ANCHOR: SAFE DIFFUSION PLANNING VIA ANALYTIC APPROXIMATION OF DOOB'S h-FUNCTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Diffusion models provide expressive, multimodal priors for offline trajectory planning, but enforcing hard safety constraints without distorting the learned trajectory distribution remains challenging. We formulate safe planning as conditional generation from the distribution of trajectories given collision avoidance, whose exact reverse process is characterized by a Doob h-transform. Since the corresponding safety posterior probability is generally intractable, we introduce ANCHOR, a training-free analytic approximation to the Doob guidance. Our method approximates the posterior using Tweedie moments and exploits signed-distance geometry for efficient computation. Under explicit regularity assumptions on the data distribution and obstacle geometry, we prove that the resulting guided diffusion is almost surely safe. On synthetic multimodal distributions, our method recovers the safety-conditioned distribution substantially more accurately than safe generative-planning baselines while avoiding boundary concentration. On multimodal navigation, locomotion, and manipulation benchmarks it achieves results that outperform or remain competitive with state-of-the-art safe diffusion and flow-matching planners.

1 INTRODUCTION

Diffusion models (Ho et al., 2020; Song et al., 2020b) have recently emerged as expressive generative planners for offline decision making (Lu et al., 2025), owing to their ability to represent complex, long-horizon, and multimodal behavior distributions rather than a single deterministic solution. Diffuser (Janner et al., 2022) showed that planning can be formulated as iterative denoising of trajectories drawn from a learned data distribution, enabling flexible conditioning on task objectives. An attractive consequence of this generative perspective is that a planner can, in principle, retain multiple qualitatively different ways of solving the same task.

Incorporating hard safety constraints into such generative planners remains challenging, yet it is a prerequisite for deploying them on autonomous systems, where planned trajectories must satisfy user-specified constraints. Recent works augment diffusion and flow-matching planners with control barrier functions (CBFs) or Lyapunov certificates (Xiao et al., 2023; Mizuta & Leung, 2024; Yang et al., 2026; Huang et al., 2026) or constrain the generative process through projection or constrained sampling (Christopher et al., 2024; Zhang et al., 2025; Römer et al., 2025). These methods enforce feasibility effectively and often come with safety guarantees. However, feasibility alone does not guarantee consistency with the learned trajectory distribution (Jung et al., 2025). When a test-time obstacle blocks a high-probability trajectory mode, locally correcting that trajectory may produce a collision-free solution in a low-probability region, even when alternative safe modes are available. This distinction matters for control learned from offline data: in offline reinforcement learning (RL), poorly supported actions can lead to unreliable value estimates (Kumar et al., 2019; 2020; Kostrikov et al., 2021; Wang et al., 2022), while in behavioral cloning, distribution shift can cause compounding errors (Ross et al., 2011; Laskey et al., 2017). Conditioning on the safe set provides an explicit distributional target: it removes unsafe probability mass and preserves the relative probabilities of safe trajectories. This motivates a complementary question: *can we approximate this safety-conditioned distribution at test time, without retraining, while retaining rigorous safety guarantees?*

We address this question by formulating safe trajectory generation as sampling from $p_0(x_0 \mid x_0 \in S)$ where S denotes the set of collision-free trajectories. The exact conditional reverse diffusion process

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

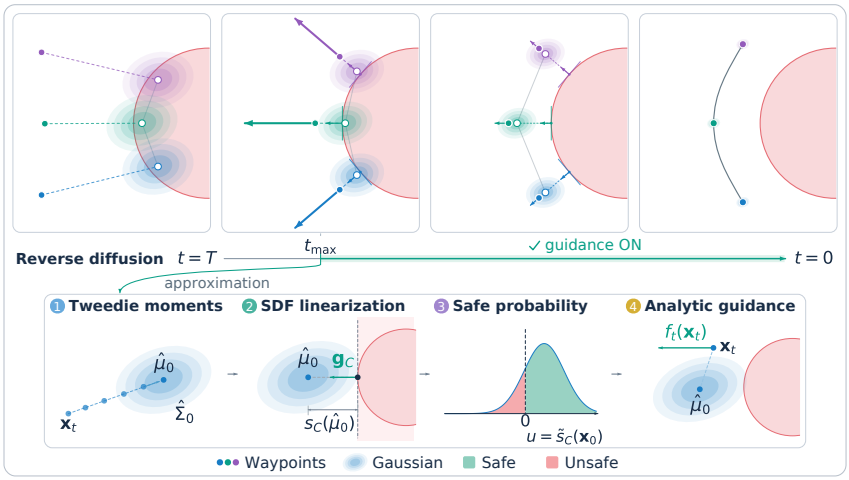


Figure 1: Overview of ANCHOR. Top: Reverse diffusion of a trajectory around an obstacle, with safety guidance activated at $t_{\max}$. Bottom: Gaussian posterior approximation and SDF linearization yield a one-dimensional Gaussian-tail estimate of the safety probability, whose log-gradient guides denoising.

is characterized by the Doob guidance $\nabla_{x_t} \log P(X_0 \in S \mid X_t = x_t)$, but the corresponding conditional probability is generally intractable for a learned data distribution. Our key observation is that it can be approximated analytically. We approximate the posterior $p(x_0 \mid x_t)$ by a Gaussian using Tweedie information that the diffusion model readily provides and encode obstacle geometry through a signed-distance function. Linearizing the signed-distance function around the posterior mean reduces the resulting Gaussian set probability to a one-dimensional Gaussian tail, yielding an efficient closed-form guidance.

The resulting method, ANCHOR, **A**nalytical **N**ormal-approximation of **C**onditional **h**-functions for **O**bstacle-aware **R**eplanning, requires no obstacle-aware retraining and can be applied to a pretrained diffusion planner at inference time. Despite the Gaussian and geometric approximations, we show under explicit regularity assumptions that the guided reverse process is almost surely safe.

Our contributions are the following: i) we formulate obstacle avoidance for diffusion planners as test-time conditional generation and derive an analytic approximation to the corresponding Doob guidance by combining Tweedie posterior moments with signed-distance geometry; ii) we show that the resulting guided diffusion is almost surely safe under explicit assumptions on the data distribution and obstacle geometry using the inexpensive isotropic covariance approximation $r_t^2 I$; iii) we introduce two multimodal Maze2D datasets to evaluate ANCHOR's ability to recover safety-conditioned distributions and generate safe trajectories: a medium-horizon task with three trajectory modes and a long-horizon task with six; iv) we demonstrate that ANCHOR achieves competitive performance against state-of-the-art safe generative planners across synthetic, navigation, locomotion, and manipulation benchmarks. We release the code and datasets upon acceptance [1].

2 Background

Diffusion models for path planning. We follow the SDE-based formulation of diffusion probabilistic models introduced in Song et al. (2020b) and apply it to trajectory planning as in Janner et al. (2022). We denote $H \in \mathbb{N}$ the horizon and $d \in \mathbb{N}$ the output dimension for each planning step. Each plan is then a trajectory $\mathbf{x}_0 \in \mathbb{R}^{H \times d}$ which we treat for the sake of simplicity as a vector $\mathbf{x}_0 \in \mathbb{R}^{Hd}$. A forward process noises the target distribution of plans p_0 and can be defined as an SDE

$$d\mathbf{x}_t = -\frac{1}{2}\mathbf{x}_t dt + dW_t, \tag{1}$$

[1] https://doobdiffusion.github.io/ANCHOR/

where W_t is a standard Brownian motion. This is an Ornstein-Uhlenbeck process, for which the marginal distribution of $\mathbf{x}_t$ is available in closed form:

$$\mathbf{x}_t \mid \mathbf{x}_0 \sim \mathcal{N}\left(\underbrace{e^{-t/2}}_{=:a_t}\mathbf{x}_0, \underbrace{(1-e^{-t})}_{=:v_t} I\right), \tag{2}$$

and we denote the ratio $r_t^2 := \frac{v_t}{a_t^2}$ as the noise-to-signal ratio. Since for large T the random variable $\mathbf{x}_T$ is approximately distributed as a standard Gaussian, we can generate samples from the target distribution by sampling from a standard Gaussian and running the reverse process. It is given by

$$d\mathbf{y}_t = \left[\frac{1}{2}\mathbf{y}_t + \nabla_{\mathbf{y}_t} \log p_t(\mathbf{y}_t)\right] dt + d\tilde{W}_t, \tag{3}$$

where $s_t(\mathbf{x}) := \nabla_x \log p_t(\mathbf{x})$ denotes the score function as was shown by Anderson (1982). We adopt the convention that the generative process is integrated from $t = T$ down to $t = 0$, and $\tilde{W}$ denotes a reverse-time Brownian motion.

The score-matching objective

$$s_t = \arg\min_{\tilde{s}_t} \mathbb{E}_{\mathbf{x}_0 \sim p_0, \varepsilon \sim \mathcal{N}(0, v_t I)} \left[\|\tilde{s}_t(a_t \mathbf{x}_0 + \varepsilon) + \varepsilon / v_t\|^2\right] \tag{4}$$

indeed yields the score function as optimizer, as shown in Hyvärinen (2005); Vincent (2011). In practice, one parameterizes the score function as a neural network s_θ and, given training data $\mathcal{D}$, minimizes the objective

$$\hat{\theta} = \arg\min_{\theta} \sum_{\mathbf{x}_0 \in \mathcal{D}} \mathbb{E}_{t \sim U(0,T], \varepsilon \sim \mathcal{N}(0, v_t I)} \left[\|s_\theta(t, a_t \mathbf{x}_0 + \varepsilon) + \varepsilon / v_t\|^2\right]. \tag{5}$$

Diffusion guidance. It is often desirable not to sample from the full target distribution p_0 but rather from it conditioned on some event. In Song et al. (2020b), the problem of controllable generation was considered with applications for class-conditional generation, image imputation, and colorization. Denote $p_0(\mathbf{x}_0 \mid \mathbf{x}_0 \in S)$ the conditional distribution given that the sample $\mathbf{x}_0$ is in some set S. Define

$$h_t(\mathbf{x}) := p_{0|t}(\mathbf{x}_0 \in S \mid \mathbf{x}_t = \mathbf{x}) = \int \mathbf{1}_S(\mathbf{x}_0) p(\mathbf{x}_0 \mid \mathbf{x}_t) d\mathbf{x}_0 = \mathbb{E}_{\mathbf{x}_0 \sim p(\mathbf{x}_0 \mid \mathbf{x}_t)} \left[\mathbf{1}_S(\mathbf{x}_0)\right] \tag{6}$$

By Bayes' rule we have that $p_t(\mathbf{x}_t \mid \mathbf{x}_0 \in S) \propto p_t(\mathbf{x}_t) h_t(\mathbf{x}_t)$ and thus get

$$\nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t \mid \mathbf{x}_0 \in S) = \nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t) + \nabla_{\mathbf{x}_t} \log h_t(\mathbf{x}_t). \tag{7}$$

Mathematically, h_t is Doob's h-transform (Särkkä & Solin, 2019). Accordingly, modifying the reverse process with the additional guidance term yields

$$d\mathbf{y}_t = \left[\frac{1}{2}\mathbf{y}_t + \nabla_{\mathbf{y}_t} \log p_t(\mathbf{y}_t) + \nabla_{\mathbf{y}_t} \log h_t(\mathbf{y}_t)\right] dt + dW_t, \tag{8}$$

whose terminal distribution can be shown to be $p_0(\mathbf{y}_0) = p_0(\mathbf{x}_0 \mid \mathbf{x}_0 \in S)$. Generally, computing $h_t(\mathbf{x}_t)$ is intractable for arbitrary data distribution p_0. Several strategies have been suggested in the literature to circumvent this problem. One approach, commonly used for conditional generation of images, is to learn the function $h_t(\mathbf{x})$ directly by training an image classifier on noisy images, as is done in Dhariwal & Nichol (2021); Song et al. (2020b). Another strategy is to implicitly learn the guidance during training, as in classifier-free guidance (Ho & Salimans, 2022). However, both approaches do not work on already available diffusion planners, as they require obstacle-aware training and lack the flexibility to adapt to arbitrary obstacles at test time.

3 Method

Let $S_0 \subseteq \mathbb{R}^d$ be a safe set over the state space that our model is going to predict. We want each of our waypoints to fulfill this condition and thus have, i.e., $S_0 \times \ldots \times S_0 =: S \subseteq \mathbb{R}^{H \times d}$. Alternatively, we want our trajectory to avoid $C := S^c$. In this section, we will introduce our method. It relies on using the diffusion guidance introduced above to sample from the conditional distribution $p_0(\mathbf{x}_0 \mid \mathbf{x}_0 \in S)$.

As already discussed, sampling exactly from this distribution would require access to $h_t(\mathbf{x}_t)$, which is in general unavailable. We do have, however, access via Tweedie's formulas to the first and second moments of $p_{0|t}(\mathbf{x}_0 \mid \mathbf{x}_t)$ and can therefore approximate this posterior with a Gaussian distribution $\mathcal{N}(\mathbf{x}_0; \hat{\mu}_0, \hat{\Sigma}_0)$, which we will discuss in subsection 3.1. It allows us to get an estimate of $h_t(\mathbf{x}_t) \approx \int_S \mathcal{N}(\mathbf{x}_0; \hat{\mu}_0, \hat{\Sigma}_0) d\mathbf{x}_0$ which we can use for guidance. As this can be computationally prohibitively expensive, we develop efficient approximations for calculating the unsafe probability over geometric sets in subsection 3.2. Lastly, in subsection 3.3 we assemble the final guidance function and provide a result that guarantees its safety.

3.1 Gaussian approximation of the posterior

Recall Tweedie's formula (Efron, 2011) for the conditional moments given a noisy sample.

Theorem 1 (Tweedie's formula)**.** *Suppose we have a prior density $p(\mathbf{x})$ and a noise distribution $q_\sigma(\tilde{\mathbf{x}} \mid \mathbf{x}) = \mathcal{N}(\tilde{\mathbf{x}}; \mathbf{x}, \sigma^2 I)$. Define the marginal of the noisy observation by $p_\sigma(\tilde{\mathbf{x}}) = \int p(\mathbf{x}) q_\sigma(\tilde{\mathbf{x}} \mid \mathbf{x}) d\mathbf{x}$. The conditional moments are then given by*

$$\mathbb{E}[\mathbf{x} \mid \tilde{\mathbf{x}}] = \tilde{\mathbf{x}} + \sigma^2 \nabla_{\tilde{\mathbf{x}}} \log p_\sigma(\tilde{\mathbf{x}}) \tag{9}$$

and

$$Cov(\mathbf{x} \mid \tilde{\mathbf{x}}) = \sigma^2 \nabla_{\tilde{\mathbf{x}}} \mathbb{E}[\mathbf{x} \mid \tilde{\mathbf{x}}] \tag{10}$$

Since we assume access to a well-trained score model, we can compute the conditional moments for each time t and state $\mathbf{x}_t$, via

$$\mathbb{E}[\mathbf{x}_0 \mid \mathbf{x}_t] = \frac{1}{a_t}(\mathbf{x}_t + v_t \nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t)), \quad \mathrm{Cov}(\mathbf{x}_0 \mid \mathbf{x}_t) = \frac{v_t}{a_t^2}(I + v_t \nabla^2 \log p_t(\mathbf{x}_t)). \tag{11}$$

The mean of the posterior requires only access to the gradient of the log density, i.e., the score function, which our neural network approximates via the empirical score matching objective (5). On the other hand, the covariance requires the Hessian of the log density, which is difficult to compute in practice (Song et al., 2020a; Martens et al., 2012). Even the choice of a Hessian-free covariance $\nabla^2_{\mathbf{x}_t} \log p_t(\mathbf{x}_t) = 0$ is a valid option, as we will derive safety guarantees for this approximation later and demonstrate its effectiveness in Section 4.

3.2 Signed Distance Functions for efficient computations of the Gaussian integral

In order to compute the Gaussian integral above efficiently, we encode the unsafe region using signed distance functions (SDF). After linearization, the multi-dimensional integration problem turns into a scalar one, once a suitable representation with an SDF is found. Recall the definition

Definition 1 (Signed distance function)**.** Let $\mathcal{C} \subset \mathbb{R}^n$ be closed with non-empty interior and boundary $\partial\mathcal{C}$. The *signed distance function* (SDF) of $\mathcal{C}$ is $s : \mathbb{R}^n \to \mathbb{R}$,

$$s_{\mathcal{C}}(\mathbf{x}) \;:=\; \begin{cases} +\inf_{\mathbf{y} \in \partial\mathcal{C}} \|\mathbf{x} - \mathbf{y}\|, & \mathbf{x} \notin \mathcal{C}, \\ -\inf_{\mathbf{y} \in \partial\mathcal{C}} \|\mathbf{x} - \mathbf{y}\|, & \mathbf{x} \in \mathcal{C}. \end{cases} \tag{12}$$

In particular we have that $S = \{s_S(\mathbf{x}) \leq 0\} = \{s_C(\mathbf{x}) \geq 0\}$ as well as $\|\nabla s_{\mathcal{C}}(\mathbf{x})\| = 1$ a.e. For general sets such as ellipses, the SDF is not as easy to obtain. We instead refer to the level function, which in applications is typically required to describe the obstacle as $C = \{L \leq 1\}$. It enables us to get a first-order approximation of the set's SDF via

$$\hat{s}(x) = \frac{L(x) - 1}{\|\nabla L(x)\|}.$$

This surrogate has the same zero level set and boundary normal as the exact signed distance and satisfies

$$\hat{s}(x) = s_C(x) + O(s_C(x)^2)$$

near ∂C; see Appendix B.1.

Using the SDF formulation, the quantity we must evaluate at every reverse step is the mass that a Gaussian places on the safe side of the obstacle,

$$h_t(\mathbf{x}_t) = \int \mathbf{1}\left[s_C(\mathbf{x}_0) \geq 0\right] p_{0|t}(\mathbf{x}_0 \mid \mathbf{x}_t) d\mathbf{x}_0 \tag{13}$$

$$\approx \underbrace{\int \mathbf{1}\left[s_C(\mathbf{x}) \geq 0\right] \mathcal{N}(\mathbf{x}; \hat{\mathbf{x}}_0(\mathbf{x}_t), \hat{\Sigma}_0(\mathbf{x}_t))\, \mathrm{d}\mathbf{x}}_{=:P(t,\mathbf{x}_t)}, \tag{14}$$

where $\hat{\mathbf{x}}_0(\mathbf{x}_t)$ is the Tweedie mean estimate and $\hat{\Sigma}(\mathbf{x_t})$ is some estimate of the covariance at state $\mathbf{x}_t$. This has not yet made the computation more efficient, as a quadrature would still integrate over a substantial part of Euclidean space. It allows us, however, by linearizing s_C, to collapse the integral to a one-dimensional Gaussian tail, which has a closed form. Denote by

$$\tilde{s}_C(\mathbf{x}) = s_C(\hat{\mathbf{x}}_0) + \mathbf{g_C}^\top (\mathbf{x} - \hat{\mathbf{x}}_0) \tag{15}$$

the first-order approximation of the SDF around the Tweedie mean $\hat{\mathbf{x}}_0$ with $\mathbf{g_C} := \nabla s_C(\hat{\mathbf{x}}_0) \neq \mathbf{0}$. We then obtain

$$P(t, \mathbf{x}_t) \approx P_1(t, \mathbf{x}_t) := \int_{\mathbb{R}^{Hd}} \mathbf{1}\left[\tilde{s}_C(\mathbf{x}) \geq 0\right] \mathcal{N}(\mathbf{x}; \hat{\mathbf{x}}_0, \hat{\Sigma}) d\mathbf{x} \tag{16}$$

$$= \int_{\mathbb{R}} \mathbf{1}\left[u \geq 0\right] \mathcal{N}(u; \tilde{s}_C(\hat{\mathbf{x}}_0), \mathbf{g}_C^\top \hat{\Sigma} \mathbf{g}_C) du = \Phi(\tilde{s}_C(\hat{\mathbf{x}}_0)/\sigma_{\text{SDF}}), \tag{17}$$

where $\sigma^2_{\text{SDF}} := \mathbf{g}_C^\top \hat{\Sigma} \mathbf{g}_C$. This follows because affine images of Gaussian random variables are again Gaussian variables. This yields a 1D Gaussian integral which we can compute efficiently. The linear approximation that allows this seems very coarse, as it turns any convex set into the hyperplane tangent to the closest point on the boundary from $\hat{\mathbf{x}}$, $u(\hat{\mathbf{x}}) := \arg\min_{\mathbf{u} \in \partial S} \|\mathbf{u} - \hat{\mathbf{x}}\|$. However, this approximation is exact on the straight path from $\hat{\mathbf{x}}$ to $u(\hat{\mathbf{x}})$ and, furthermore, for convex obstacles, $P_1(t, \mathbf{x}_t)$ is a lower bound for $P(t, \mathbf{x}_t)$ as the tangent half-plane proxy for the unsafe set encloses the actual unsafe set.

