STEERING DIFFUSION MODELS TO RARE EVENTS WITH SEQUENTIAL MONTE CARLO

Anonymous authors
Paper under double-blind review

ABSTRACT

Diffusion models are increasingly used as surrogates for expensive simulators in weather prediction, molecular dynamics, and materials design. In these models, computing the probability $p_0[E]$ of an event E is difficult, especially when the event of interest is *rare*. A stable estimate using Monte Carlo becomes computationally intractable, requiring a growing sample size $\propto 1/p_0[E]$ to compensate for an increasing rarity. In this paper, we present **D**iffusion **I**mportance Sampling of **R**are **E**vents or **DireSMC**, a sequential Monte Carlo scheme that guides a population of weighted samples towards the rare event, giving access not only to samples but also to a calibrated estimate of its probability. We set up our guidance using an analytical relaxation of the event set, allowing the method to easily extend to a wide range of user-defined rare events. We validate our method on a toy problem with analytical solutions and on a score-based climate emulator, where we obtain accurate rare-event probabilities on a range of rarities from 10^{-3} to 10^{-5}, achieving net speed-ups of $6\times$ to $651\times$ over Monte Carlo.

1 INTRODUCTION

Score-based diffusion models (Song et al., 2021b) are a powerful tool for generative modelling, and have become standard in the quantitative sciences as cheap surrogates for expensive simulators in numerical weather prediction (Couairon et al., 2026; Price et al., 2024), molecular simulation (Abramson et al., 2024) and materials design (Zeni et al., 2025). Trained on data from a distribution p_0, they generate new representative samples by running a denoising process p_t that anneals from a Gaussian reference p_1 down to the data p_0, driven by a learned score $\nabla_x \log p_t$.

Often, scientists are interested in events E lying in regions where the data distribution assigns little mass $p_0[E]$. Such *rare events* carry outsized downstream impacts, and arise across a wide range of domains, from the environmental sciences (Lancelin et al., 2026; Manshausen et al., 2026; Peard et al., 2026) and bioinformatics (Xie et al., 2026) to finance (Glasserman & Li, 2005), engineering (Au & Wang, 2014) and autonomous driving (O' Kelly et al., 2018). Practitioners typically need two things: representative samples of the event, to characterise possible failure modes, and a calibrated estimate of its probability $p_0[E]$, to inform planning and resource allocation (Kaplan & Garrick, 1981; Rubino et al., 2009). The simplest approach is to draw N independent samples and form a Monte Carlo (MC) average to estimate $p_0[E]$, which is difficult precisely because the event is rare. The relative error is $1/\sqrt{Np_0[E]}$, hence stabilising the variance requires $N \propto 1/p_0[E]$ samples, which is prohibitively expensive (Morio et al., 2014; Rubino et al., 2009). This has motivated a large body of research designed specifically to tackle the poor scaling of standard MC (Glasserman & Li, 2005; Kahn & Harris, 1951; Morio et al., 2014).

Diffusion model steering offers a natural method for generating samples from E without the high costs of simple MC. A reward r_0 defines a tilted target $\pi_0 \propto p_0\, e^{r_0}$, and inference-time guidance steers a pre-trained sampler towards it. At each denoising step, the gradient of a time-dependent reward r_t is added, so that the sampler targets $\pi_t \propto p_t\, e^{r_t}$ (Chung et al., 2023; Song et al., 2023; Uehara et al., 2025). However, designing a good reward is difficult as the ideal r_t is often intractable, and any approximations used can guide the sampler towards a different distribution to π_0. For rare-event sampling specifically, setting the tilting term $e^{r_0} \equiv \mathbf{1}_E$ recovers the rare-event tilted distribution $p_0(\cdot|E) \propto p_0 \mathbf{1}_E$ that we wish to sample from. However, this reward is a poor object for steering, as its gradient cannot be used to steer particles towards E.

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.

Recent works have attempted to address these problems. The error of using an imperfect reward r_t can be addressed using sequential Monte Carlo (SMC) by tracking importance weights along the denoising trajectory and resampling particles to focus on high probability regions (He et al., 2026; Singhal et al., 2025; Skreta et al., 2025; Wu et al., 2023). In diffusion-based molecular simulation, Xie et al. (2026) adapt ideas from Abrams & Bussi (2013); Shirts & Chodera (2008) to estimate the occupancy of a rare metastable state using collective variables and a sequence of events E_i converging to E. Manshausen et al. (2026) instead steer a diffusion model using the probability-flow ODE (PF-ODE) (Song et al., 2021a), and estimate $p_0[E]$ using importance sampling. These methods have high compute costs, which become especially problematic in the rare event settings of interest here.

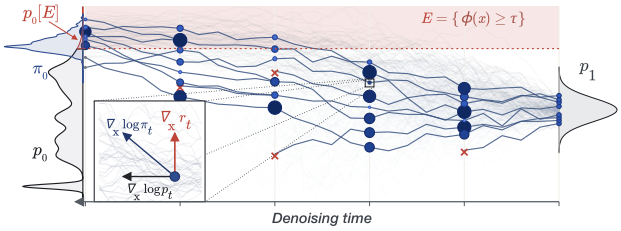


Figure 1: We steer particles (•) from p_{ref} towards π_0, which concentrates mass in the rare-event region of interest, using a guidance drift $\nabla_x r_k$. Particle size reflects the importance weights accumulated along the trajectory, which are used to resample (×) and form a rare event estimate, $p_0[E]$.

Contributions. *First*, we present DireSMC where we use a sequence of tilted marginals π_t to construct an unbiased sequential Monte Carlo estimator of $p_0[E]$ that runs within the denoising process, obtaining both samples and probability estimates. We show that the variance is governed by the quality of our steering rather than the rarity. *Second*, we show how to design a reward for rare-event simulation, and in particular for threshold-exceedance events $E = \{x : \phi(x) \geq \tau\}$, giving a smooth surrogate whose look-ahead guidance is closed-form under a Tweedie posterior approximation (Chung et al., 2023; Song et al., 2023). *Third*, we empirically validate on: (1) a Gaussian mixture where ground truth is available and we obtain probability estimates that do not degrade across 10^{-3}–10^{-5} orders of rarity. (2) a score-based climate emulator, where we draw samples and estimate probabilities of compound hot-and-dry extremes over the region of the 2021 Pacific Northwest heatwave, with net speed-ups of $6\times$ to $651\times$ over Monte Carlo across rarities from 10^{-3} to 10^{-5}.

2 Background

Diffusion models. Let p_{data} be the true data generating distribution. A diffusion model (Song et al., 2021b) progressively noises the data along a path p_t interpolating from p_{data} at $t = 0$ to a Gaussian reference p_{ref} at $t = 1$. Sampling reverses this process: a $X_1 \sim p_{\text{ref}}$ draw is evolved to $t = 0$, under the reverse-time SDE,

$$dX_t = \left[\mu_t(X_t) - \sigma_t^2 \nabla_x \log p_t(X_t)\right] dt + \sigma_t \, d\bar{W}_t, \qquad t \in [0, 1], \tag{1}$$

to generate from p_{data}. The drift μ_t and the diffusion coefficient σ_t are fixed by the noise schedule, and the score $\nabla_x \log p_t$ is approximated by a pre-trained network. In practice, Eq. (1) cannot be simulated directly. Instead, we fix a discretisation schedule $0 = t_0 < \cdots < t_K = 1$ and a numerical integrator whose k-th step $t_k \to t_{k-1}$ defines a Markov transition kernel $\bar{P}_{k-1|k}(\mathrm{d}x_{k-1} \mid x_k)$. Sampling corresponds to composing the K-step chain,

$$X_K \sim p_{\text{ref}}, \qquad X_{k-1} \mid X_k \sim \bar{P}_{k-1|k}(\mathrm{d}x_{k-1} \mid X_k), \qquad k = K, \ldots, 1. \tag{2}$$

Let p_k denote the marginal law of X_k under this chain. Hence $p_K = p_{\text{ref}}$ and $p_{k-1}(x_{k-1}) = \int p_k(x_k) \, \bar{P}_{k-1|k}(x_{k-1} \mid x_k) \, \mathrm{d}x_k$. Henceforth, we refer to the end-point of the discretised chain as p_0, with $X_0 \sim p_0$. As the discretisation is refined both $p_k \to p_{t=t_k}$ and $p_0 \to p_{\text{data}}$ at a rate given by the weak-order convergence of the integrator (Benton et al., 2024; Chen et al., 2023; Milstein & Tretyakov, 2004). Throughout, we assume that we can simulate from the transition kernels and have access to the score, but not to the densities p_t and p_k, which are generally intractable.

Rare Event Probabilities. Given a rare-event set $E \subset \mathbb{R}^d$ and the endpoint law of the chain p_0, we seek to estimate its probability $p_0[E] = \int \mathbf{1}_E(x) p_0(x) \mathrm{d}x$. A natural estimator of $p_0[E]$ is obtained by drawing N independent samples, $X_0^n \sim p_0$ and forming the Monte Carlo (MC) estimate,

$$\hat{p}_0^{\text{MC}}[E] := \frac{1}{N} \sum_{n=1}^{N} \mathbf{1}_E(X_0^n), \qquad \mathbb{E}\left[\hat{p}_0^{\text{MC}}[E]\right] = p_0[E]. \tag{3}$$

However, its relative variance is $(1 - p_0[E])/(Np_0[E])$, so when the event is rare, an impractical number $N \propto 1/p_0[E]$ of runs are needed for a fixed relative error (Rubino et al., 2009).

3 Rare-event sampling and probability estimation

Notation. For a measure μ, $\mu[f] := \int f(x)\,\mu(\mathrm{d}x)$ for a function f and $\mu[A] := \mu[\mathbf{1}_A]$ for a set A. Square brackets denote action on a function or set, and parentheses a density or kernel evaluation.

We seek to construct a low variance estimator of $p_0[E]$ with the same expectation as Eq. (3). We do so by reward tilting (Rubino et al., 2009; Uehara et al., 2025). A user-specified reward $r_0 : \mathbb{R}^d \to \mathbb{R}$ biases the base density to $\gamma_0(x) := p_0(x)\, e^{r_0(x)}$ with a normalised form given by $\pi_0 := \gamma_0/\gamma_0[1]$. Since $p_0 = \gamma_0\, e^{-r_0}$, both samples and estimates are functionals of the tilted law,

$$p_0[E] = \gamma_0\big[e^{-r_0}\mathbf{1}_E\big], \qquad p_0(x \mid E) = e^{-r_0(x)}\mathbf{1}_E(x)\,\pi_0(x)/\pi_0\big[e^{-r_0}\mathbf{1}_E\big]. \tag{4}$$

In the remainder of the section, we detail how to efficiently sample from $p_0(\cdot|E)$, construct an estimator $\hat{p}_0[E]$ using Sequential Monte Carlo (SMC), and design a reward that reduces its variance. While we present the following for discretised transition kernels, DireSMC holds automatically for the continuous time construction of a diffusion model.

3.1 Steering The diffusion Models towards Rare Events

The marginals of the diffusion model, p_k, tilted by the reward r_k, define a sequence of densities,

$$\gamma_k(x) := p_k(x)\, e^{r_k(x)}, \qquad \pi_k(x) := \gamma_k(x)/\gamma_k[1], \qquad k = K, \dots, 0, \tag{5}$$

pinned at both ends. At $k = 0$ (data), we set r_0 to be the reward of Eq. (4), and at $k = K$ (noise) we impose $r_K = 0$, such that $\gamma_K = p_{\mathrm{ref}}$. We defer reward path $r_{K-1}, \dots, r_1$ selection to Section 3.2.

Sequential Monte Carlo Sampler. We sample from the sequence of marginals of Eq. (5) using a sequential Monte Carlo sampler (Del Moral et al., 2006; Doucet et al., 2022; Vargas et al., 2023). We refer readers to App. C.2 for its general formulation; below we give a version adapted to our setting. Firstly, initialise N particles, $X_K^n \sim p_{\mathrm{ref}}$ with weights $w_K^n = 1$. At each step k, (i) propagate each particle through a backward proposal kernel, $\overleftarrow{Q}_{k-1|k}$, and (ii) update its weight by,

$$X_{k-1}^n \sim \overleftarrow{Q}_{k-1|k}(\cdot \mid X_k^n), \qquad w_{k-1}^n = w_k^n\, g_{k-1|k}\big(X_{k-1}^n, X_k^n\big), \tag{6}$$

where $g_{k-1|k}$ is the incremental weight function built from an auxiliary forward kernel $\overrightarrow{F}_{k|k-1}$,

$$g_{k-1|k}(x_{k-1}, x_k) = \left(\frac{\exp\big(r_{k-1}(x_{k-1})\big)}{\exp\big(r_k(x_k)\big)}\right) \cdot \left(\underbrace{\frac{p_{k-1}(x_{k-1})}{p_k(x_k)}}_{\text{intractable}} \cdot \frac{\overrightarrow{F}_{k|k-1}(x_k \mid x_{k-1})}{\overleftarrow{Q}_{k-1|k}(x_{k-1} \mid x_k)}\right). \tag{7}$$

The first factor is the ratio of reward-tilts at consecutive steps; the second is a likelihood ratio of the base process, involving the marginals p_k we cannot evaluate. The choice of both kernels is flexible, provided we can sample from $\overleftarrow{Q}_{k-1|k}$ and evaluate $g_{k-1|k}$ pointwise (Del Moral et al., 2006). Optionally, when the effective sample size, $\mathrm{ESS}(w_{k-1}) := \|w_{k-1}\|_1^2/\|w_{k-1}\|_2^2$ drops below a threshold ηN for $\eta \in [0, 1]$, we can resample, replacing each particle by an ancestor drawn in proportion to its weight and resetting the weights to the average weights of the ancestors. This is done through setting $X_{k-1}^n \leftarrow X_{k-1}^{a^n}$ and $w_{k-1}^n \leftarrow \frac{1}{N}\sum_n w_{k-1}^n$ with the ancestor index a^n drawn, e.g. systematically, from $\Pr[a^n = m \mid w_{k-1}] \propto w_{k-1}^m$ (Chopin et al., 2022). As resampling duplicates high-weight particles and depletes diversity, we follow it with a rejuvenation step, $X_{k-1}^n \sim M_{k-1}(\cdot \mid X_{k-1}^n)$, using a π_{k-1}-invariant MCMC kernel M_{k-1}. We use the Metropolis-adjusted diffusion model (MADM) kernel of (Lam et al., 2026), which approximates a Metropolis-adjusted Langevin kernel using only the score and r_k; both quantities that we already have access to (App. I).

Choice of Kernels. We construct our kernels to, (i) circumvent the intractable marginals and (ii) concentrate samples on E. For the first, we set $\overrightarrow{F}_{k|k-1}$ as the time reversal of the base chain,

$$\overrightarrow{F}_{k|k-1}(x_k \mid x_{k-1}) := p_k(x_k)\,\overleftarrow{P}_{k-1|k}(x_{k-1} \mid x_k)/p_{k-1}(x_{k-1}), \tag{8}$$

where $\overleftarrow{P}_{k-1|k}$ is the reverse diffusion kernel. This removes the intractable ratio and is a Markov kernel by the marginal recursion of p_{k-1} of Section 2. Our backward proposal, $\overleftarrow{Q}_{k-1|k}$ is defined by adding the gradient of the reward, or the *guidance*, to the reverse drift of Eq. (1),

$$dX_t = \big[\mu_t(X_t) - \sigma_t^2\,\big(\nabla_x \log p_t(X_t) + \nabla_x r_t(X_t)\big)\big]\,dt + \sigma_t\,d\overleftarrow{W}_t, \tag{9}$$

Algorithm 1 DireSMC: Steering diffusion models to rare events.

Require: pretrained diffusion model, transition kernels, $\overleftarrow{P}_{k-1|k}$ and $\overleftarrow{Q}_{k-1|k}$, rare event E, reward path r_k, particles N, discretisation schedule $1 = t_K > \cdots > t_0 = 0$, ESS threshold $\eta \in [0, 1]$.

$X_K^n \sim p_{\text{ref}}$, $w_K^n \leftarrow 1$ for $n = 1, \ldots, N$,
for $k = K, \ldots, 1$ **do**
  $X_{k-1}^n \sim \overleftarrow{Q}_{k-1|k}(\mathrm{d}x_{k-1} \mid X_k^n)$ ▷ One integrator step with the guided score Eq. (11)
  $w_{k-1}^n \leftarrow w_k^n \, g_{k-1|k}(X_{k-1}^n, X_k^n)$ ▷ Update weights using Eq. (12)
  if $\mathrm{ESS}(w_{k-1}) < \eta N$ **then**
    $\bar{w} \leftarrow \frac{1}{N} \sum_m w_{k-1}^m$;
    $X_{k-1}^n \leftarrow X_{k-1}^{a^n}$, $w_{k-1}^n \leftarrow \bar{w}$ ▷ Draw a^n using systematic resampling
    $X_{k-1}^n \sim M_{k-1}(\mathrm{d}x_{k-1} \mid X_{k-1}^n)$ ▷ Rejuvenate with a π_{k-1} invariant MCMC kernel
  end if
end for
return $\hat{p}_0[E]$ and $\hat{p}_0[\mathrm{d}x \mid E]$ from Eq. (13)

so that steering is exactly a tilt of the score, $\nabla_x \log p_k \to \nabla_x \log p_k + \nabla_x r_k$ and $\overleftarrow{Q}_{k-1|k}$ and $\overleftarrow{P}_{k-1|k}$ share the *same* integration steps but differ in the score. In our construction the *guidance strength* is defined within r_k (Uehara et al., 2025). Substituting into Eq. (7) gives the update function,

$$g_{k-1|k}(x_{k-1}, x_k) = \frac{e^{r_{k-1}(x_{k-1})}}{e^{r_k(x_k)}} \cdot \frac{\overleftarrow{P}_{k-1|k}(x_{k-1} \mid x_k)}{\overleftarrow{Q}_{k-1|k}(x_{k-1} \mid x_k)}. \tag{10}$$

Computing the Incremental Weights. Eq. (10) requires the ratio $\overleftarrow{P}_{k-1|k}/\overleftarrow{Q}_{k-1|k}$ pointwise. For one-step *explicit* integrators (Milstein & Tretyakov, 2004) with Gaussian transitions and a mean affine in the score, guidance shifts the mean but leaves the covariance invariant. So, for integrator dependent scalars $c_k, v_k > 0$,

$$\begin{aligned} \overleftarrow{P}_{k-1|k}(\mathrm{d}x_{k-1}|x_k) &= \mathcal{N}\big(m_k(x_k),\, v_k I\big), \\ \overleftarrow{Q}_{k-1|k}(\mathrm{d}x_{k-1}|x_k) &= \mathcal{N}\big(m_k(x_k) + c_k \nabla_x r_k(x_k),\, v_k I\big). \end{aligned} \tag{11}$$

This holds for first-order stochastic integrators such as Euler-Maruyama or SEEDS-1 of (Gonzalez et al., 2023), but not for higher-order schemes that evaluate the score at a noisy intermediate point, e.g. the stochastic Heun integrator of (Karras et al., 2022).

Lemma 1. *Under Eq. (11), let $\zeta_k := c_k/\sqrt{v_k}$, $\epsilon_k \sim \mathcal{N}(0, I)$ for sampling x_{k-1} from $\overleftarrow{Q}_{k-1|k}$, then*

$$\log g_{k-1|k}(x_{k-1}, x_k) = r_{k-1}(x_{k-1}) - r_k(x_k) - (\zeta_k^2/2)\big\|\nabla_x r_k(x_k)\big\|^2 - \zeta_k \big\langle \nabla_x r_k(x_k), \epsilon_k \big\rangle. \tag{12}$$

The proof is given in App. A; we use the analytical form for a ratio of Gaussians with a shared covariance alongside a discrete Cameron-Martin correction for the guidance term (Øksendal, 2003). The lemma states that the choice of numerical integrator only enters the weights via the scalar ζ_k. Additionally, tracking the weights requires no additional network calls as both $\nabla_x r_k$ and ϵ_k are readily available when the particle is steered. Under this formulation, for the Euler-Maruyama integrator with a VP-SDE noise schedule: $c_k = v_k = \sigma_{t_k}^2 \Delta t_k$ gives $\zeta_k = \sigma_{t_k}\sqrt{\Delta t_k}$, and Eq. (12) recovers the weights of (He et al., 2026). App. D.1 derives ζ_k for different integrators.

Estimators. With the weights reset to the average at each resampling, the particle system at $k = 0$ is a weighted approximation of γ_0, and substituting the weights into Eq. (4) gives the two outputs we are after: the probability estimate $\hat{p}_0[E] = \hat{\gamma}_0[e^{-r_0}\mathbf{1}_E]$ and weighted samples from the conditional,

$$\hat{p}_0[E] = \frac{1}{N}\sum_{n=1}^{N} w_0^n \, e^{-r_0(X_0^n)} \mathbf{1}_E(X_0^n), \; \hat{p}_0[\mathrm{d}x \mid E] = \frac{\sum_{n=1}^N w_0^n \, e^{-r_0(X_0^n)} \mathbf{1}_E(X_0^n) \, \delta_{X_0^n}[\mathrm{d}x]}{\sum_{n=1}^N w_0^n \, e^{-r_0(X_0^n)} \mathbf{1}_E(X_0^n)}. \tag{13}$$

Theorem 1. *Suppose $\overleftarrow{P}_{k-1|k} \ll \overleftarrow{Q}_{k-1|k}$ for all k, and the rejuvenation kernels M_k are π_k-invariant. Then, for any K, N, rewards $r_0, \ldots, r_K$ with $r_K = 0$ and $\gamma_0[1] < \infty$, and any adaptive resampling schedule, $\mathbb{E}[\hat{p}_0[E]] = p_0[E] = \mathbb{E}[\hat{p}_0^{\mathrm{MC}}[E]]$, and $\hat{p}_0[\cdot \mid E]$ converges almost surely to $p_0[\cdot \mid E]$ as $N \to \infty$. Moreover, if the score is exact and $p_{\text{data}}[\partial E] = 0$, then $\mathbb{E}[\hat{p}_0[E]] \to p_{\text{data}}[E]$ as $K \to \infty$.*

The proof is given in App. A and follows from the unbiasedness of the SMC normalisation constant estimator under adaptive resampling and π_{k-1} invariant moves (Del Moral et al., 2012; Syed et al., 2026). The estimator is well suited to the rare-event setting as it remains unbiased for *any* reward, however poorly it may steer. The normalisation constant, $\hat{p}_0[E]$ is accumulated along the denoising process rather than estimated separately (Del Moral et al., 2006; Doucet et al., 2022; Vargas et al., 2023) (App. C.3). The rewards only affects the variance, which is the subject of Section 3.2 below.

3.2 Reward selection

Theorem 1 shows $\hat{p}_0[E]$ is unbiased independent of reward choice, however, a poorly chosen steering inflates the variance of $\hat{p}_0[E]$ to an impractical number of required samples (Chatterjee & Diaconis, 2018). We construct a reward path and associated transition kernels that reduce the variance of $\hat{p}_0[E]$. Let $\mathbb{P}_{0:K}$ and $\mathbb{Q}_{0:K}$ be the joint laws over the base and guided chains, $(X_0, \dots, X_K)$. The ideal law to simulate from is the base chain reweighted by $e^{r_0(X_0)}/\gamma_0[1]$, since its endpoint is π_0. This is Doob's *h*-transform, $\mathbb{P}^\star_{0:K}$ of $\mathbb{P}_{0:K}$ (Dai Pra, 1991; Denker et al., 2024). Writing $\overleftarrow{P}_{0|k}$ as the denoising posterior of X_0 given X_k, the optimal reward and transitions are

$$\begin{aligned} r^\star_k(x_k) &:= \log \int \overleftarrow{P}_{0|k}(\mathrm{d}x_0 \mid x_k)\, e^{r_0(x_0)}, \\ \overleftarrow{P}^\star_{k-1|k}(\mathrm{d}x_{k-1} \mid x_k) &:= \overleftarrow{P}_{k-1|k}(\mathrm{d}x_{k-1} \mid x_k)\, e^{r^\star_{k-1}(x_{k-1}) - r^\star_k(x_k)}, \end{aligned} \tag{14}$$

and its marginal at step k is $\pi^\star_k \propto p_k\, e^{r^\star_k}$, with $\pi^\star_0 = \pi_0$ (App. A). While we refer to $r^\star_k$ as a reward for simplicity, it is also known as the value function in stochastic optimal control; see (Berner et al., 2024). The discrepancy between $\mathbb{P}^\star_{0:K}$ and $\mathbb{Q}_{0:K}$ determines the variance of our estimator.

Theorem 2. *Suppose there is no resampling ($\eta = 0$), $\overleftarrow{P}_{k-1|k} \ll \overleftarrow{Q}_{k-1|k}$ for all k, and $e^{r_0} \geq c$ on E for some $c \in (0, 1]$. Then*

$$\frac{\mathrm{Var}\big[\hat{p}_0[E]\big]}{p_0[E]^2} \;\leq\; \frac{1}{N}\left[\Big(\frac{\gamma_0[1]}{c\, p_0[E]}\Big)^2 \Big(1 + \chi^2\big(\mathbb{P}^\star_{0:K} \,\|\, \mathbb{Q}_{0:K}\big)\Big) - 1\right], \tag{15}$$

with equality when $e^{r_0} = c\,\mathbf{1}_E$, in which case $\mathbb{P}^\star_{0:K}$ is $\mathbb{P}_{0:K}$ conditioned on $X_0 \in E$. Moreover, the Doob transitions minimise Pearson's χ^2 divergence: for any $\overleftarrow{Q}_{k-1|k}$, $\chi^2(\mathbb{P}^\star_{0:K} \,\|\, \mathbb{Q}_{0:K}) \geq \chi^2(\pi^\star_K \,\|p_{\mathrm{ref}})$ with equality when $\overleftarrow{Q}_{k-1|k} = \overleftarrow{P}^\star_{k-1|k}$ for all k. The minimum is at most $\chi^2(\pi_0 \| p_0)$, and it is zero if X_K is independent of X_0 under $\mathbb{P}_{0:K}$.

The proof is in App. A and follows from the chain rule of measures and the data-processing inequality. The bound separates two design choices: (1) the endpoint reward affects $\gamma_0[1]/cp_0[E]$, which equals to 1 when $e^{r_0} \equiv 1_E$; (2) the second factor is minimised when the optimal Doob's transitions of Eq. (14) are used throughout the generation process. To elaborate, suppose $e^{r_0} = c1_E$, then the two distinct effects materialise. First, turning the steering off, the bound is exact with a relative variance equalling the Monte Carlo estimator keeping the first term. Second, if the Doob's transition kernels are instead used, then the relative variance is the initialisation error between the ideal starting distribution, $\pi^\star_K$ and the reference p_{ref}. The bound also gives us a practical target. With resampling turned off, we have $\mathrm{ESS}/N \to 1/(1 + \chi^2(\mathbb{P}^\star_{0:K} \| \mathbb{Q}_{0:K}))$ as $N \to \infty$ (Agapiou et al., 2017) which allows us to evaluate the quality of the simulated trajectories. Under resampling and sufficient rejuvenation, a more elaborate error analysis is needed, see Theorem 1(a) of (Syed et al., 2026). Intuitively, errors from a poor transition kernel $\overleftarrow{Q}_{k-1|k}$ can be pruned by resampling instead of compounding.

Rewards for threshold exceedance events. We now construct r_0 and $r_1, \dots, r_{K-1}$ for events of the form $E = \{x : \phi(x) \geq \tau\}$, for an observable $\phi : \mathbb{R}^d \to \mathbb{R}$ and threshold τ; App. E.1 extends r_0 and r_k to interval and multi-constraint events. Theorem 2 identifies the ideal terminal reward $e^{r_0} = c1_E$. However, it lacks the regularity needed to be steered with. We require a finite continuously differentiable reward so the guidance $\nabla_x r_t$ is well defined, and whose gradient is non-zero outside E, so particles outside the event set are actively pushed towards it. Further, we require that r_0 is bounded below E, so that the factor $\gamma_0[1]/cp_0[E]$ in Theorem 2 is controlled. Any CDF Φ with an everywhere-positive density gives such a regularity; writing $\delta > 0$ for a smoothing width,

$$r_{0,\delta}(x) \;:=\; \log \Phi\big((\phi(x) - \tau) \,/\, \delta\big)\,, \tag{16}$$

with $e^{r_{0,\delta}} \geq \Phi(0) = \frac{1}{2}$ on E. So $c = \frac{1}{2}$ in Theorem 2, and the induced π_0 tends to $p_0[\cdot \mid E]$ as $\delta \downarrow 0$. We adopt the Gaussian CDF because, for a linear observable $\phi(x) = a^\top x + b$, it is closed under convolution with a Gaussian (Rasmussen & Williams, 2006). See Fig. 5 for an illustration.

The reward sequence. Fixing $r_{0,\delta}$ leaves the Doob's rewards of Eq. (14) intractable through the posterior $\overleftarrow{P}_{0|k}$. Following (Chung et al., 2023; Song et al., 2023), we approximate it by a Gaussian centred at its mean $\hat{x}_{0|k}(x_k) = \mathbb{E}[X_0 \mid X_k = x_k]$,

$$\overleftarrow{P}_{0|k}(\mathrm{d}x_0 \mid x_k) \;\approx\; \mathcal{N}\big(\hat{x}_{0|k}(x_k),\, \Sigma_k\big), \tag{17}$$

with $\hat{x}_{0|k}$ approximated by Tweedie's formula from the score at t_k (Efron, 2011) (Eq. (42)) and Σ_k is approximated, e.g. via a diagonal ΠGDM posterior-covariance proxy (Song et al., 2023); we give the explicit forms of both in App. B. For the Gaussian-CDF reward and a linear observable ϕ, the resulting integral of Eq. (14) is closed-form, $r_k^\star(x_k) \approx \log \Phi\big((\phi(\hat{x}_{0|k}(x_k)) - \tau) \,/\, \sqrt{\delta^2 + a^\top \Sigma_k a}\,\big)$, with the smoothing width inflated by the convolution (App. E). The approximation is accurate near the data end and poor at high noise, where $\hat{x}_{0|k}$ carries little information about X_0, and it does not vanish at $k = K$ as the boundary condition requires. We therefore temper $r_k^\star$ by a guidance strength $\lambda_k \geq 0$, which switches the guidance on gradually as the look-ahead becomes reliable, and optionally approach the event in stages through a threshold τ_k. This gives us our reward sequence,

$$r_k(x_k) = \lambda_k \, \log \Phi\Big((\phi(\hat{x}_{0|k}(x_k)) - \tau_k) / \sqrt{\delta^2 + a^\top \Sigma_k a}\,\Big), \tag{18}$$

pinned at both ends as Eq. (5) requires: $\lambda_0 = 1$, $\tau_0 = \tau$ give $r_0 = r_{0,\delta}$ and $\lambda_K = 0$ gives $r_K = 0$. The schedules λ_k and τ_k are the two parameters of our method; App. C relates them to classical rare-event samplers. By Theorem 1, none of these approximations affect what $\hat{p}_0[E]$ estimates; by Theorem 2, they determine the variance of our estimator.

3.3 Related work

Several works use a pre-trained diffusion sampler to target a tilted distribution without retraining (Berner et al., 2024; Chung et al., 2023; Denker et al., 2024; Song et al., 2023; Uehara et al., 2025), but these do not compute probabilities. The weight construction combining diffusion models and SMC samplers (He et al., 2026; Singhal et al., 2025; Skreta et al., 2025; Wu et al., 2023) is common, but our contribution is to apply it to rare-event sampling and probability estimation. In diffusion-based molecular simulations, enhanced rare-event sampling ideas such as metadynamics and umbrella sampling have been adapted (Xie et al., 2026) to estimate the occupancy of a rare metastable state. Both methods track importance weights however, Xie et al. (2026) assemble $p_0[E]$ by combining a sequence of events E_i with MBAR (Shirts & Chodera, 2008), rather than estimating $p_0[E]$ directly. Manshausen et al. (2026) use a score-based climate emulator guidance to generate Tropical Cyclone (TC) events and use importance sampling. An ODE sampler samples the target rare event, followed by two passes of a guided and unguided Probability Flow ODE to estimate its importance weights (Song et al., 2021b). We instead estimate $p_0[E]$ by accumulating weights from a single denoising pass and benefit from resampling, which is unavailable to them; see App. J for a comparison.

4 Experiments

We validate DireSMC in two settings: a correlated two-dimensional Gaussian mixture model (GMM) and a pretrained score-based climate emulator. For the GMM, the score and guidance are known analytically so our method may be evaluated in the absence of approximation errors in addition to a learned score, see App. F. The second is a pretrained score-based climate emulator for estimating extreme compound weather events, demonstrating the scalability and applicability of our method. Estimates are evaluated against the sampled law, $p^\star$: closed-form $p_0[E]$ for the analytical score or a large MC for the learned score; the same references define the sample-quality metrics. The ratio $\hat{p}_0[E]/p^*$ measures the bias and the estimator coefficient of variation, $\mathrm{CV} = \sqrt{\mathrm{Var}(\hat{p}_0[E])}/\,p^\star$ measures the relative variance computed over independent runs.

4.1 Correlated Gaussian Mixture Model (GMM)

Let p be the seven-mode GMM shown in Figure 6 and defined in Table 4. Let $\phi(x) = x_1$ and define $E = \{x_1 \geq \tau\}$ for $\tau \in \{7, 8, 9, 10\}$. This varies the analytic mass $p_0[E]$ from 6.9×10^{-3} to 1.4×10^{-5}, defining a sequence of increasingly rare events. Unless stated otherwise $N = 1000$, $\delta = 0.01$, $\lambda(t) = 1-(t/0.9)^2$ with $\lambda(t) = 0$ for $t > 0.9$, $\tau(t) = \tau$ and $K = 1000$, using the SEEDS-1 integrator

guided by the reward of Eq. (18); where Σ_t is the ΠGDM proxy (App. B). We run Algorithm 1 with adaptive resampling at $\mathrm{ESS}/N < 0.5$, followed by three MADM rejuvenation sweeps (Lam et al., 2026) gated to a time $t \geq 0.30$. Experiments are replicated over 24 random seeds and validated against a 200 million MC reference pool (App. G).

Our method recovers the reference exceedance probability for both analytical and learned scores (full per-threshold results and sample-quality metrics in Table 7, App. G.3). For a fixed $\tau = 10$ threshold, Fig. 2 shows that the relative variance decreases as $O(1/\sqrt{N})$ (Chatterjee & Diaconis, 2018). Resampling and rejuvenation prevent accumulation of guidance mismatch (Del Moral et al., 2006; 2012) and the CV stays flat across rarities for a fixed N.

