CONSTRAINED DIFFUSION FROM BINARY EVALUATIONS VIA LEARNED DOOB h-FUNCTIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

Conditional generation under constraints that can only be checked on a finished sample, such as a molecular property filter, remains challenging for diffusion models: most guidance methods require differentiable measurement operators, paired conditional data, or rollouts of the base model at every step. We present a framework grounded in the Doob h-transform that reduces conditional generation to learning a single function, the probability that a noisy sample denoises into the constraint set. We learn this function from binary clean-sample evaluations alone, by squared-error regression on forward-noised labels; the base model is frozen, one lightweight head is trained per constraint, and noisy-state classifier guidance arises as a special case. The exact guidance step fades where the constraint signal provably vanishes and carries a factor of the inverse probability across samples, which normalized guidance discards: with the schedule and mean step size held fixed, equalizing the step costs 17.0 to 18.5 points of constraint satisfaction on CIFAR-10 at comparable FID, across independently trained estimators. Guidance is more than $50\times$ cheaper per satisfying sample than rejection sampling at a 1.7% base rate, raises a Lipinski-type property filter on QM9 from 54.3% to 96.8%, and lifts satisfaction on Stable Diffusion 1.5 from 7.2% to 60.0% at one sixteenth of the cost of derivative-free selection, which reaches 9.0%. These results demonstrate that binary evaluations of clean samples are sufficient supervision for constrained generation across pixel, latent, and molecular diffusion.

1 INTRODUCTION

Many constraints of practical interest can be verified by evaluating a program on a completed sample but cannot be expressed as differentiable functions of it: whether a molecule satisfies a property filter, whether a graph contains a required substructure, or whether an image satisfies a pixel-level predicate. Any thresholded score defines such a constraint, and filters of this kind are routinely used to screen generated samples. Each is a binary test on a clean sample that provides neither a gradient nor a conditioning signal to a pretrained model; a classifier over noisy states, where required, must first be trained from its labels. For a constraint set C, the posterior score decomposes as

$$\nabla_x \log p_t(x_t \mid C) = \nabla_x \log p_t(x_t) + \nabla_x \log h_t(x_t), \qquad h_t(x) = \Pr(X_0 \in C \mid X_t = x), \quad (1)$$

so that conditioning a frozen model reduces to estimating a single scalar function: the probability that a noisy state denoises into C.

A classifier over noisy states trained on such labels (Dhariwal & Nichol, 2021) estimates h directly, and Uehara et al. (2024) interpret classifier guidance as a Doob h-transform with h learned by regression on forward-noised labelled data. Other guidance methods approximate this decomposition using information that is unavailable in the binary setting: conditioning labels during training (Ho & Salimans, 2022), known measurement operators (Chung et al., 2023; Song et al., 2023), paired data for a learned generalized h-transform (Denker et al., 2024), text guidance or reward models for training-free h-transform editing (Nguyen et al., 2025), or differentiable evaluators (Bansal et al., 2024; Yu et al., 2023). Estimators of h itself differ along two axes. Simulation-based estimators fit h to endpoints of the model's own reverse process (Li et al., 2025; Guo et al., 2026) or estimate its gradient by Monte Carlo lookahead during sampling (Zhu et al., 2026); they require base-model evaluations or reward calls and target the model distribution rather than the data distribution. Related gradient-guidance methods use learned return or property predictors in trajectory, graph, and

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.

molecular diffusion (Janner et al., 2022; Vignac et al., 2023; Lee et al., 2023). More directly, Go et al. (2026) regress clean-sample reward-model scores onto forward-noised latents, while the guarantees of Chang et al. (2026) require the weight to be bounded away from zero, which excludes hard indicators.

We study this setting for hard indicator constraints, and our central observation concerns the size of the guidance step. Several guidance methods treat it as a free parameter and rescale the gradient to a fixed radius or norm (Yang et al., 2024; Ma et al., 2024). The Doob h-transform (Doob, 1957) does not: its drift $\nabla \log h_t = \nabla h_t / h_t$ shrinks as the constraint signal contracts under forward noising and, at a given noise level, carries a factor $1/h_t$ across samples. We characterize h as the unique bounded solution of the backward Kolmogorov Cauchy problem of the reverse-time diffusion among those attaining $\mathbf{1}_C$ along the forward process; with the exact score and h, the corresponding h-transform yields the constrained marginals when initialized from the conditional terminal law, and Appendix A bounds the practical initialization approximation. We prove that the variance of the constraint signal vanishes at high noise, so that little is left there for any estimator to act on, and we learn h by a regression whose training pairs are obtained by forward-noising labelled data in closed form; the verifier labels training data and is never called during sampling. A noisy-state classifier targets the same h and, for a binary reward, value-based decoding targets its model-law counterpart (Section 5), so the analysis applies to them as well. The novelty is therefore not that a noisy-state classifier can estimate h: it is the hard-indicator characterization, the distinction between data- and model-law estimands, and a controlled measurement of what the unnormalized $\nabla h/h$ allocation contributes when the mean guidance budget is held fixed.

This magnitude matters. With the schedule and the mean step size held fixed, equalizing the step across samples, which is what normalization does at each noise level, reduces satisfaction on CIFAR-10 by 17.0 to 18.5 points across three independently trained classifiers, at comparable constraint-conditional FID (FID against real images that satisfy the constraint), and reweighting the normalized step by $1/\hat{h}$ recovers part of the loss. With a 1000-step sampler the learned estimator has nearly identical calibration error on the model's own trajectories and on forward-noised data, and relative to rejection sampling with the verifier, guidance reduces the sampling cost per satisfying sample by factors of 3.0 and 4.2 at base rates of 30.8% and 22.5%, and by 53 to 55 at 1.7%.

Our contributions are as follows. (1) A controlled measurement showing that the magnitude of the guidance step carries information: on the evaluated predicates the unnormalized drift's step is strongly negatively correlated with $\hat{h}$, and with the schedule and mean step size held fixed, equalizing it costs 17.0 to 18.5 points of satisfaction at comparable FID across independently trained classifiers, which a $1/\hat{h}$ reweighting partially restores (Section 6.2). (2) The theory that explains it: h_t is the unique bounded solution of the reverse-time backward Kolmogorov problem attaining $\mathbf{1}_C$ along the forward process, the variance of its signal is non-increasing in the noise level and vanishes as $\bar{\alpha}_t \to 0$ without spectral-gap assumptions, a noisy-state classifier estimates the same h, and it differs from the target of rollout-based methods by an amount set by the score error of the base model (Sections 3 and 5). (3) A guidance method that needs only binary evaluations of clean samples: one lightweight estimator per constraint, trained by squared-error regression without base-model evaluations, whose log-gradient is positively aligned with a rollout reference (mean cosine 0.47 to 0.54 from $t = 400$; Tables 4 and 6), evaluated across pixel, latent and molecular diffusion, including composition (Sections 4 and 6).

2 Background and Problem Setting

Diffusion models. The forward SDE $\mathrm{d}X_t = f(X_t, t)\,\mathrm{d}t + g(t)\,\mathrm{d}W_t$ transforms data $X_0 \sim p_0$ into noise as $t \to T$, and the reverse-time dynamics $\mathrm{d}X_t = [f(X_t, t) - g(t)^2 \nabla_x \log p_t(X_t)]\,\mathrm{d}t + g(t)\,\mathrm{d}\bar{W}_t$ use the score $\nabla_x \log p_t$, approximated by $s_\theta(x, t)$ (Anderson, 1982; Song et al., 2021b). In the DDPM discretization (Ho et al., 2020), $\nabla_x \log p_t(x) \approx -\epsilon_\theta(x, t)/\sqrt{1 - \bar{\alpha}_t}$ with $\bar{\alpha}_t = \prod_{s=1}^{t}(1 - \beta_s)$.