3.3 Safety guarantee

This approximation loses its validity the more the probabilities $P(t, \mathbf{x}_t)$ and $P_1(t, \mathbf{x}_t)$ differ, which tends to grow as the diffusion time t becomes large. This is because the posterior covariance spreads for large t and thus the Gaussian integral is a blob over a larger region over which the approximation validity of the hyperplane to the unsafe set diminishes; whereas for smaller t the Gaussian integral focuses on a small region around $u(\hat{\mathbf{x}}_0)$ for which the hyperplane approximation is the most exact. In order to control for the inaccuracies of these approximations, we add a time threshold to the algorithm, $t_{\max}$, that controls when we start to apply the guidance. In total, we obtain the guidance function

$$f_t(\mathbf{x}_t) := \begin{cases} \displaystyle\sum_{j=1}^{H} \nabla_{\mathbf{x}_t} \log \Phi\left(\frac{s_{C_0}\left(\hat{\mathbf{x}}_0^{(j)}(\mathbf{x}_t)\right)}{r_t}\right) & \text{if } t \leq t_{\max}, \\ 0 & \text{if } t > t_{\max}. \end{cases} \tag{18}$$

With it, the guided reverse process is

$$d\mathbf{z}_t = \left[\frac{1}{2}\mathbf{z}_t + \nabla_{\mathbf{z}_t} \log p_t(\mathbf{z}_t) + f_t(\mathbf{z}_t)\right] dt + d\tilde{W}_t. \tag{19}$$

The safety of our method is guaranteed by the following theorem for half-space obstacles, where the signed-distance linearization (15) is exact.

Theorem 2. *Let $D := dH$ and let $\Xi : \mathbb{R}^D \to \mathbb{R}$ be differentiable with bounded gradient $\|\nabla \Xi\| \leq L$. Suppose that our data distribution is of the form*

$$p_{data}(\mathbf{x}) \propto \exp(\Xi(\mathbf{x}))\, \mathcal{N}(\mathbf{x}; \mu_0, \Sigma_0), \qquad \Sigma_0 = \operatorname{diag}\left(\Sigma_0^{(1)}, \ldots, \Sigma_0^{(H)}\right) \succ 0, \tag{20}$$

i.e., Σ_0 is block diagonal with respect to the waypoints. Let the obstacle be the half-space $C_0 := \{\mathbf{y} \in \mathbb{R}^d : a^\top \mathbf{y} < b\}$ with $\|a\| = 1$, so that $s_{C_0}(\mathbf{y}) = a^\top \mathbf{y} - b$, and construct the trajectory

safe set $S = (C_0^c)^H$. Let f_t denote the waypoint-wise guidance defined in (18), obtained from the Hessian-free isotropic posterior covariance approximation $\widehat{V}_t = r_t^2 I$. Let $\gamma > 1$ and let $\mathbf{z}_0$ denote the random variable at time 0 of the SDE defined in the reverse process guided by γf_t (19). Then

$$\mathbb{P}(\mathbf{z}_0 \in S) = 1, \tag{21}$$

i.e., the distribution produced by the guided diffusion process (19) is almost surely safe.

The block-diagonal structure of Σ_0 only restricts the Gaussian reference part of the data distribution; correlations between waypoints can still be carried by Ξ. Curved obstacles, for which the tangent half-space approximation of Section 3.2 is no longer exact, are not covered by Theorem 2, however, we conjecture that the safety of our method can be proven for more general obstacle shapes than only half-spaces and with $\gamma = 1$ as we demonstrate empirically in the following section.

4 Experiments

We evaluate ANCHOR on two synthetic mixture-generation tasks, three multimodal trajectory-planning tasks spanning synthetic and simulated settings, and three standard simulation benchmarks for robotic locomotion and manipulation. By conducting these experiments, we demonstrate the following properties: i) ANCHOR accurately reconstructs the safe conditional target distribution, rather than merely enforcing safety by concentrating samples near the constraint boundaries; ii) when a test-time constraint blocks a trajectory mode, ANCHOR can replan through the remaining safe modes rather than generate low-probability interpolations; iii) ANCHOR's safety and planning quality are competitive with state-of-the-art generative trajectory planners on standard robot benchmarks; and iv) the approximations introduced in the theory give robust distributional fidelity and performance. We compare ANCHOR against state-of-the-art generative trajectory-planning baselines: Diffuser (Janner et al., 2022), Classifier Guidance (Ho & Salimans, 2022), Truncate (Brockman et al., 2016), the RoS, ReS, and TVS variants of SafeDiffuser (Xiao et al., 2023), unconstrained Flow Matching (Lipman et al., 2022), SAD-Flower (Huang et al., 2026), as well as FlowMatcher and SafeFlowMatcher (Yang et al., 2026). Additional results, architectural and implementation details are provided in Appendices C and D.

4.1 Conditional distribution recovery

We evaluate ANCHOR's reconstruction accuracy on three synthetic tasks for which the safe conditional target distribution $p_0(x_0 \mid x_0 \in S)$ can be computed analytically. We construct one- and two-dimensional Gaussian mixtures, and a synthetic tri-modal trajectory benchmark, where an unsafe region is centered on one of the modes. Fig. 2 and Table 5 show that ANCHOR achieves the lowest distributional error under 100% safety on all benchmarks. The failure mode of the CBF-based safety method is that too much probability mass accumulates at the edge of the obstacle where the ground truth distribution barely has any support. ANCHOR, on the other hand, is able to overcome this issue due to its probabilistic grounding. Still, it is not able to completely match the conditional distribution even when fully accounting for the Hessian due to the several approximations that our method relies on. While retaining the safety as guaranteed by Theorem 2, the performance gap widens when we move from modeling the full covariance to the Hessian-free approach. We have found empirically that in certain situations, weighing the guidance by the inverse standard deviation $1/\sigma_t$ can lead to better results for the Hessian-free version; see Appendices D and D.4. Also, applying the guidance over the complete reverse process ($t_{\max} = 1.0$) worsens the performance of ANCHOR (Appendix D.4, Table 11), consistent with Section 3.3: at large diffusion times, the posterior covariance is broad, and the local linearization of the SDF around the Tweedie estimate is least accurate. Suppressing guidance for $t > 0.9$ therefore avoids committing samples to a mode before the denoised estimate provides sufficiently local geometric information.

4.2 Safe multimodal trajectory planning

We introduce two multimodal navigation benchmarks using the Maze2D simulator (Fu et al., 2020). Multimodal Maze: Medium contains 12,000 demonstrations, balanced across three modes, with a planning horizon of $H = 384$. Multimodal Maze: Hard contains 24,000 long-horizon demonstrations, balanced across six modes, with $H = 768$. Both datasets include variation within each trajectory

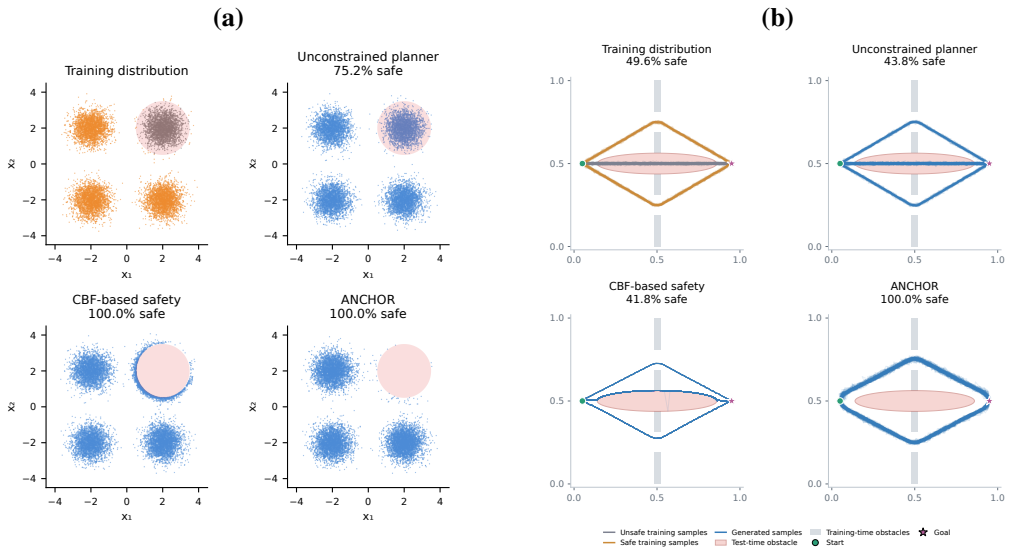


Figure 2: (a) 2-D Gaussian mixture, and (b) Sequential synthetic trajectory benchmarks. 1-D Gaussian mixture reported in Appendix D. Blue histograms show generated samples; the orange curve/samples show the exact conditional target, and red shading marks the unsafe set. In both cases, unconstrained samples ignore safety constraints; CBF-based approaches (SafeDiffuser on the left, SafeFlowMatcher on the right) tend to stack samples on the boundary, while ANCHOR models the safe conditional distribution.

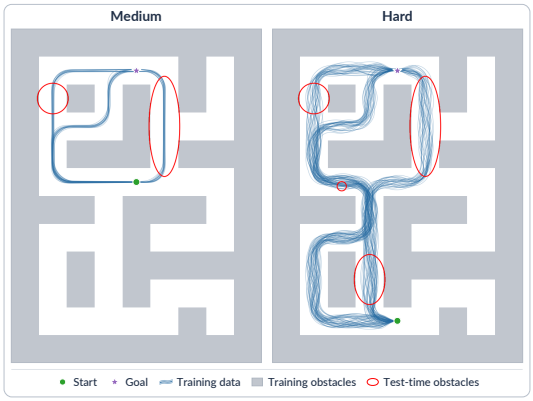


Figure 3: Multimodal Maze2d datasets: Medium (3 modes, of which 2 are blocked) and Hard (6 modes, of which 5 are blocked, plus a locally half-blocked segment).

mode. Figure 3 shows the demonstrations, start and goal positions, and test-time obstacles. The added obstacles block alternative routes while retaining safe demonstrated solutions for the same start–goal task. These settings test both selection among globally distinct routes and local adaptation, while ensuring that safe solutions are represented in the training data. Planned safety checks the complete path against the added obstacles and additionally to collision with already existing maze walls, accounting for the robot's radius. We additionally report boundary safety and cumulative reward from simulator execution. Results are presented in Table 1. ANCHOR outperforms all diffusion and flow-matching planners by a clear margin both in the default and the $1/\sigma_t$-variant. The generated trajectories in Figure 11 illustrate this difference: ANCHOR is the only method shown to reroute trajectories through the remaining obstacle-free modes, whereas the CBF-based planners locally deform trajectories and incur wall collisions, consistent with their lower safety and reward. The remaining safety violations on the long-horizon Hard benchmark motivate future work on richer posterior approximations and guidance that better accounts for global trajectory constraints.

Method	Multimodal Maze: Medium		Multimodal Maze: Hard		
	Safe/B-Safe(%) ↑	Reward ↑	Safe(%) ↑	B-Safe(%) ↑	Reward ↑
Diffuser	37.0	28.30±37.23	4.7	4.2	20.71±46.67
Flow Matching	29.2	21.58±33.81	3.6	3.6	16.88±44.22
Truncate	36.5	27.62±36.78	5.7	5.2	20.51±46.04
Classifier guidance	34.4	26.57±36.69	4.7	4.2	21.49±47.56
SafeDiffuser–RoS	0.0	33.16±39.54		Infeasible	
SafeDiffuser–ReS	37.0	28.22±37.13		Infeasible	
SafeDiffuser–TVS	43.8	33.10±37.88	1.6	1.6	16.06±40.94
FlowMatcher	0.0	0.00±0.00	0.0	0.0	7.35±13.37
SafeFlowMatcher($T_p = 16$)	0.0	0.00±0.00	0.0	0.0	2.78±8.61
SAD-Flower(U-Net)	4.7	27.95±38.46	0.0	0.0	4.35±21.22
ANCHOR (constant)	**94.3**	**70.26±18.89**	22.9	19.8	**77.73±60.05**
ANCHOR (inverse std)	93.2	68.86±20.05	**45.3**	**35.4**	73.33±60.28

Table 1: Multimodal Maze results over 192 planning attempts. Safe and B-Safe assess full-plan obstacle clearance, including maze walls; B-Safe requires > 0.01 m clearance. Reward is cumulative simulator reward (mean ± population standard deviation). Infeasible denotes no completed plans (0/192). SAD-Flower is evaluated with same architectural and sampling budget of other baselines. Further implementation details are provided in Appendix D.2.

Method	Hopper: roof 1.60 m			Walker2d: roof 1.40 m			Kuka: 90% joint limits		
	Score ± SD	P/E	B_1/B_2	Score ± SD	P/E	B_1/B_2	Reward ± SD	P/E	B_1/B_2
Diffuser	0.429±0.095	51/36	39/5	0.208±0.147	58/67	55/10	0.640±0.575	48/48	28/19
CG	0.432±0.093	0/35	0/0	0.214±0.152	**100**/71	60/**16**	0.690±0.560	**100/100**	99/60
Truncate	0.422±0.102	**100**/39	43/**6**	0.209±0.153	**100**/73	56/11	**0.710**±0.535	**100/100**	**100**/58
SFM	0.143±0.057	**100/96**	0/0	0.057±0.048	**100/95**	70/0	0.500±0.539	**100/100**	**100**/27
ANCHOR	**0.437**±0.085	**100**/39	**100**/5	**0.226**±0.154	**100**/60	**100**/13	0.680±0.581	**100/100**	**100/72**

Table 2: Robotics results over 100 completed episodes per method and benchmark. Scores and rewards are mean ± SD; safety rates are percentages. P/E lists planned and robot execution safety, respectively. B_1/B_2 lists planned boundary safety at margins of 1/5 cm for locomotion and 5/10% for Kuka. Bold indicates the highest mean score/reward and each highest safety rate among the displayed methods, including ties. CG denotes classifier guidance; SFM denotes SafeFlowMatcher. ANCHOR uses inverse-noise-standard-deviation weighting with $t_{\max} = 0.3$ for locomotion and constant weighting with $t_{\max} = 0.9$ for Kuka.

4.3 Performance on locomotion and robot manipulation benchmarks

We additionally evaluate on Hopper and Walker2d using the D4RL `medium-expert-v2` datasets (Fu et al., 2020), and on Kuka block stacking dataset (Janner et al., 2022). The planning horizons are $H = 600$ for locomotion and $H = 128$ for manipulation. We impose root-height ceilings of 1.60 m for Hopper and 1.40 m for Walker2d, and restrict Kuka to 90% of its original joint limits. Table 2 compares task performance, planned safety, execution safety, and boundary safety. ANCHOR achieves 100% planned safety on all three benchmarks. On Hopper and Walker2d, it obtains the highest mean scores among the reported methods and is the only method whose plans all satisfy the 1 cm boundary margin. This advantage does not extend uniformly to the stricter 5 cm margin. On Kuka, ANCHOR combines 100% planned and execution safety with competitive reward, while achieving the highest boundary safety at the stricter 10% margin (72%). Planned safety does not, however, fully translate into execution safety on locomotion: ANCHOR attains execution safety rates of 39% on Hopper and 60% on Walker2d. SafeFlowMatcher achieves substantially higher execution safety, but with markedly lower task scores. ANCHOR exhibits the same trade-off internally: the cutoff ablation in Appendix D.4 (Table 13) shows that guiding over the full reverse process ($t_{\max} = 1.0$) brings 5 cm boundary safety to 100%, but reduces the score to 0.125 on Hopper and -0.005 on Walker2d, matching SafeFlowMatcher, whereas $t_{\max} \leq 0.5$ preserves scores of 0.42–0.44 and 0.21–0.23 at 5–18% boundary safety. These results highlight the importance of

Benchmark / constraint	H	Total windows	Safe windows	Safe (%)
Hopper: 1.50 m roof (reference)	600	398,702	0	0.00
Hopper: 1.60 m roof	600	398,702	238,916	59.92
Walker2d: 1.30 m roof (reference)	600	727,791	38,000	5.22
Walker2d: 1.40 m roof	600	727,791	448,497	61.62
Kuka: 100% joint limits (reference)	128	716,930	716,930	100.00
Kuka: 90% joint limits	128	716,930	379,348	52.91
Maze2D (reference)	384	111	0	0.00
Multimodal Maze2D Medium	384	12,000	4,000	33.33
Multimodal Maze2D Hard	768	24,000	949	3.95

Table 3: Training-data safety support under the benchmark test constraints. Total windows include safe and unsafe training samples. References are Xiao et al. (2023); Yang et al. (2026)

assessing task performance and executed physical constraint satisfaction, which are influenced by other unmodeled factors like velocity, acceleration, and jerk of the trajectory. This motivates further work on bridging the gap between safe trajectory generation and execution. Additional results and ablations are reported in Appendix C and D.4.

4.4 Safety constraints and demonstration support

The relationship between the safety constraint and the demonstration distribution is an important part of the experimental setting. We therefore measure how many training trajectories satisfy each constraint, using the relevant planning horizon and, where applicable, the evaluation start–goal conditions. Table 3 shows that, for the original Maze2D setting used in previous works (Xiao et al., 2023; Yang et al., 2026), none of the training demonstration segments is a safe solution for the evaluation settings. Under the original constraints for the locomotion tasks, no full-length Hopper window satisfies the $1.50\,\mathrm{m}$ ceiling, while 5.22% of full-length Walker2d windows satisfy the $1.30\,\mathrm{m}$ ceiling. At the ceilings used in our comparison, these proportions increase to 59.92% and 61.62%, respectively. For Kuka, all training windows satisfy the original joint limits, whereas 52.91% satisfy the tightened 90% limits. These observations reveal two distinct evaluation settings: recovering the safe conditional distribution when safe solutions are represented in the training data, and out-of-distribution generalization when satisfying the constraints requires behavior beyond the available demonstrations. We emphasize that, in the latter setting, satisfying the specified test-time constraints alone does not establish safety with respect to other, unmodeled constraints. Locally deforming trajectories into poorly supported regions assumes that these regions are otherwise admissible, and may overlook constraints captured implicitly by the training demonstrations. Moreover, concentrating trajectories near a nominal obstacle boundary leaves little tolerance for uncertainty in its location. These considerations motivate treating constraint satisfaction and consistency with the learned behavior distribution as distinct evaluation criteria for safe trajectory planners.