DireSMC compares favourably against the importance sampling estimator of Manshausen et al. (2026). Fig. 2 (left) shows that DireSMC with resampling and MADM leads to lower CV for the same number of particles compared to their method, even though their method requires $3.5\times$ as many function evaluations as us. In this setting, rejuvenation incurs an additional 1.6% network calls over the base, which we also discount. On a compute normalised particle budget, all variants of our estimator achieve a lower variance across all rarities as seen in Fig. 2 (cost detailed in App. J and Table 24). Additionally, the estimator of Manshausen et al. (2026) assumes that the law induced by an explicit integrator is reversible. However, this often does not hold in practice, leading to a discretisation bias which decreases with step count K. To isolate the effects of discretisation bias, we ran experiments with the optimal Doob's transition kernels of Eq. (14) such that the true p^* is known and any artefacts due to approximate guidance or score estimates are removed. We also provide the PF-ODE estimator with the true Jacobian trace term, which would otherwise have to be estimated. The results in Fig. 2 (right) show their discretisation bias is significantly larger than DireSMC's with the bias decreasing with increasing K (cf. Table 23). We acknowledge that our method also incurs a discretisation bias which we characterise below.

Ablations. We isolate both the estimator bias and variance introduced by SDE integrators (Euler-Maruyama, SEEDS-1 (Gonzalez et al., 2023)) and the DDIM ODE integrator (Song et al., 2021a) used in Manshausen et al. (2026). Evaluating with both analytical and learned scores, the discretisation bias decreases approximately as $O(1/K)$ (see Fig. 7, 9 in App. G.2.2). For a fixed K we find the discretisation bias increases with rarity, highlighting the sensitivity of rare-event probability estimation to small errors. Both SDE integrators track a lower bias compared to DDIM which we postulate is due to the lower overall error in the generation process, see Theorem 3 of Xu et al. (2023). Moreover, we characterise two sources of errors due to the MADM MCMC kernel (Lam et al., 2026). It uses the $\nabla_x \log p_{t=t_k}$ to estimate Metropolis Hastings (MH) acceptance ratio, hence targets the continuous-time marginals rather than the law γ_k induced by the integrator. This introduces an error decaying as $O(1/K)$ to leading order (Fig. 13). Additionally, MADM assumes access to a conservative score to compute the ratio which degrades for a learned score network as $t \to 0$ (Vuong et al., 2025), see App. G.3.3 for details. Heuristically, we gate rejuvenation to $t^* > 0$ to account for this assumption (Table 9). Finally, we show that the method translates without modification to interval and multi-constraint events (Table 10).

4.2 Generating heatwaves with pre-trained climate model emulators

Next, we validate on the problem of estimating how the probability of extreme events evolves under a changing climate. The high cost of running physics-based climate models (Eyring et al., 2016) has motivated the practice of training cheap emulators on climate model output (Watson-Parris et al., 2022). However, as we will show, even with these cost-savings, it can be difficult to generate targeted extreme events without explicit steering.

Model. We leverage the pre-trained climate emulator of Bouabid et al. (2026) which is trained on data generated from the MPI-ESM1-2-LR model (Mauritsen et al., 2019) and released as part of the Coupled Model Intercomparison Project Phase 6 (CMIP6) (Eyring et al., 2016). The emulator generates global monthly-mean fields of near-surface temperature (`tas`), precipitation (`pr`), relative humidity (`hurs`) and wind speed (`sfcWind`), conditioned on the global mean surface temperature anomaly, ΔT, against a pre-industrial reference and the month of the year, m, to capture annual variability[1]; we refer to Bouabid et al. (2026) for an in-depth motivation of the emulator setup.

[1]As is common in the climate science research literature, the emulator outputs the change in output variables compared to a pre-industrial reference climate rather than absolute predictions.

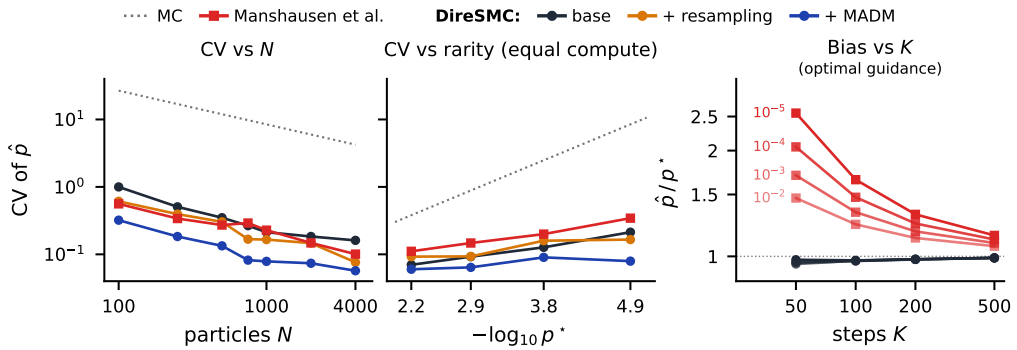


Figure 2: *Left:* CV plotted against N for a $p_0[x_1 > 10]$ problem at a fixed K. Rejuvenation tracks the lowest CV across all N. *Middle:* we compare our estimator against Manshausen et al. (2026) across rarities at a compute-normalised N. We consistently track a lower CV. *Right:* measured bias scaling with rarity using exact Doob's steering and analytical scores, highlighting the bias of Manshausen et al. (2026), DireSMC tracks the floor, see Tables 23 and App. J.1.2.

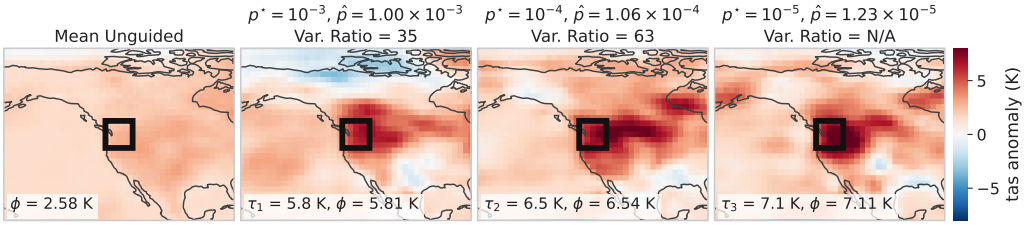


Figure 3: Surface-temperature anomaly (K) for July at $\Delta T = +1.5$ K over the Pacific Northwest (PNW) target region. *Left:* average `tas` field over 10^7 samples. *Remaining panels:* the highest-weight crossing particle produced by the guided sampler at different exceedance thresholds τ and regional average temperature $\phi(\cdot)$ of the sample. $p^\star$ are MC estimates; $\hat{p}$ is obtained from the guided estimator; full results in Table 15. The variance ratio (Var. Ratio) at matched compute is defined as $\mathrm{Var}(\hat{p}_{\text{unguided}})/\mathrm{Var}(\hat{p}_{\text{guided}})$. No samples at matched compute were generated with MC for τ_3.

Setup. We consider extremes in the temperature and humidity variables averaged within the region R defined by the bounding box 42°-52°N latitude and 235°-245°E longitude, roughly corresponding to the region affected by the 2021 Pacific Northwest (PNW) heatwave (McKinnon & Simpson, 2022). We use $N = 1024$ particles with $K = 200$ discretisation steps using the exponential integrator; error/variance metrics are computed using estimates from 8 seeds. We generate $\sim 10^7$ Monte Carlo samples to estimate ground-truth event probabilities. Further model details and hyperparameters are presented in App. H. We select the hyperparameter once to minimise the CV constrained on the τ_1 experiment (detailed below) and use it for all of our other experiments; proper per-problem tuning would lead to a larger variance reduction.

Extreme temperatures. We first verify how accurately DireSMC can estimate the probability of extreme temperatures relative to crude Monte Carlo. Assuming $\Delta T = 1.5$ K, for three different temperature thresholds τ_1, τ_2, τ_3 (exact values in App. H) we calculate the probability that the average surface temperature in the PNW exceeds a temperature threshold in the month of July. The results in Fig. 3, (full numbers in Table 15) show that our method provides significant improvements compared to standard MC estimates with at least an order of magnitude reduction in variance. For the most extreme event τ_3, MC does not sample any events that exceed this threshold, hence no samples can be used to characterise this event. Figure 3 demonstrates that the generated guided samples provide physically plausible temperature maps: the maps show more warming over land masses compared to the ocean and spatially coherent warming patterns that are not restricted solely to the target region. Additional results and ablations in App. H.

Compound hot and dry extremes. Next we probe whether our method can capture extreme events impacting multiple variables by considering events that are both extremely hot and dry because of their outsized societal impacts (Fan et al., 2023; Zscheischler et al., 2017). For these joint events it is important to not just consider the separate marginal probabilities of each variable but also

to accurately capture the correlations between variables. To do so, we compute the dependence ratio of the event, defined as $d = p_0[x > \tau_{\texttt{tas}}, x < \tau_{\texttt{hurs}}] \,/\, p_0[x > \tau_{\texttt{tas}}]\, p_0[x < \tau_{\texttt{hurs}}]$. If the events are independent the dependence ratio would reduce to $d = 1$, while $d > 1$ implies that the two events co-occur. We find that using the non-factorised form of the reward, i.e. setting $\Phi(\cdot) = \Phi_2(\texttt{hurs}, \texttt{tas})$ to be a function of both variables, leads to a better estimate (cf. Table 18). Table 1 shows we recover the correct joint probability and the dependence ratio against the ground-truth estimate with a positive net speed-up compared to crude MC. Notably, the dependence structure strengthens as the event becomes rarer from $d = 31.3 \to 1.4 \times 10^3$. This suggests that a hot-and-dry July over R is three orders of magnitude more likely than the product of its marginals would suggest.

Table 1: Probability estimates of three compound hot and dry extreme events over the PNW region. The dependence ratio d is defined in the main text, $\hat{d}$ denotes that estimated by our method. Net speed-up is the crude-MC cost of matching the CV of the guided run divided by the measured per-sample cost of guidance (Table 21); thresholds τ are given in App. H.1.

$p^\star$	$\hat{p}/p^\star$	CV	$d^\star$	$\hat{d}/d^\star$	Speed Up ↑
3.16×10^{-3}	1.00 ± 0.03	0.08	31	1.01 ± 0.04	$6\times$
2.14×10^{-4}	1.00 ± 0.03	0.09	210	1.04 ± 0.04	$82\times$
1.40×10^{-5}	1.08 ± 0.05	0.12	1393	1.03 ± 0.05	$651\times$

A warming climate. As the model can generate samples conditioned on the global temperature anomaly ΔT, it lets us ask *how do extreme events change under climate change?* We fix the temperature threshold at $\tau_{\texttt{tas}} \geq 5.81$ K and sweep $\Delta T \in [0, 4]$ K, rerunning the same recipe as above. This temperature approximately corresponds to $p_0[E] \approx 10^{-4}$ at 1.2 K (2021) of warming, mirroring the PNW heatwave (McKinnon & Simpson, 2022). The event moves from a threshold exceedance probability of $\sim 10^{-6}$ at $\Delta T = 0$ to a near-certainty at $\Delta T = 4$ K. Our estimator tracks it across all six orders of magnitude without any modification. Fig. 4 reports this probability estimate. See App. H.3 for details, including Table 19 and Fig. 19 for corresponding global temperature maps.

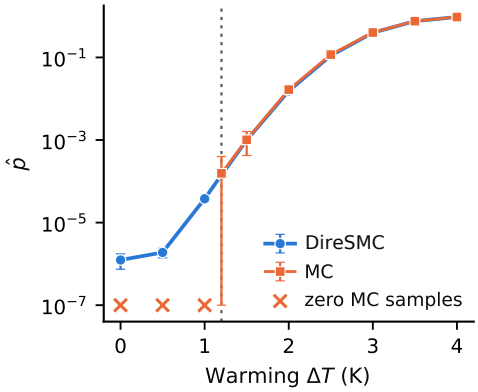


Figure 4: Estimated $\hat{p}_0[E]$ versus global temperature anomaly ΔT for our method versus unguided MC at equal compute. Points are the mean over 8 runs, bars are ± 1 std. Bars reaching the axis floor indicate runs with zero events; for $\Delta T \leq 1.0$, no MC runs record an event. See Tables 19 and 20.

5 Conclusion

We have shown that the diffusion sampler builds the same annealing bridge that traditional rare-event samplers build by hand. Using this insight, we present DireSMC and steer the generation process towards a user-defined rare event using a smooth relaxation of the event set, accumulating importance weights and estimating the probability along the way. Empirically, on synthetic examples and a score-based climate emulator, we achieve a net speedup of 6 to $651\times$ over crude Monte Carlo, improving with rarity. We believe that this is of value to the scientific community where there is a growing interest in investigating how well a generative model can capture rare events (Sun et al., 2025).

Limitations & Future Work. The variance of our estimator can be improved through better approximations of Doob's h-transform e.g. (Denker et al., 2024; Potaptchik et al., 2026; Uehara et al., 2025) as well as better reward schedule design (Syed et al., 2026); these can be readily plugged in without modification to the algorithm. Our estimates target the discretised, approximate law we sample from, and so carry a reducible discretisation bias and any errors of an approximate score. The rejuvenation kernel used (Lam et al., 2026) assumes a conservative score, which a trained network need not provide; distilling the network into an energy model (Thornton et al., 2025) could be a viable solution. Additionally the MADM kernel also does not target the law γ_k induced by the numerical integrator, with this mismatch decreasing with the weak-order convergence rate of the integrator. Algorithm 1 is agnostic to the choice of MCMC moves, and any alternatives to MADM can be used.

AI USE STATEMENT

In this work, we used generative AI tools to assist with the following tasks: help develop theoretical models or conceptual frameworks, formulate mathematical claims, provide critical ingredients for proving mathematical claims, assist in the writing of proofs, design or provide feedback on research methodology or experiments, implement methods, interpret results, create or edit software code. We also used Generative AI for sourcing/searching for information.

We did not use generative AI tools to generate synthetic datasets, propose or refine hypotheses, assist with translation, clean and reformat dataset, support qualitative or thematic data analysis.

We critically reviewed and verified all AI-assisted work and outputs either via a manual audit of generated codes and by verifying outputs against a manual review of the literature. LLM coding agents were used to assist in the code writing process. However, the core methodology, initial idea and hypothesis generation, as well as the initial exploratory code, did not involve any LLM agents. The authors made final decisions regarding theoretical, methodological, experimental details. We take full responsibility for the final content of this work, including text, claims, writing, or any artifacts produced with the aid of generative AI.

ETHICS STATEMENT

We describe the broader impact of our work to the climate sciences in Section 4.2. Cheaper access to calibrated rare-event probabilities and samples can be used to support risk assessments, but the probability we estimate is based on the discretised law of a pre trained model, so care must be taken when using them. We highlight this as a limitation throughout this work.

REPRODUCIBILITY STATEMENT

We give all theoretical results and proofs with accompanying set of assumptions in the Appendix, or refer to known literature. Additionally we provide hyperparameters used in all of our experiments in Appendices H and G. The climate model used is open-sourced, see (Bouabid et al., 2026). To aid in reproducibility, we plan to release the code and scripts to reproduce the experiments at a later date.

REFERENCES

Cameron Abrams and Giovanni Bussi. Enhanced sampling in molecular dynamics using metadynamics, replica-exchange, and temperature-acceleration. *Entropy*, 16(1):163–199, 2013.

Josh Abramson, Jonas Adler, Jack Dunger, Richard Evans, Tim Green, Alexander Pritzel, Olaf Ronneberger, Lindsay Willmore, Andrew J Ballard, Joshua Bambrick, et al. Accurate structure prediction of biomolecular interactions with alphafold 3. *Nature*, 630(8016):493–500, 2024.

Sergios Agapiou, Omiros Papaspiliopoulos, Daniel Sanz-Alonso, and Andrew M Stuart. Importance sampling: Intrinsic dimension and computational cost. *Statistical Science*, pp. 405–431, 2017.

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

Siu-Kui Au and Yu Wang. *Engineering risk assessment with subset simulation*. John Wiley & Sons, 2014.

Joe Benton, Valentin De Bortoli, Arnaud Doucet, and George Deligiannidis. Nearly d-linear convergence bounds for diffusion models via stochastic localization. In *International Conference on Learning Representations*, volume 2024, pp. 36916–36936, 2024.

Julius Berner, Lorenz Richter, and Karen Ullrich. An optimal control perspective on diffusion-based generative modeling. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=oYIjw37pTP.

Christopher M Bishop and Nasser M Nasrabadi. *Pattern recognition and machine learning*, volume 4. Springer, 2006.

Valentin De Bortoli, James Thornton, Jeremy Heng, and Arnaud Doucet. Diffusion schrödinger bridge with applications to score-based generative modeling. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=9BnCwiXB0ty.

Shahine Bouabid, Andre Nogueira Souza, and Raffaele Ferrari. Score-based generative emulation of impact-relevant earth system model outputs. *Journal of Advances in Modeling Earth Systems*, 18(3):e2025MS005558, 2026. doi: https://doi.org/10.1029/2025MS005558. URL https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2025MS005558. e2025MS005558 2025MS005558.

Benjamin Boys, Mark Girolami, Jakiw Pidstrigach, Sebastian Reich, Alan Mosca, and O Deniz Akyildiz. Tweedie moment projected diffusions for inverse problems. *arXiv preprint arXiv:2310.06721*, 2023.

Sourav Chatterjee and Persi Diaconis. The sample size required in importance sampling. *The Annals of Applied Probability*, 28(2):1099–1135, 2018.

Sitan Chen, Sinho Chewi, Jerry Li, Yuanzhi Li, Adil Salim, and Anru Zhang. Sampling is as easy as learning the score: theory for diffusion models with minimal data assumptions. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=zyLVMgsZ0U_.

Nicolas Chopin, Sumeetpal S Singh, Tomás Soto, and Matti Vihola. On resampling schemes for particle filters with weakly informative observations. *The Annals of Statistics*, 50(6):3197–3222, 2022.

Hyungjin Chung, Jeongsol Kim, Michael Thompson Mccann, Marc Louis Klasky, and Jong Chul Ye. Diffusion posterior sampling for general noisy inverse problems. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=OnD9zGAGT0k.

Guillaume Couairon, Renu Singh, Anastase Charantonis, Christian Lessig, and Claire Monteleoni. Archesweathergen: Skillful and compute-efficient probabilistic weather forecasting with machine learning. *Science Advances*, 12(17):eadx2372, 2026.

Paolo Dai Pra. A stochastic control approach to reciprocal diffusion processes. *Applied mathematics and Optimization*, 23(1):313–329, 1991.

Pierre Del Moral, Arnaud Doucet, and Ajay Jasra. Sequential monte carlo samplers. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 68(3):411–436, 2006.

Pierre Del Moral, Arnaud Doucet, and Ajay Jasra. On adaptive resampling strategies for sequential monte carlo methods. *Bernoulli*, 18(1), February 2012. ISSN 1350-7265. doi: 10.3150/10-bej335. URL http://dx.doi.org/10.3150/10-BEJ335.

Alexander Denker, Francisco Vargas, Shreyas Padhy, Kieran Didi, Simon Mathis, Vincent Dutordoir, Riccardo Barbano, Emile Mathieu, Urszula J Komorowska, and Pietro Lio. Deft: Efficient fine-tuning of diffusion models by learning the generalised h-transform. *Advances in Neural Information Processing Systems*, 37:19636–19682, 2024.

Arnaud Doucet, Simon Godsill, and Christophe Andrieu. On sequential monte carlo sampling methods for bayesian filtering. *Statistics and computing*, 10(3):197–208, 2000.

Arnaud Doucet, Will Grathwohl, Alexander G Matthews, and Heiko Strathmann. Score-based diffusion meets annealed importance sampling. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 21482–21494. Curran Associates, Inc., 2022. doi: 10.52202/068431-1561. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/86b7128efa3950df7c0f6c0342e6dcc1-Paper-Conference.pdf.

Zvi Drezner. Computation of the trivariate normal integral. *Mathematics of Computation*, 62(205): 289–294, 1994.

Zvi Drezner and George O Wesolowsky. On the computation of the bivariate normal integral. *Journal of Statistical Computation and Simulation*, 35(1-2):101–107, 1990.

Bradley Efron. Tweedie's formula and selection bias. *Journal of the American Statistical Association*, 106(496):1602–1614, 2011. doi: 10.1198/jasa.2011.tm11181. URL https://doi.org/10.1198/jasa.2011.tm11181. PMID: 22505788.

Stefan Elfwing, Eiji Uchibe, and Kenji Doya. Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. *Neural Networks*, 107:3–11, 2018. ISSN 0893-6080. doi: https://doi.org/10.1016/j.neunet.2017.12.012. URL https://www.sciencedirect.com/science/article/pii/S0893608017302976. Special issue on deep reinforcement learning.

Veronika Eyring, Sandrine Bony, Gerald A Meehl, Catherine A Senior, Bjorn Stevens, Ronald J Stouffer, and Karl E Taylor. Overview of the coupled model intercomparison project phase 6 (cmip6) experimental design and organization. *Geoscientific Model Development*, 9(5):1937–1958, 2016.

Xuewei Fan, Chiyuan Miao, Jakob Zscheischler, Louise Slater, Yi Wu, Yuanfang Chai, and Amir AghaKouchak. Escalating hot-dry extremes amplify compound fire weather risk. *Earth's Future*, 11(11):e2023EF003976, 2023. doi: https://doi.org/10.1029/2023EF003976. URL https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2023EF003976. e2023EF003976 2023EF003976.

Nial Friel and Anthony N Pettitt. Marginal likelihood estimation via power posteriors. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 70(3):589–607, 2008.

Ruiqi Gao, Emiel Hoogeboom, Jonathan Heek, Valentin De Bortoli, Kevin P. Murphy, and Tim Salimans. Diffusion meets flow matching: Two sides of the same coin. 2024. URL https://diffusionflow.github.io/.

Alan Genz. Numerical computation of multivariate normal probabilities. *Journal of Computational and Graphical Statistics*, 1(2):141–149, 1992. ISSN 10618600. URL http://www.jstor.org/stable/1390838.

Alan Genz. Numerical computation of rectangular bivariate and trivariate normal and t probabilities. *Statistics and computing*, 14(3):251–260, 2004.

Paul Glasserman and Jingyi Li. Importance sampling for portfolio credit risk. *Management science*, 51(11):1643–1656, 2005.

Paul Glasserman, Philip Heidelberger, Perwez Shahabuddin, and Tim Zajic. Multilevel splitting for estimating rare event probabilities. *Operations Research*, 47(4):585–600, 1999. ISSN 0030364X, 15265463. URL http://www.jstor.org/stable/223163.

Martin Gonzalez, Nelson Fernandez, Thuy Vinh Dinh Tran, Elies Gherbi, Hatem Hajri, and Nader Masmoudi. SEEDS: Exponential SDE solvers for fast high-quality sampling from diffusion models. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=V6IgkYKD8P.

Pieralberto Guarniero, Adam M Johansen, and Anthony Lee. The iterated auxiliary particle filter. *Journal of the American Statistical Association*, 112(520):1636–1647, 2017.

Tiankai Hang, Shuyang Gu, Chen Li, Jianmin Bao, Dong Chen, Han Hu, Xin Geng, and Baining Guo. Efficient diffusion training via min-snr weighting strategy. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 7407–7417. IEEE, 2023.

Jiajun He, José Miguel Hernández-Lobato, Yuanqi Du, and Francisco Vargas. RNE: plug-and-play diffusion inference-time control and energy-based training. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=fkf7tlkN8q.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 6840–6851. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/4c5bcfec8584af0d967f1ab10179ca4b-Paper.pdf.

Adam M Johansen and Arnaud Doucet. A note on auxiliary particle filters. *Statistics & Probability Letters*, 78(12):1498–1504, 2008.

Herman Kahn and Theodore E Harris. Estimation of particle transmission by random sampling. *National Bureau of Standards applied mathematics series*, 12:27–30, 1951.

Stanley Kaplan and B John Garrick. On the quantitative definition of risk. *Risk analysis*, 1(1):11–27, 1981.

Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 26565–26577. Curran Associates, Inc., 2022. doi: 10.52202/068431-1926. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/a98846e9d9cc01cfb87eb694d946ce6b-Paper-Conference.pdf.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Peter E. Kloeden and Eckhard Platen. *Numerical Solution of Stochastic Differential Equations*, volume 23 of *Stochastic Modelling and Applied Probability*. Springer, 1992.

Chieh-Hsin Lai, Yuhta Takida, Naoki Murata, Toshimitsu Uesaka, Yuki Mitsufuji, and Stefano Ermon. Fp-diffusion: Improving score-based diffusion models by enforcing the underlying score fokker-planck equation. In *International Conference on Machine Learning*, pp. 18365–18398. PMLR, 2023.

Kevin H Lam, Tyler Farghly, Christopher Williams, Jun Yang, Yee Whye Teh, and Arnaud Doucet. Metropolis-adjusted diffusion models. *arXiv preprint arXiv:2605.09654*, 2026.

Amaury Lancelin, Alexander Wikner, Laurent Dubus, Clément Le Priol, Dorian S. Abbot, Freddy Bouchet, Pedram Hassanzadeh, and Jonathan Weare. Ai-boosted rare event sampling to characterize extreme weather. *Phys. Rev. Lett.*, Jun 2026. doi: 10.1103/b1gc-9c2q. URL https://link.aps.org/doi/10.1103/b1gc-9c2q.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=PqvMRDCJT9t.

Xingchao Liu, Chengyue Gong, and qiang liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=XVjTT1nw5z.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Bkg6RiCqY7.

Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. DPM-solver: A fast ODE solver for diffusion probabilistic model sampling in around 10 steps. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=2uAaGwlP_V.

Peter Manshausen, Noah Brenowitz, Julius Berner, Karthik Kashinath, and Mike Pritchard. Towards accurate extreme event likelihoods from diffusion model climate emulators, 2026. URL https://arxiv.org/abs/2605.03802.

Thorsten Mauritsen, Jürgen Bader, Tobias Becker, Jörg Behrens, Matthias Bittner, Renate Brokopf, Victor Brovkin, Martin Claussen, Traute Crueger, Monika Esch, et al. Developments in the mpi-m earth system model version 1.2 (mpi-esm1. 2) and its response to increasing co2. *Journal of Advances in Modeling Earth Systems*, 11(4):998–1038, 2019.

Karen A. McKinnon and Isla R. Simpson. How unexpected was the 2021 pacific northwest heatwave? *Geophysical Research Letters*, 49(18):e2022GL100380, 2022. doi: https://doi.org/10.1029/2022GL100380. URL https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2022GL100380. e2022GL100380 2022GL100380.

Grigori N Milstein and Michael V Tretyakov. *Stochastic numerics for mathematical physics*, volume 39. Springer, 2004.

Pierre Moral. *Feynman-Kac formulae: genealogical and interacting particle systems with applications*. Springer, 2004.

Jérôme Morio, Mathieu Balesdent, Damien Jacquemart, and Christelle Vergé. A survey of rare event simulation methods for static input–output models. *Simulation Modelling Practice and Theory*, 49: 287–304, 2014.

Christian A Naesseth, Fredrik Lindsten, and Thomas B Schön. Elements of sequential monte carlo. *Foundations and Trends® in Machine Learning*, 12(3):187–306, 2019.

Radford M Neal. Annealed importance sampling. *Statistics and computing*, 11(2):125–139, 2001.

Bernt Øksendal. Stochastic differential equations. In *Stochastic differential equations: an introduction with applications*, pp. 38–50. Springer, 2003.

Brian C O'Neill, Claudia Tebaldi, Detlef P Van Vuuren, Veronika Eyring, Pierre Friedlingstein, George Hurtt, Reto Knutti, Elmar Kriegler, Jean-Francois Lamarque, Jason Lowe, et al. The scenario model intercomparison project (scenariomip) for cmip6. *Geoscientific Model Development*, 9(9):3461–3482, 2016.

Matthew O' Kelly, Aman Sinha, Hongseok Namkoong, Russ Tedrake, and John Duchi. Scalable end-to-end autonomous vehicle testing via rare-event simulation. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper_files/paper/2018/file/653c579e3f9ba5c03f2f2f8cf4512b39-Paper.pdf.

Art B. Owen. *Monte Carlo theory, methods and examples*. https://artowen.su.domains/mc/, 2013.

Donald B Owen. Tables for computing bivariate normal probabilities. *The Annals of Mathematical Statistics*, 27(4):1075–1090, 1956.

A. Peard, Y. Mo, and J. W. Hall. Simulating spatial multi-hazards with generative deep learning. *Natural Hazards and Earth System Sciences*, 26(4):1663–1683, 2026. doi: 10.5194/nhess-26-1663-2026. URL https://nhess.copernicus.org/articles/26/1663/2026/.

Emanuel Pfarr, Radu Timofte, and Frank Werner. Stochstic sampling for generative diffusion models: From euler-maruyama to higher-order schemes, 2026. URL https://arxiv.org/abs/2601.18425.

Michael K Pitt and Neil Shephard. Filtering via simulation: Auxiliary particle filters. *Journal of the American statistical association*, 94(446):590–599, 1999.

Peter Potaptchik, Adhi Saravanan, Abbas Mammadov, Alvaro Prat, Michael Samuel Albergo, and Yee Whye Teh. Meta flow maps enable scalable reward alignment. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=K5gV8Yptne.

Ilan Price, Alvaro Sanchez-Gonzalez, Ferran Alet, Tom R. Andersson, Andrew El-Kadi, Dominic Masters, Timo Ewalds, Jacklynn Stott, Shakir Mohamed, Peter Battaglia, Remi Lam, and Matthew Willson. Gencast: Diffusion-based ensemble forecasting for medium-range weather, 2024. URL https://arxiv.org/abs/2312.15796.

Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, 2006. ISBN 978-0-262-25683-4. doi: 10.7551/mitpress/3206.001.0001. URL https://doi.org/10.7551/mitpress/3206.001.0001.

Gerardo Rubino, Bruno Tuffin, et al. *Rare event simulation using Monte Carlo methods*, volume 73. Wiley Online Library, 2009.

Yuyang Shi, Valentin De Bortoli, Andrew Campbell, and Arnaud Doucet. Diffusion schrödinger bridge matching. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 62183–62223. Curran Associates, Inc., 2023. doi: 10.52202/075280-2717. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/c428adf74782c2092d254329b6b02482-Paper-Conference.pdf.

Michael R Shirts and John D Chodera. Statistically optimal analysis of samples from multiple equilibrium states. *The Journal of chemical physics*, 129(12), 2008.

Raghav Singhal, Zachary Horvitz, Ryan Teehan, Mengye Ren, Zhou Yu, Kathleen Mckeown, and Rajesh Ranganath. A general framework for inference-time scaling and steering of diffusion models. 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. 55810–55827. PMLR, 13–19 Jul 2025. URL https://proceedings.mlr.press/v267/singhal25b.html.

Marta Skreta, Tara Akhound-Sadegh, Viktor Ohanesian, Roberto Bondesan, Alán Aspuru-Guzik, Arnaud Doucet, Rob Brekelmans, Alexander Tong, and Kirill Neklyudov. Feynman-kac correctors in diffusion: Annealing, guidance, and product of experts. *arXiv preprint arXiv:2503.02819*, 2025.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021a. URL https://openreview.net/forum?id=St1giarCHLP.

Jiaming Song, Arash Vahdat, Morteza Mardani, and Jan Kautz. Pseudoinverse-guided diffusion models for inverse problems. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=9_gsMA8MRKQ.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021b. URL https://openreview.net/forum?id=PxTIG12RRHS.

Y. Qiang Sun, Pedram Hassanzadeh, Mohsen Zand, Ashesh Chattopadhyay, Jonathan Weare, and Dorian S. Abbot. Can ai weather models predict out-of-distribution gray swan tropical cyclones? *Proceedings of the National Academy of Sciences*, 122(21):e2420914122, 2025. doi: 10.1073/pnas.2420914122. URL https://www.pnas.org/doi/abs/10.1073/pnas.2420914122.

Saifuddin Syed, Alexandre Bouchard-Côté, Kevin Chern, and Arnaud Doucet. Optimized annealed sequential monte carlo samplers. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, pp. qkag082, 2026.

Denis Talay and Luciano Tubaro. Expansion of the global error for numerical schemes solving stochastic differential equations. *Stochastic analysis and applications*, 8(4):483–509, 1990.

James Thornton, Louis Béthune, Ruixiang Zhang, Arwen Bradley, Preetum Nakkiran, and Shuangfei Zhai. Composition and control with distilled energy diffusion models and sequential monte carlo. *arXiv preprint arXiv:2502.12786*, 2025.

Masatoshi Uehara, Yulai Zhao, Chenyu Wang, Xiner Li, Aviv Regev, Sergey Levine, and Tommaso Biancalani. Inference-time alignment in diffusion models with reward-guided generation: Tutorial and review. *arXiv preprint arXiv:2501.09685*, 2025.

Francisco Vargas, Will Sussman Grathwohl, and Arnaud Doucet. Denoising diffusion samplers. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=8pvnfTAbu1f.

Aki Vehtari, Daniel Simpson, Andrew Gelman, Yuling Yao, and Jonah Gabry. Pareto smoothed importance sampling. *Journal of Machine Learning Research*, 25(72):1–58, 2024.

An Vuong, Michael Thompson McCann, Javier E. Santos, and Yen Ting Lin. Are we really learning the score function? reinterpreting diffusion models through wasserstein gradient flow matching. *Transactions on Machine Learning Research*, 2025. ISSN 2835-8856. URL https://openreview.net/forum?id=CzyJqXQRhJ.

D. Watson-Parris, Y. Rao, D. Olivié, Ø. Seland, P. Nowack, G. Camps-Valls, P. Stier, S. Bouabid, M. Dewey, E. Fons, J. Gonzalez, P. Harder, K. Jeggle, J. Lenhardt, P. Manshausen, M. Novitasari, L. Ricard, and C. Roesch. Climatebench v1.0: A benchmark for data-driven climate projections. *Journal of Advances in Modeling Earth Systems*, 14(10):e2021MS002954, 2022. doi: https://doi.org/10.1029/2021MS002954. URL https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2021MS002954. e2021MS002954 2021MS002954.

Christopher Williams, Andrew Campbell, Arnaud Doucet, and Saifuddin Syed. Score-optimal diffusion schedules. *Advances in Neural Information Processing Systems*, 37:107960–107983, 2024.

Luhuan Wu, Brian Trippe, Christian Naesseth, David Blei, and John P Cunningham. Practical and asymptotically exact conditional sampling in diffusion models. *Advances in Neural Information Processing Systems*, 36:31372–31403, 2023.

Yuchen Wu, Minshuo Chen, Zihao Li, Mengdi Wang, and Yuting Wei. Theoretical insights for diffusion guidance: A case study for gaussian mixture models. In *Forty-first International Conference on Machine Learning*, 2024. URL https://openreview.net/forum?id=M1ADedSnlJ.

