantly accelerates the fine-tuning since the main computational cost in such settings stems from the simulation of trajectories.

C.3 Broader impact

This paper proposes new methodologies and theories that find numerical solutions for stochastic optimal control problems ranging from equilibrium sampling, transition path sampling to fine-tuning text-to-image generative models. Equilibrium sampling and transition path sampling are important in Bayesian statistics, physics and chemistry where they can be used to estimate free energy, understand phase transition and rare event which hold promises to accelerate drug and materials discovery. More efficient fine-tuning of text-to-image models democratizes the generation of specialized high-quality visual content for creative applications. However, these capabilities also introduce risks such as the potential for generating convincing misinformation and deepfakes.

D Background on SOC

D.1 Stochastic optimal control

In this work, we consider stochastic optimal control (SOC) problems of the form

$$\min_{u \in \mathcal{U}} \mathcal{L}_{\text{soc}}(u) = \min_{u \in \mathcal{U}} \mathbb{E} \left[\int_0^T \left(\frac{1}{2} \|u(X_s^u, s)\|^2 + f(X_s^u, s) \right) ds + g(X_T^u) \right], \quad (36)$$

with state-cost f , terminal cost g and control function $u \in \mathcal{U}$, where \mathcal{U} denotes a set of admissible controls; see App. A.2 for further details. Here, X^u is a controlled SDE of the form

$$dX_s^u = b(X_s^u, s) + \sigma(s)u(X_s^u, s)ds + \sigma(s)dW_s, \quad X_0 \sim p_0, \quad (37)$$

with base drift b , base distribution p_0 (typically a Gaussian or dirac delta distribution), and diffusion coefficient σ . We denote the path measure induced by (37) by $\mathbb{P}^u \in \mathcal{P}$. Moreover, we simply write \mathbb{P} for the path measure corresponding to the uncontrolled process, i.e.,

$$dX_s = b(X_s, s)ds + \sigma(s)dW_s, \quad X_0 \sim p_0. \quad (38)$$

Given a time t and state x , the cost functional $J(u; x, t)$ is the expected cost-to-go for a control u on the time interval $[t, T]$ and is defined as

$$J(u; x, t) = \mathbb{E} \left[\int_t^T \left(\frac{1}{2} \|u(X_s^u, s)\|^2 + f(X_s^u, s) \right) ds + g(X_T^u) \mid X_t^u = x \right]. \quad (39)$$

The value function V , or, *optimal cost-to-go* is obtained by taking the infimum over all controls in \mathcal{U} , that is,

$$V(x, t) = \inf_{u \in \mathcal{U}} J(u; x, t). \quad (40)$$

Then we have the following well-known results on representations of the value function V and solution to the SOC problem u^* ; see, e.g., [101, 102, 77, 88, 103] for details.

963 **Theorem D.1** (Optimality for SOC Problems). *Let us define the work functional as*

$$\mathcal{W}(X, t) = \int_t^T f(X_s, s) ds + g(X_T). \quad (41)$$

964 *Then we have the following representations of the value function in V in (40) and the solution u^* to*
 965 *the SOC problem in (36):*

- 966 1. (Connection between solution and value function) *The solution can be written as $u^* = -\sigma^\top \nabla V$.*
 967 2. (Optimal change of measure) *The Radon-Nikodym derivative of the optimal path measure \mathbb{Q} w.r.t.*
 968 *the uncontrolled path measure \mathbb{P} satisfies*

$$\frac{d\mathbb{Q}}{d\mathbb{P}}(X) = \frac{e^{-\mathcal{W}(X, 0)}}{\mathcal{Z}(X_0)} \quad \text{with} \quad \mathcal{Z}(X_0) = \mathbb{E}[e^{-\mathcal{W}(X, 0)} | X_0]. \quad (42)$$

- 969 3. (PDE for value function) *The value function V is the solution to the Hamilton-Jacobi-Bellman*
 970 *(HJB) equation*

$$(\partial_t + L)V(x, t) - \frac{1}{2} \|(\sigma^\top \nabla V)(x, t)\|^2 + f(x, t) = 0, \quad V(x, T) = g(x), \quad (43)$$

971 *where $L := \frac{1}{2} \sum_{i,j=1}^d (\sigma \sigma^\top)_{ij} \partial_{x_i} \partial_{x_j} + \sum_{i=1}^d b_i \partial_{x_i}$ denotes the infinitesimal generator of the*
 972 *uncontrolled SDE in (38).*

- 973 4. (Estimator for value function) *For every $(x, t) \in \mathbb{R}^d \times [0, T]$ the value function can be written as*
 974 *$V(x, t) = -\log \mathbb{E}[e^{-\mathcal{W}(X, t)} | X_t = x]$, where X is the solution of the uncontrolled SDE in (38).*

975 Combining the expressions for u^* and V in Thm. D.1, we directly obtain the path integral representa-
 976 tion of the optimal control, i.e.,

$$u^*(x, t) = \sigma(t)^\top \nabla_x \log \mathbb{E}[e^{-\mathcal{W}(X, t)} | X_t = x], \quad (44)$$

977 In practice, computing the optimal control (44) is typically impractical, as it requires running multiple
 978 simulations for each state x to obtain a Monte Carlo approximation of the expectation; see App. C.1.
 979 To address this challenge, many approaches instead learn a parameterized control function, optimized
 980 using stochastic gradient methods. These techniques are collectively referred to as iterative diffusion
 981 optimization (IDO) methods and are further discussed in the next section.

982 D.2 Iterative diffusion optimization

983 An alternative view on problem (36) is obtained by considering loss functions on path measures. By
 984 the Girsanov theorem (see App. A.3) we have

$$\frac{d\mathbb{P}}{d\mathbb{P}^u}(X^u) = \exp \left(- \int_0^T u(X_s^u, s) \cdot dW_s - \frac{1}{2} \int_0^T \|u(X_s^u, s)\|^2 ds \right). \quad (45)$$

985 Combining this with the optimal change of measure $d\mathbb{Q}/d\mathbb{P}$ in Thm. D.1, we obtain an expression
 986 for $d\mathbb{Q}/d\mathbb{P}^u$ from which we can compute the relative entropy \mathcal{L}_{RE} , i.e., the reverse Kullback-Leibler
 987 (KL) divergence

$$\mathcal{L}_{\text{RE}}(u) = D_{\text{KL}}(\mathbb{P}^u \| \mathbb{Q}) = \mathbb{E} \left[\int_0^T \left(\frac{1}{2} \|u(X_s^u, s)\|^2 + f(X_s^u, s) \right) ds + g(X_T^u) \right] + \log \mathcal{Z}. \quad (46)$$

988 Note that minimizing the stochastic optimal control problem in (36) is equal to minimizing the KL
 989 divergence, that is,

$$u^* = \arg \min_{u \in \mathcal{U}} \mathcal{L}_{\text{SOC}}(u) = \arg \min_{u \in \mathcal{U}} \mathcal{L}_{\text{RE}}(u), \quad (47)$$

990 in the sense that both have the same unique optimal control u^* as a minimizer. As such, we can
 991 consider an arbitrary divergence $D : \mathcal{P} \times \mathcal{P} \rightarrow \mathbb{R}^+$ for which holds $D(\mathbb{P}_1 | \mathbb{P}_2) = 0$ if and only if
 992 $\mathbb{P}_1 = \mathbb{P}_2$ to solve stochastic optimal control problems. More generally, we can consider any loss
 993 function for which the unique minimizer is the optimal control u^* . Iterative diffusion optimization
 994 builds on this perspective and can be seen as a common framework for solving (potentially high-
 995 dimensional) SOC problems by leveraging parameterized control functions and stochastic gradient
 996 methods to minimize different loss functions.

997 E Background on diffusion-based sampling

998 We consider the task of sampling from densities of the form

$$p_{\text{target}} = \frac{\rho_{\text{target}}}{\mathcal{Z}} \quad \text{with} \quad \mathcal{Z} := \int_{\mathbb{R}^d} \rho_{\text{target}}(x) dx, \quad (48)$$

999 where $\rho_{\text{target}} \in C(\mathbb{R}^d, \mathbb{R}_{\geq 0})$ can be evaluated pointwise, but the normalizing constant \mathcal{Z} is typically
 1000 intractable.

1001 Here, we approach the sampling problem by using denoising diffusion-based sampling based on
 1002 the work of [121]. To that end, we consider a controlled ergodic Ornstein-Uhlenbeck (OU) process
 1003 $X = (X_s)_{s \in [0, T]}$, i.e.,

$$dX_s^u = (-\zeta(s)X_s^u + u(X_s^u, s)) ds + \eta\sqrt{2\zeta(s)} dB_s, \quad X_0 \sim p_0, \quad (49)$$

1004 with noise schedule $\zeta \in C([0, T], \mathbb{R})$, $p_0(x) = \mathcal{N}(0, \eta^2 I)$ and corresponding path measure \mathbb{P}^u . The
 1005 target path space measure \mathbb{Q} is induced by an uncontrolled ergodic Ornstein-Uhlenbeck (OU) process,
 1006 starting from the target p_{target} and running backward in time, that is,

$$dX_s = \zeta(s)X_s ds + \eta\sqrt{2\zeta(s)} dB_s, \quad X_T \sim p_{\text{target}}, \quad (50)$$

1007 which fulfills $\mathbb{Q}_0 \approx p_0$ for a suitable choice of ζ . For integration, we follow [121] and use an
 1008 exponential integrator. Lastly, it can be shown that the optimal control fulfills

$$u^*(x, s) = \eta\sqrt{2\zeta(s)} \nabla_x \log \frac{\mathbb{Q}_s}{\mathbb{P}_s}(x), \quad (51)$$

1009 which is later used to analytically compute the optimal control for Gaussian mixture model target
 1010 densities. Please note that $\mathbb{P}_s = \mathcal{N}(0, \eta^2 I)$ for all $s \in [0, T]$ as the uncontrolled SDE is initialized at
 1011 its equilibrium distribution.

1012 E.1 Experimental setup

1013 Here, we provide further details on our experimental setup.

1014 **General setting.** The codebase used in this work was developed from scratch but is loosely inspired
 1015 by github.com/facebookresearch/SOC-matching. All experiments are conducted using the
 1016 Jax library [72] and are run on a single 40GB NVIDIA A40 GPU. Our default experimental setup,
 1017 unless specified otherwise, is as follows: We use the Adam optimizer [96] with a learning rate of
 1018 5×10^{-4} and gradient clipping with a value of 1. We utilized 50 discretization steps using exponential
 1019 integrators. The control function u is parameterized as a fully-connected 6-layer neural network with
 1020 256 neurons and GELU activations [92]. Time embedding is achieved via Fourier features [117]. For
 1021 all experiments, we used a time horizon of $T = 1$.

1022 The control is parameterized as

$$u^\theta(x, t) = f_1^\theta(x, t) + f_2^\theta(t) \frac{x}{\eta^2}, \quad (52)$$

1023 and for experiments using Langevin preconditioning (LP), it is parameterized as

$$u_{\text{LP}}^\theta(x, t) = f_1^\theta(x, t) + f_2^\theta(t) \left(\frac{x}{\eta^2} + \nabla_x \log \rho_{\text{target}}(x) \right), \quad (53)$$

1024 where f_1 and f_2 are neural networks parameterized by θ .

1025 For non-trust methods, we train for $60k$ gradient steps with a batch size of 2000, amounting to a total
 1026 of $120M$ target evaluations. In contrast, trust region methods use a buffer of length 50k refreshed 150
 1027 times during training, resulting in a total of $60k \times 150 = 7.5M$ target evaluations. To optimize for
 1028 the next control u_{i+1} , we perform 400 gradient steps on the replay buffer using randomly sampled
 1029 batches of size 2000. All experiments use a trust-region bound of $\varepsilon = 0.1$.

1030 For the *Many Well* target, we set the standard deviation of the prior distribution to 1 and to 2.5 for the
 1031 Gaussian mixture target. For the randomization of the mixing weights, we uniformly sample positive
 1032 values that are normalized and rescaled such that the ratio between the maximum mixing weight and
 1033 the minimum is 3. The diffusivity is scheduled according to $\zeta(t) = (C_{\text{max}} - C_{\text{min}}) \cos^2\left(\frac{t\pi}{2T}\right) + C_{\text{min}}$
 1034 with $C_{\text{min}} = 0.01$ and $C_{\text{max}} = 10$.

1035 **Evaluation protocol and model selection.** We follow the evaluation protocol of prior work [70] and
 1036 evaluate all performance criteria 100 times during training, using 2000 samples for each evaluation.
 1037 We apply a running average with a window of 5 evaluations to smooth out short-term fluctuations and
 1038 obtain more robust results within a single run. We conducted each experiment using four different
 1039 random seeds and averaged the best results for each run.

1040 **Benchmark problem details.** The *ManyWell* target involves a d -dimensional *double well* potential,
 1041 corresponding to the (unnormalized) density

$$\rho_{\text{target}}(x) = \exp \left(- \sum_{i=1}^m (x_i^2 - \delta)^2 - \frac{1}{2} \sum_{i=m+1}^d x_i^2 \right),$$

with $m \in \mathbb{N}$ representing the number of combined double wells (resulting in 2^m modes), and a separation parameter $\delta \in (0, \infty)$ (see also [124]). In our experiments, we set $m = 5$ leading to $2^m = 32$ modes. The separation parameter is set to $\delta = 4$. Since ρ_{target} factorizes across dimensions, we can compute a reference solution for $\log \mathcal{Z}$ via numerical integration, as described in [99].