5 Conclusion and future work

In this work, we have proposed ANCHOR: a diffusion-based method for safe planning that uses Gaussian approximations of Doob's h-function. We have embedded it in the larger context of modeling safe trajectory planning as the problem of sampling from the conditional $p(x_0 \mid x_0 \in S)$. We derived a theoretical safety guarantee for our method that remains valid despite the approximations introduced to make the method computationally feasible and efficient. On synthetic benchmarks, ANCHOR accurately approximates the conditional target distribution without accumulating probability mass near the obstacle boundary, as observed for the baselines. On a simulated navigation task, it also remains competitive with state-of-the-art safe trajectory planners in terms of safety, planning performance, and computational cost. Future work will investigate richer posterior distribution approximations that capture multimodality, how to expand the probabilistic framework to flow-based planners (Feng et al., 2025), and how to design SDFs that encode real-world robotic constraints. Additionally, we believe that improving the safety guarantees for arbitrary obstacles can be a fruitful research direction.

AI USE STATEMENT

In this work, we used generative AI tools for polishing the manuscript, experiment bookkeeping, rapid prototyping, assistance in proving mathematical claims, providing feedback on intermediate states of the work, creating figures, and summing up results. We have not used generative AI tools for ideation, implementing methods, reviewing literature and identifying research gaps, cleaning and reformatting data or interpreting results. We have reviewed all AI-assisted work and take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

We have taken several measures to ensure reproducibility of our results. All theoretical claims have rigorous proofs, presented in detail in the appendix. The assumptions underlying the theorems are explicitly stated, and definitions and notations are comprehensively clarified. The experimental benchmark is reproducible by independent reviewers; we will additionally provide the code for the planning framework and simulation benchmarks upon acceptance.

ETHICS STATEMENT

This work focuses on the theoretical analysis of diffusion guidance for safe trajectory planning and does not involve experiments on human subjects, sensitive personal data, or applications with direct societal risks. The simulation benchmarks used are publicly available, and no private or restricted data was used.

REFERENCES

Anurag Ajay, Yilun Du, Abhi Gupta, Joshua B. Tenenbaum, Tommi S. Jaakkola, and Pulkit Agrawal. Is conditional generative modeling all you need for decision making? In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=sP1fo2K9DFG.

Brian DO Anderson. Reverse-time diffusion equation models. *Stochastic Processes and their Applications*, 12(3):313–326, 1982.

Arpit Bansal, Hong-Min Chu, Avi Schwarzschild, Soumyadip Sengupta, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Universal guidance for diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pp. 843–852, June 2023.

S. Boucheron, G. Lugosi, and P. Massart. *Concentration Inequalities: A Nonasymptotic Theory of Independence*. OUP Oxford, 2013. ISBN 9780199535255. URL https://books.google.de/books?id=koNqWRluhP0C.

Benjamin Boys, Mark Girolami, Jakiw Pidstrigach, Sebastian Reich, Alan Mosca, and O. Deniz Akyildiz. Tweedie moment projected diffusions for inverse problems, 2024. URL https://arxiv.org/abs/2310.06721.

Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. Openai gym. *arXiv preprint arXiv:1606.01540*, 2016.

Xiaoyuan Cheng, Xiaohang Tang, and Yiming Yang. Safe and stable control via lyapunov-guided diffusion models, 2025. URL https://arxiv.org/abs/2509.25375.

Jacob K Christopher, Stephen Baek, and Ferdinando Fioretto. Constrained synthesis with projected diffusion models, 2024. URL https://arxiv.org/abs/2402.03559.

Hyungjin Chung, Jeongsol Kim, Michael Thompson Mccann, Marc Louis Klasky, and Jong Chul Ye. Diffusion Posterior Sampling for General Noisy Inverse Problems. In *International Conference on Learning Representations*, 2023. URL https://mlanthology.org/iclr/2023/chung2023iclr-diffusion/.

Alexander Denker, Francisco Vargas, Shreyas Padhy, Kieran Didi, Simon Mathis, Vincent Dutordoir, Riccardo Barbano, Emile Mathieu, Urszula Julia Komorowska, and Pietro Lio. Deft: Efficient fine-tuning of diffusion models by learning the generalised h-transform. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 19636–19682. Curran Associates, Inc., 2024. doi: 10.52202/079017-0620. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/22d258dfbdf840ccbf266bbc545dd95f-Paper-Conference.pdf.

Prafulla Dhariwal and Alex Nichol. Diffusion models beat gans on image synthesis. In *Proceedings of the 35th International Conference on Neural Information Processing Systems*, NIPS '21, Red Hook, NY, USA, 2021. Curran Associates Inc. ISBN 9781713845393.

Bradley Efron. Tweedie's formula and selection bias. *Journal of the American Statistical Association*, 106(496):1602–1614, 2011. ISSN 01621459. URL http://www.jstor.org/stable/23239562.

Ruiqi Feng, Chenglei Yu, Wenhao Deng, Peiyan Hu, and Tailin Wu. On the guidance of flow matching. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 16993–17029. PMLR, 13–19 Jul 2025. URL https://proceedings.mlr.press/v267/feng25s.html.

Justin Fu, Aviral Kumar, Ofir Nachum, George Tucker, and Sergey Levine. D4rl: Datasets for deep data-driven reinforcement learning. *arXiv preprint arXiv:2004.07219*, 2020.

Zhengyi Guo, Wenpin Tang, and Renyuan Xu. Conditional diffusion guidance under hard constraint: A stochastic analysis approach, 2026. URL https://arxiv.org/abs/2602.05533.

Yutong He, Naoki Murata, Chieh-Hsin Lai, Yuhta Takida, Toshimitsu Uesaka, Dongjun Kim, WeiHsiang Liao, Yuki Mitsufuji, Zico Kolter, Ruslan Salakhutdinov, and Stefano Ermon. Manifold preserving guided diffusion. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), *International Conference on Learning Representations*, volume 2024, pp. 44819–44850, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/c355566ce402de341c3320cf69a10750-Paper-Conference.pdf.

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

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

Tzu-Yuan Huang, Armin Lederer, Dai-Jie Wu, Xiaobing Dai, Sihua Zhang, Stefan Georg Sosnowski, Shao-Hua Sun, and Sandra Hirche. Sad-flower: Flow matching for safe, admissible, and dynamically consistent planning. In *International Conference on Machine Learning*, 2026.

Aapo Hyvärinen. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6(24):695–709, 2005. URL http://jmlr.org/papers/v6/hyvarinen05a.html.

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

Wonsuhk Jung, Utkarsh Aashu Mishra, Nadun Ranawaka Arachchige, Yongxin Chen, Danfei Xu, and Shreyas Kousik. Joint model-based model-free diffusion for planning with constraints. In Joseph Lim, Shuran Song, and Hae-Won Park (eds.), *Proceedings of The 9th Conference on Robot Learning*, volume 305 of *Proceedings of Machine Learning Research*, pp. 4328–4350. PMLR, 27–30 Sep 2025. URL https://proceedings.mlr.press/v305/jung25d.html.

S. Karaman and E. Frazzoli. Incremental sampling-based algorithms for optimal motion planning. In *Proceedings of Robotics: Science and Systems*, Zaragoza, Spain, June 2010. doi: 10.15607/RSS.2010.VI.034.

Ilya Kostrikov, Ashvin Nair, and Sergey Levine. Offline reinforcement learning with implicit q-learning. *arXiv preprint arXiv:2110.06169*, 2021.

Aviral Kumar, Justin Fu, Matthew Soh, George Tucker, and Sergey Levine. Stabilizing off-policy q-learning via bootstrapping error reduction. *Advances in neural information processing systems*, 32, 2019.

Aviral Kumar, Aurick Zhou, George Tucker, and Sergey Levine. Conservative q-learning for offline reinforcement learning. *Advances in neural information processing systems*, 33:1179–1191, 2020.

Michael Laskey, Jonathan Lee, Roy Fox, Anca Dragan, and Ken Goldberg. Dart: Noise injection for robust imitation learning. In *Conference on robot learning*, pp. 143–156. PMLR, 2017.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

Haofei Lu, Dongqi Han, Yifei Shen, and Dongsheng Li. What makes a good diffusion planner for decision making? In *International conference on learning representations*, volume 2025, pp. 68505–68534, 2025.

James Martens, Ilya Sutskever, and Kevin Swersky. Estimating the hessian by back-propagating curvature. *arXiv preprint arXiv:1206.6464*, 2012.

Kazuki Mizuta and Karen Leung. Cobl-diffusion: Diffusion-based conditional robot planning in dynamic environments using control barrier and lyapunov functions. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 13801–13808, 2024. doi: 10.1109/IROS58592.2024.10802549.

Stéphane Ross, Geoffrey Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pp. 627–635. JMLR Workshop and Conference Proceedings, 2011.

Ralf Römer, Alexander von Rohr, and Angela P. Schoellig. Diffusion predictive control with constraints, 2025. URL https://arxiv.org/abs/2412.09342.

S. Särkkä and A. Solin. *Applied Stochastic Differential Equations*. Institute of Mathematical Statistics Textbooks. Cambridge University Press, 2019. ISBN 9781316510087. URL https://books.google.de/books?id=g5SODwAAQBAJ.

Jiaming Song, Arash Vahdat, Morteza Mardani, and Jan Kautz. Pseudoinverse-Guided Diffusion Models for Inverse Problems. In *International Conference on Learning Representations*, 2023a. URL https://mlanthology.org/iclr/2023/song2023iclr-pseudoinverseguided/.

Jiaming Song, Qinsheng Zhang, Hongxu Yin, Morteza Mardani, Ming-Yu Liu, Jan Kautz, Yongxin Chen, and Arash Vahdat. Loss-guided diffusion models for plug-and-play controllable generation. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 32483–32498. PMLR, 23–29 Jul 2023b. URL https://proceedings.mlr.press/v202/song23k.html.

Yang Song, Sahaj Garg, Jiaxin Shi, and Stefano Ermon. Sliced score matching: A scalable approach to density and score estimation. In *Uncertainty in artificial intelligence*, pp. 574–584. PMLR, 2020a.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020b.

Pascal Vincent. A connection between score matching and denoising autoencoders. *Neural Computation*, 23(7):1661–1674, 07 2011. ISSN 0899-7667. doi: 10.1162/NECO_a_00142. URL https://doi.org/10.1162/NECO_a_00142.

Zhendong Wang, Jonathan J Hunt, and Mingyuan Zhou. Diffusion policies as an expressive policy class for offline reinforcement learning. *arXiv preprint arXiv:2208.06193*, 2022.

Wei Xiao, Tsun-Hsuan Wang, Chuang Gan, and Daniela Rus. Safediffuser: Safe planning with diffusion probabilistic models, 2023. URL https://arxiv.org/abs/2306.00148.

Jeongyong Yang, Seunghwan Jang, and SooJean Han. Safeflowmatcher: Safe and fast planning using flow matching with control barrier functions, 2026. URL https://arxiv.org/abs/2509.24243.

Haotian Ye, Haowei Lin, Jiaqi Han, Minkai Xu, Sheng Liu, Yitao Liang, Jianzhu Ma, James Zou, and Stefano Ermon. Tfg: Unified training-free guidance for diffusion models. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 22370–22417. Curran Associates, Inc., 2024. doi: 10.52202/079017-0704. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/2818054fc6de6dacdda0f142a3475933-Paper-Conference.pdf.

Jiwen Yu, Yinhuai Wang, Chen Zhao, Bernard Ghanem, and Jian Zhang. Freedom: Training-free energy-guided conditional diffusion model. In *ICCV 2023*, 2023.

Jichen Zhang, Liqun Zhao, Antonis Papachristodoulou, and Jack Umenberger. Constrained diffusers for safe planning and control, 2025. URL https://arxiv.org/abs/2506.12544.

APPENDIX OVERVIEW

A Related works **15**

B Theory **16**

B.1 Normalized level functions as signed-distance surrogates 16

B.2 Proof of almost sure safety 17

C Experiments **26**

C.1 Experimental details 26

C.2 Experimental setup 27

D Additional results **29**

D.1 Performance on Synthetic benchmarks 29

D.2 Performance on Multimodal Maze 30

D.3 Quality of the theoretical approximations 31

D.4 Additional ANCHOR ablations 33

A Related works

Safe planning using diffusion and flow-based generative models. Classical sampling-based planners seek feasible, low-cost paths using explicit geometric models (Karaman & Frazzoli, 2010). Recent diffusion- and flow-matching-based generative planners complement this perspective by learning expressive, multimodal trajectory priors implicitly from offline data. Diffuser (Janner et al., 2022) formulates planning as trajectory denoising, while Decision Diffuser (Ajay et al., 2023) learns condition-dependent state trajectories and extracts actions through a separately trained inverse dynamics model. To incorporate safety, SafeDiffuser, CoBL-Diffusion, and Lyapunov-guided diffusion and flow matching methods integrate control barrier or Lyapunov functions into generation (Xiao et al., 2023; Mizuta & Leung, 2024; Cheng et al., 2025; Yang et al., 2026). SAD-Flower (Huang et al., 2026) additionally addresses action admissibility and dynamic consistency through control-augmented flow matching, using predefined or data-driven system dynamics. Other approaches enforce constraints through projection or constrained Langevin sampling (Christopher et al., 2024; Römer et al., 2025; Zhang et al., 2025). JM2D (Jung et al., 2025) provides a complementary probabilistic formulation for combining learned diffusion priors with model-based constraint modules. Our approach instead treats safety directly as a conditioning event. Rather than correcting a generated trajectory toward the feasible set or forcing the intermediate, noised latents to adhere to the constraints, we aim to sample from the conditional distribution $p(x_0 \mid x_0 \in S)$ by approximating the corresponding Doob guidance, yielding a diffusion-native interpretation of obstacle avoidance.

Test-time diffusion guidance. Training-free diffusion guidance aims to adapt pretrained unconditional models to new conditions at inference time. Generic methods such as Bansal et al. (2023); Yu et al. (2023); Song et al. (2023b); He et al. (2024); Ye et al. (2024) construct guidance from differentiable predictors or losses. In inverse problems, DPS (Chung et al., 2023) and ΠGDM (Song et al., 2023a) approximate conditional scores using denoised estimates, while TMPD (Boys et al., 2024) uses Tweedie's formulas to obtain Gaussian moment approximations to the reverse posterior. In accordance with the latter work, our method also relies on Tweedie estimates to form a Gaussian moment approximation; however, theirs relies on the linearity of the forward operator and is thus explicitly designed for inverse problems, whereas our method employs signed distance functions for efficient geometric guidance. Similar to our probabilistic perspective, Denker et al. (2024) and Guo et al. (2026) also propose to use Doob's h-function for conditional generation and hard event constraints, respectively, but fundamentally require training to learn it, whereas we approximate it analytically using a moment-matched Gaussian.

B THEORY

B.1 NORMALIZED LEVEL FUNCTIONS AS SIGNED-DISTANCE SURROGATES

We justify here the signed-distance approximation used in Section 3.2. Let $C \subset \mathbb{R}^d$ have C^2 boundary and positive reach, and let s_C denote its signed-distance function, positive outside C. Let U be a neighborhood of ∂C and suppose that $L \in C^2(U)$ satisfies

$$C \cap U = \{x : L(x) \le 1\}, \qquad \partial C = \{x : L(x) = 1\}, \qquad \nabla L(x) \neq 0$$

throughout U. Define

$$\hat{s}_C(x) := \frac{L(x) - 1}{\|\nabla L(x)\|}.$$

Lemma 1. *There exists a tubular neighborhood U_ρ of ∂C such that*

$$\operatorname{sign} \hat{s}_C(x) = \operatorname{sign} s_C(x), \qquad x \in U_\rho,$$

and

$$\nabla \hat{s}_C(x) = \nabla s_C(x), \qquad x \in \partial C.$$

Moreover, uniformly for $x \in U_\rho$,

$$\hat{s}_C(x) = s_C(x) + O(s_C(x)^2).$$

Proof. Because $\nabla L \neq 0$, the denominator in the definition of $\hat{s}_C$ is strictly positive. Hence $\hat{s}_C$ and $L - 1$ have the same sign and the same zero level set. Since $C \cap U = \{L \le 1\}$, this agrees with the sign convention of s_C.

Next let $p \in \partial C$. Differentiating gives

$$\nabla \hat{s}_C = \frac{\nabla L}{\|\nabla L\|} - (L - 1)\frac{\nabla \|\nabla L\|}{\|\nabla L\|^2}.$$

On ∂C we have $L = 1$, and therefore

$$\nabla \hat{s}_C(p) = \frac{\nabla L(p)}{\|\nabla L(p)\|} = n(p) = \nabla s_C(p),$$

where $n(p)$ is the outward unit normal.

Finally, choose ρ smaller than the reach of ∂C. Then every $x \in U_\rho$ has a unique nearest boundary point $p = \pi_{\partial C}(x)$ and can be written as

$$x = p + \delta n(p), \qquad \delta = s_C(x).$$

Set

$$g := \|\nabla L(p)\| > 0.$$

Since $L \in C^2$, Taylor expansion along the normal direction yields

$$L(x) - 1 = g\delta + O(\delta^2).$$

Likewise, since ∇L is C^1,

$$\|\nabla L(x)\| = g + O(|\delta|).$$

After possibly shrinking U_ρ, g is bounded away from zero uniformly on ∂C. Hence

$$\hat{s}_C(x) = \frac{g\delta + O(\delta^2)}{g + O(|\delta|)} = \delta + O(\delta^2) = s_C(x) + O(s_C(x)^2).$$

□

B.2 PROOF OF ALMOST SURE SAFETY

We prove almost-sure safety for a *half-space* obstacle, i.e. an affine per-waypoint constraint. This is the case in which the signed-distance linearization is *exact*, so that the only remaining approximation is the Gaussian moment approximation of the posterior. We assume the data distribution p_{data} is given by

$$p_{\text{data}}(x) = \frac{1}{Z}\exp(\Xi(x)) \cdot \mathcal{N}(x; \mu_0, \Sigma_0),\ \|\nabla\Xi\| \le L, \tag{22}$$

and make the following standing assumptions, which match the deployed algorithm:

(A1) **Isotropic guidance covariance.** The covariance used in the guidance is the Hessian-free estimate $\widehat{V}_t = r_t^2 I$ (Section 3.1).

(A2) **Half-space constraints.** The safe set is a per-waypoint half-space,

$$S = \bigcap_{h=1}^{H} \{x :\ a^\top x_h \ge b\}, \qquad \|a\| = 1,$$

each constraint acting on a distinct coordinate block $x_h \in \mathbb{R}^d$.

(A3) **Tempering.** The guidance uses temperature $\gamma > 1$. (Strict tempering is what makes the leading term in the drift estimate of Step 3 below negative; see the remark after the proof.)