Yu Xie, Ludwig Winkler, Lixin Sun, Sarah Lewis, Adam E Foster, José Jiménez Luna, Tim Hempel, Michael Gastegger, Yaoyi Chen, Iryna Zaporozhets, et al. Enhanced diffusion sampling: Efficient rare event sampling and free energy calculation with diffusion models. *arXiv preprint arXiv:2602.16634*, 2026.

Yilun Xu, Mingyang Deng, Xiang Cheng, Yonglong Tian, Ziming Liu, and Tommi S. Jaakkola. Restart sampling for improving generative processes. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=wFuemocyHZ.

Claudio Zeni, Robert Pinsler, Daniel Zügner, Andrew Fowler, Matthew Horton, Xiang Fu, Zilong Wang, Aliaksandra Shysheya, Jonathan Crabbé, Shoko Ueda, et al. A generative model for inorganic materials design. *Nature*, 639(8055):624–632, 2025.

Jakob Zscheischler, Rene Orth, and Sonia I Seneviratne. Bivariate return periods of temperature and precipitation explain a large fraction of european crop yields. *Biogeosciences*, 14(13):3309–3320, 2017.

Appendix

A Proofs **20**

A.1 Notation and two identities 20

A.2 Structure of the tilted chain 20

A.3 Proof of Theorem 1 21

A.4 Proof of Theorem 2 21

A.5 The optimal reward recursion 22

A.6 Proof of Lemma 1 22

B Diffusion model details **23**

C Sequential methods **24**

C.1 Estimating the normalisation constant 24

C.2 Generic SMC algorithm 26

C.3 Why annealing becomes necessary 26

C.4 Splitting estimator 27

D Weight computation **28**

D.1 ζ_k for common integrators 29

E The guidance gradient via local linearisation **31**

E.1 Factorised guidance for multiple constraints 32

F The correlated 2D GMM benchmark **33**

G 2D GMM experiments **36**

G.1 Experimental details 36

G.2 Sources of Bias 37

G.3 The threshold-exceedance experiment 39

G.4 Box and multiple box events 45

G.5 Ablation of Parameters 47

H Climate experiments **48**

H.1 Setup and model details 48

H.2 Experiment 1: Compound hot & dry events 50

H.3 Experiment 2: A changing climate 57

H.4 Computational cost 58

H.5 Hyperparameter ablation 59

I Metropolis-adjusted Langevin corrections **59**

J Related Works Extended **61**

J.1 The PF-ODE Estimator 61

A PROOFS

A.1 NOTATION AND TWO IDENTITIES

Throughout, $(X_0, \dots, X_K)$ is the base chain of Eq. (2), $\mathbb{P}_{0:K}$ its law, p_k its marginals, and $\mathbb{Q}_{0:K}$ the law of the guided chain with kernels $\overleftarrow{Q}_{k-1|k}$; both start from p_{ref}. All densities are with respect to Lebesgue measure. For a reward r_0 with $\gamma_0[1] < \infty$, write $Z_0 := \gamma_0[1]$ and

$$\frac{d\mathbb{P}^\star_{0:K}}{d\mathbb{P}_{0:K}}(x_{K:0}) := \frac{e^{r_0(x_0)}}{Z_0}, \qquad \frac{d\mathbb{P}^E_{0:K}}{d\mathbb{P}_{0:K}}(x_{K:0}) := \frac{\mathbf{1}_E(x_0)}{p_0[E]}, \tag{19}$$

for the base chain tilted by e^{r_0} and conditioned on $X_0 \in E$ respectively; the second is the first with $e^{r_0} = \mathbf{1}_E$. The Doob rewards are

$$r^\star_k(x_k) := \log \int \overleftarrow{P}_{0|k}(\mathrm{d}x_0 \mid x_k)\, e^{r_0(x_0)}, \qquad k = 0, \dots, K, \tag{20}$$

so that $r^\star_0 = r_0$ and $Z_0 = \int p_{\mathrm{ref}}(x_K)\, e^{r^\star_K(x_K)}\, \mathrm{d}x_K$.

Two identities are used repeatedly. First, the *tower identity*: by the Markov property,

$$\int \overleftarrow{P}_{k-1|k}(\mathrm{d}x_{k-1} \mid x_k)\, e^{r^\star_{k-1}(x_{k-1})} \;=\; e^{r^\star_k(x_k)}, \qquad k = 1, \dots, K. \tag{21}$$

Second, the *telescoping identity*: the reward ratios in the incremental weight of Eq. (10) collapse along a trajectory, since $r_K = 0$:

$$\begin{aligned}\prod_{k=1}^K g_{k-1|k}(x_{k-1}, x_k) &= e^{r_0(x_0)} \prod_{k=1}^K \frac{\overleftarrow{P}_{k-1|k}(x_{k-1} \mid x_k)}{\overleftarrow{Q}_{k-1|k}(x_{k-1} \mid x_k)} \\ &= e^{r_0(x_0)}\, \frac{d\mathbb{P}_{0:K}}{d\mathbb{Q}_{0:K}}(x_{K:0}),\end{aligned} \tag{22}$$

the last equality because both chains start from p_{ref}.

A.2 STRUCTURE OF THE TILTED CHAIN

Lemma 2. *$\mathbb{P}^\star_{0:K}$ is a Markov chain. Its marginal at step k is $\pi^\star_k := p_k\, e^{r^\star_k}/Z_0$, so its initial law is $\pi^\star_K = p_{\mathrm{ref}}\, e^{r^\star_K}/Z_0$, its transitions are the twisted kernels*

$$\overleftarrow{P}^\star_{k-1|k}(\mathrm{d}x_{k-1} \mid x_k) := \overleftarrow{P}_{k-1|k}(\mathrm{d}x_{k-1} \mid x_k)\, e^{r^\star_{k-1}(x_{k-1}) - r^\star_k(x_k)}, \tag{23}$$

Consequently

$$\frac{d\mathbb{P}^\star_{0:K}}{d\mathbb{Q}_{0:K}}(x_{K:0}) \;=\; \frac{e^{r^\star_K(x_K)}}{Z_0} \prod_{k=1}^K \frac{\overleftarrow{P}^\star_{k-1|k}(x_{k-1} \mid x_k)}{\overleftarrow{Q}_{k-1|k}(x_{k-1} \mid x_k)}. \tag{24}$$

Proof. By Eq. (21), each $\overleftarrow{P}^\star_{k-1|k}$ integrates to one and is a Markov kernel. Multiplying the kernels and telescoping the exponents,

$$\prod_{k=1}^K \overleftarrow{P}^\star_{k-1|k}(x_{k-1} \mid x_k) \;=\; \frac{e^{r_0(x_0)}}{e^{r^\star_K(x_K)}} \prod_{k=1}^K \overleftarrow{P}_{k-1|k}(x_{k-1} \mid x_k), \tag{25}$$

using $r^\star_0 = r_0$. Hence

$$\begin{aligned}\frac{p_{\mathrm{ref}}(x_K)\, e^{r^\star_K(x_K)}}{Z_0} \prod_{k=1}^K \overleftarrow{P}^\star_{k-1|k}(x_{k-1} \mid x_k) &= \frac{e^{r_0(x_0)}}{Z_0}\, p_{\mathrm{ref}}(x_K) \prod_{k=1}^K \overleftarrow{P}_{k-1|k}(x_{k-1} \mid x_k) \\ &= \frac{d\mathbb{P}^\star_{0:K}}{dx_{K:0}}(x_{K:0}),\end{aligned} \tag{26}$$

which exhibits $\mathbb{P}^\star_{0:K}$ as the chain started from $\pi^\star_K$ with kernels $\overleftarrow{P}^\star_{k-1|k}$. The marginal at step k follows from the tower identity applied to the tilt: integrating out all coordinates but x_k gives $p_k(x_k) \int \overleftarrow{P}_{0|k}(\mathrm{d}x_0 \mid x_k) e^{r_0(x_0)}/Z_0 = p_k(x_k) e^{r^\star_k(x_k)}/Z_0$. Dividing the density of $\mathbb{P}^\star_{0:K}$ by that of $\mathbb{Q}_{0:K}$, $p_{\mathrm{ref}}(x_K) \prod_k \overleftarrow{Q}_{k-1|k}(x_{k-1} \mid x_k)$, gives Eq. (24). □

A.3 Proof of Theorem 1

Proof. The auxiliary kernel $\overrightarrow{F}_{k|k-1}(x_k \mid x_{k-1}) = p_k(x_k)\overleftarrow{P}_{k-1|k}(x_{k-1} \mid x_k)/p_{k-1}(x_{k-1})$ integrates to one in x_k by the marginal recursion of Section 2, so it is a Markov kernel. With targets $\gamma_k = p_k e^{r_k}$ and proposal $\overleftarrow{B}_{k-1|k} = \overleftarrow{Q}_{k-1|k}$, the generic SMC weight of Eq. (7) is

$$\begin{aligned}\frac{\gamma_{k-1}(x_{k-1})\,\overrightarrow{F}_{k|k-1}(x_k \mid x_{k-1})}{\gamma_k(x_k)\,\overleftarrow{Q}_{k-1|k}(x_{k-1} \mid x_k)} &= \frac{p_{k-1}(x_{k-1})\,e^{r_{k-1}(x_{k-1})}}{p_k(x_k)\,e^{r_k(x_k)}} \cdot \frac{\overrightarrow{F}_{k|k-1}(x_k \mid x_{k-1})}{\overleftarrow{Q}_{k-1|k}(x_{k-1} \mid x_k)} \\ &= \frac{e^{r_{k-1}(x_{k-1})}}{e^{r_k(x_k)}} \cdot \frac{p_{k-1}(x_{k-1})}{p_k(x_k)} \cdot \frac{p_k(x_k)\,\overleftarrow{P}_{k-1|k}(x_{k-1} \mid x_k)}{p_{k-1}(x_{k-1})\,\overleftarrow{Q}_{k-1|k}(x_{k-1} \mid x_k)} \\ &= \frac{e^{r_{k-1}(x_{k-1})}}{e^{r_k(x_k)}} \cdot \frac{\overleftarrow{P}_{k-1|k}(x_{k-1} \mid x_k)}{\overleftarrow{Q}_{k-1|k}(x_{k-1} \mid x_k)}, \end{aligned} \tag{27}$$

which is Eq. (10); it is finite $\overleftarrow{Q}_{k-1|k}$-a.s. because $\overleftarrow{P}_{k-1|k} \ll \overleftarrow{Q}_{k-1|k}$. The boundary conditions give $\gamma_K = p_{\mathrm{ref}} = \pi_K$, which is normalised, and $\gamma_0 = p_0 e^{r_0}$ with $Z_0 < \infty$. Algorithm 1 is therefore the SMC sampler of Section 3.1 with adaptive resampling and π_k-invariant rejuvenation, and its output is $\hat{\gamma}_0$ of Eq. (57), evaluated at $f = e^{-r_0}\mathbf{1}_E$. By the unbiasedness of $\hat{\gamma}_0[f]$ under adaptive resampling and invariant moves (Del Moral et al., 2012),

$$\mathbb{E}[\hat{p}_0[E]] = \gamma_0\big[e^{-r_0}\mathbf{1}_E\big] = \int p_0(x)\,e^{r_0(x)}e^{-r_0(x)}\mathbf{1}_E(x)\,\mathrm{d}x = p_0[E], \tag{28}$$

which is also $\mathbb{E}[\hat{p}_0^{\mathrm{MC}}[E]]$ since $X_0^n \sim p_0$ in Eq. (3). This holds for every K, N, reward sequence and resampling schedule. For the sample, $\hat{p}_0[\cdot \mid E]$ is $\hat{\pi}_0$ reweighted by $e^{-r_0}\mathbf{1}_E$ and normalised, i.e. the self-normalised estimator of Eq. (57) for the target $\gamma_0 e^{-r_0}\mathbf{1}_E \propto p_0[\cdot \mid E]$, so it converges almost surely to $p_0[\cdot \mid E]$ as $N \to \infty$ by the same result. For the limit, $p_0 \Rightarrow p_{\mathrm{data}}$ as $K \to \infty$ under the standing assumptions of Section 2, and $p_{\mathrm{data}}[\partial E] = 0$, so $p_0[E] \to p_{\mathrm{data}}[E]$ by the portmanteau theorem. □

A.4 Proof of Theorem 2

Proof. Step 1: the estimator is an importance-sampling average. Without resampling, $\hat{Z} = 1$ and $\hat{p}_0[E] = \frac{1}{N}\sum_{n=1}^N W^n$ with $W^n := W(X_{K:0}^n)$ i.i.d. under $\mathbb{Q}_{0:K}$, where by the telescoping identity Eq. (22)

$$\begin{aligned} W(x_{K:0}) &:= \prod_{k=1}^K g_{k-1|k}(x_{k-1}, x_k)\;\frac{\mathbf{1}_E(x_0)}{e^{r_0(x_0)}} \\ &= \mathbf{1}_E(x_0)\,\frac{d\mathbb{P}_{0:K}}{d\mathbb{Q}_{0:K}}(x_{K:0}) \\ &= p_0[E]\,\frac{d\mathbb{P}^E_{0:K}}{d\mathbb{Q}_{0:K}}(x_{K:0}). \end{aligned} \tag{29}$$

Hence $\mathbb{E}_{\mathbb{Q}_{0:K}}[W] = p_0[E]$ and

$$\frac{\mathrm{Var}[\hat{p}_0[E]]}{p_0[E]^2} = \frac{1}{N}\Big(\mathbb{E}_{\mathbb{Q}_{0:K}}\Big[\Big(\frac{d\mathbb{P}^E_{0:K}}{d\mathbb{Q}_{0:K}}\Big)^2\Big] - 1\Big) = \frac{\chi^2(\mathbb{P}^E_{0:K}\,\|\,\mathbb{Q}_{0:K})}{N}. \tag{30}$$

Step 2: change of measure through the tilted chain. Both $\mathbb{P}^E_{0:K}$ and $\mathbb{P}^\star_{0:K}$ are tilts of $\mathbb{P}_{0:K}$ by functions of x_0, so

$$\frac{d\mathbb{P}^E_{0:K}}{d\mathbb{P}^\star_{0:K}}(x_{K:0}) = \frac{Z_0\,e^{-r_0(x_0)}\mathbf{1}_E(x_0)}{p_0[E]} \;\le\; \frac{Z_0}{c\,p_0[E]}, \tag{31}$$

using $e^{r_0} \ge c$ on E. Therefore

$$\begin{aligned} 1 + \chi^2(\mathbb{P}^E_{0:K}\|\mathbb{Q}_{0:K}) &= \mathbb{E}_{\mathbb{Q}_{0:K}}\Big[\Big(\frac{d\mathbb{P}^E_{0:K}}{d\mathbb{P}^\star_{0:K}}\Big)^2\Big(\frac{d\mathbb{P}^\star_{0:K}}{d\mathbb{Q}_{0:K}}\Big)^2\Big] \\ &\le \Big(\frac{Z_0}{c\,p_0[E]}\Big)^2\big(1 + \chi^2(\mathbb{P}^\star_{0:K}\|\mathbb{Q}_{0:K})\big), \end{aligned} \tag{32}$$

and substituting into Eq. (30) gives Eq. (15).

Step 3: the equality case. If $e^{r_0} = c\,\mathbf{1}_E$, then $Z_0 = c\,p_0[E]$ and Eq. (31) is identically one, so $\mathbb{P}^\star_{0:K} = \mathbb{P}^E_{0:K}$ and Eq. (15) holds with equality.

Step 4: the floor. Under $\mathbb{Q}_{0:K}$, $X_K \sim p_{\mathrm{ref}}$; under $\mathbb{P}^\star_{0:K}$, $X_K \sim \pi^\star_K$ by Lemma 2. The data-processing inequality for the marginal at step K gives

$$\chi^2(\mathbb{P}^\star_{0:K} \| \mathbb{Q}_{0:K}) \;\geq\; \chi^2(\pi^\star_K \,\|\, p_{\mathrm{ref}}). \tag{33}$$

For the upper bound, $\mathbb{P}^\star_{0:K}$ and $\mathbb{P}_{0:K}$ share their transitions and differ only in the law of X_0, and $X_K \sim p_{\mathrm{ref}}$ under $\mathbb{P}_{0:K}$, so data processing for the marginal at K and then the fact that $d\mathbb{P}^\star_{0:K}/d\mathbb{P}_{0:K}$ depends on x_0 alone give

$$\chi^2(\pi^\star_K \,\|\, p_{\mathrm{ref}}) \;\leq\; \chi^2(\mathbb{P}^\star_{0:K} \,\|\, \mathbb{P}_{0:K}) = \mathbb{E}_{\mathbb{P}_{0:K}}\Big[\Big(\frac{e^{r_0(X_0)}}{Z_0}\Big)^2\Big] - 1 = \chi^2(\pi_0 \,\|\, p_0). \tag{34}$$

If $\overleftarrow{Q}_{k-1|k} = \overleftarrow{P}^\star_{k-1|k}$ for all k, the product in Eq. (24) is one, $d\mathbb{P}^\star_{0:K}/d\mathbb{Q}_{0:K}$ is a function of x_K alone, equal to $\pi^\star_K(x_K)/p_{\mathrm{ref}}(x_K)$, and the first inequality is an equality, which is Eq. (33) with its equality case. If X_K is independent of X_0 under $\mathbb{P}_{0:K}$ then $r^\star_K$ is constant, so $\pi^\star_K = p_{\mathrm{ref}}$ and the minimum is zero.

Finally, for the CDF reward of Eq. (16), $\phi \geq \tau$ on E gives $e^{r_{0,\delta}} \geq \Phi(0) = \frac{1}{2}$ there, so $c = \frac{1}{2}$, and $Z_0 = p_0[e^{r_{0,\delta}}] \to p_0[E]$ as $\delta \downarrow 0$ by dominated convergence. □

Remark 1. Initialising from $\pi^\star_K$ instead of p_{ref}, with kernels $\overleftarrow{P}^\star_{k-1|k}$ and weights $w^n_K = 1$, would make every incremental weight equal to one and the estimator deterministic, the classical zero-variance statement for Doob's h-transform. It is circular: the normaliser of $\pi^\star_K$ is Z_0, which for the indicator reward is the quantity being estimated. The boundary condition $r_K = 0$ is what allows Algorithm 1 to start from p_{ref} and Eq. (33) is the price this incurs.

A.5 The optimal reward recursion

We recall, for completeness, why the Doob rewards are the optimal intermediate targets (Guarniero et al., 2017; Johansen & Doucet, 2008; Pitt & Shephard, 1999). Fix k and r_{k-1}, let $X_{k-1} \mid X_k \sim \overleftarrow{Q}_{k-1|k}$, and define

$$r^{\mathrm{opt}}_k(x_k) := \log \int \overleftarrow{P}_{k-1|k}(\mathrm{d}x_{k-1} \mid x_k)\, e^{r_{k-1}(x_{k-1})}. \tag{35}$$

From Eq. (10), for every proposal with $\overleftarrow{P}_{k-1|k} \ll \overleftarrow{Q}_{k-1|k}$,

$$\begin{aligned}\mathbb{E}\big[g_{k-1|k} \mid X_k = x_k\big] &= e^{-r_k(x_k)} \int \overleftarrow{Q}_{k-1|k}(\mathrm{d}x_{k-1} \mid x_k)\, \frac{\overleftarrow{P}_{k-1|k}(x_{k-1} \mid x_k)}{\overleftarrow{Q}_{k-1|k}(x_{k-1} \mid x_k)}\, e^{r_{k-1}(x_{k-1})} \\ &= \exp\big(r^{\mathrm{opt}}_k(x_k) - r_k(x_k)\big).\end{aligned} \tag{36}$$

By the law of total variance,

$$\mathrm{Var}[g_{k-1|k}] = \mathrm{Var}\big[\mathbb{E}[g_{k-1|k} \mid X_k]\big] + \mathbb{E}\big[\mathrm{Var}[g_{k-1|k} \mid X_k]\big]. \tag{37}$$

The reward r_k enters $g_{k-1|k}$ only through the factor $e^{-r_k(x_k)}$, a function of x_k alone, so the second term does not depend on r_k, while the first vanishes if and only if $r_k = r^{\mathrm{opt}}_k + \mathrm{const}$. Thus r^{opt}_k minimises $\mathrm{Var}[g_{k-1|k}]$ over r_k for every proposal. The second term vanishes if and only if $x_{k-1} \mapsto e^{r_{k-1}(x_{k-1})}\overleftarrow{P}_{k-1|k}(x_{k-1} \mid x_k)/\overleftarrow{Q}_{k-1|k}(x_{k-1} \mid x_k)$ is $\overleftarrow{Q}_{k-1|k}$-a.s. constant, i.e. $\overleftarrow{Q}_{k-1|k}(\mathrm{d}x_{k-1} \mid x_k) \propto \overleftarrow{P}_{k-1|k}(\mathrm{d}x_{k-1} \mid x_k)e^{r_{k-1}(x_{k-1})}$, which with $r_k = r^{\mathrm{opt}}_k$ is the twisted kernel $\overleftarrow{P}^\star_{k-1|k}$ and gives $g_{k-1|k} \equiv 1$. Finally, if $r_{k-1} = r^\star_{k-1}$ then $r^{\mathrm{opt}}_k = r^\star_k$ by the tower identity Eq. (21), so iterating from $r_0 = r^\star_0$ recovers the Doob rewards at every step.

A.6 Proof of Lemma 1

Proof. Write $\mu := m_k(x_k)$, $\Delta := c_k \nabla_x r_k(x_k)$ and $v := v_k$. Under Eq. (11) both kernels are Gaussian in x_{k-1} with covariance vI, so

$$\log \frac{\overleftarrow{P}_{k-1|k}(x_{k-1} \mid x_k)}{\overleftarrow{Q}_{k-1|k}(x_{k-1} \mid x_k)} = -\frac{\|x_{k-1} - \mu\|^2}{2v} + \frac{\|x_{k-1} - \mu - \Delta\|^2}{2v} = \frac{\|\Delta\|^2}{2v} - \frac{\langle \Delta,\, x_{k-1} - \mu\rangle}{v}. \tag{38}$$

Substituting $x_{k-1} = \mu + \Delta + \sqrt{v}\,\epsilon_k$, the noise drawn when sampling $\overleftarrow{Q}_{k-1|k}$,

$$\log \frac{\overleftarrow{P}_{k-1|k}(x_{k-1} \mid x_k)}{\overleftarrow{Q}_{k-1|k}(x_{k-1} \mid x_k)} = \frac{\|\Delta\|^2}{2v} - \frac{\|\Delta\|^2}{v} - \frac{\langle \Delta, \epsilon_k \rangle}{\sqrt{v}}$$
$$= -\frac{\zeta_k^2}{2}\big\|\nabla_x r_k(x_k)\big\|^2 - \zeta_k \big\langle \nabla_x r_k(x_k), \epsilon_k \big\rangle, \tag{39}$$

with $\zeta_k = c_k/\sqrt{v_k}$. Adding the reward tilt $r_{k-1}(x_{k-1}) - r_k(x_k)$ of Eq. (10) gives Eq. (12). No approximation is made; the integrator enters only through (c_k, v_k). For Euler-Maruyama on a step of length Δt_k, $m_k(x_k) = x_k - [\mu_{t_k}(x_k) - \sigma_{t_k}^2 \nabla_x \log p_{t_k}(x_k)]\Delta t_k$, the guidance shifts the mean by $c_k \nabla_x r_k(x_k)$ with $c_k = \sigma_{t_k}^2 \Delta t_k$, and $v_k = \sigma_{t_k}^2 \Delta t_k$, so $\zeta_k = \sigma_{t_k}\sqrt{\Delta t_k}$. The exponential-integrator constants are derived in App. D. □

Scope. Three cases fall outside Lemma 1. Multi stage schemes such as the stochastic Heun sampler of (Karras et al., 2022) or higher order stochastic solvers of (Gonzalez et al., 2023) evaluate the network at an additional noise-dependent interval point, so the kernel mean is not affine in a single score evaluation. Deterministic solvers have $v_k = 0$, giving degenerate kernels and $\zeta_k \to \infty$. Anisotropic or per dimension noise makes v_k a matrix, and the scalar ζ_k is now a matrix.

B Diffusion model details

Training objective. The score network $s_\theta(x_t, t)$ approximates the marginal score $\nabla_{x_t} \log p_t$ and is trained by denoising score matching (Song et al., 2021b),

$$\mathcal{L}(\theta) = \mathbb{E}_t\, \mathbb{E}_{x_0 \sim p_0}\, \mathbb{E}_{x_t \sim p(x_t|x_0)} \Big[\lambda_t \big\| s_\theta(x_t, t) - \nabla_{x_t} \log p(x_t \mid x_0) \big\|^2 \Big], \tag{40}$$

with weighting $\lambda_t > 0$ and $t \sim \mathcal{U}[0, 1]$. For a Gaussian forward kernel $p(x_t \mid x_0) = \mathcal{N}(\alpha_t x_0, \bar{\sigma}_t^2 I)$ the target is closed form, $\nabla_{x_t} \log p(x_t \mid x_0) = -\epsilon/\bar{\sigma}_t$, giving the equivalent noise-prediction objective. Score-, noise-, and velocity-prediction parametrisations are related by simple bijections (Gao et al., 2024).

VP-SDE schedule. We adopt the Variance-Preserving (VP) SDE (Song et al., 2021b), with $f_t(x) = -\frac{1}{2}\beta(t)x$ and $\sigma_t = \sqrt{\beta(t)}$ for a prescribed $\beta(t)$. The forward kernel then admits the closed form,

$$X_t = \alpha_t X_0 + \bar{\sigma}_t \epsilon, \qquad \epsilon \sim \mathcal{N}(0, I)$$

with,

$$\alpha_t = \exp(-\tfrac{1}{2}\int_0^t \beta(s)\, ds), \qquad \bar{\sigma}_t^2 = 1 - \alpha_t^2. \tag{41}$$

In experiments we use the linear schedule $\beta(t) = \beta_{\min} + t(\beta_{\max} - \beta_{\min})$, $\beta_{\min} = 0.1$, $\beta_{\max} = 20$ or a cosine schedule.

Posterior mean (Tweedie). The posterior mean of the clean state given a noisy x_t is available in closed form from the score (Efron, 2011),

$$\hat{x}_{0|t}(x_t) = \frac{x_t + \bar{\sigma}_t^2\, s_\theta(x_t, t)}{\alpha_t}. \tag{42}$$

We use $\hat{x}_{0|t}$ as the mean of the Gaussian denoising-posterior approximation in Section 3.2.

Posterior-variance proxy (ΠGDM). For the covariance of the Gaussian denoising posterior we use the ΠGDM proxy, which is exact when the data distribution is a single Gaussian and is borrowed as a tractable surrogate otherwise (Song et al., 2023). Per dimension,

$$s_t^2 = \frac{\bar{\sigma}_t^2\, V_{\text{data}}}{\alpha_t^2\, V_{\text{data}} + \bar{\sigma}_t^2}, \tag{43}$$

with α_t^2 and $\bar{\sigma}_t^2 = 1 - \alpha_t^2$ the VP-SDE marginal noise variance above (note $\bar{\sigma}_t$ is the marginal noise std, distinct from the diffusion coefficient $\sigma_t = \sqrt{\beta(t)}$), and $V_{\text{data}} = \mathrm{Var}(x_0)$ the marginal

data variance (for the GMM benchmark, $\sum_k w_k(\Sigma_{k,11} + \mu_{k,1}^2) - (\sum_k w_k \mu_{k,1})^2$ along the read-out dimension). At early times ($t \to T$) $\alpha_t \to 0$, $\bar{\sigma}_t \to 1$ and $s_t^2 \to V_{\text{data}}$ (full prior uncertainty); at late times ($t \to 0$) $\bar{\sigma}_t^2 \to 0$ and $s_t^2 \to 0$ (the posterior collapses to the Tweedie point estimate). Applied per dimension this yields the diagonal posterior covariance,

$$\Sigma_t = \text{diag}\left(s_{t,1}^2, \ldots, s_{t,d}^2\right), \tag{44}$$

with V_{data} evaluated along each coordinate ($V_i = 1$ for standardised data, otherwise estimated empirically).

C Sequential Methods

For completeness we provide an overview of classical sequential methods that are used in rare-event simulation, such as annealed importance sampling, splitting and sequential Monte Carlo (SMC). These are methods used to bridge the gap from a tractable reference π_K to the tilted target π_0 of Eq. (5) through a sequence of overlapping intermediate distributions (Del Moral et al., 2006; Kahn & Harris, 1951; Neal, 2001). We briefly expand on this and highlight their similarities, but refer the readers to (Moral, 2004) for a full exposition under the *Feynman-Kac* formulation. Additionally, see (Del Moral et al., 2006; Naesseth et al., 2019) for details on the SMC machinery. Formally the sequence of distributions defines a *path-measure* over the space of all trajectories particles can take, but we omit this rigour for brevity.

Annealed Importance Sampling (AIS) and Sequential Monte Carlo (SMC) build this path by tempering the reward (Del Moral et al., 2006; Neal, 2001). One such sequence of distributions tempers the reward, (Del Moral et al., 2006; Neal, 2001),

$$\pi_k(x) = \frac{1}{Z_k} p_0(x)\, e^{\beta_k r(x)}, \qquad 1 = \beta_0 > \beta_1 > \cdots > \beta_K = 0. \tag{45}$$

Rather than targeting the constrained target π_0 directly, tempering gradually sharpens the reward landscape until the chain converges to the target distribution. Additionally, SMC utilises resampling along the trajectory to prune low-weight particles and multiply those with higher weights (Del Moral et al., 2006; Doucet et al., 2000).

Multilevel splitting takes another approach, where the sequence is constructed using a sequence of nested rare event sets (Glasserman et al., 1999; Kahn & Harris, 1951). Taking the canonical example of threshold exceedance, for a strictly decreasing ladder $\tau_0 = \tau > \tau_1 > \cdots > \tau_K$, the event set is defined as $E_k = \{x : \phi(x) \geq \tau_k\}$. The sequence is defined to be,

$$E_0 = E \subset E_1 \subset \cdots \subset E_K, \qquad \pi_k(x) = \frac{1}{Z_{k,\tau}} p_0(x) \mathbf{1}\{\phi(x) \geq \tau_k\}, \tag{46}$$

where E_K is a region readily accessible from the state space. The aim of splitting is to estimate the rare event probability at the final τ, i.e. $Z_{0,\tau} = \int p_0(x)\, \mathbf{1}\{\phi(x) \geq \tau_0\}\, dx = p_0[E]$. We show that all of these methods compute this in the same way. All methods work by transporting particles along the sequence, SMC and splitting additionally resample particles to focus computation on high-weight trajectories (i.e. trajectories that particles undertake that are more likely under the possible sequence of trajectories they could have taken)(Del Moral et al., 2006; Glasserman et al., 1999; Kahn & Harris, 1951). Intuitively, multilevel splitting is a form of SMC/AIS with a hard indicator function replacing a smoothed distribution and the schemes only differ in their construction of their sequences (Moral, 2004).

C.1 Estimating the normalisation constant

Having constructed the sequence, the quantity of interest is the terminal normalisation constant Z_0, which for the rare-event target is $Z_{0,\tau}$. We show explicitly, that splitting estimates the probability estimate in the same way as standard SMC/AIS. Please see (Del Moral et al., 2006; Glasserman et al., 1999; Kahn & Harris, 1951; Neal, 2001) for further details.

For a sequence of distributions, $\pi_K \to \cdots \to \pi_0$ of Section 3.1, let $\pi_k = \frac{1}{Z_k}\gamma_k$ be the normalised density with the normalisation constant $Z_k = \int \gamma_k$. Each step along the trajectory incurs an incremental importance weight that is, in general, a ratio of forward and backward transition kernels

between the two distributions (Del Moral et al., 2006). A particle is propagated from π_k to π_{k-1} by the proposal kernel $\vec{B}_{k-1|k}(x_{k-1} \mid x_k)$, and the auxiliary kernel $\overleftarrow{F}_{k|k-1}(x_k \mid x_{k-1})$ running in the opposite direction defines the incremental weight of Eq. (7) (Del Moral et al., 2006),

$$g_{k-1|k}(x_{k-1}, x_k) \;=\; \frac{\gamma_{k-1}(x_{k-1})\,\overleftarrow{F}_{k|k-1}(x_k \mid x_{k-1})}{\gamma_k(x_k)\,\vec{B}_{k-1|k}(x_{k-1} \mid x_k)}, \tag{47}$$

Choosing the proposal to be a π_{k-1}-invariant MCMC kernel, as in the rejuvenation move of Section 3.1, with the auxiliary kernel $\overleftarrow{F}_{k|k-1}$ taken as its time reversal, the kernels cancel and the weight reduces to the target ratio $g_{k-1|k} = \gamma_{k-1}(x_k)/\gamma_k(x_k)$ (Del Moral et al., 2006; Neal, 2001), which we use throughout this subsection. The ratio of two normalisation constants Z_{k-1}/Z_k is estimated via the Monte Carlo average of the incremental weights, i.e.,

$$\mathbb{E}_{\pi_k}[g_{k-1|k}] = \int \frac{\gamma_{k-1}(x)}{\gamma_k(x)}\,\frac{\gamma_k(x)}{Z_k}\,dx = \frac{Z_{k-1}}{Z_k}, \tag{48}$$

with the standard empirical estimator,

$$\widehat{\frac{Z_{k-1}}{Z_k}} = \frac{1}{N}\sum_{i=1}^{N} g_{k-1|k}^{(i)}, \qquad x^{(i)} \sim \pi_k. \tag{49}$$

Hence the normaliser Z_0 follows from the telescoping product with the corresponding estimator given by,

$$\hat{Z}_0 = Z_K \prod_{k=1}^{K} \Big(\frac{1}{N}\sum_{i=1}^{N} g_{k-1|k}^{(i)}\Big), \tag{50}$$

where Z_K is known, e.g. for score based models $Z_K = 1$ as $\gamma_K = p_1$ is the normalised Gaussian prior (Ho et al., 2020; Song et al., 2021b).

Tempering (AIS/SMC) Annealed importance sampling and SMC interpolate by tempering the reward, $\gamma_k = p_0\, e^{\beta_k r}$, with $1 = \beta_0 > \cdots > \beta_K = 0$, so the incremental weight is given by

$$g_{k-1|k} = e^{(\beta_{k-1}-\beta_k)\,r(x_k)}, \qquad \frac{Z_{k-1}}{Z_k} = \mathbb{E}_{\pi_k}\big[e^{(\beta_{k-1}-\beta_k)\,r}\big]. \tag{51}$$

Each normalisation ratio is the average of this factor under the current target. When consecutive temperatures are close, i.e. $\beta_{k-1} - \beta_k$ small, the weights stay close to one for all particles and the ratio is estimated reliably. A single large step, by contrast can produce weights spanning many orders of magnitude - collapsing the effective sample size where the final estimate will be dominated by only a few particles that have the largest weights (Neal, 2001).