Moreover, we consider a Gaussian mixture model (GMM) target of the form

$$p_{\text{target}}(x) = \sum_{k=1}^K \pi_k \mathcal{N}(x | \mu_k, \Sigma_k), \quad (54)$$

where $\mu_k \in \mathbb{R}^d$, $\Sigma_k \in \mathbb{R}^{d \times d}$, $\pi_k \geq 0$, and $\sum_{k=1}^K \pi_k = 1$. To compute the optimal control u^* , we exploit the fact that the optimal marginal path measures $\mathbb{Q}_t(x)$ can be derived analytically [100]:

$$\mathbb{Q}_t(x) = \sum_{k=1}^K \pi_k \mathcal{N}\left(x | \mu_k e^{-\int_t^T \zeta(s) ds}, \Sigma_k e^{-2 \int_t^T \zeta(s) ds} + \eta^2 \int_t^T 2\zeta(s) e^{-2 \int_t^s \zeta(u) du} ds\right). \quad (55)$$

and used for computing the optimal control u^* . Finally, to compute the total variation distance, we leverage the known true mixing weights π_k and define the mode partitions $S_k \subset \mathbb{R}^d$ as

$$S_k = \{x \in \mathbb{R}^d | \arg \max_j \pi_j \mathcal{N}(x | \mu_j, \Sigma_j) = k\}. \quad (56)$$

E.2 Evaluation criteria

Here, we provide further information on how our evaluation criteria are computed.

Control L^2 error. Assuming access to the optimal control u^* , we can compute the L^2 error between the optimal and the learned control, i.e.,

$$\text{control } L^2 \text{ error} := \mathbb{E} \left[\int_0^T \frac{1}{2} \|u^* - u\|^2 (X^{u^*}, s) ds \right] \quad (57)$$

where X^{u^*} is obtained by simulating the controlled process with u^* , and compute the error using a Monte Carlo estimate. Note that this quantity is equivalent to the forward Kullback-Leibler divergence

$$D_{\text{KL}}(\mathbb{Q} | \mathbb{P}^u) = \mathbb{E} \left[\log \frac{d\mathbb{Q}}{d\mathbb{P}^u}(X^{u^*}) \right]. \quad (58)$$

Via Girsanov's theorem (see App. A.3) we have that

$$\frac{d\mathbb{Q}}{d\mathbb{P}^u}(X^{u^*}) = \int_0^T (u^* - u)(X^{u^*}, s) \cdot dW_s + \int_0^T \frac{1}{2} \|u^* - u\|^2 (X^{u^*}, s) ds. \quad (59)$$

The desired equivalence follows from the fact that, under mild regularity assumptions, the stochastic integral in (59) is a martingale and has vanishing expectation.

Log-normalizing constant. By definition, the log-normalizing constant is given by

$$\mathcal{Z}(X_0) = \mathbb{E} \left[e^{-\mathcal{W}(X, 0)} | X_0 \right] = \mathbb{E} \left[\frac{d\mathbb{Q}}{d\mathbb{P}}(X) \right]. \quad (60)$$

Applying a change of measure to the controlled process yields

$$\mathcal{Z}(X_0) = \mathbb{E} \left[e^{-\mathcal{W}(X^u, 0)} \frac{d\mathbb{P}}{d\mathbb{P}^u}(X^u) | X_0 \right] = \mathbb{E} \left[e^{-\int_0^T \frac{1}{2} \|u(X_s^u, s)\|^2 ds - \int_0^T u(X_s^u, s) \cdot dW_s - \mathcal{W}(X^u, 0)} | X_0 \right], \quad (61)$$

which can be estimated via Monte Carlo using samples from the current control u .

Sinkhorn distance. We estimate the Sinkhorn distance \mathcal{W}_γ^2 [75], an entropy-regularized optimal transport distance, between model and target samples using the JAX-based ott library [76].

Total variation distance Inspired by recent work [70, 89], we assume access to ground truth mixing weights π_k , $k \in \{1, \dots, K\}$, along with a partition $\{S_1, \dots, S_K\}$ of \mathbb{R}^d , where each region $S_k \subset \mathbb{R}^d$ corresponds to the k th mode of the target distribution. We estimate the empirical mixing weights using

$$\hat{\pi}_k = \frac{\mathbb{E} [\mathbb{1}_{S_k}(X_T^u)]}{\sum_{k'=1}^K \mathbb{E} [\mathbb{1}_{S_{k'}}(X_T^u)]}. \quad (62)$$

Using these estimates, we compute the total variation distance (TVD) between the empirical and true mode weights as

$$\text{TVD} = \sum_{k=1}^K |\pi_k - \hat{\pi}_k|. \quad (63)$$

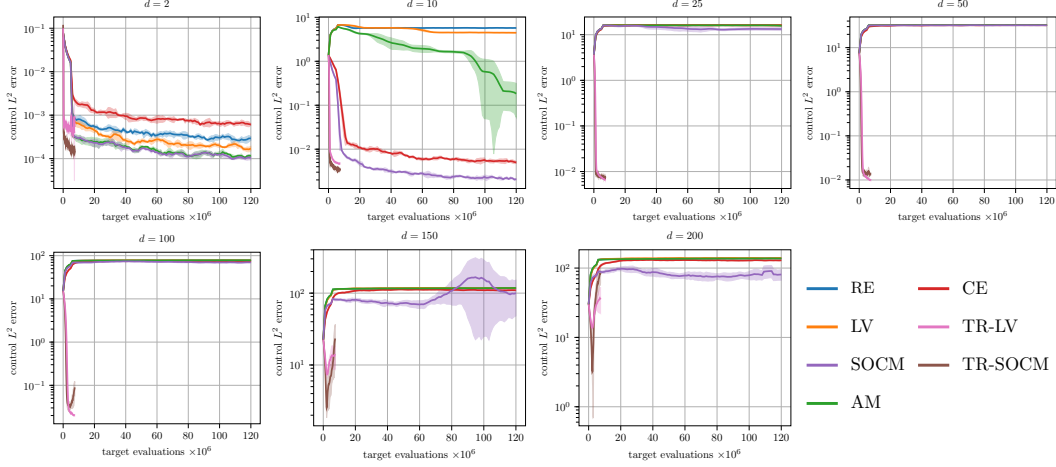


Figure 7: Control L^2 error as a function of the number of target evaluations for the GMM target across varying problem dimensionalities d . All results are averaged across four random seeds.

Details on how the ground truth mixing weights and the corresponding mode regions S_k are defined can be found in the descriptions of the target densities.

E.3 Additional experiments

Here, we provide results for additional numerical experiments.

Gaussian Mixture 40 (GMM40). We further evaluate the performance of trust-region-based losses by comparing them to existing SOC losses on the well-established GMM40 benchmark [99]. In this task, the target distribution is a Gaussian mixture model with 40 components, where the means are uniformly sampled from the interval $[-40, 40]$, and each component has an initial variance of 1. We set the prior’s standard deviation to $\eta = 30$. The results, presented in Fig. 8, show that only two losses, Cross-Entropy (CE) and trust-region with log-variance (TR-LV), can consistently learn all 40 modes. Notably, TR-LV achieves this with approximately ten times fewer target evaluations than CE.

Control L^2 error vs. target evaluations. We extend the results presented in Sec. 3.1 for the GMM benchmark by providing a detailed analysis of the control L^2 error as a function of the number of target evaluations across varying problem dimensionalities d . For $d = 2$, all SOC losses achieve low control error. However, at $d = 10$, some methods begin to exhibit elevated control error due to mode collapse. As the dimensionality increases further, only trust-region-based losses consistently maintain low control error. While these methods show partial mode collapse for $d \geq 150$, we anticipate that this issue can be mitigated by refining the control function architecture or by employing larger buffer and batch sizes. Importantly, trust-region methods also require significantly fewer target evaluations—a key advantage in many real-world applications where evaluations are costly.

Influence of trust region bounds. We further investigate the effect of different trust-region bound values ε on the GMM target using TR-LV. The results are presented in Fig. 10. The left figure shows that smaller trust-region bounds significantly improve performance: $\varepsilon = 0.01$ yields up to an order of magnitude lower control error compared to $\varepsilon = 1$. Additionally, smaller ε values help stabilize training, as evidenced by the reduced standard deviation across random seeds. In contrast, training with $\varepsilon = 1$ becomes unstable. However, this improved stability comes at the cost of slower convergence—smaller bounds require more training iterations to effectively anneal from the prior to the target path measure, as illustrated in the middle figure. Finally, the right figure shows that the empirically observed smoothed effective sample size (ESS) aligns well with its Taylor series approximation, $\text{ESS} = \left(\text{Var} \left(\frac{d\mathbb{P}^{(i+1)}}{d\mathbb{P}^{(i)}} \right) + 1 \right)^{-1} \approx \frac{1}{2\varepsilon+1}$ for small values of ε ; see App. J.3 for further details.

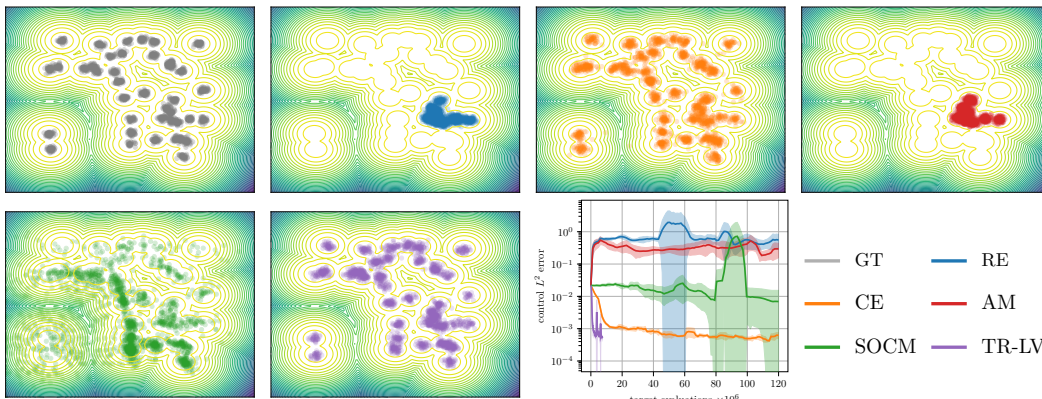


Figure 8: Qualitative and quantitative results for the GMM40 target. The qualitative plots demonstrate that only the CE (orange) and TR-LV (purple) losses successfully capture all 40 modes of the ground truth (GT, grey) distribution. This is further supported by the low L^2 control error observed for these two methods. Results are averaged across four random seeds and are not reported for the log-variance loss due to numerical instabilities.

F Transition path sampling

F.1 Experimental details

We build upon the codebase provided by TPS-DPS [112] (github.com/kiyoung98/tps-dps). Our experimental setups also follow [112] to ensure a fair comparison.

MD Simulation Setup. We run molecular dynamics simulation on the OpenMM platform. Both simulations are run at temperature 300K. For Alanine Dipeptide, we use the ‘amber99sbildn.xml’ forcefield with a VVVR integrator to simulate in vaccum. Each timestep is set as 1 femtosecond. Each path sampled is of length 1,000. For Chignolin, we use the ‘protein.ff14SBonlysc.xml’ forcefield with implicit solvent model ‘gbs2.xml’ with a VVVR integrator. Each timestep is set as 1 femtosecond. Each path sampled is of length 5,000.

Target Hit. For Alanine Dipeptide, target hit is defined over the two dihedral angles ϕ and ψ and a distance radius within 0.75Å. For Chignolin, a long MD simulation is pre-loaded with Time-lagged independent component analysis (TICA) to select the first two dimensions that capture most variance. The region is then defined over the two dimensions with a radius of 0.75.

Training process. Annealing is applied from 600K to 300K. A replay buffer is used with buffer size 1,000 and 200 for Alanine Dipeptide and Chignolin, respectively. and training over buffer per iteration is 1,000 times.

Hyperparameters: The trust-region constraint is set to $\varepsilon = 0.01$ for Alanine Dipeptide and $\varepsilon = 0.2$ for Chignolin. Batch size for both systems is set to 16, Alanine Dipeptide is trained for 2000 iterations, while Chignolin is trained for 50 iterations.

Computing Resources: Each experiment is run on a single 80GB NVIDIA H100 GPU.

F.2 Additional Experimental Result Discussion

We discuss our results in comparison to [112]. First of all, we evaluate three seed average as we notice the high variance nature of the transition path sampling problem—running several times can have huge variance in results (also evidenced in Fig. 4). We can also observe the trust-region constraint helps to stabilize the training significantly and thus have much smaller variance across three runs. Notably, for Alanine Dipeptide, both methods start with zero hitting percentage, while in Chignolin, in the beginning both methods already have some trajectories that hit the target, trust-region constraint is already effective in improving the efficiency. We use almost the exact same setup as in [112] with the only difference being the batch size for Chignolin is 16 instead of 4. We do not tune the model as our goal is to show the trust-region constraint improves the training stability and thus the efficiency and accuracy in terms of number of energy calls.