For notational convenience we indicate by the subscript t forward time and by s that the SDE is being run in reversed time, i.e., s goes from T to 0 and will clarify this notation where needed. We recall the relevant stochastic processes:

Base SDE
This process is defined via

$$dX_t = -\frac{1}{2}X_t dt + dW_t,\ p_0 = p_{\text{data}} \tag{23}$$

and the corresponding reverse process $Y_s := X_s$ is given by

$$dY_s = \left[\frac{1}{2}Y_s + \nabla \log p_s(Y_s)\right] ds + dW_s \tag{24}$$

where p_s denotes the density of X_s.

Gaussian SDE
This process is defined via

$$dX_t^G = -\frac{1}{2}X_t^G dt + dW_t,\ p_0 = \mathcal{N}(\mu_0, \Sigma_0) \tag{25}$$

and we denote this path measure by $\mathbb{P}^G$. The corresponding reverse process is given by

$$dY_s^G = \left[\frac{1}{2}Y_s^G + \underbrace{\nabla \log p_s^G(Y_s^G)}_{:=b_s^G(Y_s)}\right] ds + dW_s \tag{26}$$

where again p_t^G denotes the density of X_t^G. Further define the conditional guidance

$$g_s(x) = \nabla_x \log \mathbb{P}^G(X_0 \in S \mid X_s = x), \tag{27}$$

which makes the guided Gaussian process $dY_s^{gG} = \left[\frac{1}{2}Y_s^{gG} + b_s^G(Y_s^{gG}) + g_s(Y_s^{gG})\right] ds + d\tilde{W}_s$ safe, i.e., $\mathbb{P}(Y_0^{gG} \in S) = 1$.

Algorithm SDE
This process is defined via the reverse process

$$dZ_s = \frac{1}{2}Z_s + \underbrace{\nabla \log p_s(Z_s)}_{:=b_s(Z_s)} ds + f_s(Z_s)ds + dW_s \tag{28}$$

where f_s is defined in equation (18) and we denote this path measure by $\mathbb{Q}$.

Before we prove theorem 2 we first state one proposition and two lemmas. The overall idea is to compare the algorithm guidance f_t with the Gaussian guidance g_t: the latter is the exact Doob guidance of the Gaussian reference process and is therefore guaranteed to produce safe samples in that setting, while f_t is what the algorithm actually applies to the (non-Gaussian) data process. The proof proceeds in three steps. Proposition 1 gives the posteriors of both processes in closed form; under the tilted-Gaussian model of the data distribution, the data posterior is an exponential tilt of the Gaussian posterior. Lemma 2 uses this to show that the posterior moments of the two processes are close, with an error that vanishes as $t \to 0$, i.e., as we approach the data end of the diffusion. Lemma 3 follows the Gaussian safety probability $H(t,x) = \mathbb{P}^G(X_0 \in S \mid X_t = x)$ along the algorithm path and shows that its reciprocal $\psi = 1/H$ is a semimartingale whose drift is $\psi \langle g_t, g_t - \delta_t \rangle$, where δ_t is the deviation of the algorithm drift from the Gaussian reference drift; wherever this drift is nonpositive, ψ cannot blow up, and it suffices to know this on a closed region of space–time that the path is forced to enter. The proof of the main theorem then shows that, for a half-space obstacle and $\gamma > 1$, the drift of ψ is nonpositive whenever some waypoint lies sufficiently deep inside the obstacle and the diffusion time is small. Since an unsafe endpoint forces the path into exactly this region, while $\psi \to \infty$ there, unsafe endpoints have probability zero.
We start with the proposition on the posterior distributions.

Proposition 1 (Posterior distributions)**.** *We have*

$$p(X_0 = x_0 \mid X_t = x_t) \propto \mathcal{N}(x_0; \mu_0, \Sigma_0) \cdot \mathcal{N}(x_0; \frac{x_t}{a_t}, r_t^2 I) \cdot \exp(\Xi(x_0)) \tag{29}$$

and

$$p(X_0^G = x_0 \mid X_t^G = x_t) \propto \mathcal{N}(x_0; \mu_0, \Sigma_0) \cdot \mathcal{N}(x_0; \frac{x_t}{a_t}, r_t^2 I) \tag{30}$$

and thus $p(X_0^G = x_0 \mid X_t^G = x_t) \propto \mathcal{N}\left(x_0; \underbrace{\Sigma_t(\Sigma_0^{-1}\mu_0 + r_t^{-2}\frac{x_t}{a_t})}_{:=m_t^G(x_t)}, \Sigma_t\right)$, *where* $\Sigma_t := \left(\Sigma_0^{-1} + r_t^{-2} I\right)^{-1}$

Proof. By Bayes' rule and the transition kernel of the forward process,

$$p_{0|t}(x_0 \mid x_t) \propto p_{\text{data}}(x_0)\, p_{t|0}(x_t \mid x_0).$$

From (2),

$$p_{t|0}(x_t \mid x_0) = \mathcal{N}(x_t; a_t x_0, v_t I).$$

Since $r_t^2 = v_t / a_t^2$,

$$\frac{\|x_t - a_t x_0\|^2}{v_t} = \frac{\|x_0 - x_t/a_t\|^2}{r_t^2},$$

and hence, as a function of x_0,

$$p_{t|0}(x_t \mid x_0) \propto \mathcal{N}\left(x_0; \frac{x_t}{a_t}, r_t^2 I\right).$$

Using

$$p_{\text{data}}(x_0) \propto e^{\Xi(x_0)} \mathcal{N}(x_0; \mu_0, \Sigma_0)$$

gives

$$p_{0|t}(x_0 \mid x_t) \propto e^{\Xi(x_0)} \mathcal{N}(x_0; \mu_0, \Sigma_0) \mathcal{N}\left(x_0; \frac{x_t}{a_t}, r_t^2 I\right),$$

which proves (29).

For the Gaussian reference process, $\Xi \equiv 0$. Completing the square in x_0 gives

$$\begin{aligned} &-\frac{1}{2}(x_0 - \mu_0)^\top \Sigma_0^{-1}(x_0 - \mu_0) - \frac{1}{2r_t^2}\left\|x_0 - \frac{x_t}{a_t}\right\|^2 \\ &\quad = -\frac{1}{2}(x_0 - m_t^G(x_t))^\top \Sigma_t^{-1}(x_0 - m_t^G(x_t)) + c(x_t), \end{aligned}$$

where $c(x_t)$ does not depend on x_0 and

$$\Sigma_t = \left(\Sigma_0^{-1} + r_t^{-2} I\right)^{-1}, \qquad m_t^G(x_t) = \Sigma_t \left(\Sigma_0^{-1}\mu_0 + r_t^{-2}\frac{x_t}{a_t}\right).$$

Therefore

$$X_0^G \mid X_t^G = x_t \sim \mathcal{N}(m_t^G(x_t), \Sigma_t),$$

which proves the claim. □

Next we turn to the moment bounds of the posteriors.

Lemma 2. *There exists $C > 0$ such that when $r_t L\sqrt{d} \le 1$ for any x we have*

$$\|V_t(x) - \Sigma_t\|_{op} \le C L r_t^3, \qquad \|m_t(x) - m_t^G(x)\| \le C r_t^2 L, \qquad \|\nabla m_t(x) - \nabla m_t^G(x)\|_{op} \le \frac{CL}{a_t} r_t \tag{31}$$

Proof. We first bound the posterior means. By Proposition 1, for fixed x_t the posterior $p_{0|t}(\cdot \mid x_t)$ is an exponential tilt of $\mathcal{N}(m_t^G, \Sigma_t)$:

$$p_{0|t}(x_0 \mid x_t) = \frac{1}{Z_t} e^{\Xi(x_0)} \mathcal{N}(x_0; m_t^G, \Sigma_t).$$

Let $\gamma_t(x) = \mathcal{N}(x; m_t^G, \Sigma_t)$. Since

$$(x - m_t^G)\gamma_t(x) = -\Sigma_t \nabla \gamma_t(x),$$

Gaussian integration by parts yields

$$\begin{aligned} m_t - m_t^G &= \frac{1}{Z_t}\int (x - m_t^G) e^{\Xi(x)} \gamma_t(x)\, dx = \frac{\Sigma_t}{Z_t}\int e^{\Xi(x)} \nabla \gamma_t(x)\, dx \\ &= \frac{\Sigma_t}{Z_t}\int e^{\Xi(x)} \nabla \Xi(x) \gamma_t(x)\, dx = \Sigma_t \mathbb{E}_{p_{0|t}(\cdot|x_t)}[\nabla \Xi(X_0)]. \end{aligned}$$

Consequently, using $\|\nabla \Xi\| \le L$ and $\Sigma_t \preceq r_t^2 I$,

$$\|m_t - m_t^G\| \le \|\Sigma_t\|_{\mathrm{op}} L \le r_t^2 L.$$

Let $t \in (0, T]$ and $x_t \in \mathbb{R}^d$. Define $\psi(x) := \Xi(m_t^G(x_t) + \Sigma_t^{1/2} x)$ and let φ_d denote a standard Gaussian. Define a probability density via $\nu(d\xi) \propto \exp(\psi(\xi)) \cdot \varphi_d(\xi) d\xi$. We claim that for $\xi \sim \nu$ we have $X_0 \mid (X_t = x_t) =^d T(\xi) := m_t^G(x_t) + \Sigma_t^{1/2}\xi$, where T is a linear transform with Jacobian $J_T = \Sigma^{1/2}$. It follows that

$$\begin{aligned} \int_A \nu(\xi) d\xi &= \int_A p(T(\xi) \mid X_t = x_t) |\det J_T(\xi)| d\xi \\ &= \int_A \mathcal{N}(m_t^G(x_t) + \Sigma_t^{1/2}\xi; m_t^G, \Sigma_t) \exp\left(\Xi(m_t^G(x_t) + \Sigma_t^{1/2}\xi)\right) (\det \Sigma)^{1/2} d\xi \\ &= \int_A (2\pi)^{-d/2} (\det \Sigma)^{-1/2} \exp\left(-\frac{1}{2}\|\xi\|^2\right) \exp\left(\Xi(m_t^G(x_t) + \Sigma_t^{1/2}\xi)\right) (\det \Sigma)^{1/2} d\xi \\ &= \int_A (2\pi)^{-d/2} \exp\left(-\frac{1}{2}\|\xi\|^2\right) \exp\left(\Xi(m_t^G(x_t) + \Sigma_t^{1/2}\xi)\right) d\xi \end{aligned}$$

and thus the claim is shown. This allows us to use

$$V(x_t) = \Sigma_t^{1/2} \mathrm{Cov}_\nu(\xi) \Sigma_t^{1/2}. \tag{32}$$

Further note that we have the bound $\|\nabla \psi\|_\infty \le \|\Sigma_t^{1/2}\|_{op} \cdot \|\nabla \Xi\| \le r_t L =: \varepsilon$.
Now, we make use of the Taylor expansion of ψ and obtain

$$\psi(\xi) = \psi(0) + \nabla \psi(0) \cdot \xi + R(\xi), \tag{33}$$

with $\nabla R(\xi) = \nabla\psi(\xi) - \nabla\psi(0)$ and $R(0) = 0$. This in turn allows us to compute for the density

$$\exp(\psi(\xi))\varphi_d(\xi) = (2\pi)^{-d/2}\exp(-\frac{1}{2}\|\xi\|^2 + \psi(0) + \nabla\psi(0)\cdot\xi + R(\xi))$$
$$\propto \exp(-\frac{1}{2}\|\xi - \nabla\psi(0)\|^2)\exp(R(\xi)) = \underbrace{\mathcal{N}(\nabla\psi(0), I)}_{:=\nu_1}\exp(R(\xi)).$$

Assume for now that

$$\mathbb{E}_\nu F - \mathbb{E}_{\nu_1} F \leq C\epsilon\sqrt{\mathrm{Var}_{\nu_1} F} \tag{34}$$

holds for a random variable F. We will prove this fact later. This allows us bound for any unit vector u

$$\mathbb{E}_\nu\left(u\cdot\xi\right) - \mathbb{E}_{\nu_1}\left(u\cdot\xi\right) \leq C\epsilon\sqrt{\mathrm{Var}_{\nu_1}\left(u\cdot\xi\right)} \leq C\epsilon \tag{35}$$

as well as

$$\mathbb{E}_\nu\left(u\cdot\xi\right)^2 - \mathbb{E}_{\nu_1}\left(u\cdot\xi\right)^2 \leq C\epsilon\sqrt{\mathrm{Var}_{\nu_1}\left(u\cdot\xi\right)^2} \lesssim C\epsilon \tag{36}$$

because $u\cdot\xi$ is just a tilted Gaussian under the measure ν_1, in particular $\mathcal{N}(u\cdot\nabla\psi(0), 1)$ which has finite moments. Finally, since $u^T\mathrm{Cov}_\nu(\xi)u = \mathrm{Var}_\nu(u\cdot\xi)$ we get using $Y := u\cdot\xi$ for sake of readability

$$\begin{aligned}
\|\mathrm{Cov}_\nu\xi - I\| &= \sup_{\|u\|=1}|u^T\mathrm{Cov}_\nu(\xi)u - u^T I u| = \sup_{\|u\|=1}|\mathrm{Var}_\nu Y - 1| \\
&= \sup_{\|u\|=1}|\mathbb{E}_\nu Y^2 - (\mathbb{E}_\nu Y)^2 - 1| \\
&\leq \sup_{\|u\|=1}|\mathbb{E}_\nu Y^2 - (\mathbb{E}_\nu Y)^2 - \left(\mathbb{E}_{\nu_1} Y^2 - (\mathbb{E}_{\nu_1} Y)^2\right)| \\
&= \sup_{\|u\|=1}\left|\left(\mathbb{E}_\nu Y^2 - \mathbb{E}_{\nu_1} Y^2\right) - \left(\left(\mathbb{E}_\nu Y - \mathbb{E}_{\nu_1} Y\right)\left(\mathbb{E}_\nu Y + \mathbb{E}_{\nu_1} Y\right)\right)\right| \lesssim C\epsilon,
\end{aligned}$$

where the first equality follows from the variational formula for the operator norm of a symmetric matrix $\|A\|_{\mathrm{op}} = \sup_{\|u\|=1}|u^T A u|$, the third line follows from $\mathrm{Var}_{\nu_1}(Y) = 1$ and the last line from the bounds derived in equations (35) and (36). Lastly we note that $\|\Sigma_t\|_{op}^{1/2} \leq r_t$ because $\Sigma_t = \left(\Sigma_0^{-1} + r_t^{-2} I\right)^{-1} \preceq r_t^2 I$ and thus obtain the bound

$$\|V(x_t) - \Sigma_t\|_{op} \leq \|\Sigma_t^{1/2}\|_{op}\cdot\|\mathrm{Cov}_\nu\xi - I\|\cdot\|\Sigma_t^{1/2}\|_{op} \leq CLr_t^3 \tag{37}$$

Since $\nabla_x m(x) = \frac{a_t}{v_t}V = \frac{1}{a_t r_t^2}V$ (Boys et al., 2024), we obtain the claimed bound on the difference of the Jacobian of the means.

All that remains to show is that equation (34) holds. We have

$$\mathbb{E}_\nu F - \mathbb{E}_{\nu_1} F = \int F(x)\frac{1}{\mathbb{E}_{\nu_1}[\exp(R)]}\exp(R(x))\nu_1(dx) - \mathbb{E}_{\nu_1} F = \frac{\mathbb{E}_{\nu_1}[Fe^R] - \mathbb{E}_{\nu_1}[\exp(R)]\mathbb{E}_{\nu_1} F}{\mathbb{E}_{\nu_1}[\exp(R)]}$$
$$= \frac{\mathrm{Cov}_{\nu_1}(F, e^R)}{\mathbb{E}_{\nu_1}[\exp(R)]} = \frac{\sqrt{\mathrm{Var}_{\nu_1}(F)}\sqrt{\mathrm{Var}_{\nu_1}(e^R)}}{\mathbb{E}_{\nu_1}[\exp(R)]}.$$

We can use the Gaussian Poincaré inequality (Boucheron et al., 2013) to bound the variance of e^R by the expected value of the gradient $\nabla(e^R) = \nabla R\cdot e^R$ and obtain

$$\mathrm{Var}_{\nu_1} e^R \leq \mathbb{E}_{\nu_1}[e^{2R}\|\nabla R\|^2] \leq 4\varepsilon^2\mathbb{E}_{\nu_1} e^{2R} \tag{38}$$

In order to control the random variable e^{2R}, we note that $|R(\xi)| \leq 2\epsilon\|\xi\|$ and is thus 2ϵ-Lipschitz. By applying the Gaussian concentration inequality on the 1-Lipschitz function $\xi \mapsto \|\xi\|$ we get

$$\mathbb{E}_{\nu_1}[e^{\lambda\|\xi\|}] \leq e^{\lambda\mathbb{E}\|\xi\|}e^{\lambda^2/2} \leq e^{\lambda\sqrt{\mathbb{E}\|\xi\|^2}+\lambda^2/2} = e^{\lambda\sqrt{d+\|c\|^2}+\lambda^2/2} \tag{39}$$

for $c := \nabla\psi(0) \leq \epsilon$. Due to our requirement we have $\varepsilon\sqrt{d} = r_t L\sqrt{d} \leq 1$ Thus we get with $\lambda = 4\epsilon$

$$\mathbb{E}_{\nu_1}[e^{2R}] \leq \mathbb{E}_{\nu_1}[e^{4\epsilon\|\xi\|}] \leq \exp(4\epsilon\sqrt{d + \|c\|^2} + 8\epsilon^2). \tag{40}$$

Finally, since

$$4\varepsilon\sqrt{d+\|c\|^2} \le 4\varepsilon\sqrt{d+1} \le 4(\varepsilon\sqrt{d}+\varepsilon) \le 4(1+1) = 8 \tag{41}$$

we have

$$\mathbb{E}_\nu F - \mathbb{E}_{\nu_1} F \le \frac{\sqrt{\mathrm{Var}_{\nu_1}(F)}C}{\mathbb{E}_{\nu_1}[\exp(R)]}. \tag{42}$$

By showing that the denominator is bounded from below via

$$\mathbb{E}_{\nu_1} \exp(R) \ge \exp \mathbb{E}_{\nu_1} R \ge e^{-4} \tag{43}$$

because of Jensen's inequality and $|\mathbb{E}_{\nu_1} R| \le 2\varepsilon \mathbb{E}_{\nu_1}\|\xi\| \le 2\varepsilon\sqrt{d+\varepsilon^2} \le 4$ we have

$$\mathbb{E}_\nu F - \mathbb{E}_{\nu_1} F \le C\sqrt{\mathrm{Var}_{\nu_1} F} \tag{44}$$

which concludes the proof. □

The remaining ingredient compares the drift of the algorithm SDE with that of the Gaussian SDE. Define

$$\delta_s := [b_s + f_s] - b_s^G, \tag{45}$$

so that the algorithm process has drift $\frac{1}{2}Z_s + b_s^G(Z_s) + \delta_s(Z_s)$. The idea is to follow the safety probability of the *Gaussian reference* process, $H(s,x) := \mathbb{P}^G(X_0 \in S \mid X_s = x)$, along the *algorithm* path Z. Its reciprocal $\psi := 1/H$ is a barrier: it blows up exactly when the path ends in the unsafe set. Itô's formula shows that the drift of $\psi(Z)$ is $\psi\,\langle g, g-\delta\rangle$, so wherever $\langle g, g-\delta\rangle \le 0$ the barrier is a supermartingale and cannot blow up. It is important that this is only required on a closed region of space–time: on the safe side of the constraint the drift may be positive and its integrability near $s = 0$ depends on the very endpoint whose safety we want to establish; the localization below avoids this circularity, since an unsafe endpoint forces the path into the deep interior of the obstacle at small diffusion times.