Splitting Multilevel splitting interpolates instead with hard indicators, i.e. $\gamma_k = p_0\,\mathbf{1}\{x \in E_k\}$, for a shrinking sequence $E_K \supset \cdots \supset E_0$. The incremental weight collapses to,

$$g_{k-1|k} = \frac{p_0 \mathbf{1}\{x \in E_{k-1}\}}{p_0 \mathbf{1}\{x \in E_k\}} = \mathbf{1}\{x \in E_{k-1}\}, \tag{52}$$

for which the ratio of normalisation constant is given by,

$$\frac{Z_{k-1}}{Z_k} = \mathbb{E}_{\pi_k}\big[\mathbf{1}\{x \in E_{k-1}\}\big] = \Pr[X \in E_{k-1} \mid X \in E_k] =: \alpha_k, \tag{53}$$

so each ratio is the conditional probability of reaching the next level, estimated by the surviving fraction, $\hat{\alpha}_k = N_{X\in E_{k-1}}/N_{X\in E_k}$. Thus the telescoping product of Eq. (50) for the threshold exceedance given in Eq. (46) becomes,

$$Z_0 = Z_K \prod_{k=1}^{K} \alpha_k \;=\; \Pr[X > \tau_K] \prod_{k=1}^{K} \Pr[X > \tau_{k-1} \mid X > \tau_k], \tag{54}$$

This recovers the rare event probability estimate of the multilevel splitting algorithm (Morio et al., 2014; Rubino et al., 2009).

Both methods estimate the same object via a product of incremental ratios along the bridge where consecutive levels overlap, instead of the single rare-event ratio estimated in one step. This is precisely the failure that accumulating importance weights over the trajectory sidesteps. The diffusion process supplies the sequence of transition kernels and the incremental weights (Doucet et al., 2022; Vargas et al., 2023) carry their own normalisation avoiding a separate estimate of Z_0, see Appendix C.3.

C.2 GENERIC SMC ALGORITHM

Given an unnormalised density γ_0 with a normalised form $\pi_0 := \gamma_0/\gamma_0[1]$, we seek to estimate integrals $\gamma_0[f]$ and to sample from π_0. SMC samplers (Del Moral et al., 2006) transport a population of N weighted particles from a normalised reference γ_K down to γ_0 along a sequence $\gamma_K, \dots, \gamma_0$, with $\pi_k := \gamma_k/\gamma_k[1]$.

Algorithm. We begin by initialising N samples, $X_K^n \sim p_{\mathrm{ref}}$ with weights $w_K^n = 1$. At each step k, move each particle with a backward proposal kernel $\overleftarrow{B}_{k-1|k}$ and update its weight with the incremental weight $g_{k-1|k}$,

$$X_{k-1}^n \sim \overleftarrow{B}_{k-1|k}(\mathrm{d}x_{k-1} \mid X_k^n), \qquad w_{k-1}^n = w_k^n \, g_{k-1|k}\big(X_{k-1}^n, X_k^n\big), \tag{55}$$

$$g_{k-1|k}(x_{k-1}, x_k) = \frac{\gamma_{k-1}(x_{k-1}) \, \overrightarrow{F}_{k|k-1}(x_k \mid x_{k-1})}{\gamma_k(x_k) \, \overleftarrow{B}_{k-1|k}(x_{k-1} \mid x_k)}, \tag{56}$$

where $\overrightarrow{F}_{k|k-1}$ is a forward auxiliary kernel running from $k-1$ to k. Any kernels can be selected, provided we can sample from $\overleftarrow{B}_{k-1|k}$ and evaluate $g_{k-1|k}$ pointwise. When the effective sample size $\mathrm{ESS}(w_{k-1}) := \|w_{k-1}\|_1^2/\|w_{k-1}\|_2^2$ drops below ηN, we resample, replacing each particle by an ancestor drawn in proportion to its weight and resetting the weights to one (Chopin et al., 2022), and rejuvenate using a π_{k-1}-invariant MCMC kernel.

Estimators. Let $\hat{Z}$ be the product, over the resampling steps so far, of the average weight $\frac{1}{N}\sum_n w_k^n$ just before the reset ($\hat{Z} = 1$ if none). The particle system at step k defines the weighted approximations

$$\hat{\gamma}_0[f] := \hat{Z} \, \frac{1}{N} \sum_{n=1}^N w_0^n \, f(X_0^n), \qquad \hat{\pi}_0(\mathrm{d}x) := \frac{\sum_{n=1}^N w_0^n \, \delta_{X_0^n}(\mathrm{d}x)}{\sum_{n=1}^N w_0^n}. \tag{57}$$

$\hat{\gamma}_0[f]$ is *unbiased*, $\mathbb{E}[\hat{\gamma}_k[f]] = \gamma_k[f]$ for every f. The second is biased for finite N but converges almost surely to π_k as $N \to \infty$ (Del Moral et al., 2006).

C.3 WHY ANNEALING BECOMES NECESSARY

We now motivate why annealing is required and where importance sampling alone is insufficient. Even with *exact* samples from the tilted target π_0 of Eq. (5), taken with $r_0 = r_{0,\delta}$ or the mollified reward. The naive endpoint (self-normalised) estimator of $p_0[E]$ is badly behaved for estimating the normalisation constant, motivating the use of annealing, either via splitting or accumulated importance weights.

Given a smooth reward $r_{0,\delta}(x)$, suppose we can draw exact samples $x \sim \pi_0$ from the tilted distribution, $\pi_0 = \gamma_0/Z_0$ with $\gamma_0 = p_0 e^{r_{0,\delta}}$. The rare-event probability under the base distribution p_0 can then be written as,

$$\begin{aligned} p_0[E] = \mathbb{E}_{\pi_0}\left[\mathbf{1}\{\phi(x) \geq \tau\} \, \frac{p_0(x)}{\pi_0(x)}\right] &= \mathbb{E}_{\pi_0}\left[\mathbf{1}\{\phi(x) \geq \tau\} \, \frac{p_0(x)}{p_0(x) e^{r_{0,\delta}(x)}/Z_0}\right] \\ &= \mathbb{E}_{\pi_0}\left[\mathbf{1}\{\phi(x) \geq \tau\} \, Z_0 e^{-r_{0,\delta}(x)}\right], \end{aligned} \tag{58}$$

where $w(x) := p_0(x)/\pi_0(x) = Z_0 e^{-r_{0,\delta}(x)}$ is the importance weight, and the normaliser Z_0 is estimated from the same samples via,

$$\frac{1}{Z_0} = \frac{1}{Z_0}\int p_0 dx = \frac{1}{Z_0}\int p_0(x) e^{r_{0,\delta}(x)} e^{-r_{0,\delta}(x)} = \int \pi_0 e^{-r_{0,\delta}(x)} = \mathbb{E}_{\pi_0}[e^{-r_{0,\delta}(x)}]. \tag{59}$$

Substituting Eq. (59) into Eq. (58) yields the self-normalised importance-sampling (SNIS) estimator (Owen, 2013),

$$\hat{p}_0[E] \;=\; \frac{\sum_{i=1}^N \mathbf{1}\{\phi(x_i) \geq \tau\} \, e^{-r_{0,\delta}(x_i)}}{\sum_{i=1}^N e^{-r_{0,\delta}(x_i)}}, \qquad x_i \sim \pi_0(x) \tag{60}$$

The numerator and denominator of (60) behave very differently. The numerator is only affected by the region that π_0 approximates well, since the indicator $\mathbf{1}\{\phi(x) \geq \tau\}$ provides a "shielding" effect.

The denominator, by contrast is an expectation of the normalisation constant Z_0 which is affected by the entire domain of p_0. As the tilt is designed to concentrate on the rare event set, we are effectively using the rare-event tilted distribution π_0 to estimate a function of p_0. This naturally has a low ESS and yields a high-variance estimate of the normaliser Z_0. Even given perfect samples from π_0 we essentially have another "rare-event" expectation in the denominator where no particles of π_0 fall into the bulk. The denominator integrand $w(x) = \frac{p_0(x)}{\pi_0(x)} = Z_0 e^{-r_{0,\delta}(x)}$ is largest in the bulk, where p_0 holds most of its mass but the tilt π_0 has almost none, *exactly the region π_0 is designed to avoid*. The numerator $\mathbb{E}_q[\mathbf{1}_{\text{rare}} e^{-r_{0,\delta}}]$ stays bounded as it lives where q samples effectively, whereas the denominator $\mathbb{E}_q[e^{-r_{0,\delta}}]$ needs exactly the bulk samples that q avoids. Hence, the ESS for estimating the denominator becomes worse by design and this contributes a high variance estimate of Z_0! We could in practice break this down into a series of transition importance distributions, which leads us back to splitting/annealing of Section C (Del Moral et al., 2006; Morio et al., 2014; Neal, 2001).

C.3.1 Empirical evidence on the 2D GMM

If we draw exact samples from π_0, via Rejection Sampling or analytically, we illustrate that we still *cannot* estimate Z_0. For the 2D GMM case with a log-CDF tilting, we have access to the analytical normalisation constant, the convolution of a Gaussian with a Gaussian CDF simply inflates the variance. The normaliser of the tilted distribution as given in Eq. (98) repeated here again is given by,

$$Z_{0,\delta} = \int \Phi(f)\mathcal{N}(f \mid \mu, \sigma^2)\, df = \Phi\left(\frac{\mu}{\sqrt{1+\sigma^2}}\right), \tag{61}$$

with $f \mapsto \frac{x-\tau}{\delta}$, $\mu \mapsto \frac{\mu_i - \tau}{\delta}$ and $\sigma^2 \mapsto \frac{v_i}{\delta^2}$ (see Section F). This lets us disentangle whether the failure is in the numerator or the denominator, knowing the analytical denominator and the true probability, the true numerator is recoverable. We draw a large number of samples from p_0 and use rejection sampling to generate samples from π_0. We find that a small number of particles drawn directly from π_0 suffices to estimate the rare-event probability *provided the denominator is known*. Conversely, if the denominator is unknown we cannot estimate the probability at all - the difficulty lies entirely in the denominator as seen in Table 2, which is exactly what motivates annealing (Section 3.1 & C).

For $\tau = 7.0$ and $\delta = 0.25$ the true $Z_{0,\delta} = 7.331 \times 10^{-3}$ from Eq. (61).

Table 2: Self-normalised importance sampling (SNIS) estimates vs. sample size (N).

N	$\hat{p}_{\text{SNIS}}$	Relative Bias	$\hat{Z}_{0,\text{SNIS}}$
100	4.6035×10^{-1}	$+6,593\%$	4.873×10^{-1}
1,000	3.5854×10^{-1}	$+5,113\%$	3.770×10^{-1}
10,000	3.1680×10^{-1}	$+4,506\%$	3.367×10^{-1}
25,000	1.9804×10^{-1}	$+2,779\%$	2.102×10^{-1}
100	6.9250×10^{-3}	$+0.68\%$	(analytical $Z_{0,\delta}$)

Note that $\hat{p}_{\text{SNIS}}$ differs from $\hat{Z}_{0,\text{SNIS}}$ because some particles do not pass through the numerator's indicator by design of the smoothing in Section 3. The goal of annealing/splitting (Section 3.1) is precisely to make the numerator easier to estimate.

C.4 Splitting estimator

In the case where the path weights are not readily accessible, either because they are not computed or if a large number of discretisation is infeasible, e.g. for a second order numerical integrator. We can fall back to a guided diffusion proposal to build a splitting/annealed probability estimator. In this case, the annealing bridge becomes the nested sequence of event sets defined in Eq. (46). For the threshold exceedance example given in Sec C, the conditional ratio between adjacent levels is given by,

$$\alpha_k \;=\; p_0[X > \tau_h \mid X > \tau_\ell] = \frac{p_0[X > \tau_h \cap X > \tau_\ell]}{p_0[X > \tau_\ell]} = \frac{p_0[X > \tau_h]}{p_0[X > \tau_\ell]}. \tag{62}$$

In the denoising-diffusion setting each conditional is sampled by transporting particles from the Gaussian prior to the corresponding target, this means that the probability, $\Pr[X > \tau_h]$ is estimated

by generating samples from the tilted distribution, $p(X \mid X > \tau_h)$, as highlighted in Section 3. Within this scheme α_k admits two estimators, depending on whether a single tilt or one tilt per level is used.

Formulation I: single-tilt, two-indicator estimator. Use a *single* proposal q_ℓ targeting the lower level τ_ℓ and estimate both exceedance probabilities from the same particles. With $\tau_\ell < \tau_h$,

$$\alpha_k = \frac{\mathbb{E}_{q_\ell}\big[\mathbf{1}\{\phi(X) > \tau_h\}\, e^{-r_\ell(X)}\big]}{\mathbb{E}_{q_\ell}\big[\mathbf{1}\{\phi(X) > \tau_\ell\}\, e^{-r_\ell(X)}\big]}, \tag{63}$$

where the shared normaliser Z_ℓ cancels between numerator and denominator, so it need not be estimated. The denominator is estimated where q_ℓ has mass (it targets τ_ℓ), so it is well-conditioned; the variance is dominated by the numerator, which lives where q_ℓ is sparse ($X > \tau_h$).

Formulation II: two-tilt transitioning. Use one proposal per level, $q_\ell \propto p_0\, e^{r_\ell}$ and $q_h \propto p_0\, e^{r_h}$, and estimate the numerator under q_h and the denominator under q_ℓ,

$$\alpha_k = \frac{Z_h}{Z_\ell}\, \frac{\mathbb{E}_{q_h}\big[\mathbf{1}\{\phi(X) > \tau_h\}\, e^{-r_h(X)}\big]}{\mathbb{E}_{q_\ell}\big[\mathbf{1}\{\phi(X) > \tau_\ell\}\, e^{-r_\ell(X)}\big]}. \tag{64}$$

Each expectation is now evaluated under a tilt matched to its own threshold, so neither term suffers the ESS collapse of Formulation I; the price is that the normalisers no longer cancel and the ratio Z_h/Z_ℓ must be estimated. Annealing on the output space, i.e. between $p_0 e^{r_\ell} \to p_0 e^{r_h}$, we can compute this ratio with the tempering estimator given in Eq. (51),

$$\frac{Z_h}{Z_\ell} = \mathbb{E}_{q_\ell}\big[e^{\,r_h(X) - r_\ell(X)}\big] \approx \frac{1}{N}\sum_i e^{\,r_h(x_i) - r_\ell(x_i)}, \qquad x_i \sim q_\ell, \tag{65}$$

which is reliable when adjacent tilts overlap. Both formulations require sufficient overlap of the thresholds, the only difference is Formulation I requires overlap between two hard indicators and through the smooth reward difference, $r_h - r_\ell$, in Formulation II.

Notable Caveats: Evolving directly from $p(X \mid X > \tau_l) \to p(X \mid X > \tau_h)$ is non-trivial without additional training. Unlike traditional splitting, we do not evolve/reuse particles across different thresholds (Morio et al., 2014), hence the above formulations are not a form of sequential methods. Instead they are a sequence of independent estimators sharing a compute budget. Additionally, any importance weights used assume that within the threshold set by the indicators, samples are well distributed along the tilted distribution. As such both estimators are sensitive to the quality of the guidance. Training based methods could be introduced such as Flow Matching (Albergo & Vanden-Eijnden, 2023; Lipman et al., 2023; Liu et al., 2023) or Schrödinger's Bridges (Bortoli et al., 2021; Shi et al., 2023) but these remain beyond the scope of this work.

D Weight computation

Let $\mathbb{P}$ and $\mathbb{Q}$ be the path-measures on $[0,1]$ of the base reverse-time SDE of Eq. (1) and the guided SDE of Eq. (9) respectively. Here we do not assume that $\mathbb{Q}$ is the ideal Doob's path measure $\mathbb{P}^*$ and the gap between the two is varaince in Theorem 2. Both measures start from $X_1 \sim p_{\text{ref}}$ and differ only the drift obtained by the reward gradient term, $-\sigma_t^2 \nabla_x r_t$ so Girsanov's theorem gives us an exact form of the Radon-Nikodym Derivative (Øksendal, 2003),

$$\frac{\mathrm{d}\mathbb{P}}{\mathrm{d}\mathbb{Q}} = \exp\Big(-\int_0^1 \sigma_t\, \nabla_x r_t(X_t) \cdot \mathrm{d}\overleftarrow{W}_t \;-\; \tfrac{1}{2}\int_0^1 \sigma_t^2\, \|\nabla_x r_t(X_t)\|^2\, \mathrm{d}t\Big), \tag{66}$$

where $\overleftarrow{W}_t$ is the reverse-time Wiener process driving the guided sampler. This expression is valid provided the change of measure exists at all. A sufficient condition for existence is given by Novikov's condition,

$$\mathbb{E}_{\mathbb{Q}}\Big[\exp\Big(\tfrac{1}{2}\int_0^1 \sigma_t^2\, \|\nabla_x r_t(X_t)\|^2\, \mathrm{d}t\Big)\Big] < \infty. \tag{67}$$

The discrete estimator of Algorithm 1 does not require this condition to be fulfilled as Theorem 1 requires only $\overleftarrow{P}_{k-1|k} \ll \overleftarrow{Q}_{k-1|k}$ (App. A).

Discrete transition kernels In practice we sample on the grid $0 = t_0 < \cdots < t_K = 1$ of Section 2, with $t = 0$ the data distribution, writing x_k for the state at t_k so that step k moves $x_k \to x_{k-1}$. The path measures factorise into the backward transition kernels of Section 2 and the kernel we actually simulate (Wu et al., 2023),

$$\mathbb{P}_{0:K}(x_{0:K}) = p_{\mathrm{ref}}(x_K)\prod_{k=1}^{K}\overleftarrow{P}_{k-1|k}(x_{k-1} \mid x_k), \qquad \mathbb{Q}_{0:K}(x_{0:K}) = p_{\mathrm{ref}}(x_K)\prod_{k=1}^{K}\overleftarrow{Q}_{k-1|k}(x_{k-1} \mid x_k), \tag{68}$$

where $\overleftarrow{P}_{k-1|k}$ integrates the base reverse SDE of Eq. (1) and $\overleftarrow{Q}_{k-1|k}$ the guided SDE of Eq. (9) from t_k down to t_{k-1}. Both are initialised from the same Gaussian prior p_{ref} which cancels,

$$\frac{\mathrm{d}\mathbb{P}_{0:K}}{\mathrm{d}\mathbb{Q}_{0:K}} = \prod_{k=1}^{K}\frac{\overleftarrow{P}_{k-1|k}(x_{k-1} \mid x_k)}{\overleftarrow{Q}_{k-1|k}(x_{k-1} \mid x_k)}. \tag{69}$$

Incorporating a reward tilt $e^{r_0(x_0)}$ as a telescope product of indivual reward potentials,

$$R(x_{0:K}) = \prod_{k=1}^{K} G_k, \qquad G_k = \frac{e^{r_{k-1}(x_{k-1})}}{e^{r_k(x_k)}}, \tag{70}$$

which collapses to $R = \exp\big(r_0(x_0) - r_K(x_K)\big) = e^{r_0(x_0)}$ since $r_K = 0$. Multiply Eqs. (69) and (70) yileds,

$$e^{r_0(x_0)}\,\frac{\mathrm{d}\mathbb{P}_{0:K}}{\mathrm{d}\mathbb{Q}_{0:K}}(x_{K:0}) = \prod_{k=1}^{K} G_k\,\frac{\overleftarrow{P}_{k-1|k}(x_{k-1} \mid x_k)}{\overleftarrow{Q}_{k-1|k}(x_{k-1} \mid x_k)} = \prod_{k=1}^{K} g_{k-1|k}(x_{k-1}, x_k), \tag{71}$$

which recovers the incremental weight of Eq. (10) accumulated by Algorithm 1.

Weight Computation Recall from Lemma 1 that, for Gaussian transition kernels of Eq. (11), the incremental log weight is given by,

$$\log g_{k-1|k}(x_{k-1}, x_k) \;=\; \underbrace{r_{k-1}(x_{k-1}) - r_k(x_k)}_{\text{reward telescope}} \;-\; \underbrace{\frac{\zeta_k^2}{2}\|\nabla_x r_k(x_k)\|^2}_{\text{control cost}} \;-\; \underbrace{\zeta_k\langle \nabla_x r_k(x_k), \epsilon_k\rangle}_{\text{martingale correction}}. \tag{72}$$

As all path-measures we have introduced differ only in the drift, the change of measure between them is given by Girsanov's theorem which yields an exact Radon-Nikodym Derivative (Øksendal, 2003) and the above computation represents its discretised form. The control-cost is the price paid by the guided SDE for perturbing the base drift towards the rare-event set and a martingale correction which accounts for any "lucky" noise ϵ_k sampled along the guided step that carries the particle towards the set.

Remark. Nothing in this derivation constrains the r_k to be constant across time; and any family of schedules defined by $(\lambda(t), \tau(t), \delta(t))$ do not break the validity and the weight computation remains exact. The schedules enter the reward telescope of Eq. (70) only through the endpoints themselves $R = \exp\big(r_{t_0}(x_0) - r_{t_K}(x_K)\big)$, and the kernel ratio of Eq. (69) only through the guidance $\nabla_{x_t} r_t$.

D.1 ζ_k FOR COMMON INTEGRATORS

The first-order exponential integrator (Milstein & Tretyakov, 2004) of (Gonzalez et al., 2023) integrates the linear part of the reverse drift and the noise exactly, freezing only the network output across the step. Reparameterising the time increment in half-log Signal to Noise Ratio, $\ell_t := \log(\alpha_t/\bar{\sigma}_t)$; write $\Delta\ell_k = \ell_{t_{k-1}} - \ell_{t_k} \geq 0$ for its increment across a step.

Exponential Integrator (VP & $\hat{\epsilon}_\theta$ frozen) Restating their Eq. 14 (Gonzalez et al., 2023) in our notation and abbreviating the symbols,

$$A_k := \frac{\alpha_{t_{k-1}}}{\alpha_{t_k}}, \qquad D_k := 2\,\bar{\sigma}_{t_{k-1}}\big(e^{\Delta\ell_k} - 1\big), \qquad v_k := \bar{\sigma}^2_{t_{k-1}}\big(e^{2\Delta\ell_k} - 1\big). \tag{73}$$

The incremental reverse step is given by,

$$x_{k-1} = A_k\, x_k \;-\; D_k \hat{\epsilon}_\theta\big(x_k, t_k\big) \;+\; \sqrt{v_k}\epsilon_k, \qquad \epsilon_k \sim \mathcal{N}(0, I), \tag{74}$$

where $\hat{\epsilon}_\theta$ is the noise predicted by the network. This is related by a change of variable to the score network, $s_\theta = -\frac{\epsilon_\theta}{\bar{\sigma}}$ (Gao et al., 2024). Hence, the guidance update in score maps to the noise prediction as,

$$\begin{aligned} s_\theta &\to s_\theta + \nabla_x r_t, \\ \epsilon_\theta &\to \epsilon_\theta - \bar{\sigma}_{t_k} \nabla_x r_{t_k}(x_k). \end{aligned} \tag{75}$$

Therefore, the change of means between the two guided and unguided process is obtained by substituting Eq.(75) into Eq.(74) to get,

$$\mu_k^a - \mu_k^\nu = D_k \bar{\sigma}_{t_k} \nabla_x r_{t_k}(x_k) = c_k^{\mathrm{VP}} \nabla_x r_{t_k}(x_k), \tag{76}$$

where we have further equated this to the mean shift of Eq. (11). Hence, under the exponential first-order integrator and a VP noise schedule,

$$c_k^{\mathrm{VP}} = D_k \bar{\sigma}_{t_k} = 2\bar{\sigma}_{t_{k-1}} \bar{\sigma}_{t_k} (e^{\Delta \ell_k} - 1). \tag{77}$$

To obtain ζ_k, we make further substitutions,

$$\begin{aligned} \big(\zeta_k^{\mathrm{VP}}\big)^2 = \frac{(c_k^{\mathrm{VP}})^2}{v_k} &= \frac{4\,\bar{\sigma}_{t_{k-1}}^2 \bar{\sigma}_{t_k}^2 \big(e^{\Delta \ell_k} - 1\big)^2}{\bar{\sigma}_{t_{k-1}}^2 \big(e^{2\Delta \ell_k} - 1\big)} \\ &= \frac{4\,\bar{\sigma}_{t_k}^2 \big(e^{\Delta \ell_k} - 1\big)^2}{\big(e^{\Delta \ell_k} - 1\big)\big(e^{\Delta \ell_k} + 1\big)} \\ &= \frac{4\,\bar{\sigma}_{t_k}^2 \big(e^{\Delta \ell_k} - 1\big)}{e^{\Delta \ell_k} + 1} \\ &= 4\,\bar{\sigma}_{t_k}^2 \tanh(\Delta \ell_k / 2), \end{aligned} \tag{78}$$

so that $\zeta_k^{\mathrm{VP}} = 2\,\bar{\sigma}_{t_k} \sqrt{\tanh(\Delta \ell_k / 2)}$.

Exponential Integrator (VE) Under the VE schedule, $f_t = 0$ and $\alpha_t \equiv 1$. Following (Gonzalez et al., 2023) and rewriting the score network as a data-prediction $\hat{x}_{0|t} = x_t + \bar{\sigma}_t^2 s_\theta$ using Tweedie's formula (Efron, 2011; Gao et al., 2024) and freezing $\hat{x}_{0|t}$ along the increment, the reverse drift is given by,

$$x_{k-1} = \kappa_k x_k + (1 - \kappa_k)\hat{x}_{0|t} + \sqrt{v_k}\epsilon_k, \qquad v_k = \bar{\sigma}_{t_{k-1}}^2 (1 - \kappa_k), \tag{79}$$

where $\kappa_k := \frac{\bar{\sigma}_{t_{k-1}}^2}{\bar{\sigma}_{t_k}^2}$. Under a guided score the data prediction is perturbed to,

$$\begin{aligned} s_\theta &\to s_\theta + \nabla_x r_t, \\ \hat{x}_{0|t} &\to \hat{x}_{0|t} + \bar{\sigma}_{t_k}^2 \nabla_x r_{t_k}(x_k), \end{aligned} \tag{80}$$

hence, $c_k = (1 - \kappa_k)\bar{\sigma}_{t_k}^2$ and it follows,

$$\zeta_k^{\mathrm{VE},\hat{x}} = \bar{\sigma}_{t_k} \sqrt{e^{2\Delta \ell_k} - 1}, \tag{81}$$

with $\ell_t = -\log \bar{\sigma}_t$ when $\alpha_t \equiv 1$.

Freezing the noise-prediction $\hat{\epsilon}_k$ would lead to an exact quadrature scheme of the noise term, v_k. Under a VE schedule $f_t = 0$, so the reverse drift carries no part linear in x and, using $\sigma_t^2 / \bar{\sigma}_t = 2\dot{\bar{\sigma}}_t$, the frozen integrand may be integrated in closed form (Gonzalez et al., 2023),

$$x_{k-1} = x_k - D_k^{\mathrm{VE}} \hat{\epsilon}_\theta(x_k, t_k) + \sqrt{v_k}\,\epsilon_k, \qquad D_k^{\mathrm{VE}} := 2\big(\bar{\sigma}_{t_k} - \bar{\sigma}_{t_{k-1}}\big), \qquad v_k := \bar{\sigma}_{t_k}^2 - \bar{\sigma}_{t_{k-1}}^2, \tag{82}$$

Substituting the guided noise-prediction of Eq. (75) gives $c_k^{\mathrm{VE}} = D_k^{\mathrm{VE}} \bar{\sigma}_{t_k} = 2\bar{\sigma}_{t_k}(\bar{\sigma}_{t_k} - \bar{\sigma}_{t_{k-1}})$, and since $\bar{\sigma}_{t_{k-1}} = \bar{\sigma}_{t_k} e^{-\Delta \ell_k}$ when $\alpha_t \equiv 1$, both coefficients collapse onto the VP forms,

$$c_k^{\mathrm{VE}} = 2\,\bar{\sigma}_{t_{k-1}} \bar{\sigma}_{t_k} \big(e^{\Delta \ell_k} - 1\big), \qquad v_k = \bar{\sigma}_{t_{k-1}}^2 \big(e^{2\Delta \ell_k} - 1\big), \tag{83}$$

so that $\zeta_k^{\mathrm{VE},\,\hat{\epsilon}} = 2\,\bar{\sigma}_{t_k} \sqrt{\tanh(\Delta \ell_k / 2)}$, identical to ζ_k^{VP} of Eq. (78).

Euler–Maruyama Euler–Maruyama steps (Kloeden & Platen, 1992) discretise the reverse SDE directly, so the base and guided kernels share a common covariance, $v_k = \sigma_{t_k}^2 \Delta t_k$ and $u_t = -\sigma_t^2 \nabla_x r_t$. Hence, $c_k^{EM} = \sigma_{t_k}^2 \Delta t_k$, and,

$$\zeta_k^{EM} = \sigma_{t_k} \sqrt{\Delta t_k}. \tag{84}$$

Table 3 collects the three cases. Notably, as we increase the number of discretisations or $\Delta \ell_k \to 0$, all variants given in Table 3 recover the EM ζ_k^2.

Table 3: $\zeta_k^2 = c_k^2 / v_k$ for different integrators and noise schedules.

	c_k	v_k	$\zeta_k^2 = c_k^2/v_k$
Euler–Maruyama (VP/VE)	$\sigma_{t_k}^2 \Delta t_k$	$\sigma_{t_k}^2 \Delta t_k$	$\sigma_{t_k}^2 \Delta t_k$
exponential, VP ($\hat{\epsilon}$ frozen)	$2\bar{\sigma}_{t_{k-1}} \bar{\sigma}_{t_k} (e^{\Delta \ell_k} - 1)$	$\bar{\sigma}_{t_{k-1}}^2 (e^{2\Delta \ell_k} - 1)$	$4\,\bar{\sigma}_{t_k}^2 \tanh(\Delta \ell_k / 2)$
exponential, VE ($\hat{x}$ frozen)	$(1 - \kappa_k)\,\bar{\sigma}_{t_k}^2$	$\bar{\sigma}_{t_{k-1}}^2 (1 - \kappa_k)$	$\bar{\sigma}_{t_k}^2 (e^{2\Delta \ell_k} - 1)$
exponential, VE ($\hat{\epsilon}$ frozen)	$2\,\bar{\sigma}_{t_{k-1}} \bar{\sigma}_{t_k} \left(e^{\Delta \ell_k} - 1\right)$	$\bar{\sigma}_{t_{k-1}}^2 \left(e^{2\Delta \ell_k} - 1\right)$	$4\,\bar{\sigma}_{t_k}^2 \tanh(\Delta \ell_k / 2)$

E THE GUIDANCE GRADIENT VIA LOCAL LINEARISATION

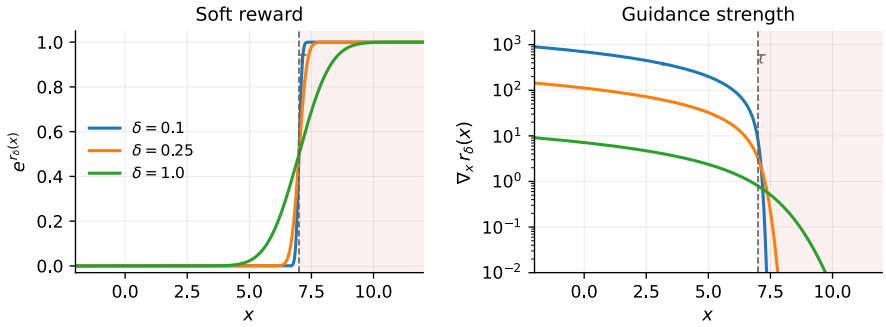


Figure 5: Soft reward and its guidance for a threshold event $E_7 = \{x : \phi(x) \geq 7\}$ shown for three smoothing levels δ. *Left.* The soft indicator relaxes the hard indicator. *Right.* The guidance $\nabla_x r_{0,\delta}(x)$ points toward the event set (shaded) and decays inside it, a smaller δ gives a stronger guidance.

The variance-aware guidance, illustrated in Fig. 5 used in Section 3 is,

$$\nabla_{x_t} \log \mathbb{E}_{p(x_0|x_t)}[e^{r_{0,\delta}(x_0)}]$$

This appendix derives the closed-form approximation used in our implementation. We consider a general vector-valued constraint $\phi : \mathbb{R}^d \to \mathbb{R}^m$ using a Gaussian approximation of the denoising posterior and a local linearisation of ϕ. The scalar affine guidance in Eq. (91), used for the one-sided threshold-exceedance experiments, is recovered as a special case $m = 1$ with a linear-ϕ. We then discuss how interval and compound events are represented by multiple one-sided constraints and the approximation induced when their guidance terms are treated independently.

For a multivariate constraint, the smooth reward is a multivariate Gaussian CDF $\Phi_m(\cdot \mid \Sigma_c)$ with constraint covariance Σ_c (the smoothing bandwidth; in the scalar case $\Sigma_c = \delta^2$). The integral is given by,

$$I(x_t) \;=\; \mathbb{E}_{p(x_0|x_t)}\left[\Phi_m\big(\phi(x_0) - \tau \,\big|\, \Sigma_c\big)\right]. \tag{85}$$

Gaussian posterior. We use the same Gaussian denoising-posterior approximation as in the main text, $p(x_0 \mid x_t) \approx \mathcal{N}(x_0 \mid \hat{x}_{0|t}, \Sigma_t)$ with $\hat{x}_{0|t}$ the Tweedie estimate and Σ_t the diagonal ΠGDM posterior-covariance proxy (Appendix B), hence,

$$I(x_t) \;\approx\; \int_{\mathbb{R}^d} \mathcal{N}(x_0 \mid \hat{x}_{0|t}, \Sigma_t)\, \Phi_m\big(\phi(x_0) - \tau \,\big|\, \Sigma_c\big)\, \mathrm{d}x_0. \tag{86}$$

Linearisation. Linearising the constraint about the Tweedie mean,

$$\phi(x_0) \;\approx\; \phi(\hat{x}_{0|t}) + J_\phi(\hat{x}_{0|t})\,(x_0 - \hat{x}_{0|t}), \qquad J_\phi(\hat{x}_{0|t}) = \nabla_{x_0}\phi\big|_{\hat{x}_{0|t}} \in \mathbb{R}^{m\times d}, \tag{87}$$

makes the integrand a Gaussian convolved with a Gaussian CDF, which integrates in closed form to a single CDF with the two covariances added (Rasmussen & Williams, 2006),

$$I(x_t) \;\approx\; \Phi_m\Big(\phi(\hat{x}_{0|t}) - \tau \,\Big|\, \Sigma_c + J_\phi(\hat{x}_{0|t})\,\Sigma_t\,J_\phi(\hat{x}_{0|t})^\top\Big)\,. \tag{88}$$

The posterior covariance Σ_t is thus pushed through the constraint Jacobian and inflates the effective observation variance.