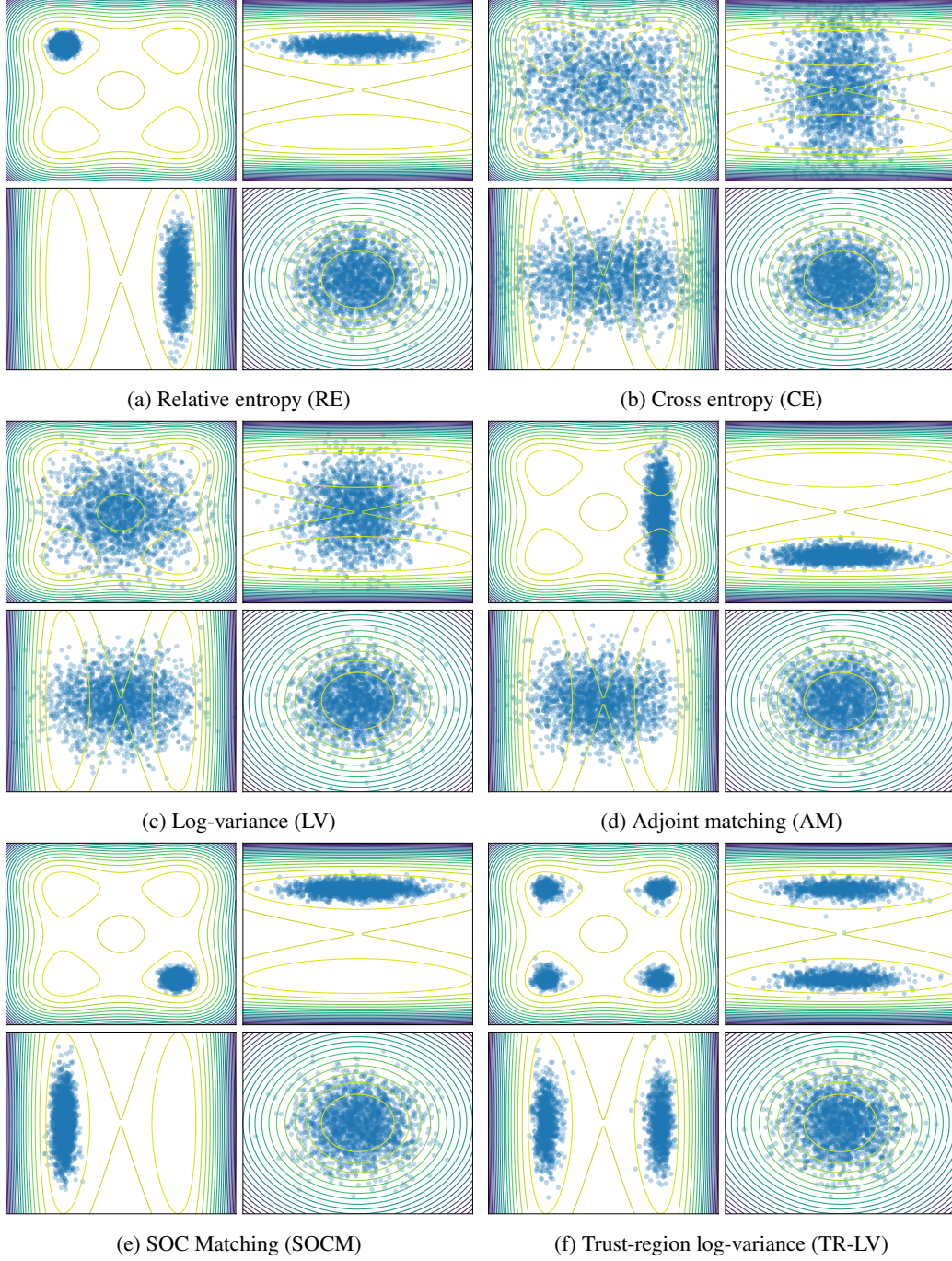


Figure 9: Qualitative results for the *Many Well* target with $d = 200$. Level plots depict the ground truth density for pairs of marginal distributions, while blue dots represent samples generated by models trained using the respective loss functions (indicated in the sub-captions). Among all methods, only the trust-region-based log-variance loss successfully avoids mode collapse and convergence issues. Interestingly, although the cross-entropy loss achieves the second-lowest estimation error for $\log \mathcal{Z}$ (see Fig. 3), the qualitative results suggest that the model fails to adequately capture the target distribution—likely due to the high variance of the importance weights. All visualizations are generated using the same random seed for consistency.

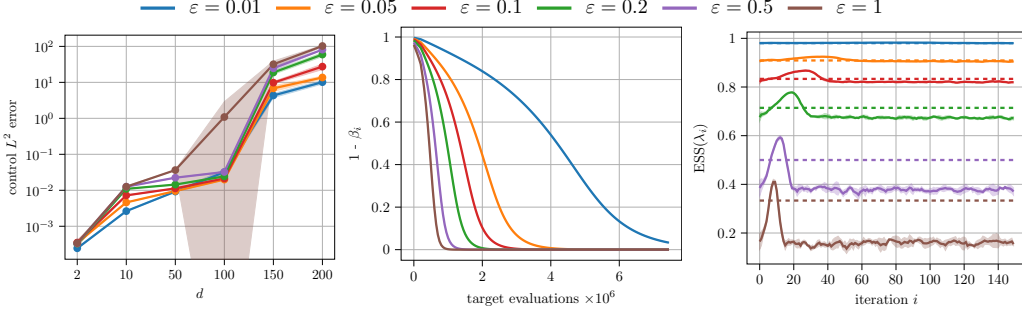


Figure 10: Influence of different trust region bound values ε on the GMM target for TR-LV. The left figure considers varying problem dimensionalities d whereas the middle and right figure report results for $d = 100$. The figure on the right shows the empirically observed smoothed effective sample size (ESS) and its approximation via Taylor series approximation, i.e., $\text{ESS} = \left(\text{Var} \left(\frac{d\mathbb{P}^{u_{i+1}}}{d\mathbb{P}^{u_i}} \right) + 1 \right)^{-1} \approx \frac{1}{2\varepsilon+1}$, with solid and dashed lines, respectively. All results are averaged across four random seeds.

G Classical SOC problems

Here, we consider classical SOC problems for which the optimal control can be computed analytically. These problems have been widely used in recent studies to compare different loss functions [101, 84, 81]. Here, we leverage them to showcase that importance sampling works in high dimensions when using trust-region based losses. To that end, we consider the comparison between the SOCM loss and its trust-region based counterpart.

G.1 Experimental Setup

The experimental setup follows the setup used for diffusion-based sampling, as explained in App. E.1, including control function architecture, hyperparameter evaluation protocol, and model selection.

For discretizing the SDE, we leverage the Euler-Maruyama scheme, i.e.,

$$\hat{X}_{n+1} = \hat{X}_n + (b + \sigma u)(\hat{X}_n, n\Delta t)\Delta t + \sigma(n)\sqrt{\Delta t}\xi_n, \quad \xi_n \sim \mathcal{N}(0, I). \quad (64)$$

Since the considered benchmark problems admit analytical solutions for the optimal control u^* , we consider the L^2 error between the learned and the optimal control for evaluating the models as explained in App. E.1.

G.2 Benchmark problem details

We consider two problems taken from [101], the *Quadratic Ornstein-Uhlenbeck (OU) easy* and *Quadratic Ornstein-Uhlenbeck (OU) hard*. For convenience, we briefly introduce them again here.

Quadratic Ornstein-Uhlenbeck (OU) The choices for the functions of the control problem are:

$$b(x, t) = Ax, \quad f(x, t) = x^\top Px, \quad g(x) = x^\top Qx, \quad \sigma(t) = \sigma_0. \quad (65)$$

where Q is a positive definite matrix. Control problems of this form are better known as linear quadratic regulator (LQR) and they admit a closed form solution [119]. The optimal control is given by:

$$u^*(x, t) = -2\sigma_0^\top F(t)x, \quad (66)$$

where $F(t)$ is the solution of the Ricatti equation

$$\frac{dF(t)}{dt} + A^\top F(t) + F(t)A - 2F(t)\sigma_0\sigma_0^\top F(t) + P = 0 \quad (67)$$

with the final condition $F(T) = Q$. Within the Quadratic OU class, we consider two settings:

- Easy: We set $A = 0.2I$, $P = 0.2I$, $Q = 0.1I$, $\sigma_0 = I$, $\lambda = 1$, $T = 1$, $x_{\text{init}} \sim 0.5\mathcal{N}(0, I)$.
- Hard: We set $A = I$, $P = I$, $Q = 0.5I$, $\sigma_0 = I$, $\lambda = 1$, $T = 1$, $x_{\text{init}} \sim 0.5\mathcal{N}(0, I)$.

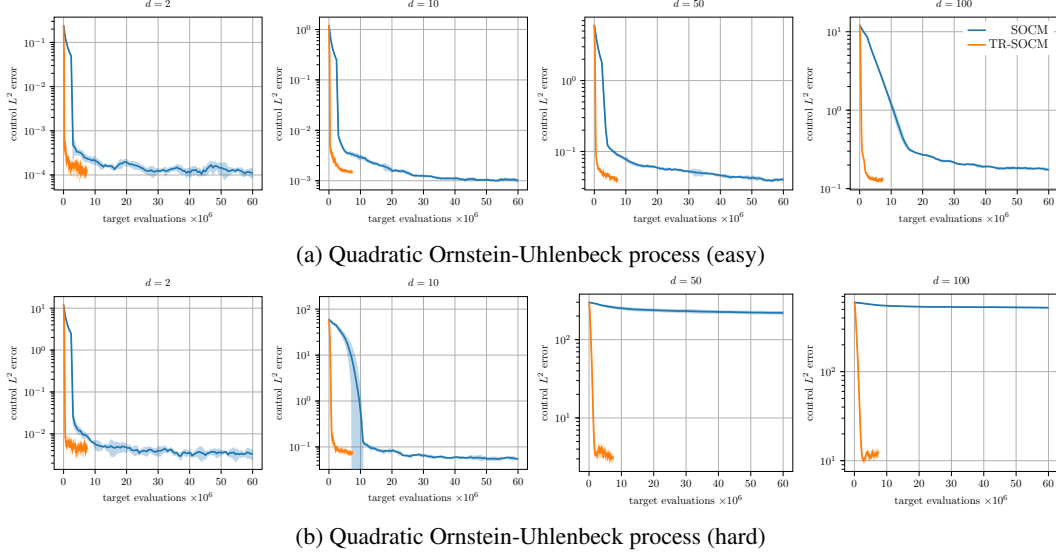


Figure 11: Control L^2 error as a function of the number of target evaluations for the quadratic OU problem across varying problem dimensionalities d . All results are averaged across four random seeds.

G.3 Results

We compare the performance of SOCM and its trust-region-based variant (TR-SOCM) on the quadratic Ornstein–Uhlenbeck (OU) problem across varying problem dimensionalities d . Both approaches rely on importance sampling, which is known to be challenging in high-dimensional settings. This experiment highlights the role of trust regions in scaling to such regimes. Results are presented in Fig. 11.

In low-dimensional settings ($d \leq 10$), both methods perform comparably, although TR-SOCM exhibits significantly better sample efficiency. As the dimensionality increases ($d \geq 50$), the performance of SOCM deteriorates markedly, while TR-SOCM continues to achieve low control error. For the more challenging variant of the quadratic OU problem, SOCM fails to meaningfully improve upon its initialization, whereas TR-SOCM demonstrates consistent error reduction.

These results suggest that trust regions are particularly beneficial in high-dimensional and difficult problem settings, where they provide stability and improved performance.

H Fine-tuning of diffusion models

We take the adjoint matching (AM) implementation in `github.com/microsoft/soc-fine-tuning-sd` as our baseline, and we modify it to implement TR-SOCM.

H.1 Fine-tuning experimental details

We generate images using classifier free guidance, with guidance scale 7.5. We use 50 inference timesteps to sample the rollouts during fine-tuning, and the evaluation samples are also generated at 50 inference timesteps.

We fine-tune using the default hyperparameters in the repo: we use AdamW, using learning rate 3×10^{-6} , beta 1 set to 0.9, beta 2 set to 0.95, and weight decay 0. We use an effective batch size of 500 trajectories, and 4 model backpropagations per trajectory. For the TR-SOCM loss, we use a buffer size of size 100, and 10 passes per buffer.

We use the 10000 fine-tuning prompts taken from the repository for [125], and the 100 validation prompts from the same repository (see <https://github.com/THUDM/ImageReward>). The two prompts used in Figure 6 are "masterpiece, best quality, realistic photograph, 8k, high detailed vintage motorcycle parked on a wet cobblestone street at dusk, neon reflections, shallow depth of field" and "masterpiece, best quality, ultra detailed, 8k renaissance cathedral interior, light streaming through stained-glass windows onto marble floors".

I Trust region SOC sequences are equispaced in the Fisher-Rao distance

For a fixed ε , suppose that we construct the sequence of controls $(u_{i+1})_{i \geq 0}$ as the solutions of the problem (4). As shown in Prop. 2.2, we have that

$$\frac{d\mathbb{P}^{u_i}}{d\mathbb{P}} \propto \left(\frac{d\mathbb{Q}}{d\mathbb{P}} \right)^{\beta_i} \left(\frac{d\mathbb{P}^{u_0}}{d\mathbb{P}} \right)^{1-\beta_i}, \quad \text{with} \quad \beta_i = 1 - \prod_{j=0}^{i-1} \frac{\lambda_j}{1+\lambda_j} \quad (68)$$

If we define the family $(\mathbb{Q}^{(\tau)})_{\tau \in [0,1]}$ such that

$$\frac{d\mathbb{Q}^{(\tau)}}{d\mathbb{P}} \propto \left(\frac{d\mathbb{Q}}{d\mathbb{P}} \right)^{\tau} \left(\frac{d\mathbb{P}^{u_0}}{d\mathbb{P}} \right)^{1-\tau}, \quad (69)$$

we can write $\mathbb{P}^{u_i} = \mathbb{Q}^{(\beta_i)}$. Hence, we can regard the sequence $(\mathbb{P}^{u_i})_{i \geq 0}$ as a discretization of the family $(\mathbb{Q}^{(\tau)})_{\tau \in [0,1]}$. Next, we characterize this discretization more precisely using tools from information geometry.

I.1 Basics on information geometry

Let $\{p(x; \theta)\}_{\theta \in \Theta}$ be a parametric family of probability densities (or mass functions) on the sample space \mathcal{X} , and let X be a random variable with distribution $p(x; \theta)$.