Lemma 3 (Localized barrier). *Let $D := Hd$ denote the trajectory dimension. Define, for $t \in (0,T]$ and $x \in \mathbb{R}^D$,*

$$H(t,x) := \mathbb{P}^G\big(X_0 \in S \mid X_t = x\big), \qquad \psi_t(x) := \frac{1}{H(t,x)}, \qquad A_t(x) := \langle g_t(x),\, g_t(x) - \delta_t(x)\rangle, \tag{46}$$

with $g_t = \nabla_x \log H(t,\cdot)$ as in (27) and δ_t as in (45). Let $\mathcal{D} \subset (0,T] \times \mathbb{R}^D$ be relatively closed and suppose that

$$A_t(x) \le 0 \qquad \text{for all } (t,x) \in \mathcal{D}. \tag{47}$$

Then, $\mathbb{Q}$-almost surely on the event

$$E_{\mathcal{D}} := \big\{\exists\, q \in (0,T] :\ (t, Z_t) \in \mathcal{D} \text{ for all } t \in (0,q]\big\}, \tag{48}$$

the limit $\lim_{t\downarrow 0} \psi_t(Z_t)$ exists and is finite.

Proof. Regularity of H. By Proposition 1, $H(t,x) = \int 1_S(x_0)\,\mathcal{N}(x_0; m_t^G(x), \Sigma_t)\,dx_0$, where m_t^G is affine in x with coefficients smooth in t and $\Sigma_t \succ 0$ is smooth in t on $(0,T]$. Hence H is $C^{1,2}$ and strictly positive on $(0,T]\times\mathbb{R}^D$, ψ is $C^{1,2}$ there, and

$$\nabla_x H(t,x) = H(t,x)\, g_t(x). \tag{49}$$

Reversed clock and backward equation. We work in the reversed clock $\tau := T - t$, in which the reverse processes are ordinary (forward-in-τ) Itô diffusions. Write $\tilde{H}(\tau,x) := H(T-\tau,x)$ and let $\beta^G(\tau,x) := \frac{1}{2}x + b_{T-\tau}^G(x)$ denote the full drift of the reverse Gaussian process Y^G in this clock, so that $dY_\tau^G = \beta^G(\tau, Y_\tau^G)\,d\tau + d\tilde{W}_\tau$. Let $(\mathcal{G}_\tau)_{\tau\in[0,T)}$ be the usual augmentation of the filtration generated by the driving Brownian motion $\tilde{W}$ (and the initial value). By the Markov property of Y^G and the tower property, $M_\tau := \mathbb{E}_{\mathbb{P}^G}[1_S(Y_T^G) \mid \mathcal{G}_\tau] = \tilde{H}(\tau, Y_\tau^G)$ is a $(\mathcal{G}_\tau)$-martingale on $[0,T)$. By Itô's formula,

$$d\tilde{H}(\tau, Y_\tau^G) = \Big[\partial_\tau \tilde{H} + \beta^G \cdot \nabla\tilde{H} + \tfrac{1}{2}\Delta\tilde{H}\Big](\tau, Y_\tau^G)\,d\tau + \nabla\tilde{H}(\tau, Y_\tau^G)\cdot d\tilde{W}_\tau,$$

and since a continuous martingale has vanishing finite-variation part, the bracket vanishes at (τ, Y_τ^G), $d\tau \otimes \mathbb{P}^G$-almost everywhere. The bracket is a continuous function of (τ, x) and Y_τ^G has a strictly positive Lebesgue density for every $\tau \in [0, T)$ (it is an Ornstein–Uhlenbeck process started from a Gaussian), hence

$$\partial_\tau \tilde{H}(\tau, x) + \beta^G(\tau, x) \cdot \nabla \tilde{H}(\tau, x) + \tfrac{1}{2}\Delta \tilde{H}(\tau, x) = 0 \qquad \text{for all } (\tau, x) \in [0, T) \times \mathbb{R}^D. \tag{50}$$

Itô decomposition along the algorithm path. Under $\mathbb{Q}$, in the same clock and with respect to the same filtration, Z solves $dZ_\tau = \big[\beta^G(\tau, Z_\tau) + \delta_{T-\tau}(Z_\tau)\big]d\tau + d\tilde{W}_\tau$ by (28) and (45). Itô's formula for $\tilde{H}(\tau, Z_\tau)$, the backward equation (50) and (49) give

$$d\tilde{H}(\tau, Z_\tau) = \delta_{T-\tau}(Z_\tau) \cdot \nabla \tilde{H}(\tau, Z_\tau)\, d\tau + \nabla \tilde{H}(\tau, Z_\tau) \cdot d\tilde{W}_\tau = \tilde{H}\, \langle \delta, g \rangle\, d\tau + \tilde{H}\, g \cdot d\tilde{W}_\tau,$$

where all functions are evaluated at $(T - \tau, Z_\tau)$, and $d\langle \tilde{H}(\cdot, Z)\rangle_\tau = \tilde{H}^2 \|g\|^2 d\tau$. Applying Itô's formula to $N_\tau := 1/\tilde{H}(\tau, Z_\tau) = \psi_{T-\tau}(Z_{T-\tau})$,

$$\begin{aligned} dN_\tau &= -\frac{d\tilde{H}}{\tilde{H}^2} + \frac{d\langle \tilde{H}\rangle_\tau}{\tilde{H}^3} = N_\tau \big(\|g\|^2 - \langle \delta, g \rangle\big) d\tau - N_\tau\, g \cdot d\tilde{W}_\tau \\ &= N_\tau\, A_{T-\tau}(Z_\tau)\, d\tau - N_\tau\, g_{T-\tau}(Z_\tau) \cdot d\tilde{W}_\tau, \qquad \tau \in [0, T), \end{aligned} \tag{51}$$

where the stochastic integral is well defined on every compact subinterval of $[0, T)$ because N and $g \circ Z$ are continuous there.

Localization. Fix a rational $q \in (0, T]$ and define

$$\tau_q := \inf\big\{\tau \in [T - q, T) :\ (T - \tau, Z_\tau) \notin \mathcal{D}\big\}, \qquad \inf \emptyset := T. \tag{52}$$

Since $\mathcal{D}$ is relatively closed and $\tau \mapsto (T - \tau, Z_\tau)$ is continuous, τ_q is the first hitting time of an open set by a continuous adapted process and therefore a $(\mathcal{G}_\tau)$-stopping time. On the stochastic interval $[T - q, \tau_q)$ we have $(T - \tau, Z_\tau) \in \mathcal{D}$, so by (47) the drift in (51) satisfies $NA \le 0$ there. For $n \in \mathbb{N}$ consider

$$N_\tau^{(n)} := \mathbf{1}_{\{N_{T-q} \le n\}}\, N_{\tau \wedge \tau_q}, \qquad \tau \in [T - q, T). \tag{53}$$

The indicator is $\mathcal{G}_{T-q}$-measurable, so $N^{(n)}$ is a nonnegative continuous local supermartingale on $[T - q, T)$ with integrable initial value $N_{T-q}^{(n)} \le n$. By Fatou's lemma applied along a localizing sequence, a nonnegative local supermartingale with integrable initial value is a supermartingale, and by Doob's supermartingale convergence theorem $\lim_{\tau \uparrow T} N_\tau^{(n)}$ exists and is finite almost surely. Since $N_{T-q} = \psi_q(Z_q) < \infty$ almost surely, letting $n \to \infty$ shows that $\lim_{\tau \uparrow T} N_{\tau \wedge \tau_q}$ exists and is finite almost surely. On the event $\{\tau_q = T\}$ we have $N_{\tau \wedge \tau_q} = N_\tau$ for all $\tau \in [T - q, T)$, so $\lim_{t \downarrow 0} \psi_t(Z_t)$ exists and is finite almost surely on $\{\tau_q = T\}$.

Finally, $E_{\mathcal{D}} \subset \bigcup_{q \in \mathbb{Q} \cap (0, T]} \{\tau_q = T\}$: if $(t, Z_t) \in \mathcal{D}$ for all $t \in (0, q]$ with some real q, the same holds for every rational $q' \le q$, and then $\tau_{q'} = T$. The claim follows since a countable union of null sets is null. □

Now we have all the tools to prove the main theorem.

Proof of theorem 2. Throughout, $D = Hd$ is the trajectory dimension (this is the dimension appearing in the hypothesis $r_t L \sqrt{d} \le 1$ of Lemma 2 and in its proof, where $x_t \in \mathbb{R}^D$), $\lambda_0 := \lambda_{\min}(\Sigma_0) > 0$, C is the constant of Lemma 2, and $\gamma > 1$ is the tempering parameter of the theorem. As in the statement of the theorem we assume that the guided reverse SDE (28) has a solution with continuous paths on $[0, T]$, so that $Z_0 = \lim_{t \downarrow 0} Z_t$ is well defined. Time arguments are indicated by t (the value of the reverse time s at which the drift is evaluated); we abbreviate $r = r_t$ and $a = a_t$ when no confusion can arise, and use throughout that $a_t = e^{-t/2} \in [e^{-T/2}, 1]$ and that $r_t = \sqrt{e^t - 1}$ is increasing with $r_t \to 0$ as $t \to 0$.

Step 1: exact structure of the two guidances. By (29) with Σ_0 block diagonal, $\Sigma_t = (\Sigma_0^{-1} + r^{-2} I)^{-1} = \mathrm{diag}(\Sigma_t^{(1)}, \dots, \Sigma_t^{(H)})$ is block diagonal with $\Sigma_t^{(h)} = \big((\Sigma_0^{(h)})^{-1} + r^{-2} I\big)^{-1}$, and $m_t^G(x) = \Sigma_t(\Sigma_0^{-1}\mu_0 + x/(ar^2))$ is affine in x with $\nabla m_t^G = \Sigma_t/(ar^2)$. Under $\mathbb{P}^G$ the blocks

$X_0[h] \mid X_t = x \sim \mathcal{N}(m_t^G(x)[h], \Sigma_t^{(h)})$ are independent, and under (A2) the safe event is the intersection of the per-block events $\{a^\top X_0[h] \geq b\}$. Since $a^\top X_0[h]$ is Gaussian, we obtain the exact factorization

$$H(t,x) = \prod_{h=1}^{H} \Phi\big(z_h(t,x)\big), \qquad z_h(t,x) := \frac{a^\top m_t^G(x)[h] - b}{\sigma_h}, \qquad \sigma_h^2 := a^\top \Sigma_t^{(h)} a. \tag{54}$$

Writing $R := \varphi/\Phi$ for the inverse Mills ratio and $a_h \in \mathbb{R}^D$ for the vector that equals a on block h and vanishes elsewhere, the Gaussian guidance is

$$g_t = \nabla_x \log H = \sum_{h=1}^{H} g^{(h)}, \qquad g^{(h)} := R(z_h)\, u_h, \qquad u_h := \nabla_x z_h = \frac{\Sigma_t a_h}{a\, r^2 \sigma_h}. \tag{55}$$

Because Σ_t is block diagonal, u_h is supported on block h, so $\langle u_h, u_{h'} \rangle = 0$ for $h \neq h'$ and $\|g_t\|^2 = \sum_h R(z_h)^2 \|u_h\|^2$.

For the half-space C_0 the signed distance $s_{C_0}(y) = a^\top y - b$ is affine, so the linearization (15) is exact. Under (A1) the guidance (18) evaluated with the exact score therefore reads

$$f_t = \gamma \sum_{h=1}^{H} R(\hat{z}_h)\, w_h, \qquad \hat{z}_h := \frac{a^\top m_t(x)[h] - b}{r}, \qquad w_h := \nabla_x \hat{z}_h = \frac{(\nabla m_t)^\top a_h}{r} = \frac{V_t a_h}{a\, r^3}, \tag{56}$$

where we used the Tweedie identity $\nabla m_t = V_t/(a_t r_t^2)$, which is exact for the exact score, and the symmetry of V_t. Thus the only difference between f_t/γ and g_t lies in the posterior moments (m_t, V_t versus m_t^G, Σ_t) and in the scale (r versus σ_h).

Step 2: elementary bounds. *(i) Inverse Mills ratio.* The function $R = \varphi/\Phi$ satisfies $R'(z) = -R(z)\big(z + R(z)\big)$ and $0 < R(z)\big(z + R(z)\big) < 1$ for all $z \in \mathbb{R}$ (these are the classical two-sided Mills-ratio inequalities). Consequently R is positive, decreasing and 1-Lipschitz, and $R(z) \geq -z$ for $z \leq 0$. Moreover $R(0) = \sqrt{2/\pi} \in (3/4, 1)$, so $R \leq 1$ on $[0, \infty)$, $R \geq 3/4$ on $(-\infty, 0]$, and, by the Lipschitz bound, $R(z) \leq R(0) + 1 < 2$ for $z \geq -1$.

(ii) Covariances and the vectors u_h. Assume $r^2 \leq \lambda_0/2$. The eigenvalues of Σ_t are $\lambda r^2/(\lambda + r^2)$ with λ ranging over the eigenvalues of Σ_0, and hence

$$r^2\Big(1 - \frac{r^2}{\lambda_0}\Big) I \preceq \Sigma_t \preceq r^2 I, \qquad \text{in particular } \Sigma_t \succeq \tfrac{r^2}{2} I. \tag{57}$$

Therefore $\sigma_h^2 \in [r^2(1 - r^2/\lambda_0), r^2]$, so that with $\kappa_h := 1 - \sigma_h/r$ we have $0 \leq \kappa_h \leq r^2/\lambda_0$ (using $\sqrt{1-x} \geq 1 - x$) and $\sigma_h \geq r/\sqrt{2}$. For u_h this gives

$$\|u_h\| \leq \frac{\|\Sigma_t^{1/2}\|_{\mathrm{op}}\, \|\Sigma_t^{1/2} a_h\|}{a\, r^2 \sigma_h} = \frac{r \sigma_h}{a\, r^2 \sigma_h} = \frac{1}{a\, r} \leq \frac{e^{T/2}}{r} \tag{58}$$

and

$$\|u_h\| \geq \langle a_h, u_h \rangle = \frac{\sigma_h^2}{a\, r^2 \sigma_h} = \frac{\sigma_h}{a\, r} \geq \frac{1}{\sqrt{2}\, r} \geq \frac{1}{2r}. \tag{59}$$

(iii) Direction error. Assume in addition $r \leq 1$ and $rL\sqrt{D} \leq 1$, so that Lemma 2 applies and $\|V_t - \Sigma_t\|_{\mathrm{op}} \leq C L r^3$. From (55) and (56),

$$w_h - u_h = \frac{1}{a\, r^2}\Big[\frac{(V_t - \Sigma_t) a_h}{r} + \Sigma_t a_h \Big(\frac{1}{r} - \frac{1}{\sigma_h}\Big)\Big], \qquad \Big|\frac{1}{r} - \frac{1}{\sigma_h}\Big| = \frac{\kappa_h}{\sigma_h} \leq \frac{r^2/\lambda_0}{r/\sqrt{2}} = \frac{\sqrt{2}\, r}{\lambda_0}.$$

Using $\|\Sigma_t a_h\| \leq r^2$ we get

$$\|w_h - u_h\| \leq \frac{1}{a\, r^2}\Big(C L r^2 + \frac{\sqrt{2}\, r^3}{\lambda_0}\Big) = \frac{C L + \sqrt{2}\, r/\lambda_0}{a} \leq C_1, \qquad C_1 := e^{T/2}\Big(CL + \frac{\sqrt{2}}{\lambda_0}\Big). \tag{60}$$

By the block-orthogonality of the u_h this yields, for all h, h',

$$\langle u_h, w_h \rangle \geq \|u_h\|^2 - C_1 \|u_h\|, \qquad |\langle u_h, w_{h'} \rangle| = |\langle u_h, w_{h'} - u_{h'} \rangle| \leq C_1 \|u_h\| \quad (h' \neq h). \tag{61}$$

(iv) Depth error. The mean bound of Lemma 2, whose proof gives $\|m_t - m_t^G\| \le r^2 L$ without any condition on r, yields

$$\hat{z}_h = \frac{\sigma_h}{r} z_h + \frac{a^\top (m_t - m_t^G)[h]}{r} = (1-\kappa_h) z_h + e_h, \qquad |e_h| \le rL. \tag{62}$$

If $z_h \le 0$, then by (i) $|z_h| \le R(z_h)$ and $R(z_h) \ge 3/4$, so $|\hat{z}_h - z_h| \le \kappa_h |z_h| + |e_h| \le \theta R(z_h)$ with $\theta := r^2/\lambda_0 + \frac{4}{3} rL$, and the 1-Lipschitz property gives

$$(1-\theta) R(z_h) \le R(\hat{z}_h) \le (1+\theta) R(z_h) \qquad (z_h \le 0). \tag{63}$$

If $z_h > 0$ and $rL \le 1$, then $\hat{z}_h \ge -|e_h| \ge -1$ and therefore

$$0 \le R(\hat{z}_h) \le 2 \qquad (z_h > 0). \tag{64}$$

(v) Tilt term. By Tweedie's formula, $\nabla \log p_t(x) = (a_t m_t(x) - x)/v_t$ and $\nabla \log p_t^G(x) = (a_t m_t^G(x) - x)/v_t$, hence, by the identity $m_t - m_t^G = \Sigma_t \mathbb{E}_{p_{0|t}(\cdot|x)}[\nabla \Xi(X_0)]$ from the proof of Lemma 2 and $\Sigma_t \preceq r^2 I$,

$$b_t - b_t^G = \frac{m_t - m_t^G}{a\, r^2} = \frac{\Sigma_t\, \mathbb{E}_{p_{0|t}(\cdot|x)}[\nabla \Xi(X_0)]}{a\, r^2}, \qquad \|b_t - b_t^G\| \le \frac{L}{a} \le L e^{T/2}. \tag{65}$$

Step 3: the barrier drift is nonpositive deep inside the obstacle. Choose $t_0 \in (0, T]$ so small that $r_0 := r_{t_0}$ satisfies

$$\begin{aligned} &\text{(T1)}\ \ r_0 \le 1 \ \text{ and } \ r_0^2 \le \lambda_0/2; \qquad \text{(T2)}\ \ r_0 L \sqrt{D} \le 1; \\ &\text{(T3)}\ \ \frac{r_0^2}{\lambda_0} + \frac{4}{3} r_0 L \le \frac{\gamma - 1}{2\gamma}; \qquad \text{(T4)}\ \ r_0 \le \frac{\gamma - 1}{128\, \gamma\, C_1 e^{T/2} H}. \end{aligned} \tag{66}$$

Such a t_0 exists because $r_t \downarrow 0$ as $t \downarrow 0$ and $\gamma > 1$. Since $r_t \le r_0$ for $t \le t_0$, all bounds of Step 2 are available on $(0, t_0]$ (note that (T2) implies $rL \le 1$), and $\theta \le (\gamma - 1)/(2\gamma) < 1$. Set

$$K := H e^T + 4\gamma C_1 e^{T/2} H^2 + \frac{8 L^2 e^{2T}}{\gamma - 1} + L e^T \sqrt{H}, \qquad M := 4 \sqrt{\frac{K}{\gamma - 1}}. \tag{67}$$

We claim that

$$A_t(x) = \langle g_t(x), g_t(x) - \delta_t(x) \rangle \le 0 \qquad \text{whenever } t \in (0, t_0] \text{ and } \min_h z_h(t,x) \le -M. \tag{68}$$