Guidance gradient. Let $\pi_t \propto p_t\, e^{r_t}$ be the continuous-time version of the tilted target π_k of Eq. (5). The guided score is then the unconditional score plus the gradient of $\log I$,

$$\nabla_{x_t}\log\pi_t(x_t) \;=\; \underbrace{\nabla_{x_t}\log p_t(x_t)}_{\text{unconditional score}} \;+\; \underbrace{\nabla_{x_t}\log\Phi_m\Big(\phi(\hat{x}_{0|t}) - \tau \,\Big|\, \Sigma_c + J_\phi(\hat{x}_{0|t})\,\Sigma_t\,J_\phi(\hat{x}_{0|t})^\top\Big)}_{\text{variance-aware guidance gradient}}\,. \tag{89}$$

Linear constraint. For an affine read-out $\phi(x_0) = Ax_0 + b$ the Jacobian is constant, $J_\phi = A$, and the guidance simplifies to

$$\nabla_{x_t}\log\pi_t(x_t) \;=\; \nabla_{x_t}\log p_t(x_t) \;+\; \nabla_{x_t}\log\Phi_m\Big(A\hat{x}_{0|t} + b - \tau \,\Big|\, \Sigma_c + A\,\Sigma_t\,A^\top\Big)\,. \tag{90}$$

Specialising further to $m = 1$ with $\phi(x) = a^\top x$ and $\Sigma_c = \delta^2$ gives the scalar guidance used in the main text,

$$\begin{aligned}\nabla_{x_t}\log\mathbb{E}_{p(x_0|x_t)}[\exp(r_{0,\delta}(x_0))] &= \nabla_{x_t}\log\int p(x_0 \mid x_t)\,\Phi\left(\frac{\phi(x_0)-\tau}{\delta}\right)dx_0 \\ &\approx\; \nabla_{x_t}\log\Phi\left(\frac{\phi(\hat{x}_{0|t})-\tau}{\sqrt{\delta^2 + a^\top\Sigma_t a}}\right),\end{aligned} \tag{91}$$

with the inflated width $\sqrt{\delta^2 + a^\top\Sigma_t a}$.

E.1 Factorised guidance for multiple constraints

Consider an event defined by the intersection of m one-sided constraints,

$$E = \bigcap_{j=1}^{m}\{\phi_j(x) \geq \tau_j\}\,, \tag{92}$$

such as the boxed intervals of App. G.4. Section E gives the corresponding look-ahead likelihood as a multivariate Gaussian CDF $\Phi_m(\,\cdot \mid \Sigma_{\text{eff}})$ with

$$\Sigma_{\text{eff}} = \Sigma_c + J_\phi(\hat{x}_{0|t})\,\Sigma_t\,J_\phi(\hat{x}_{0|t})^\top, \tag{93}$$

where Σ_c is the user-specified smoothing covariance and the second term is the denoising-posterior uncertainty projected through the read-out (Eq. (88)).

To obtain the guidance gradient, we require $\nabla_{x_t}\log\Phi_m$, which in turn requires evaluating Φ_m itself. For $m = 1$, this is available in closed form and for $m = 2, 3$ it reduces to a low-dimensional quadrature (Drezner, 1994; Genz, 2004; Owen, 1956). For $m \geq 4$ with an unstructured or full covariance matrix Σ_{eff} no such reduction exists and the function has to be approximated by (randomised) quasi-Monte Carlo techniques (Genz, 1992). The cost of that approximation is incurred once per particle per diffusion step, and unless the evaluation points are held fixed across particles and steps, the reward and its gradient become stochastic rather than deterministic functions of x_t. We therefore also provide the factorised surrogate below.

Optionally, Σ_{eff} may be approximated by its diagonal, under which Φ_m factorises into a product of univariate Gaussian CDFs,

$$I_{\text{fact}}(x_t) = \prod_{j=1}^{m} \Phi\left(\frac{\phi_j(\hat{x}_{0|t}) - \tau_j}{\delta_{\text{eff},j}(x_t)}\right), \qquad \delta^2_{\text{eff},j}(x_t) = \delta_j^2 + a_j^\top \Sigma_t \, a_j, \quad a_j = \nabla_{x_0}\phi_j\big|_{\hat{x}_{0|t}}, \tag{94}$$

with a factorised reward $r_t^{\text{fact}} := \log I_{\text{fact}}$ and guidance,

$$\nabla_{x_t} r_t^{\text{fact}} = \sum_{j=1}^{m} \nabla_{x_t} \log \Phi\left(\frac{\phi_j(\hat{x}_{0|t}) - \tau_j}{\delta_{\text{eff},j}(x_t)}\right). \tag{95}$$

This is simply m independent copies of the scalar guidance and while the factorisation retains the marginal posterior uncertainty of each constraint, it discards the off-diagonal entries of Σ_{eff}. This alters the guidance drift, and hence the implemented proposal $\mathbb{Q}$ together with the sequence of marginals π_t. Our rejuvenation kernels are made invariant with respect to the same marginals, using the factorised surrogate I_{fact}. However, the target itself remains unchanged $p_0 e^{r_{0,\delta}(x)}$ as it depends on the data-reward at $t = 0$ alone and remains unaffected by π_t or $\nabla_x r_t$. Hence, this approximation only contributes to the variance of our estimator.

Note finally that the factorisation describes the tilted marginals and not the sampler. The guided drift is,

$$\nabla_{x_t} \log p_t(x_t) + \nabla_{x_t} r_t^{\text{fact}}(x_t),$$

and the unconditional score retains the emulator's full joint dependence structure. Additionally, each term is also mapped through the dense $\partial \hat{x}_{0|t}/\partial x_t$, so a constraint supported on a single region and channel still induces a global drift across all variables. The diagonality of Σ_{eff} suppresses cross-constraint terms in the look-ahead *variance*; it does not decouple the resulting drift.

Interval Guidance A useful special case is an event defined by the interval, $E = \{\tau_a \le \phi(x) \le \tau_b\}$, which is the intersection (92) with $m = 2$, taking $\phi_1 = \phi$, $\tau_1 = \tau_a$ and $\phi_2 = -\phi$, $\tau_2 = -\tau_b$. We write the reward as a difference of two individual rewards given by,

$$e^{r_{0,\delta}(x)} = \Phi\left(\frac{\phi(x) - \tau_a}{\delta}\right) - \Phi\left(\frac{\phi(x) - \tau_b}{\delta}\right), \tag{96}$$

with the look-ahead and guidance following as before, replacing δ by $\delta^2_{\text{eff}} = \delta^2 + a^\top \Sigma_t a$ which is shared across both boundaries.

F THE CORRELATED 2D GMM BENCHMARK

This appendix specifies the synthetic benchmark used in our 2D experiments and collects the closed-form quantities it admits: the base density, the normalisation constant $Z_{0,\delta}$ of the soft tilt, the exact exceedance probability $p_0[E]$, and the exact Doob guidance $\nabla_x r_t^\star = \nabla_x \log G_t$ (Dai Pra, 1991; Denker et al., 2024). All of these are available in closed form precisely because the GMM prior is conjugate to both the Gaussian forward kernel and the Gaussian-CDF reward, which lets us isolate the behaviour of the estimator from score- and reward-approximation error (Rasmussen & Williams, 2006).

Base distribution. The data distribution is a $M = 7$-component Gaussian mixture in $\mathbb{R}^2$,

$$p_0(x) \;=\; \sum_{i=1}^{M} w_i \, \mathcal{N}(x;\, \mu_i, \Sigma_i), \qquad \sum_i w_i = 1, \tag{97}$$

where w_i $(i = 1, \dots, M)$ denote the mixture weights. The parameters are given in Table 4. The four "bulk" modes carry most of the mass near the origin; the three right-hand modes are pushed out towards large x_1 and given strong off-diagonal covariance ($\rho \approx \pm 0.85$) of opposite sign. This is by design: it makes the tail conditional $p_0(x_2 \mid x_1 > \tau)$ *bi-modal* and markedly non-Gaussian, so that hitting the rare set is not equivalent to matching a single Gaussian tail and the proposal must reproduce a genuine multi-modal conditional.

The mixture weights w_i are given in Table 4 and sum to one. The observable is the linear read-out $\phi(x) = a^\top x$ with $a = (1, 0)^\top$, i.e. the event of interest is the threshold exceedance $\{x : x_1 \ge \tau\}$.

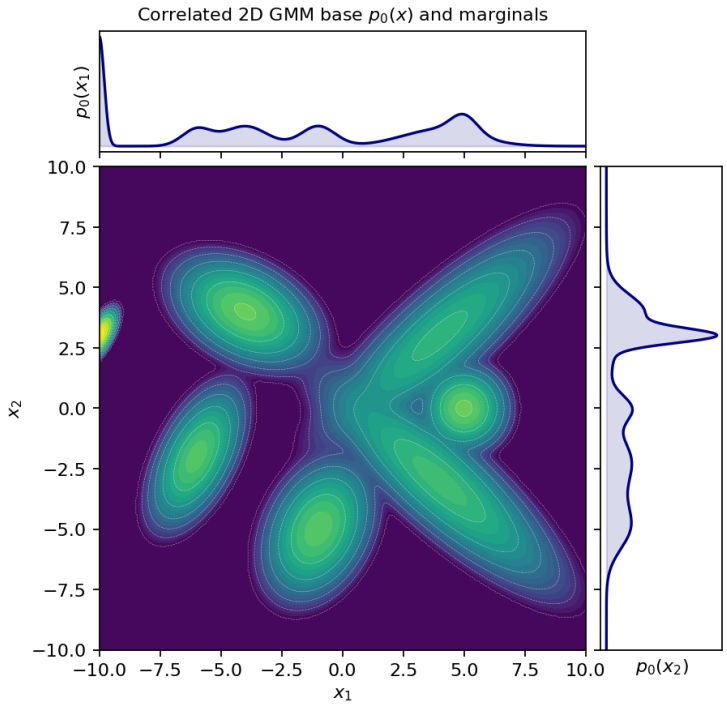


Figure 6: Base distribution $p_0(x)$ of the correlated 2D GMM (log-density, so all modes are visible) with the analytic x_1 and x_2 marginals. The two right-hand modes ($i = 5, 6$) form the tilted "X" of opposite-sign correlation that gives the rare set a bi-modal conditional.

Table 4: Parameters of the correlated 2D GMM. The right-hand modes ($i = 5, 6, 7$) carry the rare-event tail; modes 5 and 6 have correlation $\rho \approx +0.85$ and -0.85 respectively.

i	w_i	μ_i	Σ_i
1	0.10	$(-6, -2)$	$\left[\begin{smallmatrix} 0.36 & 0.30 \\ 0.30 & 0.81 \end{smallmatrix}\right]$
2	0.18	$(-4,\ 4)$	$\left[\begin{smallmatrix} 0.81 & -0.25 \\ -0.25 & 0.49 \end{smallmatrix}\right]$
3	0.14	$(-1, -5)$	$\left[\begin{smallmatrix} 0.49 & 0.20 \\ 0.20 & 0.81 \end{smallmatrix}\right]$
4	0.22	$(-10,\ 3)$	$\left[\begin{smallmatrix} 0.04 & 0.04 \\ 0.04 & 0.09 \end{smallmatrix}\right]$
5	0.10	$(\ 4,\ 3)$	$\left[\begin{smallmatrix} 2.40 & 1.85 \\ 1.85 & 2.00 \end{smallmatrix}\right]$
6	0.16	$(\ 4, -3)$	$\left[\begin{smallmatrix} 2.40 & -1.85 \\ -1.85 & 2.00 \end{smallmatrix}\right]$
7	0.10	$(\ 5,\ 0)$	$\left[\begin{smallmatrix} 0.30 & 0.00 \\ 0.00 & 0.30 \end{smallmatrix}\right]$

Normalisation constant $Z_{0,\delta}$. The soft tilt $\pi_{0,\delta}(x) \propto p_0(x)\, e^{r_{0,\delta}(x)}$ with the Gaussian-CDF reward $e^{r_{0,\delta}(x)} = \Phi\big((a^\top x - \tau)/\delta\big)$ has normaliser $Z_{0,\delta} = \mathbb{E}_{p_0}[e^{r_{0,\delta}(X)}]$. Under each Gaussian component the read-out $y = a^\top x \sim \mathcal{N}(m_i,\, v_i)$ with $m_i = a^\top \mu_i$ and $v_i = a^\top \Sigma_i a$, and a standard Gaussian–Gaussian-CDF convolution (see Section 3.9 of (Rasmussen & Williams, 2006)) gives the closed form,

$$Z_{0,\delta} \;=\; \sum_{i=1}^{M} w_i\, \mathbb{E}_{\mathcal{N}(m_i, v_i)}\Big[\Phi\big(\tfrac{y-\tau}{\delta}\big)\Big] \;=\; \sum_{i=1}^{M} w_i\, \Phi\left(\frac{m_i - \tau}{\sqrt{\delta^2 + v_i}}\right). \tag{98}$$

As $\delta \downarrow 0$ the reward sharpens to the hard indicator and $Z_{0,\delta} \to p_0[E]$.

Exact exceedance probability. Because the read-out $a^\top X$ is itself a 1D GMM with component means m_i and variances v_i, the exceedance probability is available in closed form,

$$p_0[E] \;=\; p_0[a^\top X \geq \tau] \;=\; \sum_{i=1}^{M} w_i \Big[1 - \Phi\big(\tfrac{\tau - m_i}{\sqrt{v_i}}\big)\Big] \;=\; \sum_{i=1}^{M} w_i \, \Phi\left(\frac{m_i - \tau}{\sqrt{v_i}}\right), \tag{99}$$

which we use as ground truth.

Tilted read-out marginal. While the full tilted density $\pi_{0,\delta}(x) \propto p_0(x)\,\Phi\big((a^\top x - \tau)/\delta\big)$ is not itself a Gaussian mixture, the marginal of the read-out $y = a^\top x$ is available in closed form,

$$\pi^{\mathrm{read}}_{0,\delta}(y) \;=\; \frac{1}{Z_{0,\delta}} \sum_{i=1}^{M} w_i \, \mathcal{N}(y; m_i, v_i)\, \Phi\Big(\frac{y - \tau}{\delta}\Big), \qquad y = a^\top x, \tag{100}$$

with $Z_{0,\delta}$ from Eq. (98) and (m_i, v_i) as above.

Forward process. We use the VP-SDE schedule of Appendix B with $\beta_{\min} = 0.1$, $\beta_{\max} = 20$, so that $B(t) = \int_0^t \beta(s)\, ds = t\beta_{\min} + \frac{1}{2}t^2(\beta_{\max} - \beta_{\min})$, $\alpha_t = e^{-B(t)/2}$ and $\bar{\sigma}_t^2 = 1 - e^{-B(t)}$. Because the forward kernel is Gaussian, the noised marginal p_t is again a GMM whose i-th component has mean $\alpha_t \mu_i$ and covariance (Rasmussen & Williams, 2006),

$$S_{t,i} \;=\; \alpha_t^2\, \Sigma_i + \bar{\sigma}_t^2\, I. \tag{101}$$

Exact score. The exact score of the noised marginal is the weighted sum of the per-component scores. For any mixture,

$$\nabla_x \log p_t(x) \;=\; \frac{\sum w_i \nabla_x \mathcal{N}_i}{\sum w_j \mathcal{N}_j} = \frac{\sum w_i \mathcal{N}_i\big[-S_{t,i}^{-1}(x - \alpha_t \mu_i)\big]}{\sum w_j \mathcal{N}_j}. \tag{102}$$

We use this exact score wherever analytical score is reported.

Exact look-ahead guidance. The exact guidance is the gradient of the Doob reward of Eq. (14) (Dai Pra, 1991; Denker et al., 2024), which for the soft reward reads $r_t^\star(x) = \log G_t(x)$ with $G_t(x) = \mathbb{E}_{p(x_0|x_t)}[\exp(r_{0,\delta}(x_0))]$, i.e. $\nabla_x r_t^\star = \nabla_x \log G_t(x)$. Conjugacy makes this closed-form. We can use the fact that $p(x_0|x_t, i) \propto p(x_t|x_0)p(x_0|i)$, hence the posterior under each component i is a product of two Gaussians - another Gaussian, $\mathcal{N}(m_{t,i}, C_{t,i})$ with,

$$C_{t,i} = \Big(\Sigma_i^{-1} + \tfrac{\alpha_t^2}{\bar{\sigma}_t^2} I\Big)^{-1}, \qquad m_{t,i} = C_{t,i}\Big(\Sigma_i^{-1}\mu_i + \tfrac{\alpha_t}{\bar{\sigma}_t^2} x\Big). \tag{103}$$

To evaluate the expected reward, we consider a linear observable $\phi(x_0) = a^\top x_0$. Under the i-th component posterior, this projection is a 1D Gaussian, $a^\top X_0 \sim \mathcal{N}(a^\top m_{t,i},\, s_{t,i}^2)$ with $s_{t,i}^2 = a^\top C_{t,i}\, a$, so the per-mode look-ahead reward integrates to another Gaussian CDF with,

$$\mathbb{E}_{p(x_0|x_t)}[\Phi(.)] = \Phi(\eta_{t,i}), \qquad \eta_{t,i} = \frac{a^\top m_{t,i} - \tau}{\sqrt{\delta^2 + s_{t,i}^2}}. \tag{104}$$

We can then write the global look-ahead potential $G_t(x)$ as the expected reward marginalised over all M. Using Eq.(104), we can write $G_t(x) = \sum_{i=1}^M w_{t,i}\Phi(\eta_{t,i})$, giving the exact guidance as follows.

$$\begin{aligned}
\nabla_x r_t^\star(x) &= \nabla_x \log G_t(x) \\
&= \frac{1}{G_t(x)} \nabla_x \left(\sum_{i=1}^{M} w_{t,i}(x) \Phi(\eta_{t,i}) \right) \\
&= \frac{1}{G_t(x)} \sum_{i=1}^{M} \left[\Phi(\eta_{t,i}) \nabla_x w_{t,i}(x) + w_{t,i}(x) \nabla_x \Phi(\eta_{t,i}) \right]
\end{aligned} \tag{105}$$

We can rewrite the above equation as,

$$\nabla_x r_t^\star(x) = \sum_{i=1}^{M} \widetilde{w}_i(x) \left[\frac{\nabla_x \Phi(\eta_{t,i})}{\Phi(\eta_{t,i})} + \frac{\nabla_x w_{t,i}(x)}{w_{t,i}(x)} \right] \qquad \widetilde{w}_i(x) = \frac{w_{t,i}(x)\Phi(\eta_{t,i})}{\sum_{j=1}^{M} w_{t,j}(x)\Phi(\eta_{t,j})} \tag{106}$$

Here $\widetilde{w}_i$ represents the reward weighted (posterior) responsibility of the GMM - i.e. it is the responsibility re-weighted by each mode's chance of reaching the rare event set (Bishop & Nasrabadi, 2006). A mode i is up-weighted when x_t is consistent with it (i.e. $w_{t,i}$ large) and when it is well placed to cross the threshold ($\Phi(\eta_{t,i})$ large). The first term inside the bracket is the per-mode push towards the rare set; second is the routing correction that re-weights mass between modes as the reward favours different components. Evaluating the two terms in $\nabla_x r_t^\star$, the first term is the reward push,

$$\frac{\nabla_x \Phi(\eta_{t,i})}{\Phi(\eta_{t,i})} = \frac{\varphi(\eta_{t,i})}{\Phi(\eta_{t,i})} g_{t,i} = \psi(\eta_{t,i}) g_{t,i}, \tag{107}$$

where $g_{t,i} = \nabla_x \eta_{t,i} = \frac{\alpha_t}{\bar{\sigma}_t^2} \frac{C_{t,i}a}{\sqrt{\delta^2 + s_{t,i}^2}}$ and $\psi(\cdot) = \varphi(\cdot)/\Phi(\cdot)$ is the Inverse-Mills ratio.

The second term of Eq. (106) represents the standard GMM responsibility at a noised time t, given by,

$$w_{t,i}(x) = \frac{w_i \mathcal{N}(x; \alpha_t \mu_i, S_{t,i})}{p_t(x)}.$$

Evaluating this yields,

$$\nabla_x \log w_{t,i}(x) = \nabla_x \log \mathcal{N}(x; \alpha_t \mu_i, S_{t,i}) - \nabla_x \log p_t(x). \tag{108}$$

Putting this together, we get the analytical guidance for the GMM,

$$\nabla_x r_t^\star(x) = \sum_{i=1}^{M} \widetilde{w}_i(x) \Big[\psi(\eta_{t,i}) g_{t,i} + \nabla_x \log \mathcal{N}(x; \alpha_t \mu_i, S_{t,i}) - \nabla_x \log p_t(x) \Big]. \tag{109}$$

We refer readers to (Wu et al., 2024) for a rigorous exposition of the effect of guidance on GMMs.

G 2D GMM EXPERIMENTS

G.1 EXPERIMENTAL DETAILS

The learned score. The learned score for the correlated 7-component GMM of Table 4, is a velocity prediction MLP trained under the linear VP schedule (Song et al., 2021b). The network takes $(x, \mathrm{emb}(t)) \in \mathbb{R}^{2+128}$, where t is embedded using a sinusoidal time embedding through 6 layers (hidden width 512, SiLU (Elfwing et al., 2018); 1.1M parameters) to a two-dimensional output. Targets are $v = \alpha_t \varepsilon - \bar{\sigma}_t x_0$ with $x_t = \alpha_t x_0 + \bar{\sigma}_t \varepsilon$; the score used by the sampler is recovered as $s_\theta(x,t) = -\hat{\varepsilon}/\bar{\sigma}_t$ with $\hat{\varepsilon} = \bar{\sigma}_t x + \alpha_t v_\theta(x,t)$. The loss is the per-sample v-MSE under min-SNR-γ weighting with $\gamma = 5$ and the weights given by $w(t) = \min(\mathrm{SNR}(t), \gamma)/(\mathrm{SNR}(t)+1)$ of (Hang et al., 2023), with $t \sim \mathcal{U}[0.01, 0.99]$. Training draws fresh exact GMM samples every step: 40,000 steps at batch 4096 with AdamW (Kingma & Ba, 2014; Loshchilov & Hutter, 2019) (learning rate 2×10^{-4}, weight decay 10^{-4}), 1,000-step linear warm-up followed by cosine decay, gradient clipping at 1.0, and an exponential moving average of the weights (decay 0.999); the EMA weights are used for evaluation.

The tempering ladder. Our guidance ladder used through the experiment is $\lambda(t) = 1 - (t/t_{\mathrm{on}})^2$ for $t < t_{\mathrm{on}} = 0.9$ in the case of rejuvenation and $\lambda_t \equiv 1$ without it, see App. G.5. We chose this simple form of tempering, inspired by polynomial tempered likelihood (Friel & Pettitt, 2008).

The benchmark. The mixture parameters and every closed form we compare against: the base density, the exact score, the soft-tilt normaliser $Z_{0,\delta}$ of Eq. (98), the exceedance probability $p_0[E]$ of Eq. (99), the tilted target $\pi_{0,\delta}$ of Eq. (100) and the exact Doob guidance $\nabla_x r_t^\star$ of Eq. (106) are defined in Appendix F. The right-hand modes carry strong off-diagonal covariance of opposite sign, so the tail conditional $p_0(x_2 \mid x_1 > \tau)$ is genuinely bi-modal: hitting the rare set requires reproducing a multi-modal conditional. The learned-score arm uses the v-prediction MLP described above.

Sampling parameters Unless stated otherwise every run below uses the deployed configuration of Section 4.1: $N = 1000$ particles, $K = 1000$ SEEDS-1 steps on a uniform half-log SNR noise schedule with $\delta = 0.01$, the quadratic tempering ladder $\lambda(t)$ above for the rejuvenation arm and $\lambda \equiv 1$ otherwise, adaptive resampling at $\mathrm{ESS}/N < 0.5$ with three MADM rejuvenation sweeps per resampling (Simpson's (1/3) rule with Eq. (119)), and 24 seeds. The MH step is $\rho = c(t)\,\bar{\sigma}_t^2$ with $c(t)$ fixed once from a pilot sweep targeting an acceptance ratio of ≈ 0.72 at each noise level, as prescribed in (Lam et al., 2026).

All experiments were done on a single Nvidia A100 GPU with 40 GB of VRAM.

G.2 Sources of Bias

We begin by discussing the two main contributing factors that bias our estimate against the true $p_0[E]$, followed by additional results. Neither bias is an error of the estimator, and the path-weighted estimator remains unbiased exactly for the law we sample from. The two primary sources are the learning process where the score is approximated poorly (Appendix G.2.1) and when the discretised law we sample from differs from the continuous law (Appendix G.2.2). We separate and isolate each to study their contributions to the final estimate.

G.2.1 Tail bias of the learned score

We estimate the tail bias of the learned score by comparing the model's output at a specific time and location (x_t, t) against the analytical score. Write $\Delta s(x,t) = s_\theta(x,t) - s^\star(x,t)$ for the pointwise gap between the learned and exact scores. The induced log-density bias $b(x_0) := \log p_{\mathrm{model}}(x_0) - \log p^\star(x_0)$ accumulates along the probability-flow ODE trajectory $\{x_t\}_{t\in[0,1]}$ that carries x_0 back to the prior under the *learned* velocity field, as

$$b(x_0) \;=\; -\int_0^1 \tfrac{1}{2}\beta(t)\Big[\nabla\cdot\Delta s \;+\; s^\star\cdot\Delta s\Big](x_t)\,\mathrm{d}t, \tag{110}$$

from which the tail ratio follows by reweighting the exact conditional,

$$\frac{P_{\mathrm{model}}(\phi(X)\geq\tau)}{P^\star(\phi(X)\geq\tau)} \;=\; \mathbb{E}_{p^\star(\,\cdot\,|\,\phi(X)\geq\tau)}\Big[e^{\,b(x_0)}\Big]. \tag{111}$$

Table 5: Tail bias of the learned 2D-GMM score. p_{model} (MC) is obtained from a large MC reference pool using unguided draws. Path integral measures the ratio obtained by integrating the pointwise score error along the probability-flow ODE (Eq. (110)) and reweighing the exact conditional. The two induce different laws but agree to within 7% across three decades of rarity. $p_{\mathrm{data}}[E]$ is the analytical probability.

τ	$p_{\mathrm{data}}[E]$	p_{model} (MC)	$p_{\mathrm{model}}/p_{\mathrm{data}}[E]$	rel. s.e.	path integral	ratio
7	6.88×10^{-3}	7.13×10^{-3}	1.037	0.08%	1.075	1.037
8	1.28×10^{-3}	1.38×10^{-3}	1.081	0.18%	1.156	1.069
9	1.62×10^{-4}	1.87×10^{-4}	1.153	0.50%	1.236	1.072
10	1.40×10^{-5}	1.72×10^{-5}	1.229	1.64%	1.267	1.031

In practice, we draw 2.16×10^8 samples from the model unconditionally and use it as our reference pool. We find that with both methods, the tail bias widens with rarity, and the score-estimate degrades monotonically, see Table 5. All quoted comparisons for the trained model are against the model's own rare event probability $p_0[E]$, obtained via Monte Carlo estimate.

G.2.2 The discretised law: grid and integrator

Alongside the bias due to the score estimate, which in the general case cannot be estimated, our discretisation itself induces a distinct, reducible form of bias. Let $\mathbb{P}$ be the continuous path measure defined by the diffusion process, and $\mathbb{P}_k$ bethe discretised law that we simulate. Our estimator is unbiased for the discrete law that we compute weights for under any guidance (Del Moral et al., 2006). Hence any gap that remains is a property of the base chain and the integrator alone. This section first measures this gap for two first order integrators and whether the choice of integrator reduces it at a fixed K.

Setup. Exact Doob guidance (i.e. $\nabla_x r_t^*(x)$), with resampling turned off, so neither guidance error nor adaptive resampling contributes and the residual is resolved above seed noise. We compare the Euler-Maruyama (EM) integrator against the first order exponential SDE solver of (Gonzalez et al., 2023) which we denote as SEEDS-1. Both are explicit one-step schemes of first order. The only difference is that SEEDS-1 treats the integration of the reverse SDE as a semi-linear integral, separating the linear and non-linear parts of the score, and integrating the linear and the noise term exactly, whereas EM approximates it to first order, see (Gonzalez et al., 2023; Milstein & Tretyakov, 2004; Pfarr et al., 2026) for more details. Alongside the integrator, the placement of the grid itself is a design choice: we run the sweep on a grid uniform in t and on a grid uniform in the half-log Signal-to-Noise ratio $\ell_t = \log(\alpha_t/\bar{\sigma}_t)$ of App. D.1, the integrator's own time variable. Table 6 sweeps the step count K under both placements and compares the signed bias in the probability estimate with the ground truth in $\tau = 10$.

Table 6: Signed discretisation bias versus step count K for the two integrators, under exact Doob guidance, at different grids, uniform in t (top block) and uniform in the half-log SNR $\ell_t = \log(\alpha_t/\bar{\sigma}_t)$ (bottom block). Positive values are overestimates. Paired columns are EM/SEEDS-1 throughout. $\hat{k}$ is the Pareto Smoothed Importance Sampling shape estimate (Vehtari et al., 2024) on the weights, and ESS_E is the ESS of the final particles, measured after thresholding to $\tau \geq 10$.

K	EM bias	SEEDS-1 bias	|EM|/|SEEDS|	CV ↓	$\hat{k}$ ↓	ESS_E ↑
Uniform in t						
10	+64.02% ± 11.31%	-15.98% ± 1.88%	4.01	0.338/0.109	0.85/0.46	0.032/0.102
15	+48.61% ± 8.39%	-16.23% ± 1.28%	3.00	0.277/0.075	0.70/0.45	0.063/0.189
20	+54.90% ± 16.50%	-15.13% ± 0.97%	3.63	0.522/0.056	0.68/0.43	0.099/0.248
30	+23.41% ± 1.65%	-13.62% ± 0.75%	1.72	0.065/0.043	0.56/0.40	0.176/0.375
50	+16.75% ± 1.38%	-8.36% ± 0.63%	2.00	0.058/0.034	0.55/0.45	0.315/0.465
100	+6.82% ± 0.69%	-4.99% ± 0.57%	1.37	0.032/0.029	0.40/0.45	0.600/0.634
200	+3.72% ± 0.30%	-2.71% ± 0.26%	1.38	0.014/0.013	0.40/0.40	0.771/0.802
500	+1.30% ± 0.19%	-1.27% ± 0.16%	1.02	0.009/0.008	0.39/0.40	0.901/0.906
1000	+0.75% ± 0.19%	-0.51% ± 0.17%	1.46	0.009/0.009	0.36/0.39	0.943/0.945
Uniform in $\ell_t = \log(\alpha_t/\bar{\sigma}_t)$						
10	-34.55% ± 2.77%	+256.82% ± 39.09%	0.13	0.207/0.537	0.75/0.80	0.031/0.021
15	-24.93% ± 3.12%	+28.68% ± 2.42%	0.87	0.203/0.092	0.58/0.48	0.083/0.115
20	-20.68% ± 1.76%	+12.66% ± 1.63%	1.63	0.109/0.071	0.46/0.40	0.169/0.201
30	-13.78% ± 0.88%	+2.13% ± 1.01%	6.46	0.050/0.048	0.38/0.37	0.283/0.313
50	-8.98% ± 0.59%	-2.19% ± 0.70%	4.09	0.032/0.035	0.34/0.34	0.474/0.506
100	-4.63% ± 0.42%	-2.59% ± 0.42%	1.79	0.022/0.021	0.24/0.26	0.713/0.724
200	-2.54% ± 0.36%	-1.89% ± 0.34%	1.35	0.018/0.017	0.17/0.18	0.841/0.849
500	-1.29% ± 0.17%	-1.02% ± 0.17%	1.27	0.008/0.009	0.19/0.19	0.928/0.930
1000	-0.27% ± 0.13%	-0.16% ± 0.13%	1.66	0.007/0.006	0.22/0.23	0.956/0.957

The $1/K$ law. Both integrators are one-step schemes of weak order one (Milstein & Tretyakov, 2004), for which the weak-error admits the expansion $c_1/K + O(1/K^2)$ with c_1 independent of the step size (Milstein & Tretyakov, 2004; Talay & Tubaro, 1990). While this holds for smooth test functions, it does not automatically hold for the indicator $\mathbf{1}_E$. We therefore fit the form using ordinary least-squares instead of invoking it directly as a theorem. Fitting the coefficient c_1, weighted by the per-K standard errors s_k,

$$\hat{c}_1 = \frac{\sum_k b_k K_k^{-1} s_k^{-2}}{\sum_k K_k^{-2} s_k^{-2}}, \tag{112}$$

restricted to $K \geq 30$, since $K = 10$-20 sits outside the asymptotic regime. On the uniform-t grid the fits are $\hat{c}_1 = +7.27$ for Euler-Maruyama against -4.41 for SEEDS-1, with largest residuals of 1.6 and 2.4 standard errors respectively, so the $1/K$ law holds to within roughly two standard errors over the fitted range. The signs record that the two integrators approach the continuous limit from opposite sides on this grid: EM overestimates and SEEDS-1 underestimates at every K, and the $|\mathrm{EM}|/|\mathrm{SEEDS}|$ column of Table 6 is the per-K magnitude comparison. The fitted coefficients track the measurements closely: $\hat{c}_1 = 7.27$ predicts $+3.6\%$ at $K = 200$ against a measured $+3.72\%$, and $+14.6\%$ at $K = 50$ against $+16.8\%$. On the uniform-ℓ grid the EM arm fits similarly ($\hat{c}_1 = -4.45$,

largest residual 2.4 standard errors), but the SEEDS-1 arm *changes sign* between $K = 30$ and $K = 50$: its leading coefficient is small enough that higher-order terms are not negligible over the measured range.

Grid placement. At coarse grids the two placements order the integrators differently. In uniform in t, SEEDS-1 dominates EM in every K: the magnitude ratio drops from 4.0 in $K = 10$ towards parity by $K \geq 500$. For the uniform in the ℓ grid, the order inverts in the coarsest grids $K = 10$ and $K = 15$, where we reproduce the result of (Gonzalez et al., 2023) where the performance of the integrator is poor at very small discretisation. We hypothesise that this is because, a uniform $\Delta\ell$ starves the "middle" of the denoising process, which principled methods for noise-scheduling such as (Williams et al., 2024) place a larger number of discretisation on, see Fig. 4 of (Williams et al., 2024), which adversely affects the discretised law that we sample. Given that the schedule intimately interplays with Gaussian transition kernel approximation, we leave studying the effects it has on the weight computation for future work.

From $K \geq 20$ SEEDS-1 yields a smaller discretisation bias, and the two integrators converge to each other by $K = 200$. We find that in our setting, SEEDS-1 reduces the discretisation bias at lower K on either grid placement away from the coarsest ℓ-grids, and both integrators are comparable at higher K's. We deploy SEEDS-1 at $K = 1000$ on the uniform-ℓ grid, where both integrators are within 0.2% of the continuous limit (Table 6, last row). The exponential integrator is also used in the larger climate experiments.