Reverse-time generator. For a functional ϕ of clean data, the posterior expectation $u(x, t) = \mathbb{E}[\phi(X_0) \mid X_t = x]$ is governed by the generator of the reverse-time diffusion,

$$\widehat{\mathcal{L}}_t u = \tfrac{1}{2} g(t)^2 \Delta u + \big(g(t)^2 \nabla_x \log p_t - f\big) \cdot \nabla_x u. \tag{2}$$

Unlike the forward generator, which propagates expectations of *future* states, $\widehat{\mathcal{L}}_t$ propagates expectations of the *initial* state, and depends on the data through the score.

Doob h-transform. For $h_t(x) = \Pr(X_0 \in C \mid X_t = x)$, Bayes' rule gives $p_t(x \mid C) = p_t(x)\, h_t(x) / \Pr(X_0 \in C)$, so the conditional score is the unconditional score plus $\nabla_x \log h_t$ (Doob, 1957). Adding this drift reweights reverse trajectories toward C and is exact: started from $p_T(\cdot \mid X_0 \in C)$, the resulting process has marginals $p_t(\cdot \mid X_0 \in C)$.

Problem setting. A constraint is given only through its indicator $\mathbf{1}_C : \mathcal{X} \to \{0, 1\}$, evaluable on clean samples. The learner may evaluate $\mathbf{1}_C(x_0)$ on clean data and noise those samples forward through the known kernel. It has no differentiable relaxation of $\mathbf{1}_C$, no evaluation of it on noisy states x_t, $t > 0$, no conditioning signal in the frozen base model, and does not roll the base model out to obtain labels. The goal is to sample from $p_0(\cdot \mid X_0 \in C)$ using the frozen score model and a learned estimate of h. Proofs of all results are in Appendix A.

3 Theory

The results below rest on standard tools (time reversal, Itô's formula, the Doob transform); what is specific here is the discontinuous initial condition $\mathbf{1}_C$, for which h vanishes on part of the space, and each result makes a prediction that Section 6 tests.

Proposition 1 (Regression target)**.** *For each t, the unique minimizer over measurable h of $\mathbb{E}\left[(h(X_t) - \mathbf{1}_C(X_0))^2\right]$, with (X_0, X_t) drawn from the data and the forward kernel, is $h_t(x) = \Pr(X_0 \in C \mid X_t = x)$.*

Theorem 1 (Characterization)**.** *Under standard regularity conditions (Appendix A), for any measurable C with $0 < \Pr(X_0 \in C) < 1$ the function $h_t(x) = \Pr(X_0 \in C \mid X_t = x)$ is the unique bounded classical solution of the Cauchy problem*

$$\partial_t h = \widehat{\mathcal{L}}_t h, \qquad h_0 = \mathbf{1}_C, \tag{3}$$

with $\widehat{\mathcal{L}}_t$ as in equation 2 and the initial condition attained along the forward process, $h_\varepsilon(X_\varepsilon) \to \mathbf{1}_C(X_0)$ as $\varepsilon \downarrow 0$. Started from $p_T(\cdot \mid X_0 \in C)$, the reverse-time process with drift augmented by $g(t)^2 \nabla_x \log h_t$ has marginals $p_t(\cdot \mid X_0 \in C)$ at every t.

Proposition 1 is the standard characterization of the conditional mean, and it is why binary clean labels suffice: $\mathbf{1}_C(X_0)$ is an unbiased sample of $h_t(X_t)$ at every noise level. Theorem 1 identifies the target of every guidance method in this setting as the unique bounded solution, in the sense made precise in Appendix A.2, of a parabolic problem whose initial condition is the constraint indicator.

Proposition 2 (Signal contraction)**.** *Let $\bar{h} = \Pr(X_0 \in C)$ and $\sigma^2(t) = \mathrm{Var}[h_t(X_t)]$. Then (i) $\mathbb{E}[h_t(X_t)] = \bar{h}$ for all t, and $\sigma^2(t)$ is non-increasing in t; (ii) $\sigma^2(t) \le \sqrt{I(X_0; X_t)/2}$, where I is mutual information in nats; and (iii) for the variance-preserving process on $\mathbb{R}^d$ with $m_2 = \mathbb{E}\|X_0\|^2 < \infty$, $I(X_0; X_t) \le \frac{d}{2} \log\!\left(1 + \frac{\bar{\alpha}_t\, m_2}{(1-\bar{\alpha}_t)\, d}\right)$, so $\sigma^2(t) \to 0$ as $\bar{\alpha}_t \to 0$.*

Proposition 2 makes high-noise guidance failure a property of the forward process rather than of any estimator: as t grows, h_t flattens toward $\bar{h}$ and carries progressively less information about C. On MNIST (LeCun et al., 1998) digit-3 the measured $\hat{\sigma}^2(t)$ falls by $492\times$ between $t = 100$ and $t = 900$ (Figure 3a, Appendix C.5), and the mean guidance norm falls from 2.32 to 0.225 over the same range (Appendix B.3); Figure 1 shows the same contraction for a learned h_θ on CIFAR-10.

From the regression target to the guidance field. Proposition 1 controls h_θ in squared error, while the sampler uses $\nabla \log h_\theta$. Under the variance-preserving kernel the ideal field has the closed form (Appendix A.5)

$$\nabla_{x_t} \log h_t(x_t) = \frac{\sqrt{\bar{\alpha}_t}}{1 - \bar{\alpha}_t} \Big(\mathbb{E}[X_0 \mid X_t = x_t,\, X_0 \in C] - \mathbb{E}[X_0 \mid X_t = x_t] \Big), \tag{4}$$

which can be estimated by rolling the frozen model out from x_t. Tables 4 and 6 (Appendix C.1 and C.2) compare the learned field with this reference on MNIST digit 3: the mean cosine is 0.47 to 0.54 at every measured noise level from $t = 400$, where most states have enough satisfying

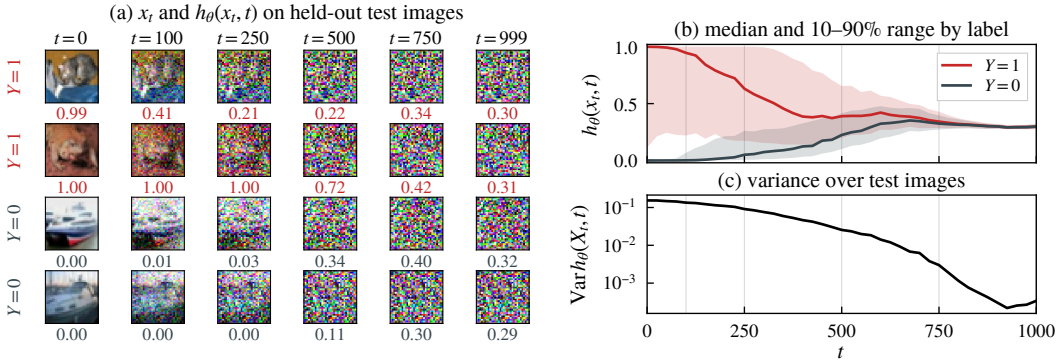

Figure 1: **The learned h_θ on held-out, forward-noised CIFAR-10 images** (red-pixel, $Y = \mathbf{1}_C(x_0)$). (a) Two test images with $Y = 1$ and two with $Y = 0$ along one forward path, with $h_\theta(x_t, t)$ under each. (b) Over 4,000 test images the label groups separate at small t and merge as t grows. (c) The variance of the learned $h_\theta(X_t, t)$ falls from 0.15 at $t = 0$ to 3×10^{-4} at $t = 900$, as Proposition 2 predicts for h_t.

rollouts to estimate the reference (Appendix C.2), against a chance level near zero. The logarithm's sensitivity to small h has little effect in practice: sweeping δ from 10^{-10} to 10^{-2} leaves satisfaction at $94.8 \pm 0.5\%$, and the fraction of held-out states with $h_\theta < 0.01$ falls from 89% at $t = 50$ to 0% from $t = 700$ (Appendix B.3).