Fix such (t, x) and abbreviate $R_h := R(z_h)$, $\hat{R}_h := R(\hat{z}_h)$, $I_- := \{h : z_h \le 0\}$, $I_+ := \{1, \dots, H\} \setminus I_-$, $P := \sum_{h \in I_-} R_h$ and $Q := \sum_{h \in I_-} R_h^2$; Cauchy–Schwarz gives $P^2 \le HQ$. Since $\delta_t = b_t - b_t^G + f_t$,

$$A_t = \|g_t\|^2 - \langle g_t, f_t \rangle - \langle g_t, b_t - b_t^G \rangle.$$

By (55), (56) and (61), and because $R_h, \hat{R}_h \ge 0$,

$$\langle g_t, f_t \rangle = \gamma \sum_{h,h'} R_h \hat{R}_{h'} \langle u_h, w_{h'} \rangle \ge \gamma \sum_h R_h \hat{R}_h \|u_h\|^2 - \gamma C_1 \Big(\sum_h R_h \|u_h\| \Big) \Big(\sum_{h'} \hat{R}_{h'} \Big),$$

so that, with (65),

$$A_t \le \underbrace{\sum_h \|u_h\|^2 R_h \big(R_h - \gamma \hat{R}_h \big)}_{=:T_1} + \underbrace{\gamma C_1 \Big(\sum_h R_h \|u_h\| \Big) \Big(\sum_{h'} \hat{R}_{h'} \Big)}_{=:T_2} + \underbrace{L e^{T/2} \|g_t\|}_{=:T_3}. \tag{69}$$

Bound on T_1. For $h \in I_-$, (63) and (T3) give $R_h - \gamma \hat{R}_h \le R_h \big(1 - \gamma(1-\theta)\big) \le -\frac{\gamma - 1}{2} R_h$, and with the lower bound in (58) the corresponding summand is at most $-\frac{\gamma - 1}{8 r^2} R_h^2$. For $h \in I_+$ we drop the nonpositive part $-\gamma \hat{R}_h$ and use $R_h \le 1$ together with the upper bound in (58) to bound the summand by e^T / r^2. Hence

$$T_1 \le -\frac{\gamma - 1}{8 r^2}\, Q + \frac{H e^T}{r^2}.$$

Bound on T_2. By (58) and $R_h \le 1$ on I_+, $\sum_h R_h\|u_h\| \le \frac{e^{T/2}}{r}(P+H)$; by (63), (64) and $\theta \le 1$, $\sum_{h'} \hat{R}_{h'} \le (1+\theta)P + 2H \le 2(P+H)$. Thus, using $(P+H)^2 \le 2(P^2+H^2)$, $P^2 \le HQ$ and (T4),

$$T_2 \le \frac{2\gamma C_1 e^{T/2}}{r}(P+H)^2 \le \frac{4\gamma C_1 e^{T/2} H}{r}\,Q + \frac{4\gamma C_1 e^{T/2} H^2}{r} \le \frac{\gamma-1}{32r^2}\,Q + \frac{4\gamma C_1 e^{T/2} H^2}{r}.$$

Bound on T_3. By (58) and $R_h \le 1$ on I_+, $\|g_t\|^2 = \sum_h R_h^2\|u_h\|^2 \le \frac{e^T}{r^2}(Q+H)$, so $\|g_t\| \le \frac{e^{T/2}}{r}(\sqrt{Q}+\sqrt{H})$. Applying $xy \le \frac{\varepsilon}{2}x^2 + \frac{1}{2\varepsilon}y^2$ with $x = \sqrt{Q}/r$, $y = Le^T$ and $\varepsilon = (\gamma-1)/16$,

$$T_3 \le \frac{Le^T\sqrt{Q}}{r} + \frac{Le^T\sqrt{H}}{r} \le \frac{\gamma-1}{32r^2}\,Q + \frac{8L^2e^{2T}}{\gamma-1} + \frac{Le^T\sqrt{H}}{r}.$$

Assembly. Summing the three bounds and using $r \le 1$ (so that $1/r \le 1/r^2$ and $1 \le 1/r^2$),

$$A_t \le -\frac{\gamma-1}{16r^2}\,Q + \frac{1}{r^2}\Big(He^T + 4\gamma C_1 e^{T/2}H^2\,r + \frac{8L^2e^{2T}}{\gamma-1}\,r^2 + Le^T\sqrt{H}\,r\Big) \le \frac{1}{r^2}\Big(-\frac{\gamma-1}{16}\,Q + K\Big). \tag{70}$$

If $\min_h z_h \le -M$, there is $h^* \in I_-$ with $R_{h^*} \ge |z_{h^*}| \ge M$ by (i), hence $Q \ge M^2 = 16K/(\gamma-1)$ and (70) gives $A_t \le 0$. This proves (68).

Note that the estimate (70) makes explicit why the quadratic-in-R term $\sum_h R_h^2\|u_h\|^2 \sim \sum_h R_h^2/r^2$ is harmless: for $\gamma > 1$ it enters with a negative sign on I_-, and it dominates every other contribution, all of which are $O(R_h/r)$ or $O(1/r^2)$, once some R_h is large. For $\gamma = 1$ the leading term vanishes and the argument gives nothing.

Step 4: conclusion. Let

$$\mathcal{D} := \Big\{(t,x) \in (0,t_0] \times \mathbb{R}^D : \min_h z_h(t,x) \le -M\Big\}.$$

Since each z_h is continuous on $(0,T] \times \mathbb{R}^D$, $\mathcal{D}$ is relatively closed, and by (68) the hypothesis of Lemma 3 holds for $\mathcal{D}$.

Suppose, for contradiction, that $\mathbb{Q}(Z_0 \notin S) > 0$. On the event $\{Z_0 \notin S\}$ there is an index h with $a^\top Z_0[h] < b$. Because Z has continuous paths, $Z_t \to Z_0$ as $t \downarrow 0$; because

$$m_t^G(x) = \big(I + r_t^2\Sigma_0^{-1}\big)^{-1}\Big(\frac{x}{a_t} + r_t^2\Sigma_0^{-1}\mu_0\Big) \longrightarrow x \qquad \text{locally uniformly in } x,$$

we get $a^\top m_t^G(Z_t)[h] - b \to a^\top Z_0[h] - b < 0$; and $\sigma_h \downarrow 0$. Hence $z_h(t,Z_t) \to -\infty$, and there is $q \in (0,t_0]$ with $\min_{h'} z_{h'}(t,Z_t) \le -M$ for all $t \in (0,q]$. In other words, $\{Z_0 \notin S\} \subset E_{\mathcal{D}}$ up to a null set. By Lemma 3, $\lim_{t\downarrow 0}\psi_t(Z_t)$ exists and is finite almost surely on $\{Z_0 \notin S\}$. But by (54),

$$\psi_t(Z_t) = \prod_{h'=1}^{H} \Phi\big(z_{h'}(t,Z_t)\big)^{-1} \;\ge\; \Phi\big(z_h(t,Z_t)\big)^{-1} \longrightarrow \infty \qquad (t \downarrow 0),$$

since $\Phi(z) \to 0$ as $z \to -\infty$. This contradiction shows $\mathbb{Q}(Z_0 \notin S) = 0$, i.e. $\mathbb{P}(z_0 \in S) = 1$. □

Remark (On the role of the assumptions). The proof uses the guidance f_t only on the near-data window $(0,t_0]$, where t_0 is determined by (66); on $(t_0,T]$ it only uses that Z_{t_0} is almost surely finite. Consequently the conclusion of Theorem 2 is unaffected by switching the guidance off for $t > t_{\max}$ as in (18), provided $t_{\max} \ge t_0$. Tempering with $\gamma > 1$ is essential, since the sign of the leading term in Step 3 is $-(\gamma-1)$. The half-space assumption (A2) is used twice: it makes the SDF linearization exact and, together with the block structure of Σ_0, it yields the exact product form (54), which is what allows the per-waypoint decomposition of g_t. The tilt Ξ enters only through Lemma 2 and through the bounded tilt term (65). Finally, the argument concerns the continuous-time guided SDE with the exact score; it makes no statement about time discretization or score-estimation error.

C EXPERIMENTS

C.1 EXPERIMENTAL DETAILS

The model architectures, training objectives, and baseline-specific inference procedures follow Diffuser (Janner et al., 2022), SafeDiffuser (also for Classifier Guidance and Truncate implementations) (Xiao et al., 2023), SafeFlowMatcher (also for FlowMatcher implementation) (Yang et al., 2026) and SAD-Flower (Huang et al., 2026). Diffuser, all three SafeDiffuser variants, and ANCHOR use the same trained score-diffusion checkpoint. FlowMatcher and all SafeFlowMatcher variants use the same trained flow-matching checkpoint. For each benchmark and model family, we train one checkpoint and evaluate it with three different sampling seeds. Diffuser, the SafeDiffuser variants, and ANCHOR use 256 reverse-diffusion steps; the SafeDiffuser correction and ANCHOR guidance are applied within these steps rather than through an additional correction horizon. FlowMatcher uses 256 Euler steps. SafeFlowMatcher uses T^p prediction steps followed by $T^c = 256$ correction steps: the default configuration therefore performs $1 + 256 = 257$ updates. As discussed in the main text, the single-step prediction produces mode averaging on our multimodal benchmarks. We therefore increase the prediction budget to $T^p = 16$, resulting in $16 + 256 = 272$ updates in total. For SAD-Flower, we retain the training objective and budget but replace the original eight-layer transformer with a conditional U-Net. On Maze2D, we use the analytical dynamics model provided in the released codebase. We do not evaluate SAD-Flower on locomotion or Kuka because we could not identify scripts specifying the dynamics models used for these tasks. To match the single-sample evaluation protocol, we evaluate the first generated trajectory rather than selecting the highest-value plan from 32 candidates. We retain the released integration grid of 1,000 points, corresponding to 999 Euler updates, with correction active during the final 29 updates ($t > 0.97$). The architectural and sample-selection changes are intended to improve comparability across methods, while preserving SAD-Flower's released integration and correction schedules. All reported experiments use one NVIDIA H100 GPU.

ANCHOR configuration. The ANCHOR configurations used in the main comparisons are listed in Table 4. Following the Gaussian-posterior approximation in Section 3.1, Hessian zero gives an isotropic Tweedie covariance, while the stopped-model gradient avoids differentiating through the score network (see Section D.3 for a detailed discussion). Restricting guidance to $t \leq t_{\max}$ avoids the high-noise regime, where the broad denoising posterior makes the local SDF approximation less accurate, as discussed in Section 3.3.

Benchmark	Posterior covariance	Tweedie gradient	Time weight m_t	t_{max}
Sequential synthetic	Hessian zero	Stopped model	$1/\sigma_t$	0.90
Multimodal Maze: Medium/Hard	Hessian zero	Through model	both	0.98
Hopper/Walker2D	Hessian zero	Stopped model	$1/\sigma_t$	0.30
Kuka	Hessian zero	Stopped model	1	0.90

Table 4: ANCHOR hyperparameters for the reported configurations.

Evaluation Metrics On the Gaussian-mixture benchmarks, we measure distributional fidelity using Wasserstein-1 (W_1) in one dimension and sliced Wasserstein-1 (SW_1) in two dimensions. We additionally report safety and boundary excess: the generated probability mass within distance 0.05 of the constraint boundary minus the corresponding mass under the safe conditional target, expressed in percentage points. For sequential synthetic navigation, we report trajectory-level SW_1, safety, and boundary safety with margin 0.01 from the added obstacle; safety also requires avoiding the existing walls.

On the simulation benchmarks, we report cumulative simulator reward for Multimodal Maze and Kuka, and D4RL-normalized cumulative return for Hopper and Walker2d. We distinguish planned safety, evaluated on the completed first plan, from execution safety, evaluated on the recorded simulator rollout. Multimodal Maze safety requires strictly positive clearance along every trajectory segment, accounting for the robot's radius, added obstacles, and existing walls. Locomotion and Kuka safety require nonnegative clearance from the roof and joint limits, respectively.

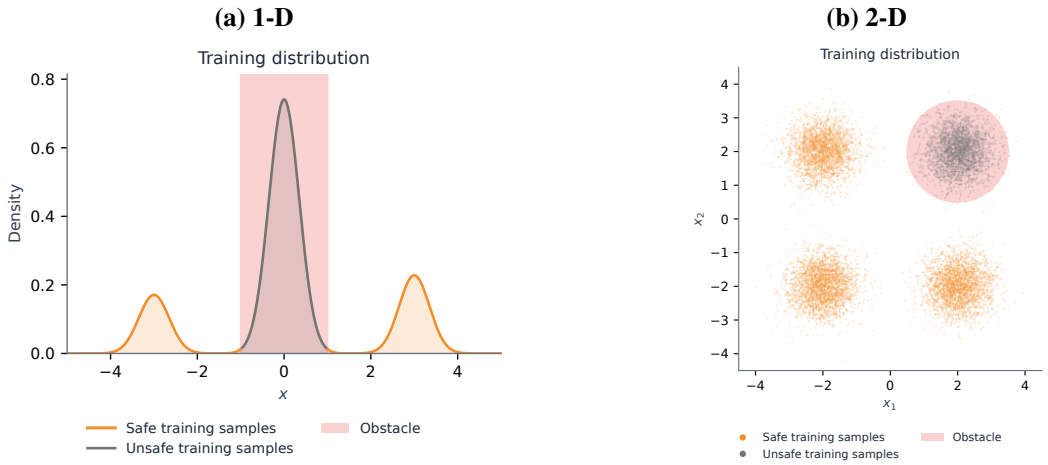


Figure 4: (a) 1-D and (b) 2-D Gaussian mixture training distributions and safe/unsafe sets.

Boundary safety requires minimum clearance greater than $b = 0.01\,\mathrm{m}$ for Multimodal Maze and $b \in \{0.01, 0.05\}\,\mathrm{m}$ for both Hopper and Walker2d. For Kuka, we use relative margins of 5% and 10% against the tightened 90% joint limits. Specifically, B-Safe$(100b\%)$ requires

$$\min_{\tau,j} \frac{0.9L_j - |q_{\tau,j}|}{0.9L_j} > b, \qquad b \in \{0.05, 0.10\},$$

where L_j is the original physical joint half-range and $q_{\tau,j}$ is the planned joint position.

Where reported for Maze2D, we additionally use the discontinuity-based trap metric following Yang et al. (2026): the percentage of plans containing an adjacent-waypoint displacement greater than 0.2 in normalized position coordinates.

C.2 Experimental setup

Gaussian mixtures. The one-dimensional distribution has three modes centered at -3, 0, and 3, with respective weights 0.15, 0.65, and 0.20 and common standard deviation 0.35. Its unsafe set is the interval $[-1, 1]$, which intersects the central mode. The two-dimensional distribution has four equally weighted modes centered at $(\pm 2, \pm 2)$, each with isotropic standard deviation 0.5. Its unsafe set is a disk of radius 1.5 centered at $(2, 2)$, and therefore intersects one of the four modes. Each model is trained on $100{,}000$ independent samples from the corresponding unconditional mixture. The data is shown in Fig. 4.

For every method and sampling seed, we generate $10{,}000$ samples. An independent reference of $10{,}000$ samples from the exact safe conditional is obtained by rejection sampling from the known mixture. We compute W_1 in one dimension and SW_1 in two dimensions between all generated samples and this safe reference; SW_1 averages 128 random one-dimensional projections. Safety is the percentage satisfying $s(x) > 0$. Boundary excess at distance 0.05 is the percentage satisfying $|s(x)| \le 0.05$ minus the corresponding percentage under the exact safe conditional. All values are mean $\pm$ sample standard deviation over the three sampling seeds.

Sequential synthetic navigation. We construct state-only trajectories with horizon $H = 60$ connecting a common start and goal through three distinct corridors. The lower, central, and upper routes have respective probabilities 0.25, 0.50, and 0.25, with within-route variation introduced by perturbing the waypoints while preserving the endpoints and avoiding the existing walls. At test time, an additional obstacle blocks the central route, leaving the two outer routes available (Fig. 5). We generate 256 trajectories per reported sampling seed. Distributional fidelity is measured by trajectory-level SW_1, using 2,048 safe-reference trajectories and 32 random projections. Safety requires positive waypoint clearance from the added obstacle and no trajectory segment intersecting an existing wall. B-Safe additionally requires clearance greater than 0.01 from the added obstacle.

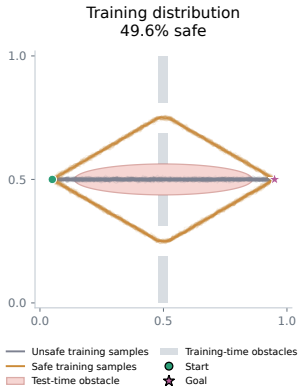


Figure 5: Sequential synthetic navigation task: training distributions and safe/unsafe sets.

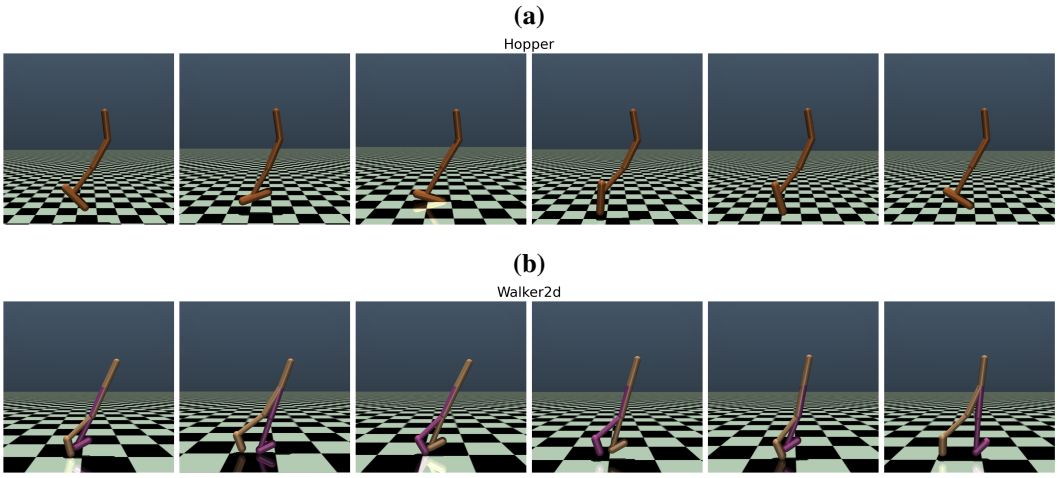


Figure 6: Rendered frames at different timesteps from demonstration rollouts in (a) Hopper and (b) Walker2d.

Multimodal Maze navigation. We use the D4RL Maze2D simulator (Fu et al., 2020), conditioning generation on the initial state and goal. The recorded Medium and Hard datasets, described in Section 4.2, contain 12,000 and 24,000 demonstrations, respectively, with horizons $H = 384$ and $H = 768$. Each method generates 64 plans for each of three sampling seeds, giving 192 planning attempts. We generate one plan per trial and execute it using the original Maze2D position–velocity tracking controller, without replanning. Reward is accumulated over the first $H - 1$ actions, giving 383 actions for Medium and 767 for Hard. Planned safety checks every segment of the generated trajectory against both the added obstacles and existing maze walls, accounting for the robot's $0.1\,\mathrm{m}$ radius. B-Safe requires an additional $0.01\,\mathrm{m}$ clearance. Execution safety is evaluated on the simulator rollout. Rewards are reported as mean $\pm$ population standard deviation.