Both channels improve together. In this experiment, the integrator does not only control the bias but also affects the variance. Across Table 6 the variance diagnostics move with it: on the uniform-t EM arm the CV falls from 0.338 to 0.009, ESS_E climbs from 0.032 to 0.943, with the PSIS $\hat{k}$ dropping below 0.5 by $K = 100$. The grid placement moves the same diagnostics: at matched K the ℓ-grid improves ESS_E and $\hat{k}$ for both integrators (e.g. at $K = 100$, EM's ESS_E is 0.713 uniform in ℓ against 0.600 uniform in t, with $\hat{k}$ of 0.24 against 0.40), the coarsest ℓ-grids under SEEDS-1 excepted. These are not two separate effects but one mechanism with two consequences. A finer grid puts the discrete chain closer to the continuous dynamics, which simultaneously shrinks the gap $\mathbb{P}_K - \mathbb{P}$ that the bias measures. As we simulate using the exact Doob's guidance, this moves the discretised proposal $\mathbb{Q}^K$ close to the variance minimising Doob's proposal which helps all other metrics. In a general setting with approximate guidance the dominant source of error is not the discretisation bias but instead the approximation error of Doob's h-transform, e.g. using DPS.

G.3 The threshold-exceedance experiment

Everything in this subsection belongs to the deployed threshold-exceedance experiment of Section 4.1: the conditional clouds behind the sample-quality columns, the no-resampling arm, the whole-path ESS diagnostics, and how the estimator scales with the particle budget N. We summarise our results in Table. 7, where we find that our method recovers the correct probability estimates as well as the correct shape statistics. We analyse all individual aspects of these results in the section below. All experiments were done with the same guidance with $K = 1000$ and the SEEDS-1 integrator on the uniform-ℓ grid unless stated otherwise.

G.3.1 Conditional-sample coverage: clouds for learned and analytical

We plot the sample clouds ($N = 2048$) using the deployed configuration of Section 4.1 for both Analytical and Learned Guidance with the same guidance/resampling/rejuvenation parameters. The red samples are drawn from the exact conditional $p(x \mid x_1 \geq \tau)$, drawn larger at (2×10^4) and the blue particles represent the guided samples. Together with the KS/EMD ratios of Table 7 the plots qualitatively highlight that samples are drawn from the conditional distribution itself. The two clouds remain visually indistinguishable as can be seen in Figure 10.

G.3.2 ESS along the trajectory

Without resampling the accumulated weight is never reset and the terminal ESS asymptotically converges to $1/(1 + \chi^2(\mathbb{P}^\star_{0:K} \| \mathbb{Q}_{0:K}))$ (Agapiou et al., 2017). Hence for the sampled path, the ESS measures the gap between the implemented guidance and the exact Doob's guidance. Figure 11 tracks

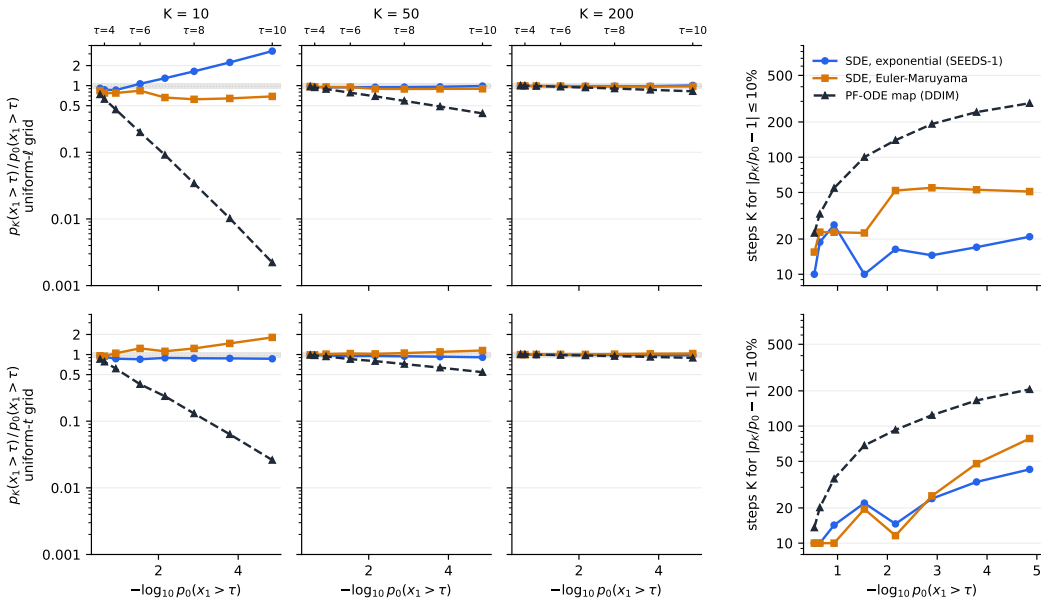


Figure 7: Discretisation bias for the three integrators, Euler-Maruyama, SEEDS-1 and DDIM. We unconditionally draw 1e8 samples from the base model at different discretisation K and evaluate the bias at different rarities. The figure on the right shows the number of discretisation required to get a $< 10\%$ bias for different rarities, showing that the SDE requires less steps than the ODE integrator. Top row uses the uniform log-SNR $\Delta\ell$ schedule and the bottom is uniform in time Δt.

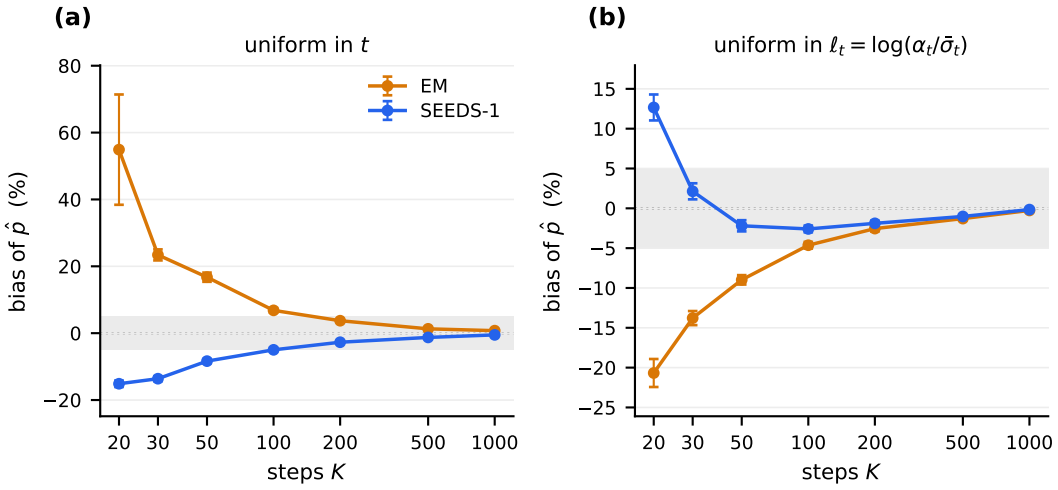


Figure 8: Measured bias versus K with 1000 particles, analytical scores and Doob's transition kernels. See App G.2.2. We show the bias of our estimate falls with a $O(1/K)$ leading order term. We compare both the EM integrator and the SEEDs-1 integrator in the noise prediction form. We compare two different noise schedules: uniform in time and uniform in log-SNR ℓ. Error-bars represent standard error of the mean. Experiment is conducted the $\tau = 10$ threshold exceed problem. See Table 6.

it along the trajectory for both scores and five thresholds. Two features that are worth noting. The trace is flat at $\mathrm{ESS}/N = 1$ until $t \approx 0.9$, which is the untilted stretch of the denoising time, where the guidance is turned off. After guidance is turned on, the ESS decays monotonically in both learned

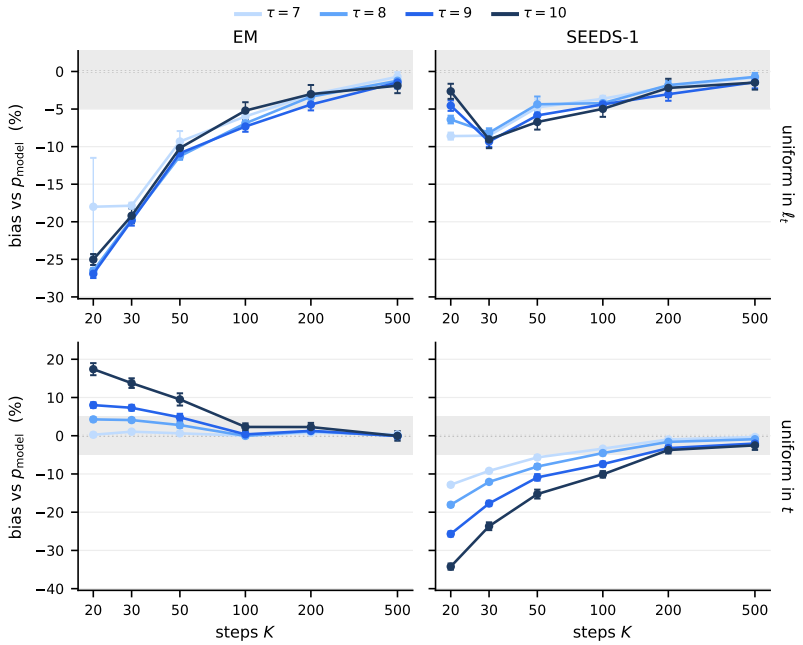


Figure 9: Measured bias versus K with 10,000 particles and the deployed guidance and no rejuvenation for the learned score. We compare the discretisation bias at different rarities, we compare against a 10^8 unconditional MC draw of the exponential integrator at K as our reference. Hence, this measures how quickly the estimates converge to the reference with K.

Table 7: Guided generation on the GMM with learned and analytical scores. Each arm is scored against the law it samples from, $p^\star$; either $p_{\text{data}}[E]$ (analytical) or an MC estimate of $p_0[E]$ (learned). Sample quality metrics are evaluated against a reference from the same law, presented as a ratio against a noise-floor. Estimates $\hat{p}/p^\star$ are biased to the MADM kernel's floor (Fig. 13). We evaluate the two-sample Kolmogorov-Smirnov (KS) test statistic, the Earth Mover's Distance (EMD) and the coefficient of variation $\text{CV}^2 = \text{Var}(\hat{p_E})/p_E^2$.

τ	score	$p^\star$	$\hat{p}/p^\star$	CV ↓	KS_{x_1}	KS_{x_2}	EMD
7	analytical	6.88×10^{-3}	0.980 ± 0.012	0.060	0.88	1.33	1.52
8	analytical	1.28×10^{-3}	0.966 ± 0.012	0.061	1.13	1.10	1.06
9	analytical	1.62×10^{-4}	0.974 ± 0.016	0.083	1.12	1.00	0.97
10	analytical	1.40×10^{-5}	0.954 ± 0.013	0.069	0.94	1.05	1.46
7	learned	7.13×10^{-3}	0.978 ± 0.012	0.060	1.11	1.20	1.14
8	learned	1.38×10^{-3}	0.985 ± 0.013	0.064	0.97	1.00	0.73
9	learned	1.87×10^{-4}	0.980 ± 0.018	0.089	0.77	0.83	0.98
10	learned	1.72×10^{-5}	0.965 ± 0.016	0.079	1.23	1.07	0.98

and analytical arms. The two panels remain close to indistinguishable, highlighting that the profile is set by the guidance geometry rather than the score itself.

G.3.3 Selection and rejuvenation

We ablate whether resampling or rejuvenation is helpful for our problem. For simplicity we provide two tables, separating the effect of each on the analytical and learned scores. Table 8 ablates the no-resampling, resampling and rejuvenation for the analytical score. Table 9 then runs the full threshold ladder on the learned score, adding the arm with no resampling at all and three noise levels where the rejuvenation move is switched off. All experiments are done using the same configuration as mentioned above. Each arm is scored against the law it samples from, $p^\star$. Weighted sample quality statistics are evaluated against a conditional reference drawn from the same law, presented as ratios to their split-half noise floor. All reported uncertainties are computed over 24 seeds.

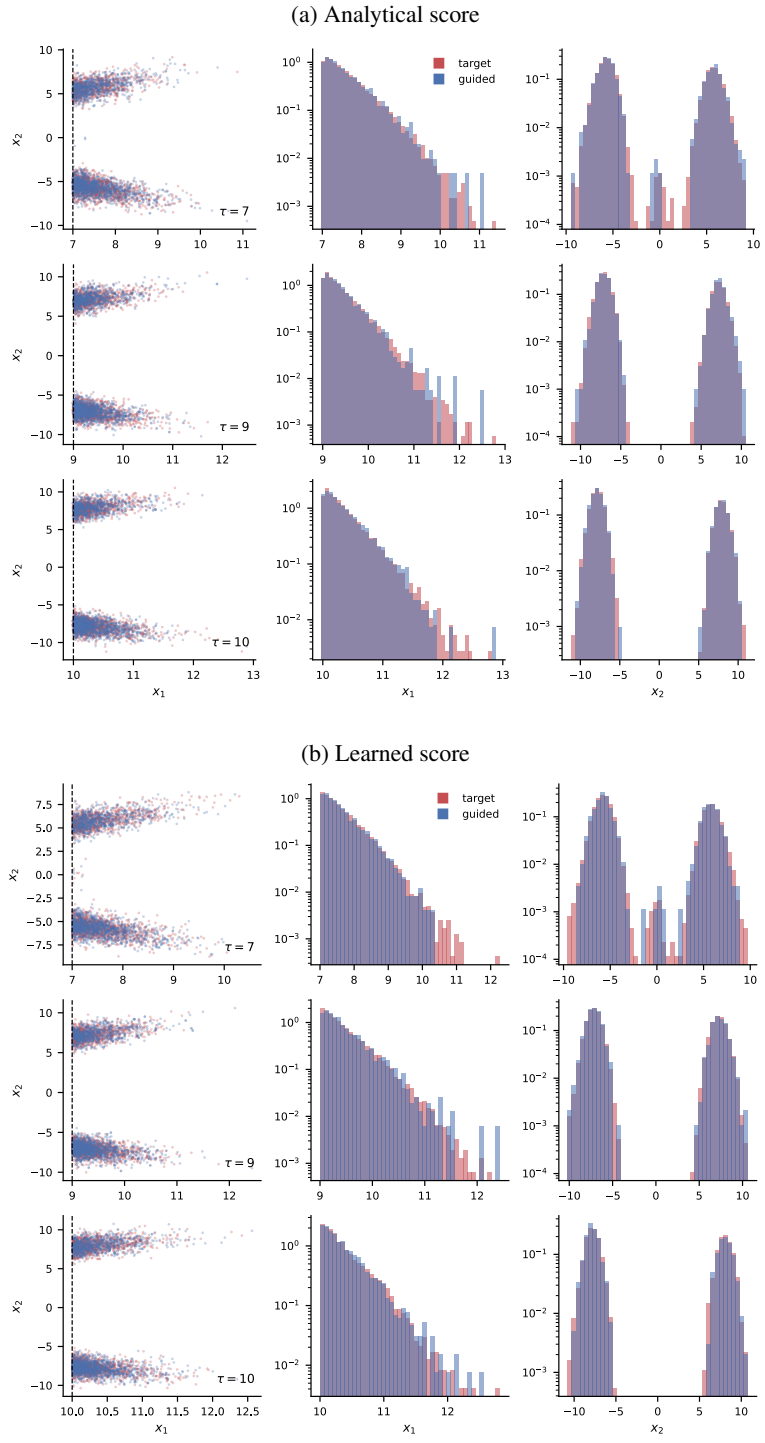


Figure 10: Conditional-sample coverage on the 2D GMM for the analytical (a) and learned (b) scores. Blue is the path-weighted guided cloud (2048 samples; pooled over 8 seeds for the learned score); red is a MC draw from the conditional $p(x \mid x_1 \geq \tau)$. *Left:* the joint cloud, with the threshold dashed. *Centre, right:* the x_1 and x_2 marginals on a log density scale. Corresponding quantitative KS/EMD metrics in Table 7.

Benefits of SMC Across, both Tables 8 and 9, resampling prunes low-weight trajectories and rejuvenation decorrelates duplicated samples. We find that rejuvenation keeps the relative variance

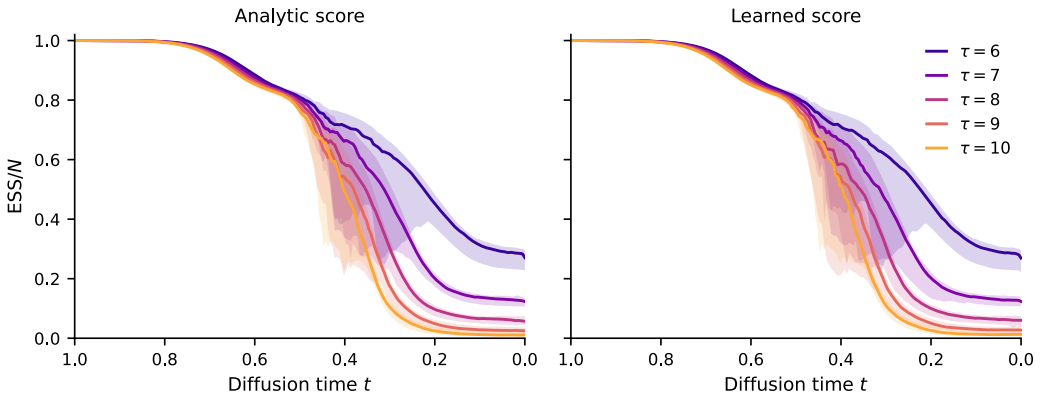


Figure 11: Effective sample size of the path weights along diffusion time, without resampling (analytic score, left; learned score, right), for $\tau \in \{6, 7, 8, 9, 10\}$. Settings as Table 7. Time runs right to left, from noise ($t = 1$) to data ($t = 0$).

Table 8: Selection and rejuvenation on the 2D GMM under the *analytical* score, 24 seeds, all other settings as in Table 7. Resampling and rejuvenation are ablated at different threshold exceedance rarities. Weighted shape-metrics are computed against a conditional reference. Rejuvenation helps reduce the CV across all rarities; without it resampling alone can adversely prune lineages.

τ	arm	CV ↓	KS_{x_1}	KS_{x_2}	EMD
7	no resampling	0.067	0.88	1.52	1.50
	resampling	0.062	1.05	0.99	0.83
	resampling + rejuv. ($M = 3$)	0.060	0.88	1.33	1.52
8	no resampling	0.083	1.14	1.35	1.17
	resampling	0.098	1.12	0.95	0.77
	resampling + rejuv. ($M = 3$)	0.061	1.13	1.10	1.06
9	no resampling	0.113	0.91	1.11	1.03
	resampling	0.150	1.15	1.47	1.14
	resampling + rejuv. ($M = 3$)	0.083	1.12	1.00	0.97
10	no resampling	0.147	0.81	1.46	2.45
	resampling	0.230	0.91	1.46	1.98
	resampling + rejuv. ($M = 3$)	0.069	0.94	1.05	1.46

constant across rarity for both analytical and learned scores, showing a noticeable decrease in CV if the base estimator is poor (Del Moral et al., 2006; 2012).

Gating the rejuvenation move by noise level For the analytical score, there is no score approximation error, hence the line-integral of the Metropolis-Hastings acceptance ratio is defined exactly up to quadrature error (Lam et al., 2026). However for a learned score it is approximated by $s_\theta \approx \nabla_{x_t} \log p_t$, and any errors in the learning process can lead to a non-conservative score approximation. The left of Fig. 12, evaluates this via the antisymmetric part of the Jacobian; here we reproduce the findings of (Lai et al., 2023; Thornton et al., 2025), where the score network becomes increasingly non-conservative as the denoising time, $t \to 0$.

A MALA transition kernel requires that M_k fulfil the Kolmogorov criteria of reversibility. As the Metropolis-Hastings acceptance ratio of MADM (Lam et al., 2026) is estimated by the line-integral of the score, i.e. Eq. (118); this means the line integral, $I(\cdot)$ must fulfil the cyclic sum (or Stokes' Theorem),

$$I(b, a) + I(c, b) + I(a, c) = 0, \tag{113}$$

Table 9: Resampling and rejuvenation ablation for the *learned* score network. Distributional metrics are presented as ratios to the reference pool's split-half noise floor. Gate levels are quoted as the half-log SNR ℓ below which the rejuvenation move triggers; the adopted gate $\ell \lesssim -0.21$ is $t \geq 0.30$. The no-resampling and resampling-only arms run at $\lambda \equiv 1$. Rejuvenation tracks the lowest CV.

event	arm	$\hat{p}/p^\star$	CV ↓	KS_0	KS_1	EMD
$x_0 > 7$	no resampling	0.952	0.070	0.74	1.00	1.09
	resampling	0.988	0.093	0.94	1.25	1.47
	resampling + rejuv. ($M = 3$)	0.985	0.061	1.01	1.17	1.28
	gated $\ell \lesssim 0.05$	0.983	0.060	0.89	1.15	1.11
	gated $\ell \lesssim -0.21$	0.978	0.060	1.11	1.20	1.14
	gated $\ell \lesssim -0.46$	0.992	0.082	0.90	1.33	1.37
$x_0 > 8$	no resampling	0.964	0.092	1.13	1.23	0.84
	resampling	0.979	0.093	1.02	1.09	1.09
	resampling + rejuv. ($M = 3$)	0.995	0.056	1.01	0.93	0.78
	gated $\ell \lesssim 0.05$	0.990	0.059	0.93	0.83	0.92
	gated $\ell \lesssim -0.21$	0.985	0.064	0.97	1.00	0.73
	gated $\ell \lesssim -0.46$	0.971	0.086	1.00	1.14	1.53
$x_0 > 9$	no resampling	0.983	0.127	0.94	0.92	0.73
	resampling	0.972	0.159	0.92	1.24	1.33
	resampling + rejuv. ($M = 3$)	0.984	0.068	0.96	1.12	0.92
	gated $\ell \lesssim 0.05$	0.985	0.073	1.10	1.08	0.84
	gated $\ell \lesssim -0.21$	0.980	0.089	0.77	0.83	0.98
	gated $\ell \lesssim -0.46$	0.978	0.109	0.99	1.53	1.74
$x_0 > 10$	no resampling	1.020	0.213	1.16	1.13	1.85
	resampling	0.986	0.166	0.75	1.74	3.90
	resampling + rejuv. ($M = 3$)	0.994	0.085	1.48	1.59	2.10
	gated $\ell \lesssim 0.05$	0.981	0.090	0.97	0.94	1.45
	gated $\ell \lesssim -0.21$	0.965	0.079	1.23	1.07	0.98
	gated $\ell \lesssim -0.46$	0.956	0.119	0.92	1.21	1.05

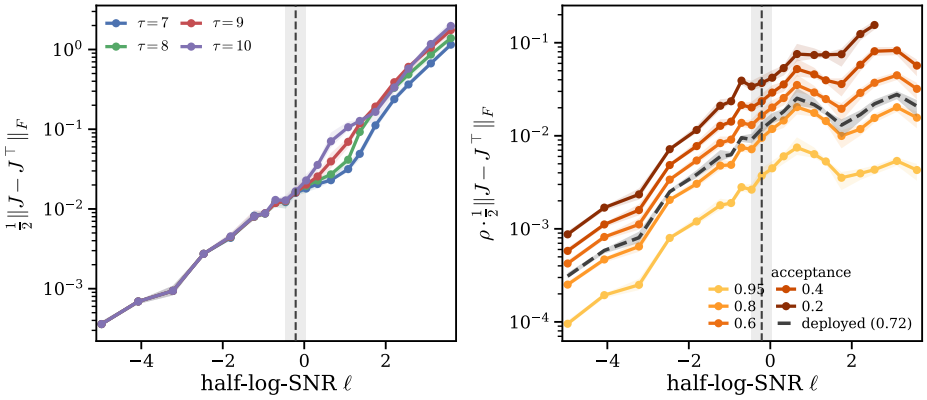


Figure 12: (Left) Magnitude of the antisymmetric part of the Jacobian of the score network, measured as a Frobenius norm for 1024 particles for the different threshold exceedance problems of Table 9. (Right) Same value as the left multiplied by ρ for $\tau > 10$ threshold exceedance problem, for different calibrated step sizes ρ, denoted by their average acceptance rate. The shaded band marks the range of gates tested and the dashed line the adopted gate.

for all triples (a, b, c). This means that the acceptance ratio of the transition: $a \to b \to c \to a$ must be equal to 1 for MADM kernel to equate to MALA. A non-conservative score does not fulfil this criterion (Vuong et al., 2025) and hence for a learned score, MADM need not be invariant to π_k.

To account for this, we turn off rejuvenation at a specific threshold in time or ℓ_t when the non-conservativity of the score increases. Table 9 ablates different thresholds. At $\tau = 10$ the ungated

move reads KS $1.48/1.59$ and EMD 2.10 against $1.23/1.07/0.98$ when gated at $t \geq 0.30$, and the same ordering holds at $\tau = 9$; we therefore adopt the gated rejuvenation for the rest of our examples, selecting the middle of the values we tested, $\ell_t \leq -0.21$ ($t \geq 0.30$). At $\tau = 7, 8$, where the corrector and resampling events were triggered less frequently, the gated and ungated rows coincide within noise, as the ungated version already sits close to the noise floor. Both gated and un-gated arms at $\tau = 10$ read 0.965 and 0.994 within 1.3 standard errors.

Step size selection As the step-size ρ defines the region where a move is proposed, i.e. $\tilde{x} - x \sim \mathcal{N}(\rho g(x)/2, \rho I)$ of Eq. (114), a larger ρ leads to, what we hypothesise, a higher exposure to the non-conservative part of the score network, which affects the estimated M_k further. For a simple visualisation, we plot the proposed step size ρ multiplied by the Frobenius norm of the antisymmetric part of the Jacobian, to the right of Fig 12. This for various ρ on the $\tau = 10$ threshold exceedance problem of Section 4.1; here the step size has been tuned to target a range of acceptance probabilities as is common (Lam et al., 2026). We elect to take smaller, more conservative steps, tuning ρ to target a 0.72 acceptance.

We hypothesise that, for a fixed number of MADM steps and ρ, there is a tradeoff between the total number of MADM steps that rejuvenate duplicated particles and the net error injected into the system by an approximate MH ratio. An adaptive schedule on $\rho(t)$ or $M(t)$ that accounts for these trade-offs could be an interesting direction for future work.

Ablation of Errors due to MADM In this setting we ablate what the discretisation error is at different K using MADM. As MADM targets the continuous-time marginal p_t instead of the discretised γ_0 law, it incurs a discretisation bias. We find the same issue while not plotted here, exists with MALA suggesting it is not a quadrature error itself. We verify this using analytical scores and the variance minimising Doob's guidance. See Fig. 13.

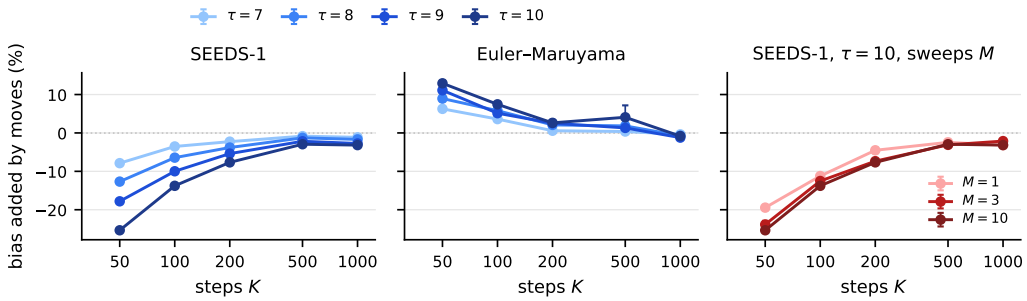


Figure 13: Bias introduced by MADM for the two integrators, using analytical scores and Doob's guidance with $M = 3$ for the first two panels. *Left & Middle* Injected bias affects rarer samples more, with all biases decreasing as $K \to \infty$. The two integrators have a different error profile. *Right* Bias plotted for the $\tau = 10$ threshold exceedance problem. We find that the error grows with M. Error bars are due to seed variability.

G.4 Box and multiple box events

We test the guidance on boxed geometries, each at two rarities, plus a bounded family. This is an analogue of an extreme defined as a set where variables exist in some range. Both families are boxes in the observable, i.e., $E = \{x : a \leq \phi(x) \leq b\}$ with $a, b \in (\mathbb{R} \cup \{\pm\infty\})^m$ and the inequalities read component-wise. We take the interval guidance of Eq. (96), see App. E. Any corner boxes that we use, takes $b = +\infty$, and it represents extremes that are joint threshold exceedances; bounded boxes have finite a and b.

For simplicity we have illustrated the box boundaries we test in Fig 14 overlaid against the true unconditional density of the analytical 2D GMM. We report performance on the learned variant in Table 10, where we ablate the effect of no resampling, resampling with gated/un-gated rejuvenation. We find that similar to the other experiments, rejuvenation lowers the CV on the rare boxes ($p^\star \leq$

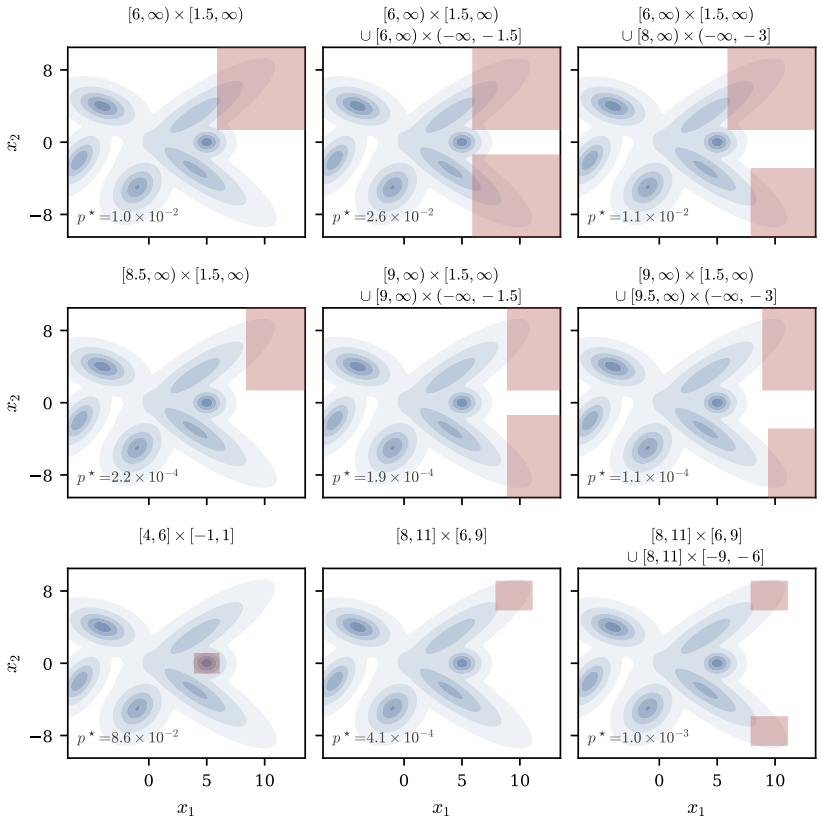


Figure 14: Illustrative example of box boundaries. Plotted against a backdrop of the unconditional density of the 2D-GMM. Quantitative results of rare-event sampling are given in Table 10 below.

2×10^{-4}, by 1.5-2.5×) and is neutral or slightly worse on the easy ones ($p^\star \geq 10^{-2}$), where resampling is triggered at most twice per run.

Table 10: Box events on the learned score. Every region is drawn on the model density in Fig. 14. $p^\star$ is the $K = 1000$ chain's own tail (Table 5 pool). We compare the estimate $\hat{p}$ and the shape metrics of the final particles, presented as a ratio against a noise floor.

event	variant	$p^\star$	$\hat{p}/p^\star$	CV ↓	KS_0	KS_1	EMD
$[6,\infty) \times [1.5,\infty)$	no resampling	9.97×10^{-3}	0.955	0.066	0.95	0.98	1.01
	resampling		1.004	0.107	1.37	0.87	0.98
	resampling + rejuv. ($M = 3$)		1.000	0.070	0.89	0.91	0.96
	gated $\ell \lesssim -0.21$		1.000	0.070	0.94	0.89	1.00
$[6,\infty) \times [1.5,\infty)$ $\cup [6,\infty) \times (-\infty,-1.5]$	no resampling	2.60×10^{-2}	0.966	0.057	1.15	1.15	1.09
	resampling		0.979	0.055	0.99	1.08	0.95
	resampling + rejuv. ($M = 3$)		0.988	0.042	0.87	1.10	1.15
	gated $\ell \lesssim -0.21$		0.992	0.044	0.96	1.44	1.50
$[6,\infty) \times [1.5,\infty)$ $\cup [8,\infty) \times (-\infty,-3]$	no resampling	1.08×10^{-2}	0.956	0.063	1.10	0.85	0.82
	resampling		0.992	0.086	0.90	1.27	1.15
	resampling + rejuv. ($M = 3$)		1.011	0.089	0.94	1.03	1.36
	gated $\ell \lesssim -0.21$		1.011	0.089	1.02	1.10	1.07
$[8.5,\infty) \times [1.5,\infty)$	no resampling	2.16×10^{-4}	0.945	0.132	1.21	0.96	0.90
	resampling		0.967	0.116	1.19	0.86	0.96
	resampling + rejuv. ($M = 3$)		0.979	0.086	1.07	0.90	0.96
	gated $\ell \lesssim -0.21$		0.982	0.105	1.17	0.91	1.02
$[9,\infty) \times [1.5,\infty)$ $\cup [9,\infty) \times (-\infty,-1.5]$	no resampling	1.87×10^{-4}	0.986	0.123	1.15	1.32	0.85
	resampling		0.990	0.133	1.10	1.17	1.28
	resampling + rejuv. ($M = 3$)		1.008	0.079	1.22	1.34	1.12
	gated $\ell \lesssim -0.21$		1.008	0.113	1.02	1.03	0.96
$[9,\infty) \times [1.5,\infty)$ $\cup [9.5,\infty) \times (-\infty,-3]$	no resampling	1.12×10^{-4}	0.972	0.164	1.11	1.21	1.26
	resampling		0.972	0.133	1.18	1.83	1.97
	resampling + rejuv. ($M = 3$)		0.998	0.066	1.15	1.00	1.29
	gated $\ell \lesssim -0.21$		1.010	0.088	1.06	1.00	0.82
$[4,6] \times [-1,1]$	no resampling	8.66×10^{-2}	0.968	0.032	1.04	1.16	1.13
	resampling		0.987	0.050	1.18	1.01	1.08
	resampling + rejuv. ($M = 3$)		0.986	0.035	1.19	0.96	1.09
	gated $\ell \lesssim -0.21$		0.992	0.035	1.04	0.89	0.98
$[8,11] \times [6,9]$	no resampling	4.11×10^{-4}	0.951	0.067	1.06	1.11	1.07
	resampling		0.996	0.092	1.02	1.06	0.96
	resampling + rejuv. ($M = 3$)		0.984	0.052	0.96	1.15	1.04
	gated $\ell \lesssim -0.21$		0.983	0.051	0.97	1.10	0.98
$[8,11] \times [6,9]$ $\cup [8,11] \times [-9,-6]$	no resampling	1.01×10^{-3}	0.964	0.057	0.81	1.41	1.35
	resampling		0.984	0.056	0.81	0.84	1.09
	resampling + rejuv. ($M = 3$)		0.986	0.050	0.86	1.12	1.57
	gated $\ell \lesssim -0.21$		0.984	0.053	0.89	0.90	1.26

G.5 Ablation of Parameters

We ablate our parameters once: one the $\tau = 10$ threshold exceedance problem and we reuse it throughout our experiments. There are two parameters we ablate over the starting point of our guidance t_{on} and the exponential p in $\lambda(t) = 1-(t/t_{\text{on}})^p$. Below are the results of the ablation for our method, separated by the base, resampling and resampling with rejuvenation; we also ablate for the lowest CV of Manshausen et al. (2026) which we use for the experiments in Fig. 2. All experiments were done with 24 seeds, $N = 1000$, $K = 1000$ using the SEEDS-1 integrator (Gonzalez et al., 2023), with $M = 3$ steps of MADM (Lam et al., 2026) after resampling.