Definition I.1 (Fisher Information Matrix). The *Fisher information matrix* at θ is defined as

$$\mathcal{I}(\theta) = \mathbb{E}_{X \sim p(\cdot; \theta)} \left[\nabla_{\theta} \log p(X; \theta) (\nabla_{\theta} \log p(X; \theta))^{\top} \right] = -\mathbb{E}_{X \sim p(\cdot; \theta)} \left[\nabla_{\theta}^2 \log p(X; \theta) \right],$$

where ∇_{θ} denotes the column gradient with respect to θ , and ∇_{θ}^2 the Hessian.

As an average of positive semi-definite matrices, $\mathcal{I}(\theta)$ is positive semi-definite, which makes it possible to define a geometric structure:

Definition I.2 (Statistical Manifold). Let $\{p(x; \theta)\}_{\theta \in \Theta}$ be a smooth parametric family of probability densities on \mathcal{X} , with parameter space $\Theta \subseteq \mathbb{R}^d$. Then Θ itself can be viewed as a d -dimensional differentiable manifold

$$\mathcal{M} = \{p(\cdot; \theta) : \theta \in \Theta\} \cong \Theta,$$

called the *statistical manifold* of the model. Endow \mathcal{M} with the Riemannian metric

$$g_{ij}(\theta) = \mathcal{I}_{ij}(\theta) = \mathbb{E}_{X \sim p(\cdot; \theta)} \left[\partial_i \log p(X; \theta) \partial_j \log p(X; \theta) \right],$$

where $\partial_i = \frac{\partial}{\partial \theta_i}$. This g is known as the *Fisher-Rao metric*, turning (\mathcal{M}, g) into the canonical information-geometric manifold of the model.

Next, we review the definition of the length of a curve on a Riemannian manifold.

Definition I.3 (Length of a Curve on a Riemannian Manifold). Let (\mathcal{M}, g) be a d -dimensional Riemannian manifold, and let $\gamma: [a, b] \rightarrow \mathcal{M}$ be a piecewise smooth curve. Choose local coordinates $\theta = (\theta^1, \dots, \theta^d)$ on an open set $\mathcal{U} \subset \mathcal{M}$ containing the image of γ , so that $\gamma(t) \mapsto \theta(t) = (\theta^1(t), \dots, \theta^d(t))$. Then the *length* of γ is

$$L(\gamma) = \int_a^b \sqrt{g_{ij}(\theta(t)) \dot{\theta}^i(t) \dot{\theta}^j(t)} dt,$$

where $\dot{\theta}^i(t) = \frac{d\theta^i}{dt}(t)$ and we employ the Einstein summation convention on repeated indices $i, j = 1, \dots, d$.

A geodesic between two points $\theta_1, \theta_2 \in \mathcal{M}$ is a piecewise smooth curve $\gamma: [a, b] \rightarrow \mathcal{M}$ such that $\gamma(a) = \theta_1, \gamma(b) = \theta_2$ that minimizes the length functional L locally. Any time reparameterization of a geodesic is also a geodesic, because The geodesic distance between θ_1, θ_2 is the infimum over the lengths of all geodesics (or all piecewise smooth curves) between θ_1, θ_2 .

Definition I.4 (Fisher-Rao distance). The geodesic distance induced by the Fisher-Rao metric is known as the *Fisher-Rao distance*.

I.1.1 A FIRST EXAMPLE: THE MANIFOLD OF SMOOTH DENSITIES

A first instantiation of this framework is on the manifold of smooth densities, defined below.

Definition I.5 (The Manifold of Smooth Densities). Let \mathcal{X} be a measurable space and denote by $\mathcal{P} = \{p: \mathcal{X} \rightarrow \mathbb{R}_{>0} \mid \int_{\mathcal{X}} p(x) dx = 1\}$ the set of smooth, strictly positive probability

densities on \mathcal{X} . Then, \mathcal{P} is an infinite-dimensional (Fréchet) manifold whose tangent space at p is $T_p\mathcal{P} = \{h: \mathcal{X} \rightarrow \mathbb{R} \mid \int_{\mathcal{X}} h(x) dx = 0\}$.

Definition I.6 (The Fisher–Rao Metric and Distance on \mathcal{P}). The *Fisher–Rao metric* G on \mathcal{P} is the Riemannian metric defined at each $p \in \mathcal{P}$ by

$$G_p(h_1, h_2) = \int_{\mathcal{X}} \frac{h_1(x) h_2(x)}{p(x)} dx, \quad h_1, h_2 \in T_p\mathcal{P}.$$

Equivalently, for a smooth path $p_t \in \mathcal{P}$,

$$\|\dot{p}_t\|_{p_t} = \sqrt{G_{p_t}(\dot{p}_t, \dot{p}_t)} = \sqrt{\int_{\mathcal{X}} \frac{(\partial_t p_t(x))^2}{p_t(x)} dx}.$$

The *Fisher–Rao distance* between two densities $p, q \in \mathcal{P}$ is the infimum of the G -lengths of all smooth paths connecting them:

$$D_{\text{FR}}(p, q) = \inf_{\substack{p_t \in \mathcal{P} \\ p_0=p, p_1=q}} \int_0^1 \sqrt{\int_{\mathcal{X}} \frac{(\partial_t p_t(x))^2}{p_t(x)} dx} dt.$$

Proposition I.7 (Closed-form of the Fisher–Rao distance and geodesics on \mathcal{P}). Under the embedding $\varphi: \mathcal{P} \rightarrow L^2(\mathcal{X})$, $\varphi(p) = \sqrt{p}$, \mathcal{P} is isometric (up to a factor of 2) to the positive orthant of the unit sphere in L^2 . Hence one obtains the closed-form

$$D_{\text{FR}}(p, q) = 2 \arccos\left(\int_{\mathcal{X}} \sqrt{p(x)q(x)} dx\right).$$

Let $\alpha = \arccos(\int_{\mathcal{X}} \sqrt{p(x)q(x)} dx)$. The same argument shows that the Fisher–Rao geodesic p_t for $t \in [0, 1]$ is given via the square-root map $\varphi(p) = \sqrt{p}$ by

$$\varphi(p_t) = \frac{\sin((1-t)\alpha)}{\sin \alpha} \varphi(p) + \frac{\sin(t\alpha)}{\sin \alpha} \varphi(q),$$

so that in the original density space

$$p_t(x) = \left(\frac{\sin((1-t)\alpha)}{\sin \alpha} \sqrt{p(x)} + \frac{\sin(t\alpha)}{\sin \alpha} \sqrt{q(x)} \right)^2.$$

In particular, $p_0 = p$, $p_1 = q$, and this path realizes the infimum in $D_{\text{FR}}(p, q) = \int_0^1 \|\dot{p}_t\|_{p_t} dt$.

I.2 A second example: exponential families

Definition I.8 (The Exponential-Family Manifold). Let

$$p(x; \theta) = \exp(\theta^i T_i(x) - A(\theta)) h(x), \quad \theta = (\theta^1, \dots, \theta^d) \in \Theta \subseteq \mathbb{R}^d$$

be a regular d -parameter exponential family on \mathcal{X} . The parameter space Θ (equipped with the atlas coming from the coordinates θ^i) is a d -dimensional differentiable manifold, which we identify with the statistical model

$$\mathcal{M} = \{p(\cdot; \theta) \mid \theta \in \Theta\}.$$

Its tangent space at θ is $T_{\theta}\mathcal{M} \cong \mathbb{R}^d$, with basis $\{\partial/\partial\theta^i\}$.

Definition I.9 (Fisher–Rao Metric). The Fisher–Rao metric on \mathcal{M} is the Riemannian metric whose components in the natural coordinate chart θ are

$$g_{ij}(\theta) = \mathbb{E}_{X \sim p(\cdot; \theta)} [\partial_i \log p(X; \theta) \partial_j \log p(X; \theta)] = -\mathbb{E}_{X \sim p(\cdot; \theta)} [\partial_{ij} \log p(X; \theta)] = \frac{\partial^2 A(\theta)}{\partial \theta^i \partial \theta^j}.$$

Equivalently, $g(\theta) = \nabla^2 A(\theta)$, the Hessian of the log-partition function.

For general exponential families, the Fisher–Rao distance and the geodesics do not admit a closed form. Yet, one-dimensional families can be handled explicitly, because geodesics are trivial:

Proposition I.10 (One-Parameter Exponential Family). If $d = 1$ then $\theta \in (a, b) \subseteq \mathbb{R}$, and $g(\theta) = A''(\theta)$. Hence

$$\text{FR}(\theta_1, \theta_2) = \left| \int_{\theta_1}^{\theta_2} \sqrt{A''(\theta)} d\theta \right|.$$

In particular, for a Poisson family $A(\theta) = e^{\theta}$ one gets $D_{\text{FR}}(\theta_1, \theta_2) = |e^{\theta_2/2} - e^{\theta_1/2}|$.

1253 I.3 Local expansion of the Kullback–Leibler divergence

1254 **Proposition I.11** (Second-Order Expansion of KL). *Let $\{p(x; \theta)\}_{\theta \in \Theta}$ be a smooth parametric family*
 1255 *of densities, and fix $\theta \in \Theta$. For a small increment $\delta \in \mathbb{R}^d$, consider*

$$\text{KL}(p(\cdot; \theta + \delta) \| p(\cdot; \theta)) = \int_{\mathcal{X}} p(x; \theta + \delta) \log \frac{p(x; \theta + \delta)}{p(x; \theta)} dx.$$

1256 *Then one has the Taylor expansion*

$$\text{KL}(p(\theta + \delta) \| p(\theta)) = \underbrace{0}_{\text{constant term}} + \underbrace{0}_{\text{linear term}} + \frac{1}{2} \delta^i I_{ij}(\theta) \delta^j + O(\|\delta\|^3),$$

1257 *where*

$$I_{ij}(\theta) = \mathbb{E}_{X \sim p(\cdot; \theta)} [\partial_i \log p(X; \theta) \partial_j \log p(X; \theta)]$$

1258 *is the Fisher information matrix. Equivalently,*

$$\left. \frac{\partial \text{KL}}{\partial \delta^i} \right|_{\delta=0} = 0, \quad \left. \frac{\partial^2 \text{KL}}{\partial \delta^i \partial \delta^j} \right|_{\delta=0} = I_{ij}(\theta).$$

1259 *Sketch.* Expand both $p(x; \theta + \delta)$ and $\log p(x; \theta + \delta)$ to second order in δ , substitute into the integral,
 1260 and use $\int p \partial_i \log p dx = 0$ and $\int p \partial_i \partial_j \log p dx = -I_{ij}(\theta)$ to verify cancellation of constant and
 1261 linear terms, leaving the stated quadratic form. \square

1262 I.4 Information geometry on the exponential family of path measures

1263 We can view the family $(\mathbb{Q}^{(\tau)})_{\tau \in [0,1]}$ defined in (69) as a one-parameter exponential family by
 1264 rewriting $\mathbb{Q}^{(\tau)}$ as

$$\frac{d\mathbb{Q}^{(\tau)}}{d\mathbb{P}^{u_0}} = \exp \left(\tau \left(\log \frac{d\mathbb{Q}}{d\mathbb{P}^{u_0}} \right) - A(\tau) \right), \quad (70)$$

1265 where the log-partition function $A(\tau)$ is defined as

$$A(\tau) = \log \mathbb{E}_{\mathbb{P}^{u_0}} \left[\left(\frac{d\mathbb{Q}}{d\mathbb{P}^{u_0}} \right)^\tau \right]. \quad (71)$$

1266 Equivalently, we can write it as an exponential family centered on an arbitrary $\tau \in [0, 1]$:

$$\frac{d\mathbb{Q}^{(\tau+\Delta\tau)}}{d\mathbb{Q}^{(\tau)}} = \exp \left(\Delta\tau \left(\log \frac{d\mathbb{Q}}{d\mathbb{P}^{u_0}} \right) - A_\tau(\Delta\tau) \right), \quad (72)$$

1267 where

$$A_\tau(\Delta\tau) := \log \mathbb{E}_{\mathbb{Q}^{(\tau)}} \left[\left(\frac{d\mathbb{Q}}{d\mathbb{P}^{u_0}} \right)^{\Delta\tau} \right]. \quad (73)$$

1268 **Deriving an expression for the Fisher information.** Observe that by construction

$$\begin{aligned} A_\tau(\Delta\tau) &:= \log \mathbb{E}_{\mathbb{P}^{u_0}} \left[\left(\frac{d\mathbb{Q}}{d\mathbb{P}^{u_0}} \right)^{\Delta\tau} \frac{d\mathbb{Q}^{(\tau)}}{d\mathbb{P}^{u_0}} \right] = \log \mathbb{E}_{\mathbb{P}^{u_0}} \left[\left(\frac{d\mathbb{Q}}{d\mathbb{P}^{u_0}} \right)^{\Delta\tau} \left(\frac{d\mathbb{Q}}{d\mathbb{P}^{u_0}} \right)^\tau \exp(-A(\tau)) \right] \\ &= A(\tau + \Delta\tau) - A(\tau), \end{aligned} \quad (74)$$

1269 which means that $A'_\tau(0) = A'(\tau)$ for all $\tau \in (0, 1)$. Thus, by Prop. I.10, we conclude that the Fisher
 1270 information matrix, which is a scalar because the manifold is one-dimensional, reads

$$\mathcal{I}(\tau) = A''(\tau) = A''_\tau(0). \quad (75)$$

1271 Computing the first and second derivatives of A_τ is straightforward:

$$A'_\tau(\Delta\tau) = \frac{\mathbb{E}_{\mathbb{Q}^{(\tau)}} \left[\log \left(\frac{d\mathbb{Q}}{d\mathbb{P}^{u_0}} \right) \left(\frac{d\mathbb{Q}}{d\mathbb{P}^{u_0}} \right)^{\Delta\tau} \right]}{\mathbb{E}_{\mathbb{Q}^{(\tau)}} \left[\left(\frac{d\mathbb{Q}}{d\mathbb{P}^{u_0}} \right)^{\Delta\tau} \right]}, \quad (76)$$

$$A''_\tau(0) = \mathbb{E}_{\mathbb{Q}^{(\tau)}} \left[\log \left(\frac{d\mathbb{Q}}{d\mathbb{P}^{u_0}} \right)^2 \right] - \mathbb{E}_{\mathbb{Q}^{(\tau)}} \left[\log \left(\frac{d\mathbb{Q}}{d\mathbb{P}^{u_0}} \right) \right]^2,$$