4 Method

Objective. We parameterize $h_\theta(x, t) \in (0, 1)$ by a small network with a sigmoid output and minimize $\mathcal{L} = \mathcal{L}_{\text{bdry}} + \mathcal{L}_{\text{sup}} + \lambda_{\text{fc}}\mathcal{L}_{\text{fc}}$ with $\lambda_{\text{fc}} = 0.1$:

$$\mathcal{L}_{\text{sup}} = \mathbb{E}_{x_0,t,\epsilon}\big[(h_\theta(x_t, t) - \mathbf{1}_C(x_0))^2\big], \qquad \mathcal{L}_{\text{bdry}} = \mathbb{E}_{x_0}\big[(h_\theta(x_0, 0) - \mathbf{1}_C(x_0))^2\big], \tag{5}$$

$$\mathcal{L}_{\text{fc}} = \mathbb{E}_{x_0,t}\big[(h_\theta(x_t, t) - \mathrm{sg}[h_\theta(x'_{t+1}, t+1)])^2\big], \tag{6}$$

where x_t and x'_{t+1} are independent forward noisings of the same x_0 and sg is stop-gradient. $\mathcal{L}_{\text{sup}}$ alone identifies h (Proposition 1) and suffices in practice (Appendix C.7); it is the core of the method. The other two are optional regularizers, used for every head reported in Section 6: $\mathcal{L}_{\text{bdry}}$ places weight on the discontinuous initial condition, which uniform timestep sampling rarely visits, and $\mathcal{L}_{\text{fc}}$ smooths the estimate across noise levels (Appendix B.1). Every training pair is produced in closed form from labelled data, so training requires no base-model evaluations. Learning the guidance field directly would amount to training a score model on the constraint-satisfying samples alone, a regression in the data dimension from the positive fraction of the data (6.1% on Stable Diffusion); the scalar head is trained on every labelled sample, positive and negative, and its gradient is a conservative field by construction.

Sampling. In the DDPM parameterization we replace the noise prediction by

$$\tilde{\epsilon}(x_t, t) = \epsilon_\theta(x_t, t) - \gamma\sqrt{1 - \bar{\alpha}_t}\,\nabla_{x_t} \log\big(h_\theta(x_t, t) + \delta\big), \tag{7}$$

with guidance scale γ and $\delta = 10^{-8}$ unless stated. At $\gamma = 1$ and $\delta = 0$, with an exact score and an exact h, this is the h-transform of Theorem 1 (Appendix A.6); as is standard for guidance scales (Dhariwal & Nichol, 2021), we use $\gamma > 1$, and δ keeps $\log h$ finite where $h \approx 0$. Each step costs one forward and one backward pass of h_θ; there is no candidate set and no rollout. Algorithm 1 (Appendix C.1) gives both procedures.

Composition. For $C_1 \cap C_2$ with separately trained heads, we guide with $w_1 \nabla \log h^{(1)} + w_2 \nabla \log h^{(2)}$. With $w_1 = w_2 = 1$ this is the exact joint correction when $\mathbf{1}_{C_1}(X_0)$ and $\mathbf{1}_{C_2}(X_0)$ are conditionally independent given X_t at every t (Appendix A.6); otherwise the weights trade off the two constraints.

5 DATA AND MODEL ESTIMANDS OF h

With a binary terminal reward, h is the value $\mathbb{E}[\mathbf{1}_C(X_0) \mid X_t]$ targeted by value-learning methods (Li et al., 2025; Yang & Klein, 2021; Lawson et al., 2022; Uehara et al., 2024), and our $\mathcal{L}_{\text{sup}}$ shares its gradient in expectation with consistency value regression. The difference lies in which conditional law the expectation is taken under. Rollout-based estimators (Li et al., 2025; Guo et al., 2026) regress onto endpoints of the *model's* reverse process started at x, and so estimate $h_t^{\text{model}}(x) = \Pr_{\text{model}}(X_0 \in C \mid X_t = x)$. Forward noising of labelled data, as here and in Chang et al. (2026), estimates $h_t^{\text{data}} = h_t$.

Proposition 3 (Estimand gap)**.** *Under the assumptions of Appendix A.4, h^{model} solves equation 3 with $\nabla_x \log p_t$ replaced by the model score s_θ in $\widehat{\mathcal{L}}_t$. Consequently $h^{\text{model}} = h^{\text{data}}$ for every C if the model's reverse process reproduces the data posterior, and in general the two differ by an amount governed by the score error $s_\theta - \nabla_x \log p_t$ rather than by either estimator.*

With an exact score, guiding with h^{data} samples the data distribution restricted to C, the reference of constraint-conditional FID. With the model's score only h^{model} gives an exact h-transform of the process being sampled; Section 6.3 compares the estimator's calibration under the two laws for a real model. The two targets also differ in cost: forward noising gives every labelled sample an independent training pair at every noise level with no base-model evaluations, whereas the SVDD-MC targets of Section 6.5 consumed 5,000,000 base-model forward passes for 5,000 independent endpoint outcomes. Equal gradients in expectation do not imply equal estimator variance at finite sample size.

Estimators of h therefore fall into a two-by-two grid: rollout targets with soft rewards (Li et al., 2025), rollout targets with hard constraints (Guo et al., 2026), forward-noising targets with weights bounded below (Chang et al., 2026), and forward-noising targets for hard indicators, the cell of noisy-state classifier guidance, whose target this work characterizes and studies where h approaches zero (Table 5, Appendix C.1). The analysis concerns h^{data} itself, so it applies to any estimator of it, including a classifier trained by cross-entropy on the same labels, whose loss has the same minimizer: $\nabla \log h = \nabla h / h$ shrinks at high noise (by equation 4, $\|\nabla \log h_t\| \le \sqrt{\bar\alpha_t}\, D/(1-\bar\alpha_t)$ for data of diameter D) and carries a factor $1/h$ at each noise level, whereas normalized guidance applies a step of fixed magnitude to every sample at every step. Section 6.2 measures both effects.

6 EXPERIMENTS

We evaluate in three settings where binary constraints arise: image predicates that no pretrained model conditions on, where we compare against a matched classifier and against rejection sampling and measure how the drift allocates the step (Sections 6.1–6.5); property filters in molecular design, including composition of independently trained constraints (Section 6.6); and text-to-image generation with Stable Diffusion 1.5 (Section 6.7).

Protocol. The base model is frozen throughout. Satisfaction is the fraction of samples that pass the predicate used for labelling (for the MNIST digits and the CIFAR-10 classes, a separately trained classifier judges the samples), with the standard deviation over sampling seeds for a single trained head (Wilson 95% intervals (Wilson, 1927) where stated); FID (Heusel et al., 2017) uses real constraint-satisfying images (constraint-conditional) and a general reference set (full-distribution). Classifier-guidance baselines use a classifier over noisy states trained on the identical labels, optimizer and budget, at matched capacity, and guide with the standard rule $\epsilon \leftarrow \epsilon - \gamma\sqrt{1-\bar\alpha_t}\, g$, $g = \nabla_{x_t} \log p_\phi(y{=}1 \mid x_t)$; the normalized variant divides g by $\|g\|$ per sample. DPS requires a differentiable measurement operator, which a binary evaluation does not provide. Appendix B gives every setting.

6.1 MATCHED-CAPACITY COMPARISON WITH CLASSIFIER GUIDANCE

We define four predicates on CIFAR-10 (Krizhevsky, 2009) images as deterministic functions of pixel statistics, with positive rates of 24.5–26.8% on the training set: red-pixel fraction, Sobel edge density, upper-half luminance, and low saturation (Appendix B.4). None exists in the conditioning space of the base model, the DDPM of Ho et al. (2020). Our h_θ has 675,969 parameters and the

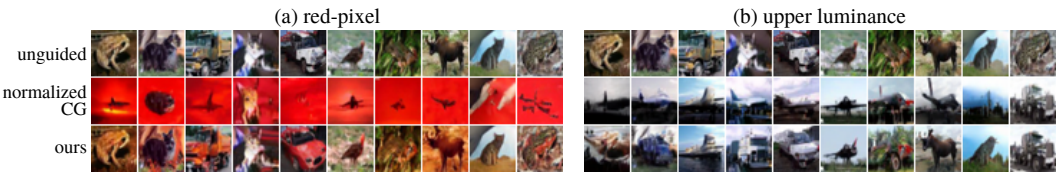

Figure 2: **Paired CIFAR-10 samples at $\gamma = 5$.** Each column starts from the same x_T and uses the same sampling noise in every row. The figure shows the first ten samples generated without selection; across the full 64-sample batch, every guided sample in both rows satisfies the constraint. The middle row, the matched classifier under the normalized rule, achieves this by collapsing to a few image types: a red field with a small object in (a), dark-ground aircraft scenes in (b). Guidance with h_θ keeps the content of the unguided sample in (a) and produces varied scenes in (b).