Table 11: Schedule ablation at $\tau = 10$, $K = 1000$, $N = 1000$: CV of $\hat{p}$ for the tempering schedule $\lambda(t) = 1 - (t/t_{\text{on}})^p$. Bold marks the minimum of each column. We find that the no-rejuvenation variants converged on the same tempering schedule $\lambda_t \equiv 1$ compared to the MADM variant. All schedules with rejuvenation yield a lower CV compared to without.

schedule	base ↓	+ resampling↓	+ MADM ↓	(Manshausen et al., 2026) ↓
analytic score				
$\lambda \equiv 1$	**0.15**	**0.23**	0.09	**0.29**
$t_{\text{on}} = 1.0,\ p = 2$	0.28	0.32	0.08	0.68
$t_{\text{on}} = 0.9,\ p = 2$ (deployed)	0.35	0.33	**0.07**	0.83
$t_{\text{on}} = 0.7,\ p = 2$	0.62	0.40	0.10	1.26
$t_{\text{on}} = 0.5,\ p = 2$	1.73	0.91	0.13	1.94
$t_{\text{on}} = 0.9,\ p = 1$	0.71	0.48	0.11	1.45
$t_{\text{on}} = 0.9,\ p = 4$	0.20	0.23	0.10	0.56
learned score				
$\lambda \equiv 1$	**0.21**	**0.17**	0.11	**0.17**
$t_{\text{on}} = 1.0,\ p = 2$	0.32	0.26	0.13	0.51
$t_{\text{on}} = 0.9,\ p = 2$ (deployed)	0.40	0.29	**0.08**	0.68
$t_{\text{on}} = 0.7,\ p = 2$	0.70	0.36	0.15	1.18
$t_{\text{on}} = 0.5,\ p = 2$	2.00	0.75	0.25	1.90
$t_{\text{on}} = 0.9,\ p = 1$	0.78	0.37	0.18	1.28
$t_{\text{on}} = 0.9,\ p = 4$	0.23	0.18	0.16	0.45

H CLIMATE EXPERIMENTS

H.1 SETUP AND MODEL DETAILS

We use a pretrained score-based diffusion climate emulator that acts as a surrogate for physics-based Earth System Models (Bouabid et al., 2026) (licensed under MIT Licence), trained on large-ensemble runs from the Coupled Model Intercomparison Project (CMIP6) on different climate change (forcing) scenarios (O'Neill et al., 2016). Our results are based on the MPI-ESM1-2-LR emulator. The model outputs monthly-averaged fields conditioned on the global mean surface temperature anomaly (GMST), which captures global warming against preindustrial warming, and on the month of the year, which captures annual variability.

The emulator generates four surface fields jointly across the globe on every draw, listed in Table 12 by their CMIP6 short names. All fields are monthly means and are handled throughout as *anomalies* against the pre-industrial (piControl) climatology, matching the convention of Table 13.

Table 12: Emulated variables with their corresponding CMIP6 short names, descriptions and units.

Short name	Description	Units
`tas`	2 m air temperature	K
`hurs`	2 m relative humidity	%
`pr`	Precipitation rate	mm day^{-1}
`sfcWind`	10 m wind speed	m s^{-1}

We compute a large Monte Carlo average of the rare event probability with $N = 9{,}011{,}200$ unguided draws at a GMST of $\Delta T = 1.5$ K for the month of July to validate our results against. For the region R, with $\phi(\cdot)$ as a per-variable regional average, the associated mean, standard deviation and tail values at different rarities of interest are given in Table 13.

H.1.1 IMPLEMENTATION DETAILS

For all experiments throughout we use $K = 200$ discretisation of the exponential integrator in the noise-prediction mode, see App. D and Fig. 15. We schedule $\lambda(t)$ and $\delta(t)$ for all experiments below. For a VE-schedule, $\alpha_t \equiv 1$, so the half-log SNR is $\ell_t = -\log \bar{\sigma}_t$ and we specify both schedule in $\bar{\sigma}_t$: piecewise linear in λ and δ_t is interpolated log-linearly between two endpoints. Similar to

Table 13: Tail thresholds for the four emulator variables, from $N = 9{,}011{,}200$ unguided Monte Carlo draws over R at $\Delta T = +1.5$ K in July. Upper-tail entries satisfy $P(\phi \geq \tau) = p$, lower-tail entries $P(\phi \leq \tau) = p$. Binomial relative standard errors are 1.1%, 3.3% and 10.5% at $p = 10^{-3}, 10^{-4}, 10^{-5}$ respectively. The guided runs use thresholds from an earlier (8.9 million) reference pool, e.g. $\tau_{\texttt{tas}} = 5.8073$ K and $\tau_{\texttt{hurs}} = -15.8307\%$ at $p = 10^{-3}$ (Table 17), all within $< 1.5\%$. Every $p^\star$ in the paper is evaluated at the thresholds below and the comaprision remain valid, but a small bias remains.

		Marginal		Upper tail			Lower tail		
Variable	Unit	Mean	SD	10^{-3}	10^{-4}	10^{-5}	10^{-3}	10^{-4}	10^{-5}
`tas`	K	+2.577	1.023	5.813	6.498	7.096	−0.539	−1.174	−1.727
`pr`	mm/day	−0.250	0.450	1.698	2.311	2.842	−1.334	−1.517	−1.650
`hurs`	%	−3.275	4.220	10.519	13.485	16.054	−15.838	−18.387	−20.554
`sfcWind`	m/s	+0.193	0.186	0.773	0.895	1.004	−0.387	−0.508	−0.613

GMM example, we turn guidance, i.e. $\lambda(t) \equiv 0$ off for the first 15% ($\bar{\sigma} > 40$) of the denoising process to account for any approximation errors in $\hat{x}_{0|t}$, converging to $\lambda(\bar{\sigma}) \to 1$ after adding a lambda overshoot at a maximum of 1.5. $\delta(t)$ is held at 0.5 until $\bar{\sigma} = 0.3$ SD and sharpens to 0.01 by $\bar{\sigma} = 0.03$SD, where SD is the standard deviation of each observable approximated by the same method as the ΠGDM mentioned in App B. We decrease δ towards the end of the denoising process to reduce the variance of our estimator due to mismatch in the generated π_0. For simplicity, we do not anneal $\tau(\bar{\sigma})$.

Correctness of our method is independent of these choices, so they are selected for variance alone. We set them empirically on the $p^* \approx 10^{-3}$ rarity, PNW hot threshold exceedance problem using a pilot sweep at the same setup as described above. We provide an ablation of the values we tested below (Table. 22), but adopt the *same* hyper-parameters for all experiments, with $N = 1024$, $K = 200$ and 8-seeds throughout, see Table 14.

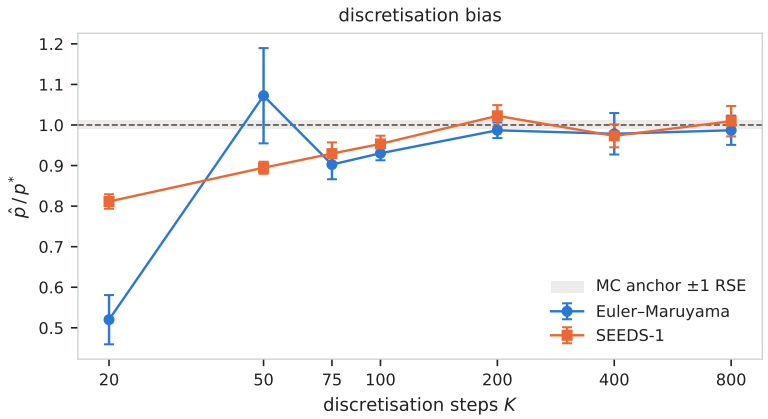


Figure 15: Discretisation bias for the integrator for the τ_1 PNW heatwave event using the SEEDS-1 integrator, with 1024 samples at each point and without any rejuvenation. Dotted line represents the reference pool. We find that for SEEDS-1 and Euler-Maruyama integrators using $K = 200$ saturates the discretisation bias. We could not compare against a large K unconditional reference due to prohibitive costs. estimate. Bars represent ±1 standard error of the mean over 8 shared seeds across both integrators.

All experiments were done on a single Nvidia H100 GPU with 96 GB of VRAM. We used approximately 370 GPUh to generate the 9 million reference samples. A single batch of 1024 unconditional draws at $K = 200$ steps takes approximately 155 seconds, with the method and its variants multiplying this cost; we summarise this in Table 21.

Table 14: Deployed schedule and sampler configuration for the climate emulator experiments. $\bar{\sigma}$ is the marginal noise standard deviation. Selected once on the 10^{-3} PNW temperature event and applied unchanged to other experiments. $^{\ddagger}\delta$ is quoted in units of the observable's unconditional standard deviation.

	Symbol	Value
Schedules		
Tempering onset	$\bar{\sigma}_{\text{on}}$	40
Tempering peak	$\lambda_{\max}$	1.5 at $\bar{\sigma} = 2$
Tempering relaxation	$\bar{\sigma}_{\text{end}}$	0.3
Smoothing width, transport	δ_{hi}	0.5 SD
Smoothing width, floor	δ_{lo}	0.01 SD
Ramp window		$\bar{\sigma} \in [0.03, 0.3]$
Ramp exponent		$\log_{10} 50 \approx 1.70$
Observable SD, frozen‡ (`tas`, `hurs`)	SD	1.017 K, 4.224%
Threshold	τ	constant
Sampler		
Particles	N	1024
Integration steps	K	200
Integrator		exponential (noise-prediction)
Resampling		adaptive, ESS$/N < 0.5$
Rejuvenation	M	0
Noise range	$[\bar{\sigma}_{\min}, \bar{\sigma}_{\max}]$	$[0.01, 146.09]$
Posterior proxy	s_t^2	$\bar{\sigma}^2/(1+\bar{\sigma}^2)$
Look-ahead bandwidth	δ_{eff}^2	$\delta(\bar{\sigma})^2 + s_t^2\|a\|^2$

Table 15: Marginal extreme-event probabilities over the PNW target region in July at $\Delta T = +1.5$ K, for hot ($\phi_{\texttt{tas}} \geq \tau$) and dry ($\phi_{\texttt{hurs}} \leq \tau$) events. For each variable, τ_1, τ_2, τ_3 are chosen so that the event is successively rarer (exact values in App. H). Ground-truth $p^\star$ are computed with $\sim 10^7$ MC samples. Unguided Monte Carlo and our guided method use the same computational budget. Estimated probabilities, $\hat{p}$ are reported as a ratio $\hat{p}/p^*$ with mean ± 1 standard deviation over 8 seeds. The variance ratio (VR) is $\text{Var}(\hat{p}_{\text{unguided}})/\text{Var}(\hat{p}_{\text{guided}})$. The hot results are shown in Fig. 3.

	Hot: $\phi_{\texttt{tas}}(x) \geq \tau$				Dry: $\phi_{\texttt{hurs}}(x) \leq \tau$			
		$\hat{p}/p^\star$				$\hat{p}/p^\star$		
	$p^\star$	Unguided	Guided	VR	$p^\star$	Unguided	Guided	VR
τ_1	1.02e-3	1.10 ± 0.22	$\mathbf{0.99 \pm 0.04}$	35×	1.01e-3	1.42 ± 0.49	$\mathbf{0.98 \pm 0.07}$	43×
τ_2	1.01e-4	0.26 ± 0.72	$\mathbf{1.06 \pm 0.09}$	63×	1.00e-4	2.08 ± 1.92	$\mathbf{1.00 \pm 0.10}$	397×
τ_3	1.01e-5	0.00 ± 0.00	$\mathbf{1.21 \pm 0.26}$	N/A	1.01e-5	0.00 ± 0.00	$\mathbf{1.20 \pm 0.30}$	N/A

H.2 Experiment 1: Compound hot & dry events

This subsection supports the first climate experiment of Section 4.2: the ablation behind the deployed recipe, and the fields the guided sampler produces at each rarity. We begin with the marginal threshold exceedance problem, ith the hot (`tas`) and dry (`hurs`) marginal results reported in Table 15, followed by the joint.

H.2.1 Selection and rejuvenation

Similar to Appendix G.3.3, we ablate the effect of resampling and rejuvenation on the hot and dry threshold exceedance events and the joint hot-and-dry event that we report the net speed-up on. This is because while resampling does not cost extra wall clock time, rejuvenation does. We evaluate whether the reduction in CV justifies the added computation cost. For this, we use a fixed $M = 5$ MADM steps using 2 degrees of quadrature. Additionally, we cache the estimate of $\log \Gamma_\pi$ in Eq. (119) for either the current x or proposal $\tilde{x}$ so that it may be reused between moves. For the same reasons as

highlighted in Appendix G, we set the acceptance target at 0.72 and gate the rejuvenation move to run only at noise levels $\bar{\sigma} \geq 0.30$.

Table 16 captures this ablation. We report the CV which is the spread of the estimated $\hat{p}$ across 8 seeds; $\hat{k}$ is the generalised-Pareto shape parameter of the importance weights, fitted with the top 96 ($\min(0.2N, 3\sqrt{N})$) (Vehtari et al., 2024), within a run, reported as the median over seeds. We find that without resampling all weights fall into $\hat{k} \geq 0.5$ (no finite second moment) and most outside of the $\hat{k} \geq 0.7$ (unreliable importance sampling estimate) range. This suggests that the weights are very heavy-tailed and the convergence of the no-resampling estimator is slow to converge, with an accompany CV that is optimistic.

Throughout the experiments, resampling improves the CV and rejuvenation helps in the deeper rarities where the CV remains high after resampling. W_1 is the weighted Wasserstein-1 distance of the quantity of interest $\phi(\cdot)$ between the generated samples and the Monte Carlo reference, reported against a null floor. All variants report a similar W_1 and there is no discernible issue with any. However, it is worth noting that rejuvenation adds $\sim$1.2–1.3$\times$ wall-clock time compared to resampling alone, hence there is a trade-off between a larger N versus more rejuvenation steps at finite compute budget. In the experiments below, unless stated otherwise we run with zero rejuvenation steps.

H.2.2 Conditional fields for temperature events

We plot both the local (Fig. 16) and global (Fig. 17) guided fields for the threshold exceedance problem to visualise representative draws from the conditional model obtained using guidance. In the example below, guidance is only applied to the region R for the `tas` variable, any structure or perturbation of other variables or away from R is due to the unguided score.

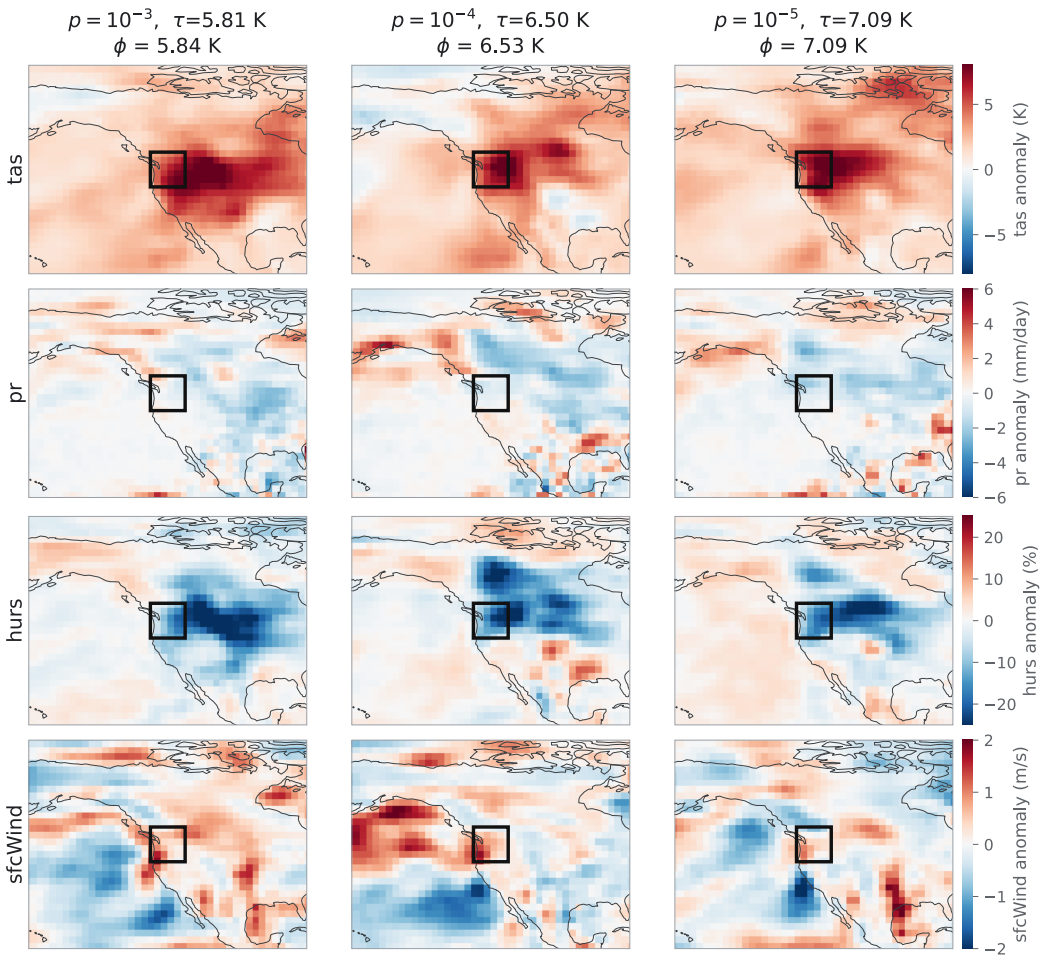


Figure 16: Top-weight crossing fields over the Pacific Northwest (PNW) target region produced by the guided sampler at different exceedance threshold τ. Guided samples based on the `tas` variable alone. Same setup as reference Table 15. p denotes $p_0[E]$ and $\phi(\cdot)$ is the regional average temperature.

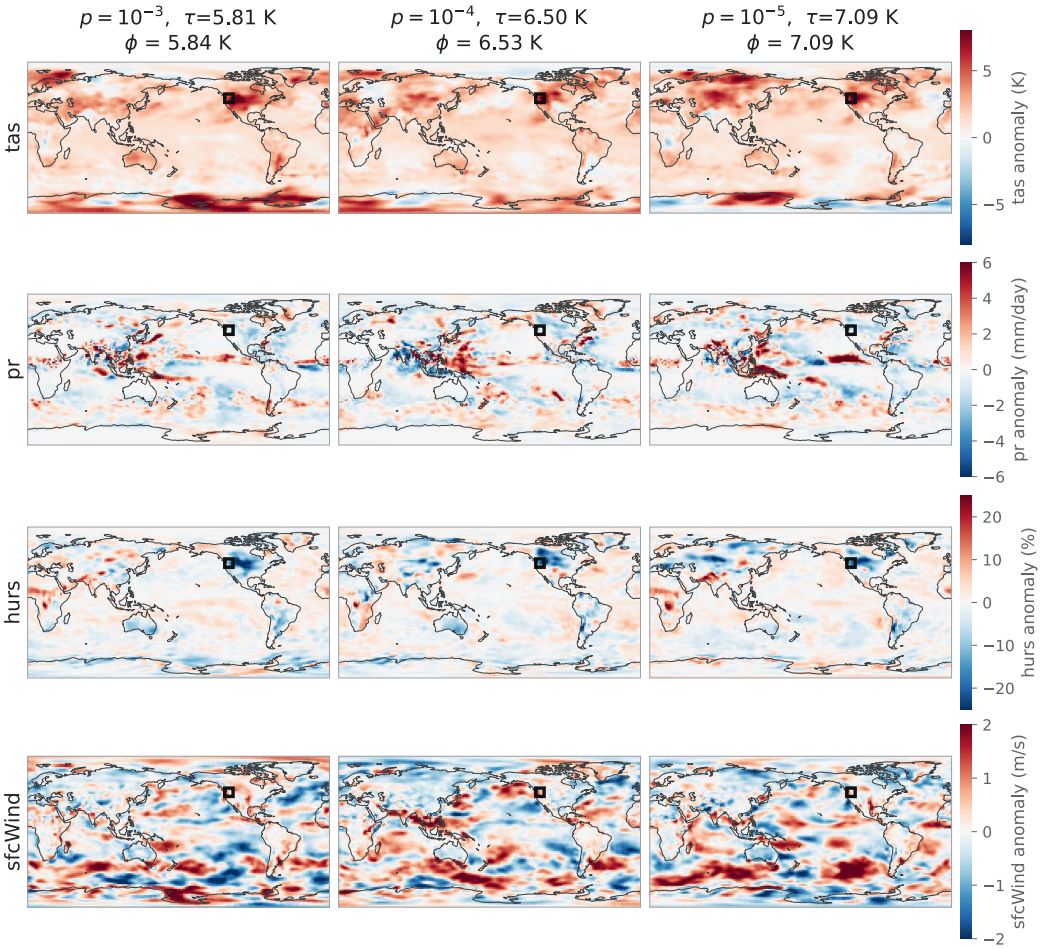


Figure 17: The same fields as Fig. 16 on the global domain.

Table 16: Ablation of resampling and rejuvenation on the climate emulator for the hot, dry, and joint hot-and-dry events. Same parameters as in the main text; details in App. H.2.1. Ratios are mean ± one standard error over 8 seeds. W_1 is a weighted distance between the $\phi(\cdot)$ of generated samples and draws from a reference, scaled against a null floor. The relative error of $p^\star$ is $1.1/3.3/10.5\%$ at the three marginal rarities and $0.6/2.3/8.9\%$ at the three joint rarities. CV is estimated across $\hat{p}$ estimates and $\hat{k}$ is estimated per seed and averaged across 8. $\hat{k} > 0.7$ represents weights that have no finite second moment (Vehtari et al., 2024), so the CV estimates are optimistic.

	$p^\star$	arm	$\hat{p}/p^\star$	CV ↓	ESS/N	$\hat{k}$	W_1 tas	W_1 hurs
Hot	10^{-3}	no resampling	0.963 ± 0.047	0.137	0.07	0.62	0.86	–
		adaptive resampling	0.989 ± 0.013	0.037	–		1.77	–
		+ MADM moves	1.008 ± 0.017	0.048	–		1.47	–
	10^{-4}	no resampling	0.935 ± 0.043	0.130	0.04	0.67	1.04	–
		adaptive resampling	1.056 ± 0.033	0.087	–		1.17	–
		+ MADM moves	0.993 ± 0.033	0.093	–		1.06	–
	10^{-5}	no resampling	0.921 ± 0.056	0.171	0.02	0.82	1.00	–
		adaptive resampling	1.213 ± 0.090	0.210	–		0.69	–
		+ MADM moves	1.143 ± 0.054	0.134	–		0.78	–
Dry	10^{-3}	no resampling	0.973 ± 0.045	0.131	0.07	0.64	–	1.28
		adaptive resampling	0.977 ± 0.026	0.076	–		–	1.28
		+ MADM moves	1.005 ± 0.021	0.060	–		–	1.49
	10^{-4}	no resampling	0.948 ± 0.069	0.205	0.03	0.80	–	1.31
		adaptive resampling	1.004 ± 0.034	0.096	–		–	1.18
		+ MADM moves	1.043 ± 0.033	0.090	–		–	1.28
	10^{-5}	no resampling	0.973 ± 0.110	0.319	0.03	0.79	–	1.28
		adaptive resampling	1.199 ± 0.103	0.244	–		–	0.75
		+ MADM moves	1.100 ± 0.051	0.132	–		–	0.66
Joint	10^{-2}	no resampling	0.952 ± 0.023	0.069	0.09	0.54	0.98	1.00
		adaptive resampling	1.001 ± 0.029	0.081	–		1.74	1.86
		+ MADM moves	0.996 ± 0.019	0.053	–		1.81	1.57
	10^{-3}	no resampling	0.948 ± 0.027	0.080	0.04	0.75	0.97	0.94
		adaptive resampling	1.001 ± 0.030	0.085	–		1.14	1.28
		+ MADM moves	1.047 ± 0.042	0.113	–		1.18	1.04
	10^{-4}	no resampling	1.012 ± 0.072	0.201	0.02	0.90	1.39	1.00
		adaptive resampling	1.080 ± 0.045	0.118	–		1.12	1.00
		+ MADM moves	1.126 ± 0.050	0.125	–		1.05	0.88

H.2.3 Estimating the marginals and the joint

We expand on the multivariate hazard experiment of Section 4.2. d-values are computed using three guided runs for the guided process and read out as a fraction of joint exceedance $p(\text{hot \& dry})$ from the Monte Carlo reference. For the joint guided runs we ablate between two variants for the CDF reward, one with a joint 2 dimensional CDF across the two variables $\Phi(\texttt{hurs}, \texttt{tas})$ and another as a product of two individual CDFs $\Phi(\texttt{hurs})\Phi(\texttt{tas})$. We replace the ΠGDM's isotropic covariance estimate, $\text{Cov}(x_0|x_t)$ with the model's by computing the Jacobian Vector Product of $a^\top \Sigma_t a$ (Boys et al., 2023), additionally we use a joint 2-dimensional bandwidth-inflated CDF $\Phi_2(\cdot)$ (Eq. (88)) as the reward estimated using Drezner's algorithm with a 16-point Gauss-Legendre quadrature (Drezner & Wesolowsky, 1990).

Table 17 summarises the probability estimate obtained via the Monte Carlo reference which we compare against. In this setting we define p to be the probability a marginal exceeds a specific threshold, with the joint probability given by p^2 under an independence assumption.

Table 17: Monte Carlo estimate and dependence for the compound hot-dry event at matched marginal quantiles. p represents the compound hot-dry event at matched marginal quantiles: at each level p both thresholds are the pool's own p-quantiles, so $p_{\text{indep}} = p^2$ exactly; quoted errors are binomial, and the 10^{-4} row rests on 126 joint crossers.

marginal p	τ_{tas} (K)	τ_{hurs} (%)	p_{joint}	p_{indep}	$d^\star$
10^{-1}	3.8879	-8.6109	$(5.041 \pm 0.008) \times 10^{-2}$	1.00×10^{-2}	5.02 ± 0.01
10^{-2}	4.9862	-12.7843	$(3.159 \pm 0.019) \times 10^{-3}$	1.00×10^{-4}	31.3 ± 0.2
10^{-3}	5.8073	-15.8307	$(2.141 \pm 0.049) \times 10^{-4}$	1.00×10^{-6}	209.7 ± 4.8
10^{-4}	6.4960	-18.3927	$(1.398 \pm 0.124) \times 10^{-5}$	1.00×10^{-8}	1393 ± 124

H.2.4 Factorised versus multivariate guidance

We ablate the two reward variants - factorised (Eq. (94)) and joint (Eq. (88)) of the compound weather event of Section 4.2. Across rarities, we compute the guidance based on a factorised reward (i.e. a product of two independent 1-dimensional CDFs) and its correlated Φ_2 variant. Both variants yield a different guidance drift as well as look-ahead reward $r_{0,\delta}$. Additionally, for the Gaussian surrogate of $p(x_0|x_t) \approx N(\hat{x}_{0|t}, \Sigma)$, where $\hat{x}_{0|t}$ is Tweedie's estimate, we replace the isotropic Σ approximation of (Song et al., 2023), with a JVP estimate of $\sigma^2 A^T J A$, where A is defined by the linear function, J is the Jacobian of the network and σ^2 is the current noise level (Appendix E.1). The joint estimator costs roughly $1.64\times$ wall-clock time (Appendix H.4) of the factorised but yields a lower variance of $p_0[E]$, which we discuss below.

The experiment was conducted with identical seeds and settings as the main text and compared against the MC reference. We use the same 8 seeds across both variants.

Table 18: Comparison of two reward estimates for compound hot & dry events. Joint reward yields a lower variance estimate at all thresholds, whereas at $N = 1024$ the factorised reward underestimates the probability due to finite-N effects of SMC. Notably the 10^{-4} reference itself carries an 8.9% standard error. Setup remains identical to Table 1, MC reference: Table 17.

	Factorised reward			Joint Φ_2 reward		
marginal p	$\hat{p}/p^\star$	CV $\downarrow$	$\hat{d}/d^\star$	$\hat{p}/p^\star$	CV $\downarrow$	$\hat{d}/d^\star$
10^{-2}	0.919 ± 0.044	0.135	0.92 ± 0.04	$\mathbf{1.001 \pm 0.029}$	**0.081**	$\mathbf{1.01 \pm 0.04}$
10^{-3}	0.828 ± 0.069	0.228	0.85 ± 0.05	$\mathbf{1.001 \pm 0.030}$	**0.085**	$\mathbf{1.04 \pm 0.04}$
10^{-4}	0.881 ± 0.107	0.234	0.84 ± 0.06	$\mathbf{1.080 \pm 0.045}$	**0.118**	$\mathbf{1.03 \pm 0.05}$

We find that across all of the variables, the correlated reward yields a better CV, corresponding to a better estimate of the guidance drift resulting in a lower as seen in Table 18. The uncorrelated reward underestimates the probability. However, this is a finite N behaviour of an improper proposal. As the true distribution p_t is highly correlated between `tas` and `hurs`, measured at $\rho \approx 0.72$, uncorrelated rewards undersample the correlated high weight region, where $p(x_i)/q(x_i) > 1$. These regions would contribute a large weight to the estimator and increase the estimated $p_0[E]$. As $\pi_t(x) \propto p_t e^{r_{0,\delta}(x)}$, a large fraction of the inherent correlation is already present in the proposal through p_t, hence at this sample size we mostly recover the dependence $\hat{d}/d^\star \approx 0.84$ at a marginal $p = 10^{-4}$.

H.2.5 Conditional fields for compound events

The compound event as a field. Figure 18 contrasts the conditional mean field under the hot-only constraint with the one under the joint hot & dry constraint of Table 1. Guiding on the temperature alone, reduces the humidity across the region, illustrating why the dependency ratio $d > 1$.

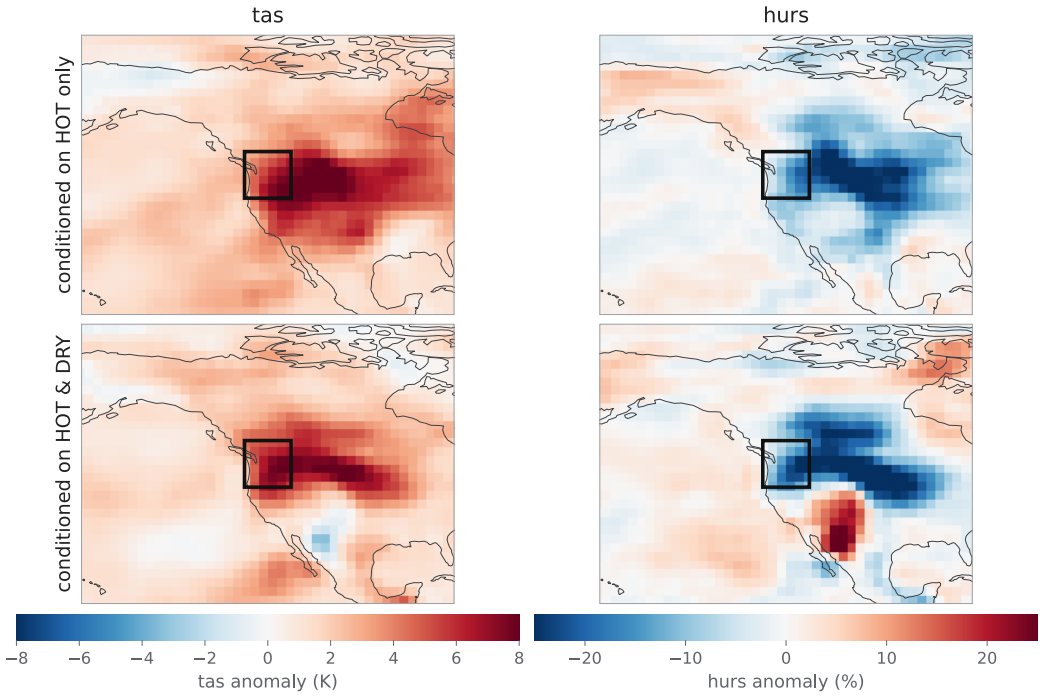


Figure 18: Conditional mean fields for the compound event of Table 1, over the PNW box in July at GMST $+1.5$ K. *Top:* conditioned on the hot constraint alone. *Bottom:* conditioned jointly on hot & dry. *Left:* near-surface temperature. *Right:* relative humidity. Both constraints are at marginal 10^{-3}; the box is the region of interest. Guiding on temperature alone reduces the humidity across a region.

H.3 EXPERIMENT 2: A CHANGING CLIMATE

Table 19: Warming sweep at the frozen event $\tau = 5.81$ K over the region computed with 8 seeds per point. p^* is obtained from a large Monte Carlo references which exists only at $\Delta T \geq 1.5$ K (relative errors 1.1%, 3.2%, 0.7%, 0.13%) and the guided estimates match all four.