1272 and this implies that

$$\mathcal{I}(\tau) = \text{Var}_{\mathbb{Q}^{(\tau)}} \left[\log \left(\frac{d\mathbb{Q}}{d\mathbb{P}^{u_0}} \right) \right]. \quad (77)$$

1273 **Connecting the trust region constraint to the Fisher information.** Applying Proposition I.11, we
 1274 obtain that

$$\text{KL}(\mathbb{Q}^{(\tau+\Delta\tau)}|\mathbb{Q}^{(\tau)}) = \frac{\Delta\tau^2}{2}\mathcal{I}(\tau) + O(\Delta\tau^3), \quad (78)$$

1275 When we set $\tau + \Delta\tau = \beta_{i+1}$, $\tau = \beta_i$, we have that

$$\varepsilon = \text{KL}(\mathbb{P}^{u_{i+1}}|\mathbb{P}^{u_i}) = \frac{\Delta\tau^2}{2}\mathcal{I}(\tau) + O(\Delta\tau^3). \quad (79)$$

1276 Thus,

$$\Delta\tau = \sqrt{\frac{2\varepsilon}{\mathcal{I}(\tau)}} + O(\Delta\tau^{3/2}), \quad (80)$$

1277 Moreover, the Fisher-Rao distance between \mathbb{P}^{u_0} and $\mathbb{P}^{(i)}$, or rather, between 0 and β_i ,

$$\text{FR}(0, \beta_i) = \int_0^{\beta_i} \sqrt{\mathcal{I}(\tau)} d\tau. \quad (81)$$

1278 Then, the difference between Fisher-Rao distances $\text{FR}(0, \beta_{i+1})$ and $\text{FR}(0, \beta_i)$ which is equal to the
 1279 Fisher-Rao distance $\text{FR}(\beta_i, \beta_{i+1})$ is

$$\begin{aligned} \text{FR}(0, \beta_{i+1}) - \text{FR}(0, \beta_i) &= \text{FR}(\beta_i, \beta_{i+1}) = \int_{\beta_i}^{\beta_{i+1}} \sqrt{\mathcal{I}(\tau)} d\tau \\ &= (\sqrt{\mathcal{I}(\beta_i)} + O(\beta_{i+1} - \beta_i))(\beta_{i+1} - \beta_i) = \sqrt{\mathcal{I}(\beta_i)}\Delta\tau + O(\Delta\tau^2) \\ &= \sqrt{\mathcal{I}(\beta_i)}\sqrt{\frac{2\varepsilon}{\mathcal{I}(\beta_i)}} + O(\Delta\tau^{3/2}) = \sqrt{2\varepsilon} + O(\Delta\tau^{3/2}). \end{aligned} \quad (82)$$

1280 In continuous time, we have a curve $\beta : \mathbb{R}^{>0} \rightarrow [0, 1]$, and

$$\frac{d}{dt}\text{FR}(0, \beta(t)) = \sqrt{\mathcal{I}(\beta(t))}\beta'(t) = \sqrt{\mathcal{I}(\beta(t))}\sqrt{\frac{2}{\mathcal{I}(\beta(t))}} = \sqrt{2} \quad (83)$$

1281 Thus, we have shown the following result:

1282 **Proposition I.12.** *Up to high order terms, the elements of sequence $(\mathbb{P}^{u_i})_{0 \leq i \leq I-1}$ are equispaced in*
 1283 *the Fisher-Rao distance. The last term \mathbb{P}^{u_I} is equal to the target distribution \mathbb{Q} .*

1284 **A Monte Carlo estimate for the Fisher information.** By equation (70), we have that $\log \frac{d\mathbb{Q}^{(\tau)}}{d\mathbb{P}^{u_0}} =$
 1285 $\tau \left(\log \frac{d\mathbb{Q}}{d\mathbb{P}^{u_0}} \right) - A(\tau)$. Hence, we can rewrite (77) as

$$\mathcal{I}(\tau) = \frac{1}{\tau^2} \text{Var}_{\mathbb{Q}^{(\tau)}} \left[\log \left(\frac{d\mathbb{Q}^{(\tau)}}{d\mathbb{P}^{u_0}} \right) \right], \quad (84)$$

1286 which provides a way to estimate $\mathcal{I}(\tau)$, leveraging the Girsanov theorem to estimate $\log \left(\frac{d\mathbb{Q}^{(\tau)}}{d\mathbb{P}^{u_0}} \right) =$
 1287 $\log \left(\frac{d\mathbb{Q}^{(\tau)}}{d\mathbb{P}} \right) - \log \left(\frac{d\mathbb{P}^{u_0}}{d\mathbb{P}} \right)$.

1288 J Details on trust region SOC algorithms

1289 J.1 Characterizing the solutions of the trust region optimization problem

1290 **Proposition J.1** (Characterizing the solutions of the trust region optimization problem). *The solution*
 1291 *$\mathbb{P}^{u_{i+1}}$ of the problem (9) is unique and it satisfies*

- 1292 • *If $D_{\text{KL}}(\mathbb{P}^{u_i}|\mathbb{Q}) \leq \varepsilon$, then $\mathbb{P}^{u_{i+1}} = \mathbb{Q}$.*
- 1293 • *If $D_{\text{KL}}(\mathbb{P}^{u_i}|\mathbb{Q}) \geq \varepsilon$, then $D_{\text{KL}}(\mathbb{P}^{u_{i+1}}|\mathbb{P}^{u_i}) = \varepsilon$, i.e. $\mathbb{P}^{u_{i+1}}$ is also the unique solution of the*
 1294 *problem*

$$\arg \min_{u \in \mathcal{U}} D_{\text{KL}}(\mathbb{P}^u|\mathbb{Q}) \quad \text{s.t.} \quad D_{\text{KL}}(\mathbb{P}^u|\mathbb{P}^{u_i}) = \varepsilon. \quad (85)$$

1295 *Proof.* To prove the first case, observe that \mathbb{Q} is the only solution of the unconstrained problem
 1296 $\arg \min_{\mathbb{P} \in \mathcal{P}} D_{\text{KL}}(\mathbb{P}|\mathbb{Q})$, which means that it is also the unique solution of the problem (9) since it
 1297 satisfies its constraint. To prove the second case, by the Karush-Kuhn-Tucker (KKT) conditions, we
 1298 have that either $\lambda = 0$, or $D_{\text{KL}}(\mathbb{P}^{u_{i+1}}|\mathbb{P}^{u_i}) = \varepsilon$. Thus, if $D_{\text{KL}}(\mathbb{P}^{u_{i+1}}|\mathbb{P}^{u_i}) < \varepsilon$, then we must have
 1299 that $\lambda = 0$. The first order optimality condition for the problem is as follows: for any perturbation v
 1300 of the control u_{i+1} , we have that

$$0 = \frac{d}{d\eta} (D_{\text{KL}}(\mathbb{P}^{u_{i+1}+\eta v}|\mathbb{Q}) + \lambda (D_{\text{KL}}(\mathbb{P}^{u_{i+1}+\eta v}|\mathbb{P}^{u_i}) - \varepsilon))|_{\eta=0} = \frac{d}{d\eta} D_{\text{KL}}(\mathbb{P}^{u_{i+1}+\eta v}|\mathbb{Q})|_{\eta=0}, \quad (86)$$

Algorithm 2 Trust Region SOC with buffer

Require: Neural network u_θ with parameters θ , target path measure \mathbb{Q} , buffer size K , time discretization $S = (s_j)_{j=0}^J \subset [0, T]$, number of steps N per annealing, termination threshold δ
Initialize $i = 0$ and $\lambda_0 = \infty$
for $i = 0, 1, \dots$ **do**
 Define $u_i = u_\theta$ (detached)
 Simulate K trajectories $(X_s^{(k)})_{s \in S}$ of the SDE in (1) with Brownian motion $W_s^{(k)}$ and control u_i
 Compute importance weights $w^{(k)} = \frac{d\mathbb{Q}}{d\mathbb{P}^{u_i}}(X^{(k)}) \propto \exp(-\mathcal{W}_i(X^{(k)}, 0))$ as in (12)
 Initialize buffer $\mathcal{B} = \{(W^{(k)}, X^{(k)}, w^{(k)})\}_{k=1}^K$
 Compute multiplier $\lambda_i = \arg \max_{\lambda \in \mathbb{R}^+} \mathcal{L}_{\text{Dual}}^{(i)}(\lambda)$ as in (14) using \mathcal{B} and a 1-dim. non-linear solver
 if $\lambda_i \leq \delta$ **then**
 return control u_i with $\mathbb{P}^{u_i} \approx \mathbb{Q}$
 if adjoint matching loss **then**
 Compute annealing $\beta_{i+1} = 1 - \prod_{j=0}^i \frac{\lambda_j}{1+\lambda_j}$ as in Prop. 2.2
 Compute lean adjoint states $a_s^{(k)} = a_{i+1}(X_s^{(k)}, s)$, $s \in S$, as in (18) and store in \mathcal{B}
 for $n = 1, \dots, N$ **do**
 if adjoint matching loss **then**
 Estimate $\mathcal{L}(\theta) = \mathbb{E}_{(X, w, a) \sim \mathcal{B}, s \sim \text{Unif}(S)} [\|\sigma^\top a_s - u_\theta(X_s, s)\| w^{\frac{1}{1+\lambda_i}}]$ as in (19)
 if log-variance loss **then**
 Estimate $\mathcal{L}(\theta) = \text{Var}_{(W, X, w) \sim \mathcal{B}} [\sum_{j=1}^J \frac{\|\Delta_j\|^2 (s_j - s_{j-1})}{2} + \Delta_j \cdot (W_{s_j} - W_{s_{j-1}}) + \frac{1}{1+\lambda_i} \log w]$
 with $\Delta_j = u_i(X_{s_j}, s_j) - u_\theta(X_{s_j}, s_j)$ as in (16)
 Perform a gradient-descent step on $\mathcal{L}(\theta)$

1301 which means that u_{i+1} satisfies the first-order optimality condition for the relative entropy loss
1302 $u \mapsto D_{\text{KL}}(\mathbb{P}^u | \mathbb{Q})$. By [83, Prop. 2], the only control that satisfies the first-order optimality condition
1303 for the relative entropy loss is the optimal control u^* , which implies that $\mathbb{P}^{u_{i+1}} = \mathbb{Q}$, which is
1304 a contradiction because $\varepsilon > D_{\text{KL}}(\mathbb{P}^{u_{i+1}} | \mathbb{P}^{u_i}) = D_{\text{KL}}(\mathbb{P} | \mathbb{Q}) \geq \varepsilon$. Hence, we conclude that
1305 $D_{\text{KL}}(\mathbb{P}^{u_{i+1}} | \mathbb{P}^{u_i}) = \varepsilon$. To show that the solution $\mathbb{P}^{u_{i+1}}$ is unique, we use that $\mathbb{P} \mapsto D_{\text{KL}}(\mathbb{P} | \mathbb{P}^{u_i})$
1306 is strictly convex, and that $\{\mathbb{P} | D_{\text{KL}}(\mathbb{P} | \mathbb{P}^{u_i}) \leq \varepsilon\}$ is a convex set because it is the sublevel set of a
1307 convex functional. \square

1308 J.2 Implementation

1309 We provide a detailed version of Algorithm 1 in Algorithm 2. The hyperparameters and repositories
1310 for the experiments on unnormalized densities, transition path sampling, and fine-tuning can be found
1311 in the respective sections in Apps. E, F and H.