Table 1: Matched classifier under the standard and normalized rules, and ours, on four CIFAR-10 predicates (matched in capacity, labels, optimizer, budget and inference cost). Satisfaction (%) and FID against constraint-satisfying test images (FID_c); FID_f in Tables 7 and 8. Mean over 3 seeds $\times$ 1,000 samples. Unguided satisfaction in parentheses.

		Satisfaction			FID_c		
Predicate	γ	CG	CG-norm	Ours	CG	CG-norm	Ours
Red-pixel (31.3)	5	100.0	100.0	100.0	62.8	187.0	58.2
	15	100.0	100.0	100.0	98.2	212.3	93.1
Upper luminance (23.1)	5	100.0	100.0	100.0	43.4	81.4	43.9
	15	100.0	100.0	100.0	50.1	87.7	52.7
Edge density (16.5)	5	99.7	100.0	100.0	52.5	96.9	59.3
	15	100.0	100.0	100.0	72.7	109.5	78.8
Low saturation (16.8)	5	99.5	100.0	95.9	48.6	101.6	45.9
	15	100.0	100.0	99.9	71.2	116.8	73.2

classifier 676,098, and a guided sample costs 0.404 s with either. Table 1 compares our head with the matched classifier under both rules. With the standard unnormalized drift, the two estimators reach at least 95.9% satisfaction from $\gamma = 5$ and constraint-conditional FID within 6.8 of each other, as Section 5 predicts for two estimates of the same h. Normalizing the classifier's drift keeps satisfaction at 100% from $\gamma = 5$ but gives 1.6 to 3.2$\times$ the constraint-conditional FID of our drift at $\gamma = 5$ (paired samples in Figure 2); unlike the unnormalized drift, whose magnitude decays as the signal contracts, the normalized step is largest at $t = 999$ (Section 6.2).

6.2 How the unnormalized drift allocates the step

By equation 4 and $\nabla \log h = \nabla h / h$, the unnormalized log-probability drift has two properties that a normalized step removes: its size varies over the noise schedule, and at each step it varies across samples, with a factor $1/h$. We measure both using the matched classifier's estimate of this drift, so that the result concerns the sampling rule rather than our estimator, on red-pixel and upper luminance at $\gamma = 2$ (Appendix C.4). Over the schedule, the mean step per unit γ, $\bar{m}(t)$, the mean of $\sqrt{1-\bar{\alpha}_t}\,\|g\|$ over samples, peaks near $t = 500$ on red-pixel and $t = 700$ on upper luminance and is 4.4 and 3.7$\times$ smaller at $t = 999$, whereas the normalized step $\sqrt{1-\bar{\alpha}_t}$ is largest at $t = 999$. Across samples, step size and $\hat{h}(x_t)$ have Spearman correlation between -0.60 and -0.998 at $t \in \{799, 599, 399, 199\}$ on both predicates (250 trajectories; $p < 10^{-24}$; Table 11).

To isolate the allocation across samples, we keep the normalized direction and give every sample the step $\bar{m}(t)$ measured along the standard-rule trajectories, so that the schedule and the mean step are those of the standard rule (Table 2). Satisfaction falls by 18.4, 18.5 and 17.7 points on red-pixel and by 17.2, 17.3 and 17.0 on upper luminance for three independently trained classifiers, with FID_c within 1.2 of the standard rule in every case: the loss is a property of the rule rather than of one classifier. Weighting each sample's step by $(1/\hat{h}_i)/\text{mean}_j(1/\hat{h}_j)$, which keeps the mean step,

Table 2: The step at a fixed schedule and mean step size, CIFAR-10, $\gamma = 2$, for independently trained classifiers (same recipe; training seeds 0, 1 and 2). All rules use the classifier's gradient direction; "equal" and "$1/\hat{h}$" give each sample a step whose mean over samples is that classifier's standard-rule $\bar{m}(t)$. Classifier 1 is the matched classifier of Section 6.1: mean over sampling seeds 0 and 1 ($1/\hat{h}$ upper luminance: seed 0); classifiers 2 and 3: sampling seed 0; 1,000 samples per run.

		Satisfaction (%)			FID_c		
Predicate	Classifier	Standard	Equal	$1/\hat{h}$	Standard	Equal	$1/\hat{h}$
Red-pixel	1	99.0	80.6	89.0	52.60	51.85	53.06
	2	98.2	79.7	86.2	53.25	53.04	54.11
	3	98.5	80.8	88.0	53.21	53.34	53.75
Upper luminance	1	99.7	82.5	94.3	39.61	40.73	39.34
	2	98.5	81.2	91.6	39.49	40.29	39.76
	3	99.7	82.7	94.5	39.55	40.32	40.13

recovers 35 to 46% and 60 to 69% of the loss, again at comparable FID_c; the rest is carried by the variation of $\|\nabla\hat{h}\|$ across samples.

6.3 Calibration under the data and model laws

We compare the calibration of the learned head under the two laws of Proposition 3 for a real model. Along 2,000 unguided trajectories we record $h_\theta(x_t, t)$ at ten noise levels and score it against each trajectory's outcome $\mathbf{1}_C(x_0)$, which measures its calibration to h^{model}; on forward-noised test images at the same t, the same score measures calibration to h^{data}. With the 1000-step sampler the head has nearly identical calibration error under the two laws: mean expected calibration error 0.051 and 0.051 on red-pixel, and 0.021 and 0.024 on upper luminance. Sampling with 100 steps moves the model's red-pixel rate from 31.5% to 46.0% and raises the model-law error to 0.155, while the data-law error is 0.049; upper luminance, whose rate stays at 21.8%, keeps 0.023 against 0.024. The difference in calibration follows the shift of the model's law on the constraint, and it does not prevent guidance: with 100 steps, $\gamma = 2$ reaches 99.5% on both predicates (Appendix C.4).

6.4 Guidance and rejection sampling

Rejection sampling with the verifier draws exactly from the model's constrained law, at $1/p$ samples per accepted one for a pass rate p. On red-pixel and upper luminance ($p = 30.8\%$ and 22.5% over 4,000 and 4,500 unguided samples), the first 1,000 accepted samples have FID_c 48.4 and 42.4 (Appendix C.4); our guidance at $\gamma = 2$ reaches 97.2% and 99.5% at FID_c 48.8 and 39.2 (Table 7), and since a guided step costs about 6% more than an unguided one on the same GPU, it needs 3.0 and 4.2× less sampling time per satisfying sample. For a rare constraint we raise the red-pixel threshold to the 99th percentile of the training statistic (1.00% positive; Appendix C.4). The model passes it 1.7% of the time (Wilson interval 1.2–2.4%, $n = 2{,}000$), so rejection draws about 59 samples per accepted one; guidance at $\gamma = 2$ reaches 95.5 to 99.0% over two training seeds of the head (the matched classifier: 95.4 to 96.0%), 53 to 55× less time per satisfying sample (38 to 77× over the Wilson interval). Bounds on the guidance error grow with the inverse probability of the condition (Sahu et al., 2026); at 1% both estimators still exceed 95%.

6.5 Supervision and inference cost

Against SVDD (Li et al., 2025) on MNIST digit-3, with value networks matched at 122,657 parameters, our training targets need no base-model evaluations, where the rollout targets of SVDD-MC consume 5,000,000 base-model forward passes (340 s against 625 s of training). Our sampler runs at 0.050 s/sample with 95.1% satisfaction, against 0.294 and 1.048 s/sample for SVDD-PM with $M = 5$ and $M = 20$ candidates drawn in proportion to value (92.5% and 95.2%); keeping the best candidate reaches 100.0% at 5.7× our cost (Appendix C.6).