ΔT (K)	$\hat{p}$	s.e. $\downarrow$	CV $\downarrow$	$p^\star$	$\hat{p}/p^\star$
1.5	9.93×10^{-4}	$\pm0.24\times10^{-4}$	0.069	1.015×10^{-3}	0.979
2.0	1.55×10^{-2}	$\pm0.04\times10^{-2}$	0.068	1.508×10^{-2}	1.025
3.0	3.96×10^{-1}	$\pm0.02\times10^{-1}$	0.017	3.956×10^{-1}	1.002
4.0	9.52×10^{-1}	$\pm0.01\times10^{-1}$	0.004	9.508×10^{-1}	1.001

Table 19 reports the sweep at a frozen threshold against the large Monte Carlo reference, the generated fields behind it are in Fig. 19. As the global mean surface temperature increases, the degree of warming across regions varies per region which is expected under the modelling condition of (Bouabid et al., 2026). As such, the regional mean temperature is greater than the GMST which is reflected at $\Delta T = 4.0$K, where the estimated $p_0[E] = 0.95$. Table 20 reports the performance of DireSMC compared against a compute normalised MC estimate. Due to computational constraints we could not validate the estimates at $\Delta T \leq 1.2$ K; to obtain a MC probability estimate at a 5% relative error we would require on the order of $10^8/10^7/10^6$ draws from the unconditional model at $\Delta T = 0.0/1.0/1.2$ K respectively.

Table 20: Warming sweep at the frozen event $\tau = 5.81$ K over the region computed with 8 seeds per point. We compare against an unguided MC estimate at matched clock time allocating it $4.7\times$ particles. Zero counts the number of seeds that reported no generated samples. See Table. 19 for DireSMC validated against a large MC pool. Var. Ratio = $\mathrm{Var}(\hat{p}_{\mathrm{MC}})/\mathrm{Var}(\hat{p})$.

	DireSMC		Crude MC			
ΔT (K)	$\hat{p}$	CV	$\hat{p}_{\mathrm{MC}}$	CV	Zero	Var. Ratio
0.0	1.24×10^{-6}	0.403	0	—	8/8	N/A
0.5	1.89×10^{-6}	0.259	0	—	8/8	N/A
1.0	3.78×10^{-5}	0.108	0	—	8/8	N/A
1.2	1.42×10^{-4}	0.097	1.56×10^{-4}	1.553	5/8	308
1.5	9.93×10^{-4}	0.069	1.01×10^{-3}	0.585	0/8	75
2.0	1.55×10^{-2}	0.068	1.66×10^{-2}	0.154	0/8	5.9
2.5	1.11×10^{-1}	0.026	1.17×10^{-1}	0.033	0/8	1.8
3.0	3.96×10^{-1}	0.017	4.01×10^{-1}	0.014	0/8	0.7
3.5	7.56×10^{-1}	0.007	7.56×10^{-1}	0.009	0/8	1.9
4.0	9.52×10^{-1}	0.004	9.51×10^{-1}	0.003	0/8	0.6

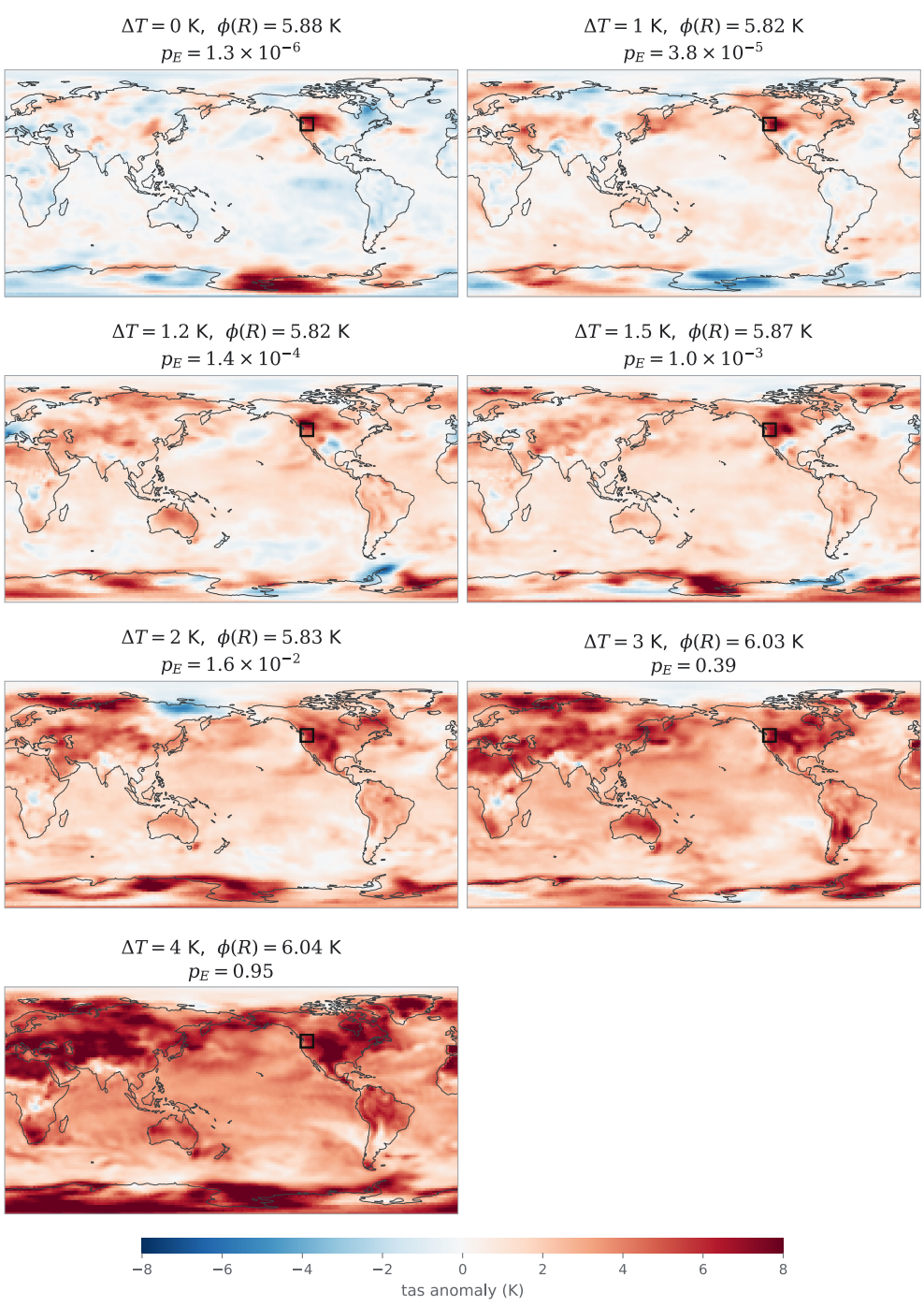


Figure 19: Highest weight samples under the no resampling regime. We guide the sampler to the same $\tau_{\texttt{tas}} \geq 5.81$K for a range of global average temperatures ΔT. See Table 19 for the qualitative analysis.

H.4 Computational cost

We measure the net-speed-up of the path-weighted estimator by comparing the net wall-clock time required for each example in Table 21. We compare our results against the number of N_{MC} samples required to form a Monte Carlo estimate at equal-CV, discounted by the added cost of computing

the guidance. An unguided draw is one denoiser call per step (0.151 s per sample at $K = 200$); the production sampler measures $4.7\times$ that per sample for a univariate or factorised reward (marginal threshold exceedances and multivariate exceedance under uncorrelated reward), and the correlated variant of the reward measures a larger $7.7\times$. In general, any m-coupled reward requires m vector Jacobian product calls per denoising step, which contributes a linearly growing computational cost to the wall-clock time.

Table 21: Net speed-up over crude Monte Carlo at matched CV for $N = 1024$ guided draws against unguided draws. Computed using CVs obtained from Tables 15, 1 and 18.

	Threshold Exceedances		Joint hot & dry	
p^*	hot	dry	factorised reward	Joint reward
Per-sample cost	$4.7\times$	$4.7\times$	$4.7\times$	$7.7\times$
10^{-1}	$1.4\times$	$3.3\times$	$0.8\times$	$0.5\times$
10^{-2}	$0.7\times$	$5.5\times$	$3.6\times$	$6.1\times$
10^{-3}	$149\times$	$36\times$	$19\times$	$82\times$
10^{-4}	$273\times$	$226\times$	$271\times$	$651\times$
10^{-5}	$467\times$	$350\times$	—	—

We find that for most of the generations, the net speedup of the path-weighted sampler out-weighs the added clock time for guided steps. Additionally, at deeper rarities where Monte Carlo has zero samples within the extreme event set of interest (e.g. for very rare events), the method becomes a point of feasibility rather than speed-up alone. These speed-ups scale with the quality of the guidance itself.

H.5 Hyperparameter ablation

We test a range of schedules of $(\lambda(\bar{\sigma}), \delta(\bar{\sigma}))$ on the threshold exceedance problem at a marginal $p^* = 10^{-3}$. We caveat that the parameters we present were chosen heuristically, and we do not independently tune the hyperparameters for any other setting presented in our paper. We also only tuned this on one specific problem, so we believe there are further performance improvements that can be made by better schedule tuning. While we tune only on the hot $p^* = 10^{-3}$ problem, we will also show what the corresponding parameters/ablations affect the 10^{-3} dry problem as well. All ablations are done with a single change to the default parameters described in App. H.1.1. We report $\hat{p}_0[E]/p^*$ against the Monte Carlo anchor, with errors displayed as the standard errors over the eight seeds, which are shared across all arms.

I Metropolis-adjusted Langevin corrections

We use Metropolis-Adjusted Diffusion Model (MADM, (Lam et al., 2026)) for the rejuvenation kernel M_k of Section 3.1. From the predictor-corrector framework of (Song et al., 2021b) the guided score can be used as a predictor and M MADM steps act as our corrector, targeting the tilted marginals π_k of Eq. (5). The unadjusted Langevin algorithm (ULA) update is,

$$\tilde{x} \;=\; x + \tfrac{\rho}{2}\, g(x) + \sqrt{\rho}\, z, \quad z \sim \mathcal{N}(0, I), \quad g(x) := \nabla_x \log \pi_k(x) = s_\theta(x, t_k) + h_{t_k}(x). \tag{114}$$

Here g is the guided score. ULA accepts all proposals $x \to \tilde{x}$ and is biased for every finite step size ρ, and only leaves π_k invariant in the $\rho \to 0$ limit.

A Metropolis-Hastings correction removes this bias while keeping the same proposal: accept $\tilde{x}$ with probability $\mathcal{A}(x, \tilde{x}) = 1 \wedge \big[\Gamma_\pi(x, \tilde{x})\, H(x, \tilde{x})\big]$, otherwise stay at x, where the acceptance ratio splits into a target ratio Γ_π and a proposal ratio H. The proposal ratio is Gaussian and closed-form,

$$\log H(x, \tilde{x}) \;=\; \frac{1}{2\rho}\big\|\tilde{x} - x - \tfrac{\rho}{2} g(x)\big\|^2 \;-\; \frac{1}{2\rho}\big\|x - \tilde{x} - \tfrac{\rho}{2} g(\tilde{x})\big\|^2, \tag{115}$$

while the target ratio is exactly the tilted-density ratio at t_k,

$$\Gamma_\pi(x, \tilde{x}) \;=\; \frac{\pi_k(\tilde{x})}{\pi_k(x)} \;\propto\; \frac{p_{t_k}(\tilde{x})\, e^{r_{t_k}(\tilde{x})}}{p_{t_k}(x)\, e^{r_{t_k}(x)}}. \tag{116}$$

Table 22: One-knob ablation of the guidance schedules at $p^* = 10^{-3}$. We vary a single parameter reference in Table 14; the same 8 seeds are used in every row.

			Hot (tas)		Dry (hurs)	
Knob	Deployed	Ablated	$\hat{p}/p^*$	CV	$\hat{p}/p^*$	CV
Deployed recipe			0.989	0.037	0.977	0.076
Tempering schedule $\lambda(\bar{\sigma})$						
Peak $\lambda_{\max}$	1.5	1.0 (no overshoot)	0.955	0.076	0.960	0.069
Peak $\lambda_{\max}$	1.5	2.0	1.010	0.094	1.018	0.145
Onset $\bar{\sigma}_{\text{on}}$	40	20 (later)	0.972	0.095	0.973	0.103
Onset $\bar{\sigma}_{\text{on}}$	40	80 (earlier)	1.035	0.069	1.043	0.123
Smoothing width $\delta(\bar{\sigma})$						
Transport δ_{hi}	0.5 SD	0.25 SD	1.088	0.209	1.387	0.398
Floor δ_{lo}	0.01 SD	0.05 SD	1.040	0.053	1.043	0.077

Only Γ_π remains intractable for a diffusion model: the reward factor $e^{r_{t_k}}$ is available in closed form, but we do not have direct access to the base marginal p_{t_k}, only its score $s_\theta(x, t_k)$. Writing the target ratio as

$$\log \Gamma_\pi(x, \tilde{x}) \;=\; \underbrace{\log \frac{p_{t_k}(\tilde{x})}{p_{t_k}(x)}}_{\text{intractable}} \;+\; \underbrace{\big(r_{t_k}(\tilde{x}) - r_{t_k}(x)\big)}_{\text{exact}} \tag{117}$$

isolates the one hard piece: the reward term is known in closed form and needs no approximation at all. MADM (Lam et al., 2026) sidesteps the marginal-density problem by computing the remaining p_{t_k}-ratio, under a well trained network where $s_\theta = \nabla_x \log p_{t_k}$, using a line-integral,

$$\log \frac{p_{t_k}(\tilde{x})}{p_{t_k}(x)} \;=\; \int_0^1 \big\langle\, s_\theta\big(x + u(\tilde{x} - x),\, t_k\big),\; \tilde{x} - x \,\big\rangle \, \mathrm{d}u, \tag{118}$$

valid when s_θ is a gradient field, which holds exactly for an analytic score but is not enforced by the denoising objective for a learned network. This is the conservation error of the learned score, which is small in high noise regions, where we apply rejuvenation. The line integral can be approximated by quadrature, for instance by Simpson's rule at the midpoint $\bar{x} = (x + \tilde{x})/2$,

$$\widehat{\log \Gamma_\pi} \;=\; \tfrac{1}{6}\big\langle\, s_\theta(x, t_k) + 4\, s_\theta(\bar{x}, t_k) + s_\theta(\tilde{x}, t_k),\; \tilde{x} - x \,\big\rangle \;+\; \big(r_{t_k}(\tilde{x}) - r_{t_k}(x)\big), \tag{119}$$

giving the practical acceptance probability $\widehat{\mathcal{A}}(x, \tilde{x}) = 1 \wedge \exp\big(\widehat{\log \Gamma_\pi}\big) H(x, \tilde{x})$. The endpoint scores $s_\theta(x, t_k), s_\theta(\tilde{x}, t_k)$ and rewards $r_{t_k}(x), r_{t_k}(\tilde{x})$ are shared with the proposal and its acceptance test, so the quadrature costs additional *unconditional* score evaluations.

The quadrature carries a truncation error of order $\mathcal{O}(\rho^{5/2})$ on the p_{t_k}-ratio, so the adjusted kernel is not exactly π_k-invariant but is accurate to far higher order than the $\mathcal{O}(\rho)$ bias of the unadjusted chain (Lam et al., 2026). An exact alternative replaces the Metropolis rule with Barker acceptance, at the cost of additional score evaluations (Lam et al., 2026); but we use the Simpson-rule variant throughout.

We can summarise a single MADM move under the Simpson's-rule variant of (Lam et al., 2026) as (i) propose a transition with Eq. (114) and (ii) accept the proposal with an acceptance probability approximated with Eqs. (115) and (119). Algorithm 2 states this as the rejuvenation kernel M_k called by Algorithm 1.

Algorithm 2 One MADM rejuvenation move at level k, targeting $\pi_k \propto p_{t_k} e^{r_{t_k}}$.

Require: state x, level k, score $s_\theta(\cdot, t_k)$, reward r_{t_k}, step size ρ.

1: $g(x) \leftarrow s_\theta(x, t_k) + \nabla_x r_{t_k}(x)$ ▷ Guided score

2: $\tilde{x} \leftarrow x + \frac{\rho}{2} g(x) + \sqrt{\rho}\, z, \quad z \sim \mathcal{N}(0, I)$ ▷ Langevin proposal, Eq. (114)

3: $\log \widehat{\mathcal{A}} \leftarrow \min\left\{0,\ \widehat{\log \Gamma_\pi}(x, \tilde{x}) + \log H(x, \tilde{x})\right\}$ ▷ Eqs. (119) and (115)

4: With probability $\widehat{\mathcal{A}}$ set $x \leftarrow \tilde{x}$; otherwise keep x.

5: **return** x

J Related Works Extended

We compare our method for generating the rare event probabilities against the method introduced in (Manshausen et al., 2026). Both methods guide a pre-trained diffusion model towards the extremes, and reweight the resulting samples to estimate $p_0[E]$ under the model density p_θ. The methods differ at the integrator. The method of (Manshausen et al., 2026) samples with an ODE and estimates the rare-event probability using the Probability Flow ODE (PF-ODE) (Song et al., 2021b) for both unguided and guided processes, which allows them to weight each sample by its exact terminal likelihood. We sample with a reverse SDE and weight each trajectory by the Girsanov density of the guided path measure, computed against the unguided process. Neither object is available to the other: a stochastic integrator has no PF-ODE likelihood, and a deterministic flow induces path measures that are mutually singular, so Girsanov's theorem is not available.

Resampling mid-trajectory is only available to the path-weights formulation as a terminal-only weight defines no intermediate targets to resample against. During the denoising process, we measure how far the sampled trajectory $\mathbb{Q}$ deviates from one defined by the tilted process $\mathbb{P}_{r_{0,\delta}}$, and adaptively resample within a Sequential Monte Carlo scheme. Under the operating guidance of (Manshausen et al., 2026), this yields us a far larger fraction of particles from the rare-event set E.

Notation. The *arrow* on a density is the direction in which it is evaluated: backward ($\leftarrow$) means along the sampling run itself (i.e. from a Gaussian prior $\rightarrow$ the data distribution); forward ($\rightarrow$) is the noising direction (i.e. data $\rightarrow$ prior) obtained via integrating the ODE from the generated sample back to noise. K is the number of integrator steps; arrow-free symbols (p_0, q_0) are the continuous-time objects, untouched by the discretisation. The key point is that, at a finite K, the two directions assign *different* values to the same density (dependent on the integrator of choice).

$p_0,\ q_0$	model's true density and its guided (tilted) counterpart; target $p_0[E] = \int_E p_0$
$\overleftarrow{q}$	discretised density of the guided draws $x \sim \overleftarrow{q}$.
$\overrightarrow{q}$	the same guided density evaluated by the forward ODE run - their denominator
$\overleftarrow{p}$	discretised density of the *unguided* process - our estimator targets this quantity
$\overrightarrow{p}$	the unguided density evaluated by a forward ODE - their numerator
$\mathcal{E}_0 = \log(\overrightarrow{p}/\overleftarrow{p})$	*arrow gap* of the unguided density: forward and backward evaluation of the same process.
$\mathcal{E}_\gamma = \log(\overrightarrow{q}/\overleftarrow{q})$	arrow gap of the guided density
$D_{\mathrm{map}} = \log(\overleftarrow{p}/p_0)$	the discrete chain's gap to the model

The three discrepancies $\mathcal{E}_0, \mathcal{E}_\gamma, D_{\mathrm{map}}$ are all $O(1/K)$ discretisation quantities and vanish as $K \to \infty$.

J.1 The PF-ODE Estimator

Both methods target the same quantity, $p_0[E]$ by importance sampling with a guided proposal: draw $x \sim \overleftarrow{q}$, then reweight,

$$p_0[E] \;=\; \int p_0\, \mathbf{1}_E \,\mathrm{d}x \;=\; \mathbb{E}_{x\sim\overleftarrow{q}}\left[\frac{p_0}{\overleftarrow{q}}\,\mathbf{1}_E\right]. \tag{120}$$

The two methods differ only in how this ratio is estimated. We use the Girsanov path ratio $\mathrm{d}\mathbb{P}/\mathrm{d}\mathbb{Q}$, accumulated during the sampling run itself, which equals the backward ratio, $\overleftarrow{p}/\overleftarrow{q}$. The method of (Manshausen et al., 2026) is a ratio of two *forward-evaluated* endpoint densities:

$$\text{ours: } \mathbb{E}_{x\sim\overleftarrow{q}}\left[\frac{\overleftarrow{p}}{\overleftarrow{q}}\,\mathbf{1}_E\right], \qquad \text{(Manshausen et al., 2026)} : \mathbb{E}_{x\sim\overleftarrow{q}}\left[\frac{\overrightarrow{p}}{\overleftarrow{q}}\,\mathbf{1}_E\right]. \tag{121}$$

Both methods approximate p_0 by either $\overleftarrow{p}$ (us) or $\vec{p}$ as this is not readily evaluable for a score-based model. Notably, importance sampling asks us to divide by the density of the draws, i.e. the *backward* object $\overleftarrow{q}$ instead of $\vec{q}$.

The method of (Manshausen et al., 2026) first generates a sample by integrating a deterministic ODE from $t = 1$ down to data at $t = 0$,

$$x_0 \;=\; x_1 \;-\; \int_0^1 \Big(f_t\, x_t \;-\; \tfrac{1}{2}\sigma_t^2 \big(s_\theta(x_t, t) + \gamma\, h_t(x_t) \mathbf{1}\{t \in \Delta\} \big) \Big)\, \mathrm{d}t, \tag{122}$$

where h_t is their classifier guidance term and γ is their guidance strength.

After sampling, samples that belong to the rare event set of interest $x_E \sim \overleftarrow{q} \in E$ are selected and their likelihoods are evaluated: from each sample x_E, they re-integrate the same dynamics in the noising direction $t = 0 \to 1$, accumulating the integral,

$$\log \vec{p}(x_E) \;=\; \log p_1(x_1) \;+\; \int_0^1 \Big(d\, f_t \;-\; \tfrac{1}{2}\sigma_t^2\, \nabla \cdot s_\theta(x_t, t) \Big)\, \mathrm{d}t, \tag{123}$$

where d is the dimension and x_1 the endpoint of the re-integration (noise), $\log \vec{q}$ is obtained similarly via replacing the score in Eq. 123 with its guided variant, $s_\theta \to s_\theta + \gamma h_t(x_t)\mathbf{1}\{t \in \Delta\}$. Their importance weight is given by,

$$\log \frac{p_1(x_1)}{p_1(x_1^\gamma)} - \frac{1}{2} \int_0^1 \sigma_t^2 \Big(\nabla \cdot s_\theta(x_t, t) \;-\; \nabla \cdot s_\theta(x_t^\gamma, t) \Big)\, \mathrm{d}t + \frac{\gamma}{2} \int_\Delta \sigma_t^2\, \nabla \cdot h_t(x_t^\gamma)\, \mathrm{d}t, \tag{124}$$

where x_t and x_t^γ are the unguided and guided re-integration trajectories from the same sample x_E, with endpoints x_1 and x_1^γ. The first integral vanishes if the two trajectories coincide, $x_t = x_t^\gamma$, which is everywhere before the guidance window.

The assumption of (Manshausen et al., 2026). The weight is exact iff (i) $\vec{p} = p_0$, or when the forward density equals the true model density, where $D_{\mathrm{map}} = 0$, and (ii) $\vec{q} = \overleftarrow{q}$, where the forward re-integration of the guided path reproduces the density of the draws, i.e, $\mathcal{E}_\gamma = 0$. Neither holds at finite K for the *explicit* Euler integrator used, as the reverse of this integrator is the *implicit* Euler which is not used. Hence the forward density $\vec{q}$ does not retrace the same trajectory as the backward process $\overleftarrow{q}$.

The bias (the expectation over guided draws in E) of this is given by,

$$\frac{\hat{p}_0[E]}{p_0[E]} \;=\; \mathbb{E}\big[\exp\left(\mathcal{E}_0 - \mathcal{E}_\gamma + D_{\mathrm{map}}\right)\big]\,, \tag{125}$$

which is unbiased iff the right-hand side is one. This happens trivially, when $\mathcal{E}_0 - \mathcal{E}_\gamma = 0$ (i.e. no guidance) *and* $D_{\mathrm{map}} = 0$, i.e. $\overleftarrow{p} = p_0$. The non-trivial solution requires $\overleftarrow{p} = \vec{p} = p_0$ and $\vec{q} = \overleftarrow{q}$, such that $\mathcal{E}_0 = \mathcal{E}_\gamma = 0$ which occurs exactly at $K \to \infty$. In a non-analytical setting, D_{map} is unknown and cannot be evaluated, hence, both methods assume $\overleftarrow{p} = p_0$. At finite K, an accurate estimator therefore requires the three terms to cancel. Within the finite limit, the system becomes unbiased if $\mathcal{E}_\gamma = \mathcal{E}_0 + D_{\mathrm{map}}$ such that individual biases cancel under addition. Our counterpart is a single condition, $\overleftarrow{p} \to p_0$: the law of the K-step chain must approach the model's, or $D_{\mathrm{map}} \to 0$. We too need large K for this to be satisfied. In the absence of p_0 and D_{map}, our condition is trivially satisfied, whereas (Manshausen et al., 2026) requires $\mathcal{E}_0 = \mathcal{E}_\gamma$ or the cancellation of individual biases.

J.1.1 Empirical results

We compare these methods using the 2D GMM threshold exceedance problem at $\tau = 10$. As both methods use first order integrators, we elect to use the exponential integrator for both ODE and SDE integrators with analytical scores and Doob's guidance on the tested equal ℓ discretisation (see App. G.2.2). In the ODE setting of (Manshausen et al., 2026), this is equivalent to the DDIM (Gonzalez et al., 2023; Lu et al., 2022; Song et al., 2021a) integrator, but under the EDM's parametrisation (Karras et al., 2022) this corresponds to the Euler integrator used in (Manshausen et al., 2026). Additionally, we clip the guidance to ensure that the crossing fraction remains 10% for both methods (under no resampling) which mirrors the operational regime of (Manshausen et al., 2026). We calibrate the clipping for both methods at each K separately.

Remark To isolate the effect of discretisation bias, we provide the true $\nabla \cdot s(x_t, t)$ in Eq. (124) for the PF-ODE computation of (Manshausen et al., 2026) instead of approximating it via the Hutchinson trace estimator. However, in practice the trace estimator contributes variance to the estimate, shrinking with the number of probes in exchange for a larger computational overhead.

Table 23: Signed percentage bias for the rare-event probability estimated against the analytical ground truth for the 2D GMM threshold exceedance problem at $\tau = 10$. Positive % reflects an overestimate. We guide with the optimal r_k^* (Doob's guidance) as well as a clipped reward that yields 10% particles in E, the regime employed in (Manshausen et al., 2026). Positive % reflects an overestimate; we have a lower discretisation bias at all K.

guidance	K	$\vec{p}/\vec{q}$ ((Manshausen et al., 2026)) ↓	ours ↓
exact Doob	50	$+155.7\%_{\pm 0.3}$	$\mathbf{-2.2}\%_{\pm 0.7}$
exact Doob	100	$+65.3\%_{\pm 0.2}$	$\mathbf{-2.6}\%_{\pm 0.4}$
exact Doob	200	$+31.7\%_{\pm 0.2}$	$\mathbf{-1.9}\%_{\pm 0.3}$
exact Doob	500	$+14.7\%_{\pm 0.1}$	$\mathbf{-1.0}\%_{\pm 0.2}$
clipped Doob, ∼10% cross	50	$+42.8\%_{\pm 4.2}$	$\mathbf{+2.7}\%_{\pm 2.5}$
clipped Doob, ∼10% cross	100	$+21.2\%_{\pm 3.4}$	$\mathbf{-2.1}\%_{\pm 1.9}$
clipped Doob, ∼10% cross	200	$+10.7\%_{\pm 3.0}$	$\mathbf{-1.6}\%_{\pm 1.8}$
clipped Doob, ∼10% cross	500	$+4.4\%_{\pm 2.8}$	$\mathbf{-1.9}\%_{\pm 2.2}$

Under the exact Doob's guidance, the discretisation bias from $\vec{p}/\vec{q}$ survives, as the guidance is employed throughout the trajectory. We report the discretisation bias against K in Table 23. Exact guidance highlights that this is a discretisation bias itself which converges as $\vec{p} \to \overleftarrow{p}$ and $D_{\text{map}} = 0$. For our method, any error lives in D_{map}. The discretisation bias of (Manshausen et al., 2026) is dependent on a specific choice of γ and h_t, reflected in $\mathcal{E}_\gamma$. Hence the two proposals in Table 23 give a different estimate for $p_0[E]$. Additionally, as we adaptively resample, our crossing fraction remains high even with the clipped guidance at $86.5\%/80.4\%/81.3\%/79.4\%$ (averaged across seeds) for $K = 50/100/200/500$ respectively compared to theirs at 10%, which results in a higher particle population concentration on the event of interest. We need not discard 90% of samples in this regime, and this is reflected in a lower seed-level standard error of our method.

Table 24: Network evaluations per particle. For a K step denoising process, with R rejuvenation steps and M MCMC moves of Algorithm 1. P denotes the number of Hutchinson's probes in Eq. (124).

	score function	r	∇r	JVP
ours (Alg. 1)	$K{+}2RM$	$K{+}RM$	$K{+}RM$	–
odds ratio (Eq. 124)	$3K$	$2K$	$2K$	$2PK$

Remark on computational costs. Let N be the number of particles, R the number of resampling steps, M the number of MCMC move of Algorithm 1, and P the number of Hutchinson trace estimator probes used per step in Eq. (124). Both samplers pay a different number of function evaluations per particle, and have a different compute budget. Table 24 summarises this. K is fixed and set by the discretisation bias tolerance versus compute. Which has the same weak-order convergence for both methods (first-order) (Gonzalez et al., 2023; Manshausen et al., 2026).

As an illustrative example, take the $\tau = 10$ example of Section 4, coupled with the guidance term of Eq. (18) and the ΠGDM Σ_t proxy (Song et al., 2023). Averaged over 24 seeds, resampling is triggered $R = 3.5$ times, followed by $M = 3$ MADM rejuvenation steps. If we set $P = 1$, which is a conservative estimate as typically a larger P results in a lower variance estimate of the Jacobian trace. If we were to count every single score evaluation, reward and reward-gradient evaluation as one call, our estimator uses $2K + 3RM \approx 2032$ function calls per particle, of which rejuvenation moves account for 32, whereas the odds-ratio estimator with a single probe uses $5K + 2PK = 7000$. As a reward and its gradient calls are often paired together, at an equal step count Manshausen et al. (2026) receive $N/3.5$ particles at the same compute budget, which is the comparison drawn in Fig 2.

J.1.2 Detailed Comparison

Below we collate the tables used to generate Fig. 2. In the first two panels, we use a learned score network, wheras the final uses an analytical score with Doob's transition kernels. We motivate why this is selected in Section 4.1. All hyperparameters are kept the same unless they are ablated. The only exception is that each arm uses the best $\lambda(t)$ tempering parameter as selected in App. G.5.

CV vs N. We keep the same $K = 1000$ and ablate at $\tau = 10$ with learned scores. Table 25 summarises these results, where we find that the CV of DireSMC (w/ rejuvenation) outperforms Manshausen et al. (2026) at all K.

Table 25: CV of $\hat{p}$ against particle count N (Fig. 2, left), at $\tau = 10$, $K = 1000$, with the learned score, over 24 seeds. Measured against model's tail mass p^* (Table 7).

		DireSMC		
N	Manshausen et al. (2026)	base	+ resampling	+ MADM
100	0.631	0.999	0.611	**0.321**
250	0.385	0.506	0.394	**0.184**
500	0.259	0.349	0.303	**0.134**
750	0.206	0.266	0.168	**0.082**
1000	0.167	0.213	0.166	**0.079**
2000	0.140	0.184	0.147	**0.074**
4000	0.094	0.161	0.076	**0.057**

CV vs Rarity. We keep the same $K = 1000$ and $N = 1000$ and ablate the rarities using approxiamte guidance and learned scores. We discount the number of particles available based on number of function evaluations compared to DireSMC base. For Manshausen et al. (2026) this is $3.5\times$ less particles (286) on this problem. Table 26 summarises these results, where we find that at matched compute all variants of DireSMC outperform Manshausen et al. (2026).

Table 26: CV of $\hat{p}$ relative to models threshold exceedance probability $p^\star$ (Table 7). $\hat{p}$ across rarities at equal compute (Fig. 2, middle), with $K = 1000$, the learned score and 24 seeds. We use $N = 1000$ and Manshausen et al. (2026) use $n = N/3.5 = 286$ (Table 24). All variants of DireSMC outperform Manshausen et al. (2026).

			DireSMC		
τ	$p^\star$	Manshausen et al. (2026)	base	+ resampling	+ MADM
7	7.13×10^{-3}	0.129	0.070	0.093	**0.060**
8	1.38×10^{-3}	0.167	0.092	0.093	**0.064**
9	1.87×10^{-4}	0.221	0.127	0.159	**0.090**
10	1.72×10^{-5}	0.349	0.213	0.166	**0.079**

Bias vs K. We measure the discretisation bias using analytical scores, Doob's transition kernels of DireSMC with no resampling, selected to isolate the errors introduced in Eq. 124. Table 27 summarises these results.

Table 27: Discretisation bias against the step count K (Fig. 2, right), shown as the mean $\hat{p}/p^\star \pm$ s.e. over 24 seeds. Computed using analytical socres, Doob's transition kernels and withut resampling at $N = 1000$. $p^\star$ is $p_{\text{data}}[E]$. Manshausen et al. (2026) are given the exact divergence.

τ	$p^\star$	K	Manshausen et al. (2026)	DireSMC
7	6.88×10^{-3}	50	1.466 ± 0.001	$\mathbf{0.950} \pm 0.006$
		100	1.233 ± 0.001	$\mathbf{0.972} \pm 0.003$
		200	1.128 ± 0.001	$\mathbf{0.981} \pm 0.003$
		500	1.069 ± 0.001	$\mathbf{0.990} \pm 0.002$
8	1.28×10^{-3}	50	1.701 ± 0.002	$\mathbf{0.957} \pm 0.005$
		100	1.336 ± 0.001	$\mathbf{0.970} \pm 0.003$
		200	1.178 ± 0.001	$\mathbf{0.980} \pm 0.003$
		500	1.091 ± 0.001	$\mathbf{0.990} \pm 0.002$
9	1.62×10^{-4}	50	2.049 ± 0.002	$\mathbf{0.963} \pm 0.006$
		100	1.473 ± 0.002	$\mathbf{0.972} \pm 0.004$
		200	1.240 ± 0.001	$\mathbf{0.981} \pm 0.003$
		500	1.117 ± 0.001	$\mathbf{0.990} \pm 0.002$
10	1.40×10^{-5}	50	2.557 ± 0.003	$\mathbf{0.978} \pm 0.007$
		100	1.653 ± 0.002	$\mathbf{0.974} \pm 0.004$
		200	1.317 ± 0.002	$\mathbf{0.981} \pm 0.003$
		500	1.147 ± 0.001	$\mathbf{0.990} \pm 0.002$