Locomotion: Hopper and Walker2D. We train on the D4RL `hopper-medium-expert-v2` and `walker2d-medium-expert-v2` datasets (Fu et al., 2020). The models jointly generate state and action sequences with horizon $H = 600$, conditioned on the current observation. At each control step, we execute the first predicted action and replan from the resulting simulator state. Episodes end upon environment termination or after 500 actions. We evaluate 100 episodes per method using paired environment and sampling seeds, and report D4RL-normalized cumulative return. The test-time root-height ceilings are $1.60\,\mathrm{m}$ for Hopper and $1.40\,\mathrm{m}$ for Walker2D. Planned safety is measured on the completed first plan, while execution safety covers the rollout. Planned boundary safety uses roof-clearance margins of 1 and 5 cm.

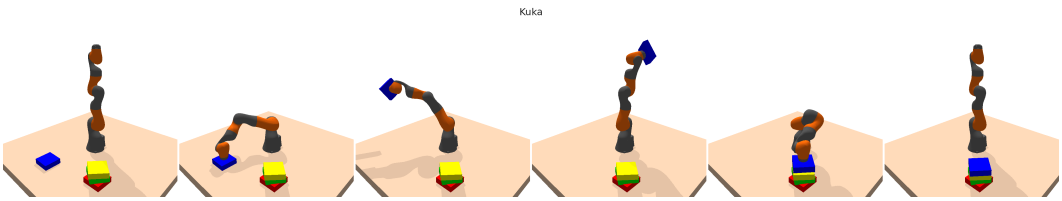


Figure 7: KUKA simulator: rendered images at different timesteps from a demonstration rollout.

	1-D Gaussian mixture		2-D Gaussian mixture	
Method	$W_1 \downarrow$	Safe [%] $\uparrow$	$\mathrm{SW}_1 \downarrow$	Safe [%] $\uparrow$
Diffuser (unguided)	1.776 ± 0.007	35.1 ± 0.3	0.664 ± 0.004	75.1 ± 0.1
Truncate	0.610 ± 0.015	93.7 ± 0.2	0.345 ± 0.009	98.1 ± 0.0
Classifier guidance	1.756 ± 0.005	36.4 ± 0.3	0.653 ± 0.004	75.8 ± 0.1
SafeDiffuser–RoS	0.629 ± 0.029	100.0 ± 0.0	0.332 ± 0.009	100.0 ± 0.0
SafeDiffuser–ReS	5.118 ± 1.032	100.0 ± 0.0	0.614 ± 0.010	100.0 ± 0.0
SafeDiffuser–TVS	1.453 ± 0.027	35.7 ± 0.6	0.602 ± 0.004	75.2 ± 0.2
FlowMatcher (unguided)	1.766 ± 0.007	35.2 ± 0.3	0.652 ± 0.027	75.6 ± 0.9
SafeFlowMatcher ($T_p = 1$, default)	2.843 ± 0.000	0.2 ± 0.0	1.521 ± 0.002	100.0 ± 0.0
SafeFlowMatcher ($T_p = 16$)	0.855 ± 0.009	100.0 ± 0.0	0.575 ± 0.026	100.0 ± 0.0
ANCHOR, Hessian zero, constant	0.237 ± 0.002	100.0 ± 0.0	0.184 ± 0.017	100.0 ± 0.0
ANCHOR, Hessian zero, $1/\sigma_t$	$\mathbf{0.059 \pm 0.005}$	100.0 ± 0.0	0.200 ± 0.018	100.0 ± 0.0
ANCHOR, full Hessian, constant	0.121 ± 0.019	100.0 ± 0.0	$\mathbf{0.071 \pm 0.007}$	100.0 ± 0.0

Table 5: Gaussian-mixture results. W_1 / SW_1: (sliced) Wasserstein-1 distance between all generated samples and the exact safe conditional; Safe: fraction of samples with positive signed distance. Mean±sample SD over three sampling seeds with 10,000 samples each. All diffusion-based methods share one score checkpoint per task and use 256 reverse steps; the flow-matching methods share one flow checkpoint, with T_p prediction and 256 correction steps All ANCHOR rows use the guidance window $t_{\max} = 0.9$. Hessian zero denotes the isotropic covariance $r_t^2 I$ with the stopped-model gradient; full Hessian the exact Tweedie covariance (11) with the probit differentiated through the Tweedie mean and covariance.

KUKA. We use the reference block-stacking dataset containing 11,649 demonstrations and generate state-only plans with horizon $H = 128$, conditioned on the initial robot and block configuration. Each trial executes one plan without replanning. The predicted joint positions are tracked using the reference PyBullet position controller, with 10 physics steps per target; predicted attachment states control block grasping and release. We evaluate 100 episodes per method and report cumulative stacking reward, including the final release. Safety requires all seven joints to remain within 90% of their original limits. Relative boundary margins of 5% and 10% are measured against these tightened limits, corresponding to 85.5% and 81% of the original joint half-ranges, respectively. Planned safety is evaluated over the generated states, and execution safety over the recorded simulator states.

D Additional Results

D.1 Performance on Synthetic Benchmarks

On the Gaussian mixtures (Table 5), SafeDiffuser–RoS and ReS achieve 100% safety but retain excess boundary mass and substantially larger distributional errors than ANCHOR. ANCHOR with inverse-standard-deviation weighting instead achieves 100% safety without excess boundary accumulation.

Sequential (Table 6) exposes a different failure: local obstacle avoidance does not reliably produce safe trajectories. The added obstacle blocks the central corridor, and locally deforming trajectories around it can cause collisions with the surrounding walls. Safety therefore requires rerouting through the remaining safe trajectory modes. No baseline exceeds 52.3% safety, whereas ANCHOR achieves

Method	$SW_1 \downarrow$ ($\times 10^{-3}$)	Safe [%]↑	B-Safe(0.01) [%]↑
Diffuser	39.5±3.2	50.9±3.9	50.9±3.9
Truncate	37.9±2.7	51.4±3.8	51.3±3.8
CG	59.8±19.3	50.5±3.8	50.5±3.8
SafeDiffuser–RoS	60.2±0.1	3.3±1.5	0.0±0.0
SafeDiffuser–ReS	38.0±3.6	52.3±4.2	50.4±3.7
SafeDiffuser–TVS	32.2±2.1	50.0±3.7	50.0±3.7
Flow Matching	42.2±3.9	48.6±4.9	48.6±4.9
FlowMatcher	83.67±0.04	0.0±0.0	0.0±0.0
SFM ($T_p = 1$)	66.5±0.7	0.0±0.0	0.0±0.0
SFM ($T_p = 16$)	46.1±4.5	46.6±5.2	46.6±5.2
ANCHOR (inverse std)	10.7±1.8	99.9±0.2	99.9±0.2
ANCHOR (constant)	3.2±1.7	100.0±0.0	100.0±0.0

Table 6: Sequential three-mode benchmark: 256 trajectories per sampling seed. Values are mean ± sample standard deviation over three sampling seeds. SW_1 is reported in units of 10^{-3} and uses 2,048 safe-reference trajectories and 32 projections. Safe requires positive red-obstacle waypoint clearance and no polyline–wall collision. B-Safe(0.01) additionally requires 0.01 clearance from the red obstacle only. CG denotes classifier guidance; SFM denotes SafeFlowMatcher, with T_p prediction steps. Both ANCHOR variants use Hessian-zero covariance, stopped-model gradients, $t_{\max} = 0.9$, and relative norm cap 2, with time weights specified in parentheses.

	Multimodal Maze: Medium				**Multimodal Maze: Hard**				
Method	Safe/B-Safe(%) ↑	Reward ↑	Exec.(%) ↑	Trap(%) ↓	Safe(%) ↑	B-Safe(%) ↑	Reward ↑	Exec.(%) ↑	Trap(%) ↓
Diffuser	37.0	28.30±37.23	37.0	0.0	4.7	4.2	20.71±46.67	11.5	0.0
Flow Matching	29.2	21.58±33.81	29.7	0.0	3.6	3.6	16.88±44.22	8.3	0.0
Truncate	36.5	27.62±36.78	37.5	63.5	5.7	5.2	20.51±46.04	11.5	65.1
Classifier guidance	34.4	26.57±36.69	34.9	25.5	4.7	4.2	21.49±47.56	13.0	10.9
SafeDiffuser–RoS	0.0	33.16±39.54	41.7	100.0			Infeasible		
SafeDiffuser–ReS	37.0	28.22±37.13	37.5	63.0			Infeasible		
SafeDiffuser–TVS	43.8	33.10±37.88	44.3	56.2	1.6	1.6	16.06±40.94	8.9	81.2
FlowMatcher	0.0	0.00±0.00	1.0	0.0	0.0	0.0	7.35±13.37	0.5	0.0
SafeFlowMatcher	0.0	0.00±0.00	1.6	0.0	0.0	0.0	2.78±8.61	0.0	28.6
SAD-Flower	4.7	27.95±38.46	22.4	84.4	0.0	0.0	4.35±21.22	0.5	99.0
ANCHOR (constant)	**94.3**	**70.26±18.89**	**94.3**	0.0	22.9	19.8	**77.73±60.05**	39.1	0.0

Table 7: Full Multimodal Maze results with trap rates over 192 planning attempts. Trap is the percentage of plans with at least one adjacent-waypoint displacement greater than 0.2 in normalized position coordinates. Infeasible denotes no completed plans (0/192).

99.9% with inverse-standard-deviation weighting and 100% with constant weighting, together with substantially lower distributional errors.

Compared with constant weighting, inverse-standard-deviation weighting reduces the 1-D W_1 from 0.238 to 0.061 and removes positive boundary excess on both mixtures, while maintaining comparable 2-D distributional fidelity. This is consistent with the factor $1/\sigma_t$ strengthening low-noise corrections and moving samples farther from the constraint boundary. The preferred weighting is nevertheless task-dependent, as on Sequential, constant weighting performs slightly better than inverse-standard-deviation weighting.

D.2 Performance on Multimodal Maze

In Maze2D, we empirically observe that the guidance vector $f_t(\mathbf{x}_t)$ in Eq. (18) can attain large magnitudes. To prevent excessively large updates during reverse sampling, we clip its ℓ_2 norm to 2. Table 7 additionally reports the trap rate, which detects large jumps between adjacent waypoints. On Medium, the SafeDiffuser variants have trap rates between 56.2% and 100%, while Truncate and SAD-Flower reach 63.5% and 84.4%, respectively. On Hard, the rates increase to 81.2% for SafeDiffuser–TVS and 99.0% for SAD-Flower. Figure 11 shows the associated artifacts: trajectories accumulate near obstacle boundaries and contain abrupt segments crossing the surrounding walls, rather than switching to an available safe route. The reported ANCHOR configuration has zero trap

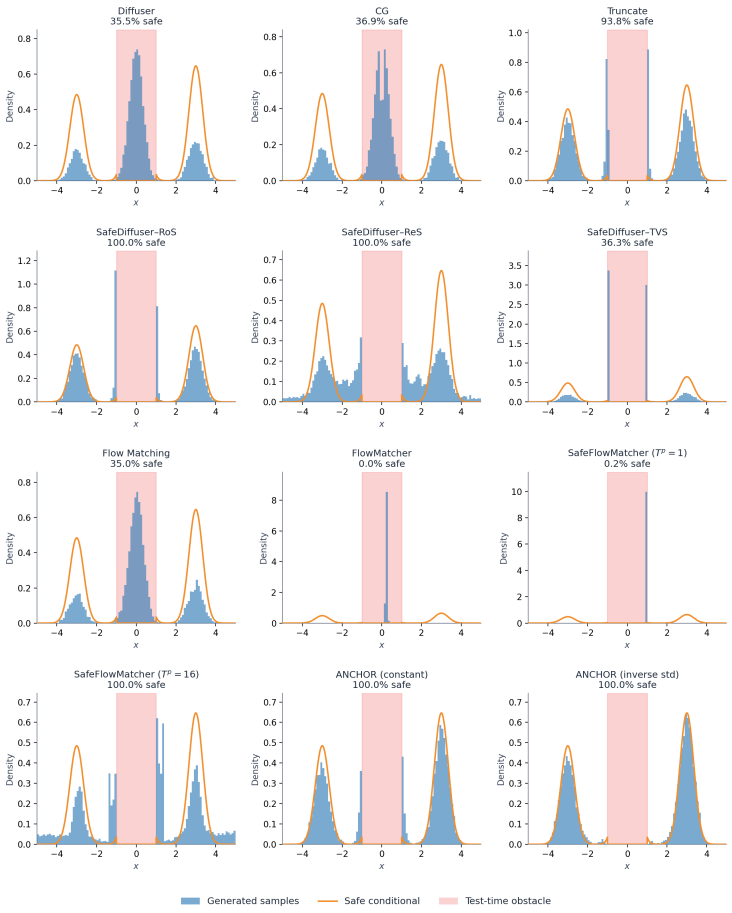


Figure 8: 1D Gaussian mixture: model-generated samples.

rate on both benchmarks and visibly reroutes trajectories through the unblocked corridor on Medium, achieving 94.3% planned safety.

Additionally, we note that, on Hard, SafeDiffuser–RoS and –ReS encounter infeasible CBF projections: the simultaneous linearized safety constraints admit no feasible correction at intermediate sampler states. Consequently, neither method completes a plan in any of the 192 attempts.

D.3 Quality of the theoretical approximations

Posterior covariance and guidance gradients. We compare two posterior-covariance approximations: Hessian zero and diagonal covariance. Hessian zero assigns the same variance to every coordinate, while diagonal covariance estimates each coordinate's variance separately but ignores correlations:

$$\widehat{\Sigma}_t^{\text{zero}} = r_t^2 I, \qquad \widehat{\Sigma}_t^{\text{diag}} = \text{diag}(\hat{d}_{t,1}, \ldots, \hat{d}_{t,D}),$$

where $\hat{d}_{t,i}$ estimates the posterior variance of coordinate i.

We also compare three gradient approximations: denoised-coordinate guidance omits the Jacobian of the Tweedie mean; stopped-model guidance holds the score-network output fixed, giving a mean Jacobian I/a_t; through-model guidance differentiates the Tweedie mean through the network.

Table 8 evaluates all combinations on both Gaussian mixtures under constant and inverse-standard-deviation weighting. We fix $t_{\max} = 0.9$, $\lambda = 1$, $t_{\min} = 0$, and uncapped guidance, and report three sampling seeds with 10,000 samples per seed. Diagonal covariance with stopped-model gradients reduces the one-dimensional constant-weight W_1 from 0.238 to 0.165, but increases the

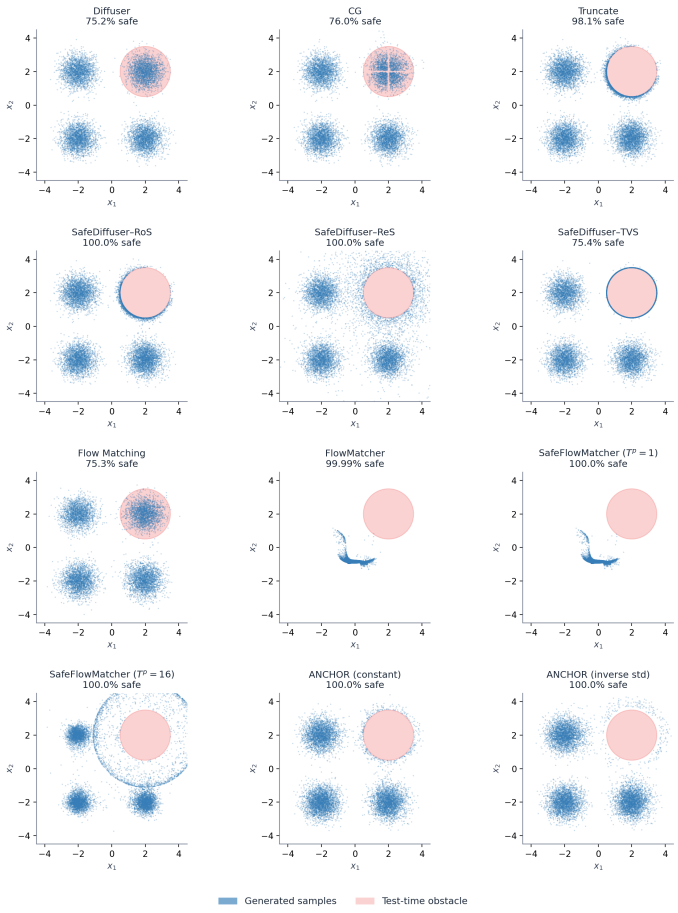


Figure 9: 2D Gaussian mixture: model-generated samples.

inverse-weight W_1 from 0.061 to 0.174. In two dimensions, diagonal covariance with through-model gradients achieves the lowest errors: 0.098 with constant weighting and 0.070 with inverse weighting. All inverse-weight configurations attain 100% empirical safety.

Covariance approximations on Multimodal Maze: Hard. We also compare Hessian-zero and diagonal covariance on the Hard maze at eight shared cutoffs, $t_{\max} \in \{0.1, 0.3, 0.5, 0.7, 0.9, 0.95, 0.98, 1.0\}$. All configurations use through-model gradients, $\lambda = 1$, with both time weights. Each configuration is evaluated on 192 plans across three sampling seeds. Table 9 shows the 48-configuration comparison of reward, planned safety, and boundary safety. At the shared constant-weight cutoff $t_{\max} = 0.95$, diagonal covariance increases planned safety from 26.6% to 49.0% relative to Hessian zero, while reducing mean reward from 76.96 to 58.72. In general, these results show that the approximation can change the safety–reward trade-off, and that retaining more covariance information does not by itself ensure better plans on the Hard Maze2D benchmark.

Approximation quality on Kuka. Table 10 compares covariance and gradient approximations on Kuka under the tightened 90% joint limits. Covariance variants use stopped-model gradients; gradient variants use Hessian-zero covariance. All settings share the same checkpoint and 100 matched episodes, with inverse weighting, $t_{\max} = 0.5$, $\lambda = 1$, and no norm cap. With stopped-model gradients, diagonal covariance raises mean reward from 0.45 to 0.58 compared with Hessian zero, but lowers planned safety from 100% to 36%.