Table 3: QM9, 3 seeds $\times$ 1,000 molecules. Satisfaction, uniqueness and novelty (%). The composed rows guide with $\log h_{\text{Lip}} + \log h_{\text{ring}}$ from separately trained heads; their satisfaction is the joint rate. All rows use the same Euler–Maruyama sampler. SVDD-PM: 1 seed $\times$ 500 molecules (Appendix B.6).

Constraint	Method	γ / M	Sat.	Uniq.	Novel
Lipinski-type	Unguided	0	54.3	98.0	98.3
	Ours	2	88.7	92.4	96.2
	Ours	10	96.8	83.2	94.5
	Ours	50	99.5	63.2	92.8
	SVDD-PM	$M = 5$	96.0	97.8	97.1
Lipinski $\wedge$ ring	Unguided	0	39.6	98.0	98.3
	Composed	5	72.4	92.3	96.0
	Composed	20	78.3	90.8	95.9

6.6 Molecular property constraints

On QM9 (Ramakrishnan et al., 2014) with the GDSS base model (Jo et al., 2022), we use RDKit[1] predicates on decoded molecules: ring presence, and a Lipinski-type filter (Lipinski et al., 1997) (molecular weight below 125, at most one hydrogen-bond donor, at most two acceptors, thresholds scaled to QM9's size range and giving a 35.3% positive rate). Guidance is added to the atom-feature score of GDSS; validity is at least 99.9% in every run (Appendix C.10). The unguided baseline and all guided runs use the same Euler–Maruyama discretization of the GDSS reverse SDE (Appendix B.6). Table 3 supports three observations. First, on the Lipinski-type filter our method raises satisfaction from 54.3% to 96.8% at $\gamma = 10$ while retaining 83.2% uniqueness, with a satisfaction–diversity trade-off controlled by γ. Second, SVDD-PM reaches 96.0% at 7.4$\times$ the cost of GDSS's predictor–corrector sampler measured on the same GPU (8.24 against 1.12 s/mol), since every step decodes and scores $M = 5$ candidates, whereas our sampler adds one evaluation of the head and its gradient per step. Third, composition works without joint training: guiding with two independently trained heads raises the joint rate from 39.6% to 78.3%, and at $\gamma = 20$ it matches the product of the guided marginals (96.3% $\times$ 81.6% = 78.6%; Table 23), consistent with the two properties being nearly independent under the model (unguided: 39.6% joint, 41.6% product).

6.7 Scale: Stable Diffusion 1.5

On Stable Diffusion 1.5 (Rombach et al., 2022) with 50-step DDIM (Song et al., 2021a) and classifier-free guidance scale 7.5, we train h_θ on a fixed 30,000-image COCO (Lin et al., 2014) subset labelled by a grayscale predicate (6.1% positive) and guide in latent space. Guidance scales are larger than in pixel space because classifier-free guidance inflates the noise prediction. Satisfaction rises from 7.2% to 60.0%, monotonically in γ (Tables 19 and 20; samples in Figure 5, appendix), with constraint-conditional FID improving from 109.9 to 98.2. Derivative-free candidate selection reaches 9.0% at 16$\times$ our sampling cost. The normalized rule is more sensitive to the scale: the matched classifier reaches 65.1% at $\gamma = 10$, its smallest tested scale, and declines to 24.7% at $\gamma = 75$, while our drift rises monotonically (Appendix C.9). On a separate set of 200 prompts (one seed, 500 images; Table 21), the classifier under the standard rule reaches 28.8, 46.6 and 54.4% at $\gamma = 10$, 35 and 75, against 30.0, 48.8 and 58.0% for our head, while the normalized rule falls from 61.8% at $\gamma = 10$ to 25.4% at $\gamma = 75$.

7 Related work

Guidance with gradients or labels. Classifier guidance (Dhariwal & Nichol, 2021) and classifier-free guidance (Ho & Salimans, 2022) require noisy-state classifiers or conditional training; DPS and ΠGDM (Chung et al., 2023; Song et al., 2023) require a known measurement operator, and Universal

[1] https://www.rdkit.org

Guidance and FreeDoM (Bansal et al., 2024; Yu et al., 2023) differentiable evaluators. The decomposition equation 1 underlies conditional sampling in Song et al. (2021b), and Didi et al. (2023) cast conditional generation, including motif scaffolding, as a Doob h-transform; DEFT (Denker et al., 2024) learns a generalized h-transform from paired data, h-Edit (Nguyen et al., 2025) applies the h-transform to editing, and ensemble Kalman guidance (Zheng et al., 2025) is derivative-free for black-box inverse problems but propagates an ensemble of particles.

Analyses of guidance. Guidance does not in general sample the intended tilted distribution (Chidambaram et al., 2024), and restricting it to a noise interval improves quality (Kynkäänniemi et al., 2024); training-free methods often rescale the guidance gradient, for instance to a fixed radius (Yang et al., 2024; Ma et al., 2024), the choice whose cost Section 6.2 measures. Recent theory characterizes when linear score combinations realize projective composition and when they fail (Bradley et al., 2025).

Value functions and twisted sampling. SVDD (Li et al., 2025) guides diffusion by soft value functions with candidate selection at each step; FUDGE (Yang & Klein, 2021), SIXO (Lawson et al., 2022), controlled decoding (Mudgal et al., 2024), twisted SMC, including for indicator targets (Zhao et al., 2024), and its amortized form in discrete diffusion (Kim et al., 2026) learn values or twists from base-model samples or SMC approximations of the target, and Value Matching (Jensen et al., 2026) learns a value for reward tilting from on-policy rollouts. These methods differ in how the value or twist is estimated and in whether inference uses candidate selection, particles, or token-level reweighting.

Learning h directly. Guo et al. (2026) learn h for hard constraints with a martingale loss on trajectories of the pretrained model and give total-variation and Wasserstein guarantees; Chang et al. (2026) learn h by gradient-penalized regression on forward-noised samples with Wasserstein guarantees for weights bounded below; Go et al. (2026) regress neural reward-model scores on forward-noised latents and guide with the value gradient; Uehara et al. (2024) also describe guidance with the gradient of a value learned by Monte Carlo regression on samples of the pretrained model; and Zhu et al. (2026) estimate the Doob correction for an event by Monte Carlo lookahead without training. Section 5 and Table 5 position this work.

8 Discussion and conclusion

Guidance from binary labels is a Doob h-transform, and its effectiveness depends less on the estimator of h than on the drift constructed from it: $\nabla h/h$ acts at noise levels where the signal has not contracted, its step is negatively correlated with $\hat{h}$ across samples, and at a matched budget this allocation accounts for 17 to 18 points of satisfaction on CIFAR-10 that equal per-sample steps forgo, for each of the three classifiers we trained. In the regimes we evaluate, guidance is substantially cheaper per satisfying sample than rejection sampling, with the advantage increasing for the rare constraint we study. One estimator per constraint, trained once from binary labels and combined by adding log-gradients, guides a frozen model at a modest overhead over unguided sampling (5% on CIFAR-10, 27% on Stable Diffusion) without invoking the verifier during sampling. In our experiments, rescaling the guidance gradient to a fixed size discards the information carried by the step size, and reweighting a normalized step by $1/\hat{h}$ restores part of it. Natural extensions are costly verifiers, such as docking, and rarer constraints.

AI use statement

Generative AI tools were used to aid and polish the writing and to help draft sections of the paper from the authors' notes, results and instructions. The research idea and the method originate with the authors, who ran all experiments and produced all reported results. All AI-assisted text was reviewed and edited by the authors. Reported numerical results were checked against the corresponding experimental records, and references and citation-supported claims were manually reviewed against cited public records. The authors take full responsibility for the content of this work, including all text, claims, and results.

Ethics statement

This work uses public datasets (MNIST, CIFAR-10, COCO and QM9) and publicly released pre-trained models, and involves no human subjects. The method steers a frozen generator toward outputs that pass a user-supplied check. Like other guidance methods it could be pointed at checks that encode harmful properties, and it inherits the biases of the base model and of the labelled data used to train each head.