1312 J.3 Variance of the importance weights and trust region bounds

1313 As mentioned in Remark 2.1, one motivation of the trust region constrain $D_{\text{KL}}(\mathbb{P}^u | \mathbb{P}^{u_i}) \leq \varepsilon$ defined
1314 in (4) is to keep the variance of the importance weights between two consecutive measures \mathbb{P}^{u_i} and
1315 $\mathbb{P}^{u_{i+1}}$ small. This can be motivated by the inequality

$$\text{Var}_{\mathbb{P}^{u_i}} \left(\frac{d\mathbb{P}^{u_{i+1}}}{d\mathbb{P}^{u_i}} \right) = \mathbb{E}_{\mathbb{P}^{u_i}} \left[\left(\frac{d\mathbb{P}^{u_{i+1}}}{d\mathbb{P}^{u_i}} \right)^2 - 1 \right] = \mathbb{E}_{\mathbb{P}^{u_{i+1}}} \left[\frac{d\mathbb{P}^{u_{i+1}}}{d\mathbb{P}^{u_i}} - 1 \right] \quad (87a)$$

$$\geq \exp \left(\mathbb{E}_{\mathbb{P}^{u_{i+1}}} \left[\log \frac{d\mathbb{P}^{u_{i+1}}}{d\mathbb{P}^{u_i}} \right] \right) - 1 = \exp(D_{\text{KL}}(\mathbb{P}^{u_{i+1}} | \mathbb{P}^{u_i})) - 1, \quad (87b)$$

1316 which follows by Jensen's inequality. While a lower bound on the variance is not straight forward for
1317 path space measures (cf. [91]), we can consider the following heuristics. Let us assume that

$$\frac{d\mathbb{P}^{u_{i+1}}}{d\mathbb{P}^{u_i}} \approx 1 \quad (88)$$

1318 \mathbb{P}^{u_i} - and $\mathbb{P}^{u_{i+1}}$ -almost surely, which is reasonable if $D_{\text{KL}}(\mathbb{P}^{u_{i+1}} | \mathbb{P}^{u_i}) \leq \varepsilon$ with $\varepsilon \ll 1$. By a Taylor
1319 approximation it then holds

$$\left(\frac{d\mathbb{P}^{u_{i+1}}}{d\mathbb{P}^{u_i}} \right)^2 = \exp \left(2 \log \frac{d\mathbb{P}^{u_{i+1}}}{d\mathbb{P}^{u_i}} \right) \approx 1 + 2 \log \frac{d\mathbb{P}^{u_{i+1}}}{d\mathbb{P}^{u_i}}. \quad (89)$$

Now, taking expectations w.r.t. $\mathbb{P}^{u_i} \approx \mathbb{P}^{u_{i+1}}$, respectively, using computations similar to (87), and assuming $\bar{D}_{\text{KL}}(\mathbb{P}^{u_{i+1}}|\mathbb{P}^{u_i}) = \varepsilon$, as argued in App. J.1, yields

$$\text{Var}_{\mathbb{P}^{u_i}} \left(\frac{d\mathbb{P}^{u_{i+1}}}{d\mathbb{P}^{u_i}} \right) \approx 2\varepsilon. \quad (90)$$

1322 J.4 Lagrangian formulation

Using the Girsanov theorem (see App. A.3), we first note that we can write the Lagrangian as

$$\mathcal{L}_{\text{TR}}^{(i)}(u, \lambda) = \mathbb{E} \left[\int_0^T \left(\frac{1}{2} \|u(X_s^u, s)\|^2 + f(X_s^u, s) \right) ds + g(X_T^u) \right] + \lambda (D_{\text{KL}}(\mathbb{P}^u|\mathbb{P}^{u_i}) - \varepsilon) \quad (91)$$

$$= \mathbb{E} \left[\int_0^T \left(\frac{1}{2} \|u(X_s^u, s)\|^2 + \frac{\lambda}{2} \|u(X_s^u, s) - u_i(X_s^u, s)\|^2 + f(X_s^u, s) \right) ds + g(X_T^u) \right] - \lambda \varepsilon \quad (92)$$

$$= \mathbb{E} \left[\int_0^T \left(\frac{1+\lambda}{2} \|u(X_s^u, s) - \frac{\lambda}{1+\lambda} u_i(X_s^u, s)\|^2 + f_i(X_s^u, s) \right) ds + g(X_T^u) \right] - \lambda \varepsilon \quad (93)$$

$$= \mathcal{L}_{\text{TRC}}^{(i)}(u, \lambda) - \lambda \varepsilon, \quad (94)$$

where $\mathcal{L}_{\text{TRC}}^{(i)}(u, \lambda)$ is defined as in (11), $\lambda \in \mathbb{R}^+$ is the Lagrangian multiplier for the trust region constraint, and we abbreviate $f_i := \frac{\lambda}{2(1+\lambda)} \|u_i\|^2 + f$. For fixed λ , optimizing the Lagrangian $\mathcal{L}_{\text{TR}}^{(i)}(u, \lambda)$ with respect to u is again a SOC. As such, for a given u_i , λ , we can define the value function as

$$V_{i+1}^\lambda(x, t) := \inf_{u \in \mathcal{U}} \mathbb{E} \left[\int_t^T \left(\frac{1+\lambda}{2} \|u(X_s^u, s) - \frac{\lambda}{1+\lambda} u_i(X_s^u, s)\|^2 + f_i(X_s^u, s) \right) ds + g(X_T^u) \middle| X_t = x \right]. \quad (95)$$

The next proposition provides representations for the value function and the solution to the SOC problem.

Proposition J.2 (Optimality for trust region SOC problems). *For fixed λ , let us define by*

$$V_{i+1}^\lambda(x, t) := \inf_{u \in \mathcal{U}} \mathbb{E} \left[\int_0^T \left(\frac{1+\lambda}{2} \|u - \frac{\lambda}{1+\lambda} u_i\|^2 + \frac{\lambda}{2(1+\lambda)} \|u_i\|^2 + f \right) (X_s^u, s) ds + g(X_T^u) \middle| X_t = x \right]$$

the value function of the SOC problem $\inf_{u \in \mathcal{U}} \mathcal{L}_{\text{TRC}}^{(i)}(u, \lambda)$ corresponding to (11) and by u_{i+1}^λ its solution. Then it holds that

$$(i) \text{ (Estimator for value function) } V_{i+1}^\lambda(x, t) = -(1 + \lambda) \log \mathbb{E} \left[e^{-\frac{1}{1+\lambda} \mathcal{W}_i(X^{u_i}, t)} \middle| X_t^{u_i} = x \right],$$

$$\text{where } \mathcal{W}_i(X^{u_i}, t) = \int_t^T \frac{1}{2} \|u_i(X_s^{u_i}, s)\|^2 ds + \int_t^T u_i(X_s^{u_i}, s) \cdot dW_s + \mathcal{W}(X^{u_i}, t).$$

$$(ii) \text{ (Connection between solution and value function) It holds } u_{i+1}^\lambda = \frac{\lambda}{1+\lambda} u_i - \frac{1}{1+\lambda} \sigma^\top \nabla V_{i+1}^\lambda.$$

Moreover, for $u_0 = \mathbf{0}$ and the optimal Lagrange multiplier λ_i , let us define the value function

$$\tilde{V}_{i+1}(x, t) := \inf_{u \in \mathcal{U}} \mathbb{E} \left[\int_0^T \left(\frac{1}{2} \|u\|^2 + \beta_{i+1} f \right) (X_s^u, s) ds + \beta_{i+1} g(X_T^u) \middle| X_t = x \right]$$

of the SOC problem given by the optimal change of measure

$$\frac{d\mathbb{P}^{u_{i+1}}}{d\mathbb{P}}(X) \propto \left(\frac{d\mathbb{Q}}{d\mathbb{P}}(X) \right)^{\beta_{i+1}} \propto e^{-\beta_{i+1} \mathcal{W}(X, 0)} \quad (96)$$

as in Prop. 2.2 and (3). Then it holds that

$$(iii) \text{ (Estimator for value function) } \tilde{V}_{i+1}(x, t) = -\log \mathbb{E} \left[e^{-\beta_{i+1} \mathcal{W}(X_t, t)} \middle| X_t = x \right],$$

$$(iv) \text{ (Connection between solution and value function) } u_{i+1} = u_{i+1}^{\lambda_i} = -\sigma^\top \nabla \tilde{V}_{i+1}.$$

Proof. For notational convenience, we abbreviate $V = V_{i+1}^\lambda$ in this proof. From the verification theorem (see, e.g., [103, Theorem 3.5.2]), we obtain that the value function is the solution to the HJB

1342 equation

$$(\partial_t + L)V = - \inf_{\alpha \in \mathbb{R}^d} \left\{ f_i + \frac{1+\lambda}{2} \|\alpha - \frac{\lambda}{1+\lambda} u_i\|^2 + \sigma \alpha \cdot \nabla V \right\} \quad (97)$$

$$= -f_i - \inf_{\alpha \in \mathbb{R}^d} \left\{ \frac{1+\lambda}{2} \|\alpha - \frac{\lambda}{1+\lambda} u_i\|^2 + \sigma \alpha \cdot \nabla V \right\}, \quad V(\cdot, T) = g, \quad (98)$$

1343 where the infimum is pointwise for every $(x, t) \in \mathbb{R}^d \times [0, T]$ and the optimal α^* defines the solution
1344 u^* . Solving for α yields $\alpha^* = \frac{\lambda}{1+\lambda} u_i - \frac{1}{1+\lambda} \sigma^\top \nabla V$, which proves Item (ii).

1345 Plugging this result back into the HJB equation, we obtain

$$(\partial_t + L)V = -f - \frac{\lambda}{2(1+\lambda)} \|u_i\|^2 - \frac{1}{2(1+\lambda)} \|\sigma^\top \nabla V\|^2 - \sigma \left(\frac{\lambda}{1+\lambda} u_i - \frac{1}{1+\lambda} \sigma \nabla V \right) \cdot \nabla V \quad (99)$$

$$= -f - \frac{\lambda}{2(1+\lambda)} \|u_i\|^2 + \frac{1}{2(1+\lambda)} \|\sigma^\top \nabla V\|^2 - \sigma \frac{\lambda}{1+\lambda} u_i \cdot \nabla V \quad (100)$$

$$= -f - \frac{\lambda}{2(1+\lambda)} \|u_i\|^2 + \frac{1}{2(1+\lambda)} \|\sigma^\top \nabla V\|^2 - \sigma u_i \cdot \nabla V + \sigma \frac{1}{1+\lambda} u_i \cdot \nabla V \quad (101)$$

1346 Now, we define the infinitesimal generator of the SDE

$$dX_s^{u_i} = (b(X_s^{u_i}, s) + \sigma u_i(X_s^{u_i}, s)) ds + \sigma dW_s \quad (102)$$

1347 as

$$\bar{L} := \frac{1}{2} \sum_{i,j} (\sigma \sigma^\top)_{ij} \partial_{x_i} \partial_{x_j} + \sum_i (b_i + \sigma(u_i)_i) \partial_{x_i} = L + \sum_i \sigma(u_i)_i \partial_{x_i}. \quad (103)$$

1348 Using (103), we can rewrite (101) as

$$(\partial_t + \bar{L})V = -f - \frac{\lambda}{2(1+\lambda)} \|u_i\|^2 + \frac{1}{2(1+\lambda)} \|\sigma^\top \nabla V\|^2 + \sigma \frac{1}{1+\lambda} u_i \cdot \nabla V \quad (104)$$

$$= -f - \frac{1}{2} \|u_i\|^2 + \frac{1}{2(1+\lambda)} \|u_i + \sigma^\top \nabla V\|^2 \quad (105)$$

1349 By Itô's formula (see App. A.3), we have

$$dV(X_s^{u_i}, s) = (\partial_s + \bar{L})V(X_s^{u_i}, s) ds + \sigma^\top \nabla V(X_s^{u_i}, s) \cdot dW_s. \quad (106)$$

1350 Plugging Eq. (105) in Eq. (106) and defining $Y_s := V(X_s^{u_i}, s)$ and $Z_s := (-u_i - \sigma \nabla V)(X_s^{u_i}, s)$,
1351 we obtain the pair of forward-backward SDEs (FBSDEs)

$$dX_s^{u_i} = (b(X_s^{u_i}, s) + \sigma u_i(X_s^{u_i}, s)) ds + \sigma dW_s, \quad X_0^{u_i} \sim p_0, \quad (107)$$

$$dY_s = \left(-f(X_s^{u_i}, s) - \frac{1}{2} \|u_i(X_s^{u_i}, s)\|^2 + \frac{1}{2(1+\lambda)} \|Z_s\|^2 \right) ds - (u_i(X_s^{u_i}, s) + Z_s) \cdot dW_s, \quad (108)$$

1352 with $Y_T = g(X_T^{u_i})$. This shows that

$$g(X_T^{u_i}) = Y_t - \int_t^T \left(f_i(X_s^{u_i}, s) + \frac{1}{2} \|u_i(X_s^{u_i}, s)\|^2 - \frac{1}{2(1+\lambda)} \|Z_s\|^2 \right) ds - \int_t^T (u_i(X_s^{u_i}, s) + Z_s) \cdot dW_s,$$

1353 which can be rewritten as

$$\mathcal{W}_i(X^{u_i}, t) = Y_t + \int_t^T \frac{1}{2(1+\lambda)} \|Z_s\|^2 ds - \int_t^T Z_s \cdot dW_s. \quad (109)$$

1354 Using the definition of Y_t , we can now write

$$\begin{aligned} \mathbb{E} \left[e^{-\frac{1}{1+\lambda} \mathcal{W}_i(X^{u_i}, t)} \middle| X_t^{u_i} = x \right] &= e^{-\frac{1}{1+\lambda} V(X_t^{u_i}, t)} \mathbb{E} \left[e^{\frac{1}{1+\lambda} \int_t^T Z_s \cdot dW_s - \frac{1}{(1+\lambda)^2} \int_t^T \frac{1}{2} \|Z_s\|^2 ds} \middle| X_t^{u_i} = x \right] \\ &= e^{-\frac{1}{1+\lambda} V(X_t^{u_i}, t)}, \end{aligned}$$

1355 where leveraged Novikov's theorem to show that the Doléans-Dade exponential is a martingale with
1356 vanishing expectation. This concludes the proof of Item (i). The proof of Items (iii) and (iv) follows
1357 directly from Thm. D.1. \square

1358 J.5 Log-variance with buffer and trust regions

1359 Here, we provide further details on the trust-region log-variance loss introduced in Sec. 2.2 and given
1360 by

$$\mathcal{L}_{LV}(u) = \text{Var} \left[\log \left(\frac{d\mathbb{P}^{u_{i+1}}}{d\mathbb{P}^u}(X^{u_i}) \right) \right] = \text{Var} \left[\log \left(\frac{d\mathbb{P}^{u_{i+1}}}{d\mathbb{P}^{u_i}}(X^{u_i}) \frac{d\mathbb{P}^{u_i}}{d\mathbb{P}^u}(X^{u_i}) \right) \right]. \quad (110)$$

1361 The Girsanov's theorem (see App. A.3) shows that

$$\log \left(\frac{d\mathbb{P}^{u_i}}{d\mathbb{P}^u}(X^{u_i}) \right) = \int_0^T \frac{1}{2} \|u_i(X_s^{u_i}, s) - u(X_s^{u_i}, s)\|^2 ds + \int_0^T (u_i - u)(X_s^{u_i}, s) \cdot dW_s. \quad (111)$$

1362 Combining this result for $u = 0$ with Prop. 2.2, we obtain

$$\frac{d\mathbb{P}^{u_{i+1}}}{d\mathbb{P}^{u_i}}(X^{u_i}) \propto \frac{1}{1+\lambda_i} \log \left(\frac{d\mathbb{Q}}{d\mathbb{P}} \frac{d\mathbb{P}}{d\mathbb{P}^{u_i}}(X^{u_i}) \right) \quad (112)$$

$$= e^{-\frac{1}{1+\lambda_i} \left(\int_0^T \frac{1}{2} \|u_i(X_s^{u_i}, s)\|^2 ds + \int_0^T u_i(X_s^{u_i}, s) \cdot dW_s + \mathcal{W}(X^{u_i}, 0) \right)}. \quad (113)$$

1363 Noting that the variance is shift-invariant, (111) and (112) imply that

$$\mathcal{L}_{LV}(u) = \text{Var} \left[-\frac{1}{1+\lambda_i} \left(\int_0^T \frac{1}{2} \|u_i(X_s^{u_i}, s)\|^2 ds + \int_0^T u_i(X_s^{u_i}, s) \cdot dW_s + \mathcal{W}(X^{u_i}, 0) \right) \right] \quad (114)$$

$$+ \int_0^T \frac{1}{2} \|u_i(X_s^{u_i}, s) - u(X_s^{u_i}, s)\|^2 ds + \int_0^T (u_i - u)(X_s^{u_i}, s) \cdot dW_s \Big], \quad (115)$$

1364 which can be implemented by discretizing the integrals; see App. J.2.