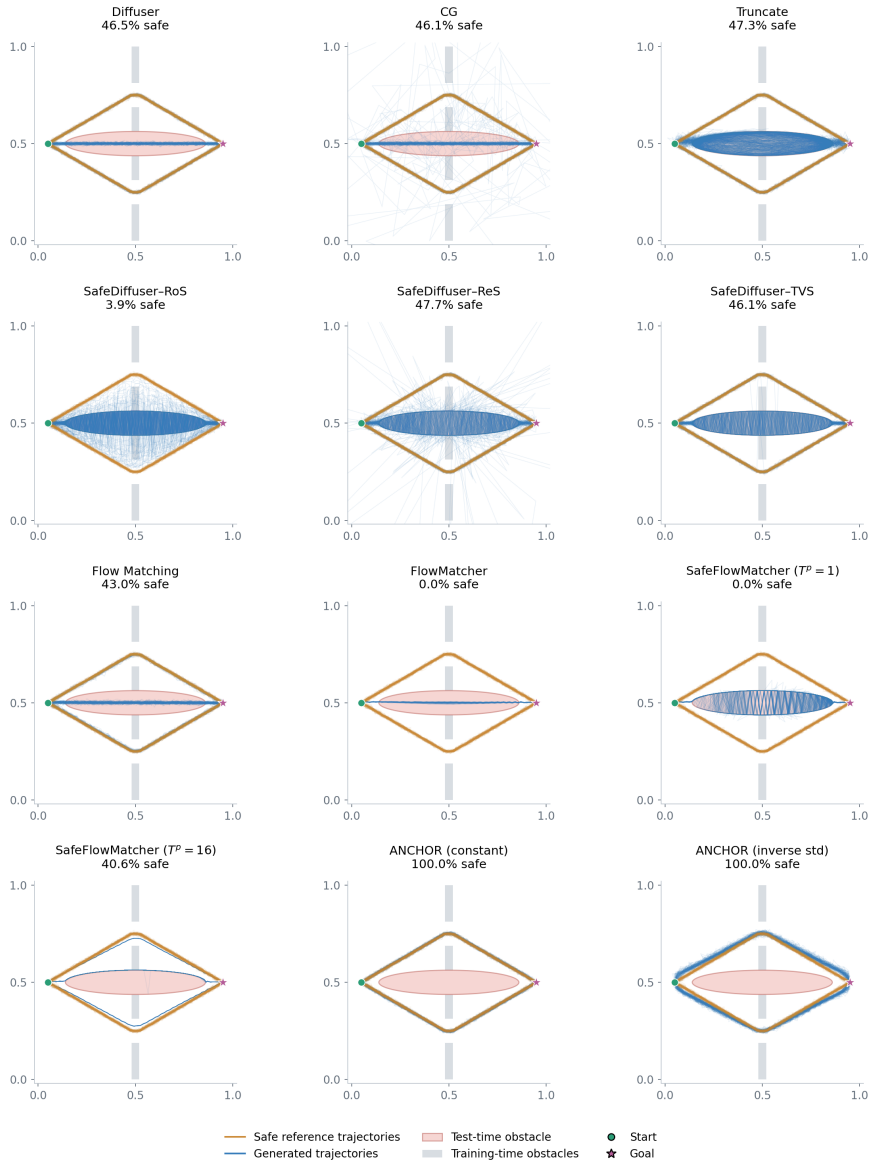


Figure 10: Sequential synthetic trajectory: model-generated samples.

D.4 Additional ANCHOR ablations

We investigate the guidance cutoff $t_{\max}$ under constant and inverse-standard-deviation weighting. Within each benchmark, we hold the checkpoint, covariance approximation, and gradient convention fixed.

Gaussian mixtures. Table 11 compares both weighting strategies using Hessian-zero covariance and stopped-model gradients. The results expose the same trade-off anticipated by the theory in Section 3.3: increasing $t_{\max}$ gives the guidance more time to act, but eventually activates it at high diffusion times, where the local SDF and Gaussian-posterior approximations are less accurate. Small cutoffs achieve safety but leave substantial distributional error. Increasing $t_{\max}$ from 0.9 to 1.0 indeed raises the one-dimensional W_1 from 0.238 to 1.578 with constant weighting and from 0.061 to 1.691 with inverse weighting.



Figure 11: All 192 planned trajectories, colored by completion, of each baseline on the Multimodal Maze benchmarks: (a) Medium and (b) Hard. We show both guidance scaling strategies for ANCHOR: constant scaling and inverse std scaling. Black regions denote local traps Yang et al. (2026).

Multimodal Maze. Table 12 compares the t_{max} guidance cutoffs and scaling stategies on Medium and Hard using Hessian-zero covariance, and through-model gradients. Both tasks benefit from activating guidance over a larger portion of the reverse process. On Medium, constant-weight planned safety increases from 37.0% at $t_{\max} = 0.3$ to 94.3% at $t_{\max} = 0.98$. Hard remains substantially more challenging, and higher reward does not necessarily coincide with higher planned or boundary safety.

Locomotion. Table 13 compares five guidance cutoffs on Hopper and Walker2d. All configurations achieve 100% planned safety, so the main trade-off concerns boundary clearance and task score. For $t_{\max} \in [0.05, 0.5]$, mean scores remain between 0.422 and 0.437 on Hopper and between 0.208 and 0.233 on Walker2d. Within this range, inverse-standard-deviation weighting raises boundary safety at $1\,\mathrm{cm}$ to 100%, compared with 56–59% under constant weighting, without a consistent reduction in mean score. However, boundary safety at $5\,\mathrm{cm}$ remains limited to 5% on Hopper and 10–18% on Walker2d. Applying guidance throughout the reverse process ($t_{\max} = 1$) raises boundary safety at $5\,\mathrm{cm}$ to 100% under both weighting strategies, but reduces mean scores to 0.125 on Hopper and approximately zero on Walker2d. Thus, extending guidance to the highest-noise steps increases clearance at a substantial cost to locomotion performance, without improving nominal planned safety. The intermediate cutoff $t_{\max} = 0.3$ used in the main comparison avoids this score collapse: with

		1-D Gaussian mixture			2-D Gaussian mixture		
Gradient	m_t	$W_1 \downarrow$ ($\times 10^{-2}$)	Safe↑ [%]	$\Delta_\partial \to 0$ [pp]	$\mathrm{SW}_1 \downarrow$ ($\times 10^{-2}$)	Safe↑ [%]	$\Delta_\partial \to 0$ [pp]
Hessian-zero covariance							
Denoised	1	67.2±1.7	100	11.4±0.4	27.0±1.0	100	2.8±0.1
Denoised	$1/\sigma_t$	29.3±1.9	100	−0.3±0.0	20.5±1.1	100	−0.1±0.0
Stopped	1	23.8±0.2	100	4.4±0.2	18.4±1.7	100	0.6±0.1
Stopped	$1/\sigma_t$	6.1±0.4	100	−0.3±0.0	19.9±1.8	100	−0.1±0.0
Through	1	51.0±1.7	100	13.0±0.2	14.9±1.4	100	1.9±0.1
Through	$1/\sigma_t$	23.3±2.2	100	−0.3±0.0	11.0±1.2	100	−0.1±0.0
Diagonal covariance							
Denoised	1	45.1±1.7	100	5.3±0.2	17.3±0.9	100	1.6±0.1
Denoised	$1/\sigma_t$	18.3±1.3	100	−0.3±0.0	12.0±0.9	100	−0.1±0.0
Stopped	1	16.5±1.5	100	1.2±0.1	18.8±1.4	100	0.3±0.1
Stopped	$1/\sigma_t$	17.4±2.0	100	−0.3±0.0	20.3±1.5	100	−0.1±0.0
Through	1	34.4±2.0	100	5.6±0.2	9.8±0.9	100	0.6±0.0
Through	$1/\sigma_t$	11.9±2.9	100	−0.3±0.0	7.0±1.0	100	−0.1±0.0

Table 8: Covariance and gradient comparison at $t_{\max} = 0.9$. Values are mean ± sample standard deviation over three seeds, with 10,000 samples per seed. Distances are reported in units of 10^{-2}; Δ_∂ denotes boundary excess in percentage points. Safety entries of 100 abbreviate $100.0 \pm 0.0\%$. Denoised, Stopped, and Through denote denoised-coordinate, stopped-model, and through-model gradients, respectively; m_t is the time weight. All configurations use 256 sampling steps.

		Hessian zero			Diagonal		
Weight m_t	$t_{\max}$	Reward	Safe [%]	B-Safe [%]	Reward	Safe [%]	B-Safe [%]
1	0.10	20.72±46.41	9.4	7.8	20.72±46.51	16.7	16.7
1	0.30	20.17±45.96	9.4	8.3	21.16±46.39	16.1	16.1
1	0.50	21.27±46.74	9.4	7.8	23.31±47.71	17.2	17.2
1	0.70	21.72±46.88	9.9	8.9	30.61±54.61	20.8	20.8
1	0.90	54.86±61.39	19.8	17.2	63.64±61.29	46.9	46.9
1	0.95	76.96±61.03	26.6	23.4	58.72±56.81	49.0	47.4
1	0.98	77.73±60.05	22.9	19.8	55.59±57.64	43.2	42.7
1	1.00	80.19±59.34	27.1	24.5	32.68±51.67	25.5	25.5
$1/\sigma_t$	0.10	21.47±45.90	13.0	12.0	20.72±46.51	9.4	9.4
$1/\sigma_t$	0.30	21.58±45.30	11.5	10.9	21.54±46.33	11.5	11.5
$1/\sigma_t$	0.50	22.82±45.85	12.5	11.5	25.13±49.04	12.5	12.0
$1/\sigma_t$	0.70	23.52±46.57	13.0	12.5	30.35±54.44	18.8	18.8
$1/\sigma_t$	0.90	54.38±58.77	28.1	22.9	58.28±58.49	38.5	37.0
$1/\sigma_t$	0.95	74.12±59.29	41.7	34.9	57.03±56.38	42.2	41.7
$1/\sigma_t$	0.98	73.33±60.28	45.3	35.4	45.46±56.58	34.4	33.9
$1/\sigma_t$	1.00	75.74±58.93	43.2	37.5	25.21±47.44	17.7	17.7

Table 9: Covariance approximations on Multimodal Maze: Hard. Every setting uses 64 plans across 3 seeds. All rows use the same guidance implementation. Through-model gradients differentiate the posterior mean with covariance derivatives held fixed.

inverse-standard-deviation weighting, it attains mean scores of 0.437 and 0.226, respectively, while maintaining 100% planned safety and $1\,\mathrm{cm}$ boundary safety.

Kuka. Table 14 reports the corresponding comparison under the tightened 90% joint limits, retaining Hessian-zero covariance, and stopped-model gradients. Both weights attain 100% nominal planned safety at the shared cutoffs, but inverse weighting produces larger relative boundary margins

Covariance	Gradient	Reward ↑	Safe [%]↑	B-Safe(5%)	B-Safe(10%)
Hessian zero	Stopped	0.45 ± 0.54	100	100	98
Diagonal	Stopped	0.58 ± 0.55	36	36	27
Hessian zero	Denoised	0.48 ± 0.57	100	100	98
Hessian zero	Through	0.49 ± 0.54	100	100	91

Table 10: Kuka ANCHOR covariance and gradient comparison. Inverse-standard-deviation weighting, tmax=0.5. Each row contains 100 episodes.

		One-dimensional mixture			Two-dimensional mixture		
$t_{\max}$	Time weight	$W_1 \downarrow$	Safe [%]↑	Δ_∂ [pp]$\to 0$	$\mathrm{SW}_1 \downarrow$	Safe [%]↑	Δ_∂ [pp]$\to 0$
0.05	1	1.392 ± 0.005	100.0 ± 0.0	64.31 ± 0.31	0.616 ± 0.003	100.0 ± 0.0	24.67 ± 0.08
0.05	$1/\sigma_t$	1.068 ± 0.003	100.0 ± 0.0	-0.33 ± 0.00	0.586 ± 0.003	100.0 ± 0.0	-0.13 ± 0.00
0.10	1	1.389 ± 0.007	100.0 ± 0.0	56.59 ± 0.41	0.616 ± 0.003	100.0 ± 0.0	21.98 ± 0.10
0.10	$1/\sigma_t$	1.265 ± 0.007	100.0 ± 0.0	-0.31 ± 0.01	0.605 ± 0.004	100.0 ± 0.0	-0.13 ± 0.01
0.15	1	1.374 ± 0.009	100.0 ± 0.0	43.61 ± 0.24	0.613 ± 0.004	100.0 ± 0.0	16.59 ± 0.34
0.15	$1/\sigma_t$	1.194 ± 0.009	100.0 ± 0.0	-0.32 ± 0.01	0.600 ± 0.005	100.0 ± 0.0	-0.12 ± 0.01
0.20	1	1.362 ± 0.011	100.0 ± 0.0	35.11 ± 0.16	0.611 ± 0.003	100.0 ± 0.0	12.56 ± 0.36
0.20	$1/\sigma_t$	1.070 ± 0.010	100.0 ± 0.0	-0.33 ± 0.01	0.592 ± 0.003	100.0 ± 0.0	-0.13 ± 0.01
0.30	1	1.316 ± 0.006	100.0 ± 0.0	25.86 ± 0.71	0.606 ± 0.004	100.0 ± 0.0	8.22 ± 0.28
0.30	$1/\sigma_t$	0.794 ± 0.008	100.0 ± 0.0	-0.33 ± 0.00	0.568 ± 0.005	100.0 ± 0.0	-0.13 ± 0.01
0.40	1	1.183 ± 0.014	100.0 ± 0.0	20.01 ± 0.49	0.588 ± 0.008	100.0 ± 0.0	6.02 ± 0.09
0.40	$1/\sigma_t$	0.621 ± 0.020	100.0 ± 0.0	-0.33 ± 0.00	0.535 ± 0.007	100.0 ± 0.0	-0.13 ± 0.01
0.50	1	0.968 ± 0.012	100.0 ± 0.0	14.93 ± 0.39	0.542 ± 0.007	100.0 ± 0.0	4.48 ± 0.05
0.50	$1/\sigma_t$	0.483 ± 0.025	100.0 ± 0.0	-0.33 ± 0.00	0.483 ± 0.006	100.0 ± 0.0	-0.13 ± 0.00
0.60	1	0.747 ± 0.010	100.0 ± 0.0	10.99 ± 0.16	0.454 ± 0.004	100.0 ± 0.0	3.19 ± 0.07
0.60	$1/\sigma_t$	0.377 ± 0.005	100.0 ± 0.0	-0.33 ± 0.00	0.391 ± 0.004	100.0 ± 0.0	-0.13 ± 0.00
0.70	1	0.550 ± 0.004	100.0 ± 0.0	8.00 ± 0.16	0.320 ± 0.009	100.0 ± 0.0	2.31 ± 0.06
0.70	$1/\sigma_t$	0.267 ± 0.018	100.0 ± 0.0	-0.33 ± 0.00	0.260 ± 0.013	100.0 ± 0.0	-0.13 ± 0.00
0.80	1	0.381 ± 0.008	100.0 ± 0.0	5.84 ± 0.13	0.151 ± 0.011	100.0 ± 0.0	1.33 ± 0.06
0.80	$1/\sigma_t$	0.155 ± 0.016	100.0 ± 0.0	-0.33 ± 0.00	0.100 ± 0.010	100.0 ± 0.0	-0.13 ± 0.00
0.90	1	0.238 ± 0.002	100.0 ± 0.0	4.39 ± 0.17	0.184 ± 0.017	100.0 ± 0.0	0.62 ± 0.05
0.90	$1/\sigma_t$	0.061 ± 0.004	100.0 ± 0.0	-0.33 ± 0.00	0.199 ± 0.018	100.0 ± 0.0	-0.13 ± 0.00
1.00	1	1.578 ± 0.018	100.0 ± 0.0	3.21 ± 0.15	0.585 ± 0.020	100.0 ± 0.0	0.18 ± 0.08
1.00	$1/\sigma_t$	1.691 ± 0.016	100.0 ± 0.0	-0.33 ± 0.00	0.605 ± 0.019	100.0 ± 0.0	-0.13 ± 0.00

Table 11: Cutoff comparison with Hessian-zero covariance and stopped-model gradients; sampling uses 256 steps.

with lower reward. At $t_{\max} = 0.5$, constant and inverse weighting give mean rewards of 0.63 and 0.45, and B-Safe(10%) rates of 63% and 98%, respectively.

	Constant $m_t = 1$			Inverse std $m_t = 1/\sigma_t$		
$t_{\max}$	Reward	Safe [%]	B-Safe [%]	Reward	Safe [%]	B-Safe [%]
Multimodal Maze: Medium						
0.30	28.21±37.12	37.0	37.0	28.05±36.91	37.0	37.0
0.50	28.09±36.97	37.0	37.0	27.84±36.65	37.0	37.0
0.70	28.31±36.88	37.5	37.5	28.42±36.63	37.5	37.5
0.90	44.01±36.93	59.4	59.4	43.54±36.57	59.4	59.4
0.95	65.73±25.37	88.0	88.0	64.92±25.08	88.0	88.0
0.98	70.26±18.89	94.3	94.3	68.86±20.05	93.2	93.2
1.00	68.90±20.69	92.7	92.7	69.68±18.77	94.3	94.3
Multimodal Maze: Hard						
0.10	20.72±46.41	9.4	7.8	21.47±45.90	13.0	12.0
0.30	20.17±45.96	9.4	8.3	21.58±45.30	11.5	10.9
0.50	21.27±46.74	9.4	7.8	22.82±45.85	12.5	11.5
0.70	21.72±46.88	9.9	8.9	23.52±46.57	13.0	12.5
0.90	54.86±61.39	19.8	17.2	54.38±58.77	28.1	22.9
0.95	76.96±61.03	26.6	23.4	74.12±59.29	41.7	34.9
0.98	77.73±60.05	22.9	19.8	73.33±60.28	45.3	35.4
1.00	80.19±59.34	27.1	24.5	75.74±58.93	43.2	37.5

Table 12: ANCHOR cutoff sweeps on Multimodal Maze with constant and inverse-standard-deviation weighting. All settings use Hessian-zero covariance, through-model mean gradients, and 256 sampling updates.

	Constant weighting				Inverse-standard-deviation weighting			
$t_{\max}$	Score ↑	Safe ↑	B(1 cm)	B(5 cm)	Score ↑	Safe ↑	B(1 cm)	B(5 cm)
Hopper: 1.60 m roof								
0.05	0.425±0.096	100	56	5	0.425±0.096	100	100	5
0.10	0.426±0.092	100	56	5	0.433±0.096	100	100	5
0.30	0.422±0.104	100	57	5	0.437±0.085	100	100	5
0.50	0.430±0.092	100	57	5	0.422±0.103	100	100	5
0.80	0.419±0.101	100	61	7	0.427±0.095	100	100	10
1.00	0.125±0.019	100	100	100	0.125±0.019	100	100	100
Walker2d: 1.40 m roof								
0.05	0.211±0.148	100	56	10	0.211±0.148	100	100	10
0.10	0.216±0.148	100	56	10	0.215±0.147	100	100	10
0.30	0.217±0.154	100	57	11	0.226±0.154	100	100	13
0.50	0.233±0.154	100	59	13	0.208±0.151	100	100	18
0.80	0.217±0.143	100	62	23	0.213±0.149	100	100	25
1.00	−0.005±0.017	100	100	100	−0.003±0.021	100	100	100

Table 13: Locomotion ANCHOR cutoff and time-weight ablation. Each setting uses 100 episodes, Hessian-zero covariance, stopped-model gradients, and 256 sampling updates. Hopper and Walker2d use roofs of 1.60 m and 1.40 m.

	Constant weighting				Inverse-standard-deviation weighting			
$t_{\max}$	Reward ↑	Safe ↑	B(5%)	B(10%)	Reward ↑	Safe ↑	B(5%)	B(10%)
0.10	0.52 ± 0.57	100	100	22	0.43 ± 0.57	100	100	97
0.20	0.55 ± 0.61	100	100	29	0.45 ± 0.55	100	100	98
0.30	0.53 ± 0.61	100	100	43	0.47 ± 0.56	100	100	98
0.50	0.63 ± 0.59	100	100	63	0.45 ± 0.54	100	100	98
1.00	0.63 ± 0.59	100	100	78	0.41 ± 0.53	100	100	100

Table 14: Kuka ANCHOR cutoff and time-weight ablation. Each setting uses 100 matched episodes, Hessian-zero covariance, and stopped-model gradients.