Reproducibility statement

Appendix A proves every result of Sections 3 and 5 under the assumption stated there. Appendix B specifies each base model, head architecture, optimizer, training budget, sampler, predicate with its thresholds, FID reference set and timing protocol, and Appendix C reports the full sweeps behind every table of the main text, with seeds and sample counts in each caption. Code and trained heads will be released upon publication.

References

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

Arpit Bansal, Hong-Min Chu, Avi Schwarzschild, Roni Sengupta, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Universal guidance for diffusion models. In *International Conference on Learning Representations*, 2024.

Arwen Bradley, Preetum Nakkiran, David Berthelot, James Thornton, and Joshua M. Susskind. Mechanisms of projective composition of diffusion models. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 5319–5354, 2025.

Jinyuan Chang, Chenguang Duan, Yuling Jiao, Yi Xu, and Jerry Zhijian Yang. Inference-time alignment for diffusion models via variationally stable Doob's matching. *arXiv preprint arXiv:2601.06514*, 2026.

Muthu Chidambaram, Khashayar Gatmiry, Sitan Chen, Holden Lee, and Jianfeng Lu. What does guidance do? A fine-grained analysis in a simple setting. In *Advances in Neural Information Processing Systems*, volume 37, pp. 84968–85005, 2024.

Hyungjin Chung, Jeongsol Kim, Michael T. McCann, Marc L. Klasky, and Jong Chul Ye. Diffusion posterior sampling for general noisy inverse problems. In *International Conference on Learning Representations*, 2023.

Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. John Wiley & Sons, second edition, 2006.

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

Prafulla Dhariwal and Alexander Nichol. Diffusion models beat GANs on image synthesis. In *Advances in Neural Information Processing Systems*, volume 34, pp. 8780–8794, 2021.

Kieran Didi, Francisco Vargas, Simon Mathis, Vincent Dutordoir, Emile Mathieu, Urszula Julia Komorowska, and Pietro Lio. A framework for conditional diffusion modelling with applications in motif scaffolding for protein design. arXiv preprint arXiv:2312.09236, 2023.

J. L. Doob. Conditional Brownian motion and the boundary limits of harmonic functions. *Bulletin de la Société Mathématique de France*, 85:431–458, 1957.

Bradley Efron. Tweedie's formula and selection bias. *Journal of the American Statistical Association*, 106(496):1602–1614, 2011.

Hyojun Go, Hyungjin Chung, Prune Truong, Goutam Bhat, Li Mi, Zhaochong An, Zixiang Zhao, Dominik Narnhofer, Serge Belongie, Federico Tombari, and Konrad Schindler. Stitched value model for diffusion alignment. *arXiv preprint arXiv:2605.19804*, 2026.

Zhengyi Guo, Wenpin Tang, and Renyuan Xu. Conditional diffusion guidance under hard constraint: A stochastic analysis approach. *arXiv preprint arXiv:2602.05533*, 2026.

U. G. Haussmann and E. Pardoux. Time reversal of diffusions. *The Annals of Probability*, 14(4): 1188–1205, 1986.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs trained by a two time-scale update rule converge to a local Nash equilibrium. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. arXiv preprint arXiv:2207.12598, 2022. A short version appeared at the NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications.

Jonathan Ho, Ajay N. Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 33, pp. 6840–6851, 2020.

Michael Janner, Yilun Du, Joshua B. Tenenbaum, and Sergey Levine. Planning with diffusion for flexible behavior synthesis. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 9902–9915, 2022.

Cristian Perez Jensen, Luca Schaufelberger, Riccardo De Santi, Kjell Jorner, and Andreas Krause. Value matching: Scalable and gradient-free reward-guided flow adaptation. In *International Conference on Learning Representations*, 2026.

Jaehyeong Jo, Seul Lee, and Sung Ju Hwang. Score-based generative modeling of graphs via the system of stochastic differential equations. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 10362–10383, 2022.

Jaihoon Kim, Taehoon Yoon, Prin Phunyaphibarn, Seungjun Kim, Morteza Mardani, and Minhyuk Sung. Contrastive distribution matching for amortized sequential Monte Carlo in discrete diffusion. *arXiv preprint arXiv:2605.23346*, 2026.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

Tuomas Kynkäänniemi, Miika Aittala, Tero Karras, Samuli Laine, Timo Aila, and Jaakko Lehtinen. Applying guidance in a limited interval improves sample and distribution quality in diffusion models. In *Advances in Neural Information Processing Systems*, volume 37, pp. 122458–122483, 2024.

Dieterich Lawson, Allan Raventós, Andrew Warrington, and Scott Linderman. SIXO: Smoothing inference with twisted objectives. In *Advances in Neural Information Processing Systems*, volume 35, pp. 38844–38858, 2022.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Seul Lee, Jaehyeong Jo, and Sung Ju Hwang. Exploring chemical space with score-based out-of-distribution generation. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 18872–18892, 2023.

Xiner Li, Yulai Zhao, Chenyu Wang, Gabriele Scalia, Gokcen Eraslan, Surag Nair, Tommaso Biancalani, Shuiwang Ji, Aviv Regev, Sergey Levine, and Masatoshi Uehara. Derivative-free guidance in continuous and discrete diffusion models with soft value-based decoding. In *Advances in Neural Information Processing Systems*, volume 38, pp. 105884–105922, 2025.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common objects in context. In *Computer Vision – ECCV 2014*, volume 8693 of *Lecture Notes in Computer Science*, pp. 740–755. Springer, 2014.

Christopher A. Lipinski, Franco Lombardo, Beryl W. Dominy, and Paul J. Feeney. Experimental and computational approaches to estimate solubility and permeability in drug discovery and development settings. *Advanced Drug Delivery Reviews*, 23(1–3):3–25, 1997.

Jiajun Ma, Tianyang Hu, Wenjia Wang, and Jiacheng Sun. Elucidating the design space of classifier-guided diffusion generation. In *International Conference on Learning Representations*, 2024.

Sidharth Mudgal, Jong Lee, Harish Ganapathy, Yaguang Li, Tao Wang, Yanping Huang, Zhifeng Chen, Heng-Tze Cheng, Michael Collins, Trevor Strohman, Jilin Chen, Alex Beutel, and Ahmad Beirami. Controlled decoding from language models. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 36486–36503, 2024.

Toan Nguyen, Kien Do, Duc Kieu, and Thin Nguyen. h-Edit: Effective and flexible diffusion-based editing via Doob's h-transform. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 28490–28501, 2025.

Anton Obukhov, Maximilian Seitzer, Po-Wei Wu, Semen Zhydenko, Jonathan Kyl, and Elvis Yu-Jing Lin. High-fidelity performance metrics for generative models in PyTorch. https://github.com/toshas/torch-fidelity, 2020.

Bernt Øksendal. *Stochastic Differential Equations: An Introduction with Applications*. Universitext. Springer, sixth edition, 2003.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763, 2021.

Raghunathan Ramakrishnan, Pavlo O. Dral, Matthias Rupp, and O. Anatole von Lilienfeld. Quantum chemistry structures and properties of 134 kilo molecules. *Scientific Data*, 1:140022, 2014.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10684–10695, 2022.

Sharan Sahu, Arisina Banerjee, and Yuchen Wu. Provably reliable classifier guidance via cross-entropy control. arXiv preprint arXiv:2601.21200, 2026.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021a.

Jiaming Song, Arash Vahdat, Morteza Mardani, and Jan Kautz. Pseudoinverse-guided diffusion models for inverse problems. In *International Conference on Learning Representations*, 2023.

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.

Masatoshi Uehara, Yulai Zhao, Tommaso Biancalani, and Sergey Levine. Understanding reinforcement learning-based fine-tuning of diffusion models: A tutorial and review. arXiv preprint arXiv:2407.13734, 2024.

Clément Vignac, Igor Krawczuk, Antoine Siraudin, Bohan Wang, Volkan Cevher, and Pascal Frossard. DiGress: Discrete denoising diffusion for graph generation. In *International Conference on Learning Representations*, 2023.