1365 Please note that the loss reduces to

$$\mathcal{L}_{LV}(u) = \text{Var} \left[\log \left(\frac{d\mathbb{Q}}{d\mathbb{P}^{u_i}}(X^{u_i}) \frac{d\mathbb{P}^{u_i}}{d\mathbb{P}^u}(X^{u_i}) \right) \right] = \text{Var} \left[\log \left(\frac{d\mathbb{Q}}{d\mathbb{P}^u}(X^{u_i}) \right) \right] \quad (116)$$

1366 for $\lambda_i = 0$, which is how the loss is mostly used in the literature, where the variance is computed
1367 using the most recent control, see e.g. [106].

1368 J.6 Trust-region stochastic optimal control matching via adjoint method

1369 Here, we provide further details on the trust-region version of the stochastic optimal control matching
1370 (SOCM) loss introduced in [84]. We start from the cross-entropy loss, i.e., the forward KL divergence
1371 between u_{i+1} and u , that is,

$$\mathcal{L}_{CE}(u) = D_{KL}(\mathbb{P}^{u_{i+1}} | \mathbb{P}^u) = \mathbb{E} \left[\log \frac{d\mathbb{P}^{u_{i+1}}}{d\mathbb{P}^u}(X^{u_{i+1}}) \right]. \quad (117)$$

1372 Using Girsanov's theorem (see App. A.3), the cross-entropy loss can be written as

$$\mathcal{L}_{CE}(u) = \mathbb{E} \left[\frac{1}{2} \int_0^T \|u_{i+1}(X_s^{u_{i+1}}, s) - u(X_s^{u_{i+1}}, s)\|^2 ds \right] \quad (118)$$

$$= \mathbb{E} \left[\frac{1}{2} \int_0^T \|u_{i+1}(X_s^{u_i}, s) - u(X_s^{u_i}, s)\|^2 ds \frac{d\mathbb{P}^{u_{i+1}}}{d\mathbb{P}^{u_i}} \right]. \quad (119)$$

1373 Using the expression for the optimal control $u_{i+1} = \frac{\lambda_i}{1+\lambda_i} u_i - \frac{1}{1+\lambda_i} \nabla V_{i+1}$, see Prop. 2.5, yields

$$\mathcal{L}_{CE}(u) = \mathbb{E} \left[\frac{1}{2} \int_0^T \left\| \frac{\lambda_i}{1+\lambda_i} u_i(X_s^{u_i}, s) - \frac{1}{1+\lambda_i} \sigma^\top \nabla V_{i+1}(X_s^{u_i}, s) - u(X_s^{u_i}, s) \right\|^2 ds \frac{d\mathbb{P}^{u_{i+1}}}{d\mathbb{P}^{u_i}} \right] \quad (120)$$

1374 with

$$\nabla_x V_{i+1}(x, t) = -(1+\lambda_i) \frac{\nabla_x \mathbb{E} \left[e^{-\frac{1}{1+\lambda_i} \mathcal{W}_i(X^{u_i}, t)} \middle| X_t^{u_i} = x \right]}{\mathbb{E} \left[e^{-\frac{1}{1+\lambda_i} \mathcal{W}_i(X^{u_i}, t)} \middle| X_t^{u_i} = x \right]}. \quad (121)$$

1375 We use the adjoint method [83, see Lemma 5] to evaluate the conditional expectation (121)⁸, giving

$$\nabla_x \mathbb{E} \left[e^{-\frac{1}{1+\lambda_i} \mathcal{W}_i(X^{u_i}, t)} \middle| X_t^{u_i} = x \right] = \mathbb{E} \left[\tilde{a}(t, u_i, X^{u_i}) e^{-\frac{1}{1+\lambda_i} \mathcal{W}_i(X^{u_i}, t)} \middle| X_t^{u_i} = x \right] \quad (122)$$

1376 where the adjoint state $\tilde{a}(t, u_i, X^{u_i})$ satisfies the ordinary differential equation (ODE)

$$\frac{d}{ds} \tilde{a}(s, u_i, X_s^{u_i}) = - \left[(\nabla(b(X_s^{u_i}, s) + \sigma u_i(X_s^{u_i}, s)))^\top \tilde{a}(u_i, X_s^{u_i}, s) \right. \quad (123)$$

$$\left. + \frac{1}{1+\lambda_i} \nabla(f(X_s^{u_i}, s) + \frac{1}{2} \|u_i(X_s^{u_i}, s)\|^2) \right] \quad (124)$$

1377 with $\tilde{a}(T, u_i, X_T^{u_i}) = \frac{1}{1+\lambda_i} \nabla g(X_T)$. Using the argument from [85, Theorem 1], replacing the
1378 path-wise reparameterization trick with the adjoint method, we arrive at the trust-region version of
1379 the stochastic optimal control loss given by

$$\mathcal{L}_{SOCM}(u) = \mathbb{E} \left[\frac{1}{2} \int_0^T \left\| \frac{\lambda_i}{1+\lambda_i} u_i(X_s^{u_i}, s) - \sigma^\top \tilde{a}(u_i, X_s^{u_i}, s) - u(X_s^{u_i}, s) \right\|^2 ds \frac{d\mathbb{P}^{u_{i+1}}}{d\mathbb{P}^{u_i}} \right] + K \quad (125)$$

⁸Note that there exist other methods for computing derivatives of functionals of stochastic processes. We refer the interested reader to [85].

for some K independent of u . However, the adjoint state contains the Jacobian ∇u_i and the derivative $\nabla \|u_i\|$, which can be expensive in practice. In what follows, we rewrite the objective such that we can get rid of these terms.

1383 J.7 Trust-region stochastic optimal control matching via lean adjoint method

1384 Starting again from the cross-entropy loss, we now employ the alternative expression for the optimal
1385 control as stated in Item (ii). This yields the objective

$$\mathcal{L}_{\text{CE}}(u) = \mathbb{E} \left[\frac{1}{2} \int_0^T \|\sigma^\top \nabla \tilde{V}_{i+1}(X_s, s) - u(X_s, s)\|^2 ds \frac{d\mathbb{P}^{u_{i+1}}}{d\mathbb{P}} \right]. \quad (126)$$

1386 where the gradient of the smoothed value function is given by

$$\nabla_x \tilde{V}_{i+1}(x, t) = - \frac{\nabla_x \mathbb{E} \left[e^{-\beta_i \mathcal{W}(X_t, 0)} | X_t = x \right]}{\mathbb{E} \left[e^{-\beta_i \mathcal{W}(X_t, 0)} | X_t = x \right]} \quad (127)$$

1387 We evaluate the conditional expectation using the adjoint method:

$$\nabla_x \mathbb{E} \left[e^{-\beta_i \mathcal{W}(X_t, 0)} | X_t = x \right] = \mathbb{E} \left[a_{i+1}(X_s, s) e^{-\beta_i \mathcal{W}(X_t, 0)} | X_t = x \right]. \quad (128)$$

1388 where $a_{i+1}(X_s, s)$ denotes the lean adjoint state [83], which satisfies the backward differential
1389 equation:

$$\frac{d}{ds} a_{i+1}(X_s, s) = - \left[(\nabla b(X_s, s))^\top a_{i+1}(X_s, s) + \beta_i \nabla f(X_s, s) \right] \quad (129)$$

1390 with terminal condition $a_{i+1}(X_T, T) = \beta_i \nabla g(X_T)$. Following the derivations in [85], we arrive at
1391 the objective:

$$\mathcal{L}_{\text{SOCM}}(u) = \mathbb{E} \left[\frac{1}{2} \int_0^T \|\sigma^\top a_{i+1}(X_s, s) - u(X_s, s)\|^2 ds \frac{d\mathbb{P}^{u_{i+1}}}{d\mathbb{P}}(X) \right]. \quad (130)$$

1392 Finally, performing a change of measure to the previous control u_i gives the expression:

$$\mathcal{L}_{\text{SOCM}}(u) = \mathbb{E} \left[\frac{1}{2} \int_0^T \|\sigma^\top a_{i+1}(X_s^{u_i}, s) - u(X_s^{u_i}, s)\|^2 ds \frac{d\mathbb{P}^{u_{i+1}}}{d\mathbb{P}^{u_i}} \right]. \quad (131)$$

1393 We remark that the adjoint ODE in (129) can be solved as

$$a_{i+1}(X_s, s) = \beta_i \exp \left(\int_s^T \nabla b(X_t, t)^\top dt \right) \nabla g(X_T) \quad (132)$$

1394 if $f = 0$ and $\nabla b(X_t, t) \nabla b(X_s, s) = \nabla b(X_s, s) \nabla b(X_t, t)$ for all $s, t \in [0, T]$ (i.e., the matrices at
1395 different times commute). This allows us to solve the adjoint ODE exactly for our applications to
1396 sampling from unnormalized densities; see App. E.

1397 K Trust regions for probability measures

1398 Our goal is to sample from a probability density of the form

$$p_{\text{target}}(x) = \frac{\rho_{\text{target}}(x)}{\mathcal{Z}}, \quad \text{with } \mathcal{Z} = \int \rho_{\text{target}}(x) dx, \quad (133)$$

1399 where we can evaluate ρ_{target} but typically do not have access to samples from p_{target} . To tackle this
1400 problem, one can again formulate this problem as a variational problem by minimizing a divergence
1401 between some q and the target density p_{target} . However, one can again incorporate an additional trust
1402 region constraint, that is, an upper bound on the change of the variational distribution q within a
1403 single update step. Formally, we are trying to solve the following problem

$$q_{i+1} = \arg \min_q D_{\text{KL}}(q \| p_{\text{target}}) \quad \text{s.t.} \quad D_{\text{KL}}(q \| q_i) \leq \varepsilon, \quad \int dq = 1 \quad (134)$$

1404 where q_i is the variational distribution from the previous iteration. We again tackle the constrained
1405 optimization problem in (134) using Lagrangian multipliers. The Lagrangian is given by

$$\mathcal{L}_{\text{TR}}^{(i)}(q, \lambda, \omega) = D_{\text{KL}}(q \| p_{\text{target}}) + \lambda (D_{\text{KL}}(q \| q_i) - \varepsilon) + \omega \left(\int dq - 1 \right) \quad (135)$$

1406 with Lagrangian multipliers λ, ω . Taking the functional derivative $\delta \mathcal{L}_{\text{TR}}^{(i)}(q, \lambda, \omega) / \delta q$ and setting it to
1407 zero admits a closed-form solution for the optimal density q_{i+1} as the geometric average between the

old distribution and the (unnormalized) optimal distribution, that is,⁹

$$q_{i+1}(\lambda) = \arg \min_q \mathcal{L}_{\text{TR}}^{(i)}(q, \lambda) = \frac{q_i^{\frac{\lambda}{1+\lambda}} \rho_{\text{target}}^{\frac{1}{1+\lambda}}}{\mathcal{Z}_i(\lambda)}, \quad \text{with} \quad \mathcal{Z}_i(\lambda) = \int q_i^{\frac{\lambda}{1+\lambda}}(x) \rho_{\text{target}}^{\frac{1}{1+\lambda}}(x) dx. \quad (136)$$

Plugging the optimal distribution back into the Lagrangian yields the dual function

$$\mathcal{L}_{\text{Dual}}^{(i)}(\lambda) = \mathcal{L}_{\text{TR}}^{(i)}(q_{i+1}(\lambda), \lambda) = -(1 + \lambda) \log \mathcal{Z}_i(\lambda) - \lambda \varepsilon. \quad (137)$$

Note that we can use any non-linear optimizer for solving for the optimal Lagrangian multiplier by maximizing the dual function, i.e.,

$$\lambda_i = \arg \max_{\lambda \in \mathbb{R}^+} \mathcal{L}_{\text{Dual}}^{(i)}(\lambda). \quad (138)$$

⁹Note the dependence of $\mathcal{L}_{\text{TR}}^{(i)}$ on ω vanishes as q_{i+1} satisfies the normalization constraint.