Edwin B. Wilson. Probable inference, the law of succession, and statistical inference. *Journal of the American Statistical Association*, 22(158):209–212, 1927.

Kevin Yang and Dan Klein. FUDGE: Controlled text generation with future discriminators. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 3511–3535, 2021.

Lingxiao Yang, Shutong Ding, Yifan Cai, Jingyi Yu, Jingya Wang, and Ye Shi. Guidance with spherical Gaussian constraint for conditional diffusion. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 56071–56095, 2024.

Jiwen Yu, Yinhuai Wang, Chen Zhao, Bernard Ghanem, and Jian Zhang. FreeDoM: Training-free energy-guided conditional diffusion model. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 23174–23184, 2023.

Stephen Zhao, Rob Brekelmans, Alireza Makhzani, and Roger Baker Grosse. Probabilistic inference in language models via twisted sequential Monte Carlo. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 60704–60748, 2024.

Hongkai Zheng, Wenda Chu, Austin Wang, Nikola B. Kovachki, Ricardo Baptista, and Yisong Yue. Ensemble Kalman diffusion guidance: A derivative-free method for inverse problems. *Transactions on Machine Learning Research*, 2025.

Qijie Zhu, Zeqi Ye, Han Liu, Zhaoran Wang, and Minshuo Chen. Training-free adaptation of diffusion models via Doob's h-transform. In *International Conference on Machine Learning*, 2026.

A Proofs

Setting and standing assumption. Throughout, $(X_t)_{t\in[0,T]}$ solves $\mathrm{d}X_t = f(X_t,t)\,\mathrm{d}t + g(t)\,\mathrm{d}W_t$ with $X_0 \sim p_0$ independent of W, the set C is measurable with $\bar{h} = \Pr(X_0 \in C) \in (0,1)$, and $Y = \mathbf{1}_C(X_0)$. We write $q_t(\cdot \mid x_0)$ for the transition density of X_t given $X_0 = x_0$, p_t for the marginal density, $\mathcal{G}_t = \sigma(X_r : r \geq t)$, and

$$m_t(x) = \int_C q_t(x \mid x_0)\, p_0(\mathrm{d}x_0), \qquad h_t(x) = \frac{m_t(x)}{p_t(x)},$$

so that $h_t(x) = \Pr(X_0 \in C \mid X_t = x)$ by Bayes' rule, and $m_t/\bar{h} = p_t(\cdot \mid X_0 \in C)$. The regularity conditions of Theorem 1 are the following.

(A) The forward SDE has a unique strong solution, $f(\cdot,t)$ is C^1 and $g > 0$. For $t \in (0,T]$, p_t and m_t are positive, of class $C^{1,2}$ in (t,x), and solve the forward equation $\partial_t \rho = \mathcal{F}_t \rho := -\nabla\cdot(f\rho) + \frac{1}{2}g^2\Delta\rho$ (for m_t this follows from the equation for $q_t(\cdot \mid x_0)$ by integrating over $x_0 \in C$). For $\rho = p$ and for $\rho = m/\bar{h}$, the reverse-time SDE

$$\mathrm{d}\bar{X}_s = \big[-f(\bar{X}_s, T-s) + g(T-s)^2\,\nabla \log \rho_{T-s}(\bar{X}_s)\big]\,\mathrm{d}s + g(T-s)\,\mathrm{d}\bar{W}_s, \qquad s \in [0,T), \tag{8}$$

is well posed from every initial law, and its solution started from ρ_T has the law of the time reversal of the forward process started from ρ_0 (Anderson, 1982; Haussmann & Pardoux, 1986).

For the variance-preserving and variance-exploding processes used in this paper the forward kernels are Gaussian, so p_t and m_t are smooth and strictly positive for $t > 0$; (A) is the usual assumption behind the reverse-time formulation of score-based models (Song et al., 2021b).

A.1 Proof of Proposition 1

Proof. Fix t and let h be measurable with $\mathbb{E}[h(X_t)^2] < \infty$. Expanding the square around $h_t(X_t)$,

$$\begin{aligned}\mathbb{E}\big[(h(X_t) - Y)^2\big] &= \mathbb{E}\big[(h(X_t) - h_t(X_t))^2\big] + \mathbb{E}\big[(h_t(X_t) - Y)^2\big] \\ &\quad + 2\,\mathbb{E}\big[(h(X_t) - h_t(X_t))\,\mathbb{E}[h_t(X_t) - Y \mid X_t]\big].\end{aligned}$$

The last term vanishes because $\mathbb{E}[Y \mid X_t] = h_t(X_t)$, and the second does not depend on h. The objective is therefore minimized exactly by the functions that agree with h_t p_t-almost everywhere, and by h_t alone among continuous functions, since $p_t > 0$. $\mathcal{L}_{\sup}$ averages this objective over noise levels drawn uniformly from the time grid, so its minimizer is h_t at every level. Only pairs (X_0, X_t) enter, and the forward kernel generates them in closed form. □

A.2 Proof of Theorem 1

The initial condition $h_0 = \mathbf{1}_C$ is discontinuous, so we state in which sense it is attained: along the process. Theorem 1 is the combination of the four parts below.

Theorem 1 (precise form). *Assume (A). Then (a) $h \in C^{1,2}((0,T] \times \mathbb{R}^d)$, $0 \le h \le 1$, and $\partial_t h = \widehat{\mathcal{L}}_t h$ on $(0,T] \times \mathbb{R}^d$; (b) $h_t(X_t) \to \mathbf{1}_C(X_0)$ almost surely and in L^2 as $t \downarrow 0$; (c) if $u \in C^{1,2}((0,T] \times \mathbb{R}^d)$ is bounded, satisfies $\partial_t u = \widehat{\mathcal{L}}_t u$, and $u(\varepsilon, X_\varepsilon) \to \mathbf{1}_C(X_0)$ in L^1 as $\varepsilon \downarrow 0$, then $u = h$; (d) the reverse-time dynamics with drift augmented by $g(t)^2 \nabla \log h_t$, started from $p_T(\cdot \mid X_0 \in C)$, have marginal $p_t(\cdot \mid X_0 \in C)$ at every $t \in (0,T]$, and their limit as $t \downarrow 0$ exists almost surely and has law $p_0(\cdot \mid X_0 \in C)$.*

Proof. (a) The equation. By (A), $h = m/p$ is of class $C^{1,2}$ with values in $[0,1]$. For $C^{1,2}$ functions the product rule gives

$$\mathcal{F}_t(hp) = h\,\mathcal{F}_t p - p\,f \cdot \nabla h + g^2\,\nabla p \cdot \nabla h + \tfrac{1}{2} g^2 p\,\Delta h.$$

Since $\partial_t m = \mathcal{F}_t m$ and $\partial_t p = \mathcal{F}_t p$,

$$\partial_t h = \frac{\partial_t m - h\,\partial_t p}{p} = \frac{\mathcal{F}_t(hp) - h\,\mathcal{F}_t p}{p} = \tfrac{1}{2} g^2 \Delta h + \left(g^2 \nabla \log p_t - f\right) \cdot \nabla h = \widehat{\mathcal{L}}_t h.$$

The same computation with $\int \phi(x_0)\, q_t(\cdot \mid x_0)\, p_0(\mathrm{d}x_0)$ in place of m_t shows that $u(x,t) = \mathbb{E}[\phi(X_0) \mid X_t = x]$ solves $\partial_t u = \widehat{\mathcal{L}}_t u$ for bounded ϕ with the corresponding regularity, as stated in Section 2.