Appendix References

- [64] T. Akhound-Sadegh, J. Rector-Brooks, A. J. Bose, S. Mittal, P. Lemos, C.-H. Liu, M. Sendera, S. Ravanbakhsh, G. Gidel, Y. Bengio, et al. Iterated denoising energy matching for sampling from boltzmann densities. *arXiv preprint arXiv:2402.06121*, 2024.
- [65] M. S. Albergo and E. Vanden-Eijnden. Nets: A non-equilibrium transport sampler. *arXiv preprint arXiv:2410.02711*, 2024.
- [66] C. Beck, S. Becker, P. Grohs, N. Jaafari, and A. Jentzen. Solving the kolmogorov pde by means of deep learning. *Journal of Scientific Computing*, 88:1–28, 2021.
- [67] J. Berner, M. Dablander, and P. Grohs. Numerically solving parametric families of high-dimensional kolmogorov partial differential equations via deep learning. *Advances in Neural Information Processing Systems*, 33:16615–16627, 2020.
- [68] J. Berner, L. Richter, and K. Ullrich. An optimal control perspective on diffusion-based generative modeling. *arXiv preprint arXiv:2211.01364*, 2022.
- [69] K. Black, M. Janner, Y. Du, I. Kostrikov, and S. Levine. Training diffusion models with reinforcement learning. In *The Twelfth International Conference on Learning Representations*, 2024.
- [70] D. Blessing, X. Jia, J. Esslinger, F. Vargas, and G. Neumann. Beyond elbos: A large-scale evaluation of variational methods for sampling. *arXiv preprint arXiv:2406.07423*, 2024.
- [71] P. G. Bolhuis, D. Chandler, C. Dellago, and P. L. Geissler. Transition path sampling: Throwing ropes over rough mountain passes, in the dark. *Annual review of physical chemistry*, 53(1):291–318, 2002.
- [72] J. Bradbury, R. Frostig, P. Hawkins, M. J. Johnson, C. Leary, D. Maclaurin, G. Necula, A. Paszke, J. VanderPlas, S. Wanderman-Milne, et al. Jax: Autograd and xla. *Astrophysics Source Code Library*, pages ascl–2111, 2021.
- [73] R. Chetrite and H. Touchette. Variational and optimal control representations of conditioned and driven processes. *Journal of Statistical Mechanics: Theory and Experiment*, 2015(12):P12001, 2015.
- [74] K. Clark, P. Vicol, K. Swersky, and D. J. Fleet. Directly fine-tuning diffusion models on differentiable rewards. In *The Twelfth International Conference on Learning Representations*, 2024.
- [75] M. Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 26, 2013.
- [76] M. Cuturi, L. Meng-Papaxanthos, Y. Tian, C. Bunne, G. Davis, and O. Teboul. Optimal transport tools (OTT): A jax toolbox for all things Wasserstein. *arXiv preprint arXiv:2201.12324*, 2022.
- [77] P. Dai Pra. A stochastic control approach to reciprocal diffusion processes. *Applied mathematics and Optimization*, 23(1):313–329, 1991.
- [78] A. Das, D. C. Rose, J. P. Garrahan, and D. T. Limmer. Reinforcement learning of rare diffusive dynamics. *The Journal of Chemical Physics*, 155(13), 2021.
- [79] C. Dellago, P. G. Bolhuis, and D. Chandler. Efficient transition path sampling: Application to lennard-jones cluster rearrangements. *The Journal of chemical physics*, 108(22):9236–9245, 1998.
- [80] Z. Ding, Y. Jiao, X. Lu, Z. Yang, and C. Yuan. Sampling via Föllmer flow. *arXiv preprint arXiv:2311.03660*, 2023.
- [81] C. Domingo-Enrich. A taxonomy of loss functions for stochastic optimal control. *arXiv preprint arXiv:2410.00345*, 2024.

- [82] C. Domingo-Enrich, M. Drozdal, B. Karrer, and R. T. Chen. Adjoint matching: Fine-tuning flow and diffusion generative models with memoryless stochastic optimal control. *arXiv preprint arXiv:2409.08861*, 2024.
- [83] C. Domingo-Enrich, M. Drozdal, B. Karrer, and R. T. Q. Chen. Adjoint matching: Fine-tuning flow and diffusion generative models with memoryless stochastic optimal control. In *The Thirteenth International Conference on Learning Representations*, 2025.
- [84] C. Domingo-Enrich, J. Han, B. Amos, J. Bruna, and R. T. Chen. Stochastic optimal control matching. *arXiv preprint arXiv:2312.02027*, 2023.
- [85] C. Domingo-Enrich, J. Han, B. Amos, J. Bruna, and R. T. Q. Chen. Stochastic optimal control matching. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.
- [86] Y. Du, M. Plainier, R. Brekelmans, C. Duan, F. Noe, C. P. Gomes, A. Aspuru-Guzik, and K. Neklyudov. Doob’s lagrangian: A sample-efficient variational approach to transition path sampling. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.
- [87] Y. Fan, O. Watkins, Y. Du, H. Liu, M. Ryu, C. Boutilier, P. Abbeel, M. Ghavamzadeh, K. Lee, and K. Lee. Dpok: Reinforcement learning for fine-tuning text-to-image diffusion models. *arXiv preprint arXiv:2305.16381*, 2023.
- [88] W. H. Fleming and H. M. Soner. *Controlled Markov processes and viscosity solutions*, volume 25. Springer Science & Business Media, 2006.
- [89] L. Grenioux, M. Noble, and M. Gabri  . Improving the evaluation of samplers on multi-modal targets. *arXiv preprint arXiv:2504.08916*, 2025.
- [90] J. Han, A. Jentzen, and W. E. Solving high-dimensional partial differential equations using deep learning. *Proceedings of the National Academy of Sciences*, 115(34):8505–8510, 2018.
- [91] C. Hartmann and L. Richter. Nonasymptotic bounds for suboptimal importance sampling. *SIAM/ASA Journal on Uncertainty Quantification*, 12(2):309–346, 2024.
- [92] D. Hendrycks and K. Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016.
- [93] L. Holdijk, Y. Du, F. Hooft, P. Jaini, B. Ensing, and M. Welling. Stochastic optimal control for collective variable free sampling of molecular transition paths. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.
- [94] J. Huang, Y. Jiao, L. Kang, X. Liao, J. Liu, and Y. Liu. Schr  dinger-f  llmer sampler: sampling without ergodicity. *arXiv preprint arXiv:2106.10880*, 2021.
- [95] X. Huang, H. Dong, Y. Hao, Y. Ma, and T. Zhang. Monte carlo sampling without isoperimetry: A reverse diffusion approach. *arXiv preprint arXiv:2307.02037*, 2023.
- [96] D. P. Kingma. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- [97] J. Liu, G. Liu, J. Liang, Y. Li, J. Liu, X. Wang, P. Wan, D. Zhang, and W. Ouyang. Flow-GRPO: Training flow matching models via online RL, 2025.
- [98] Z. Liu, T. Z. Xiao, W. Liu, Y. Bengio, and D. Zhang. Efficient diversity-preserving diffusion alignment via gradient-informed GFlowNets, 2024.
- [99] L. I. Midgley, V. Stimper, G. N. Simm, B. Sch  lkopf, and J. M. Hern  ndez-Lobato. Flow annealed importance sampling bootstrap. *arXiv preprint arXiv:2208.01893*, 2022.
- [100] M. Noble, L. Grenioux, M. Gabri  , and A. O. Durmus. Learned reference-based diffusion sampling for multi-modal distributions. *arXiv preprint arXiv:2410.19449*, 2024.

- [101] N. Nüsken and L. Richter. Solving high-dimensional Hamilton–Jacobi–Bellman pdes using neural networks: perspectives from the theory of controlled diffusions and measures on path space. *Partial differential equations and applications*, 2:1–48, 2021.
- [102] M. Pavon. Stochastic control and nonequilibrium thermodynamical systems. *Applied Mathematics and Optimization*, 19(1):187–202, 1989.
- [103] H. Pham. *Continuous-time Stochastic Control and Optimization with Financial Applications*. Stochastic Modelling and Applied Probability. Springer Berlin Heidelberg, 2009.
- [104] L. Richter. *Solving high-dimensional PDEs, approximation of path space measures and importance sampling of diffusions*. PhD thesis, BTU Cottbus-Senftenberg, 2021.
- [105] L. Richter and J. Berner. Robust SDE-based variational formulations for solving linear PDEs via deep learning. In *International Conference on Machine Learning*, pages 18649–18666. PMLR, 2022.
- [106] L. Richter, J. Berner, and G.-H. Liu. Improved sampling via learned diffusions. *arXiv preprint arXiv:2307.01198*, 2023.
- [107] L. Richter, L. Sallandt, and N. Nüsken. Solving high-dimensional parabolic pdes using the tensor train format. In *International Conference on Machine Learning*, pages 8998–9009. PMLR, 2021.
- [108] L. Richter, L. Sallandt, and N. Nüsken. From continuous-time formulations to discretization schemes: tensor trains and robust regression for bsdes and parabolic pdes. *arXiv preprint arXiv:2307.15496*, 2023.
- [109] D. C. Rose, J. F. Mair, and J. P. Garrahan. A reinforcement learning approach to rare trajectory sampling. *New Journal of Physics*, 23(1):013013, 2021.
- [110] M. Sabate Vidales, D. Šiška, and L. Szpruch. Unbiased deep solvers for linear parametric PDEs. *Applied Mathematical Finance*, 28(4):299–329, 2021.
- [111] M. Sendera, M. Kim, S. Mittal, P. Lemos, L. Scimeca, J. Rector-Brooks, A. Adam, Y. Bengio, and N. Malkin. Improved off-policy training of diffusion samplers. *Advances in Neural Information Processing Systems*, 37:81016–81045, 2024.
- [112] K. Seong, S. Park, S. Kim, W. Y. Kim, and S. Ahn. Transition path sampling with improved off-policy training of diffusion path samplers. *arXiv preprint arXiv:2405.19961*, 2024.
- [113] A. N. Singh, A. Das, and D. T. Limmer. Variational path sampling of rare dynamical events. *Annual Review of Physical Chemistry*, 76, 2025.
- [114] A. N. Singh and D. T. Limmer. Variational deep learning of equilibrium transition path ensembles. *The Journal of Chemical Physics*, 159(2), 2023.
- [115] J. Sun, J. Berner, L. Richter, M. Zeinhofer, J. Müller, K. Azizzadenesheli, and A. Anandkumar. Dynamical measure transport and neural PDE solvers for sampling. *arXiv preprint arXiv:2407.07873*, 2024.
- [116] H. Y. Tan, S. Osher, and W. Li. Noise-free sampling algorithms via regularized wasserstein proximals. *arXiv preprint arXiv:2308.14945*, 2023.
- [117] M. Tancik, P. Srinivasan, B. Mildenhall, S. Fridovich-Keil, N. Raghavan, U. Singhal, R. Ramamoorthi, J. Barron, and R. Ng. Fourier features let networks learn high frequency functions in low dimensional domains. *Advances in neural information processing systems*, 33:7537–7547, 2020.
- [118] M. Uehara, Y. Zhao, K. Black, E. Hajiramezanali, G. Scalia, N. L. Diamant, A. M. Tseng, T. Biancalani, and S. Levine. Fine-tuning of continuous-time diffusion models as entropy-regularized control. *arXiv preprint arXiv:2402.15194*, 2024.
- [119] R. Van Handel. Stochastic calculus, filtering, and stochastic control. *Course notes.*, URL <http://www.princeton.edu/rvan/acm217/ACM217.pdf>, 14, 2007.

- 1549 [120] E. Vanden-Eijnden et al. Transition-path theory and path-finding algorithms for the study of
1550 rare events. *Annual review of physical chemistry*, 61:391–420, 2010.
- 1551 [121] F. Vargas, W. Grathwohl, and A. Doucet. Denoising diffusion samplers. *arXiv preprint*
1552 *arXiv:2302.13834*, 2023.
- 1553 [122] F. Vargas, A. Ovsianas, D. Fernandes, M. Girolami, N. D. Lawrence, and N. Nüsken. Bayesian
1554 learning via neural schrödinger–föllmer flows. *Statistics and Computing*, 33(1):3, 2023.
- 1555 [123] F. Vargas, S. Padhy, D. Blessing, and N. Nüsken. Transport meets variational inference:
1556 Controlled Monte Carlo diffusions. In *The Twelfth International Conference on Learning*
1557 *Representations*, 2024.
- 1558 [124] H. Wu, J. Köhler, and F. Noé. Stochastic normalizing flows. *Advances in neural information*
1559 *processing systems*, 33:5933–5944, 2020.
- 1560 [125] J. Xu, X. Liu, Y. Wu, Y. Tong, Q. Li, M. Ding, J. Tang, and Y. Dong. Imagereward: Learning
1561 and evaluating human preferences for text-to-image generation. In *Thirty-seventh Conference*
1562 *on Neural Information Processing Systems*, 2023.
- 1563 [126] J. Yan, H. Touchette, and G. M. Rotskoff. Learning nonequilibrium control forces to character-
1564 ize dynamical phase transitions. *Physical Review E*, 105(2):024115, 2022.
- 1565 [127] D. Zhang, R. T. Chen, C.-H. Liu, A. Courville, and Y. Bengio. Diffusion generative flow
1566 samplers: Improving learning signals through partial trajectory optimization. In *The Twelfth*
1567 *International Conference on Learning Representations*, 2024.
- 1568 [128] D. Zhang, Y. Zhang, J. Gu, R. Zhang, J. Susskind, N. Jaitly, and S. Zhai. Improving GFlowNets
1569 for text-to-image diffusion alignment. *arXiv preprint arXiv:2406.00633*, 2024.
- 1570 [129] Q. Zhang and Y. Chen. Path Integral Sampler: a stochastic control approach for sampling. In
1571 *International Conference on Learning Representations*, 2022.