(b) The initial condition. Because (X_t) is Markov, X_0 is conditionally independent of $(X_r)_{r>t}$ given X_t, so $h_t(X_t) = \mathbb{E}[Y \mid X_t] = \mathbb{E}[Y \mid \mathcal{G}_t]$. As $t \downarrow 0$ the σ-fields $\mathcal{G}_t$ increase to $\mathcal{G}_{0+} = \bigvee_{t>0} \mathcal{G}_t$, which contains $\sigma(X_0)$ because $X_0 = \lim_{t \downarrow 0} X_t$ by continuity of paths. By (a) and continuity of paths, $s \mapsto h_{T-s}(X_{T-s}) = \mathbb{E}[Y \mid \mathcal{G}_{T-s}]$ is a continuous bounded martingale for the increasing filtration $(\mathcal{G}_{T-s})_{s \in [0,T)}$, so it converges almost surely as $s \uparrow T$. Along any sequence $t_n \downarrow 0$, Lévy's upward theorem identifies the limit as $\mathbb{E}[Y \mid \mathcal{G}_{0+}] = Y$. Hence $h_t(X_t) \to Y$ almost surely, and in L^2 by bounded convergence.

(c) Uniqueness. Let $\bar{X}_s = X_{T-s}$ and $v(s,x) = u(T-s,x)$ for $s \in [0,T)$. By (A), $\bar{X}$ solves equation 8 with $\rho = p$, whose generator at time s is $\widehat{\mathcal{L}}_{T-s}$, and $\partial_s v = -\widehat{\mathcal{L}}_{T-s} v$. By Itô's formula (Øksendal, 2003), $v(s, \bar{X}_s)$ is a local martingale on $[0, T-\varepsilon]$. It is bounded, hence a martingale, also with respect to the filtration $\sigma(\bar{X}_r : r \le s) = \mathcal{G}_{T-s}$ generated by $\bar{X}$; this property depends only on the law of $\bar{X}$, so it holds for the time reversal of X. Therefore $u(t, X_t) = \mathbb{E}[u(\varepsilon, X_\varepsilon) \mid \mathcal{G}_t]$ for $0 < \varepsilon < t$. Letting $\varepsilon \downarrow 0$ and using that conditional expectation is a contraction in L^1, $u(t, X_t) = \mathbb{E}[Y \mid \mathcal{G}_t] = h_t(X_t)$ almost surely. Hence $u(t,\cdot) = h_t$ p_t-almost everywhere, and everywhere because both are continuous and $p_t > 0$.

(d) Marginals. Conditionally on $X_0 \in C$, the process solves the same forward SDE (the Brownian motion is independent of X_0) from the initial law $p_0(\cdot \mid X_0 \in C) = \mathbf{1}_C p_0/\bar{h}$, so its marginals are $m_t/\bar{h} = p_t\, h_t/\bar{h} = p_t(\cdot \mid X_0 \in C)$. By (A), its time reversal solves equation 8 with $\rho = m/\bar{h}$, whose drift contains $g^2 \nabla \log(m_t/\bar{h}) = g^2 \nabla \log p_t + g^2 \nabla \log h_t$: these are the reverse-time dynamics with drift augmented by $g(t)^2 \nabla \log h_t$. Started from $p_T(\cdot \mid X_0 \in C)$, they have marginal $p_t(\cdot \mid X_0 \in C)$ at every $t \in (0,T]$. The time reversal of the conditioned forward process extends continuously to $s = T$ with value X_0; since the solution has the same law on $C([0,T), \mathbb{R}^d)$, its limit as $s \uparrow T$ exists almost surely and has law $p_0(\cdot \mid X_0 \in C)$. □

Initialization. Samplers start the reverse dynamics from p_T, or its Gaussian approximation, rather than from $p_T(\cdot \mid X_0 \in C)$. Since $p_T(\cdot \mid X_0 \in C)/p_T = h_T/\bar{h}$, the bound $\mathbb{E}|h_T(X_T) - \bar{h}| \le$

$\sqrt{I(X_0;X_T)/2}$ from the proof of Proposition 2(ii) gives

$$\mathrm{TV}\big(p_T(\cdot \mid X_0 \in C),\, p_T\big) = \frac{\mathbb{E}\,|h_T(X_T) - \bar{h}|}{2\bar{h}} \le \frac{\sqrt{I(X_0;X_T)/2}}{2\bar{h}},$$

and the guided reverse dynamics, being a Markov kernel, cannot increase this distance; starting from $\mathcal{N}(0,I)$ instead of p_T adds $\mathrm{TV}(p_T,\mathcal{N}(0,I))$ by the triangle inequality. The error due to initialization therefore vanishes as the forward process forgets X_0.

A.3 Proof of Proposition 2

Proof. (i) $\mathbb{E}[h_t(X_t)] = \mathbb{E}\big[\mathbb{E}[Y \mid X_t]\big] = \mathbb{E}[Y] = \bar{h}$. For $0 \le s < t$, $X_0 \to X_s \to X_t$ is a Markov chain, so $\mathbb{E}[Y \mid X_s, X_t] = \mathbb{E}[Y \mid X_s] = h_s(X_s)$ and, by the tower property, $h_t(X_t) = \mathbb{E}[h_s(X_s) \mid X_t]$. The law of total variance gives $\sigma^2(s) = \sigma^2(t) + \mathbb{E}\big[\mathrm{Var}(h_s(X_s) \mid X_t)\big] \ge \sigma^2(t)$.

(ii) Since $h_t(X_t)$ and $\bar{h}$ lie in $[0,1]$, $\sigma^2(t) \le \mathbb{E}|h_t(X_t) - \bar{h}|$. The joint law of (Y, X_t) has density $h_t p_t$ at $Y=1$ and $(1-h_t)p_t$ at $Y=0$, and the product of its marginals has densities $\bar{h}\,p_t$ and $(1-\bar{h})\,p_t$. Hence

$$\mathrm{TV}\big(P_{Y,X_t},\, P_Y \otimes P_{X_t}\big) = \tfrac{1}{2}\int \big(|h_t - \bar{h}| + |(1-h_t) - (1-\bar{h})|\big)\, p_t = \mathbb{E}\,|h_t(X_t) - \bar{h}|.$$

Pinsker's inequality bounds this by $\sqrt{\mathrm{KL}(P_{Y,X_t} \,\|\, P_Y \otimes P_{X_t})/2} = \sqrt{I(Y;X_t)/2}$, and the data-processing inequality for $Y \to X_0 \to X_t$ gives $I(Y;X_t) \le I(X_0;X_t)$ (Cover & Thomas, 2006).

(iii) For the variance-preserving process, $X_t = \sqrt{\bar{\alpha}_t}\,X_0 + \sqrt{1-\bar{\alpha}_t}\,\varepsilon$ with $\varepsilon \sim \mathcal{N}(0, I_d)$ independent of X_0. Writing H for differential entropy, $I(X_0;X_t) = H(X_t) - H(X_t \mid X_0)$ with $H(X_t \mid X_0) = \frac{d}{2}\log\big(2\pi e(1-\bar{\alpha}_t)\big)$. Among densities on $\mathbb{R}^d$ with second moment $s_t = \mathbb{E}\|X_t\|^2 = \bar{\alpha}_t m_2 + (1-\bar{\alpha}_t)d$, the isotropic Gaussian has the largest entropy (Cover & Thomas, 2006), so $H(X_t) \le \frac{d}{2}\log(2\pi e\, s_t/d)$. Subtracting,

$$I(X_0;X_t) \le \frac{d}{2}\log\frac{\bar{\alpha}_t m_2 + (1-\bar{\alpha}_t)d}{(1-\bar{\alpha}_t)d} = \frac{d}{2}\log\Big(1 + \frac{\bar{\alpha}_t\, m_2}{(1-\bar{\alpha}_t)\,d}\Big),$$

which tends to 0 as $\bar{\alpha}_t \to 0$; by (ii), so does $\sigma^2(t)$. □

A.4 Proof of Proposition 3

Let $\hat{X}$ be the model's reverse-time process, equation 8 with $\nabla \log \rho_{T-s}$ replaced by $s_\theta(\cdot, T-s)$, and write $\hat{X}_{(t)}$ for its state at diffusion time t. A rollout-based estimator regresses $\mathbf{1}_C(\hat{X}_{(0)})$ on the state $\hat{X}_{(t)} = x$ from which the rollout starts, so it targets $h_t^{\mathrm{model}}(x) = \Pr(\hat{X}_{(0)} \in C \mid \hat{X}_{(t)} = x)$. We assume that this SDE is well posed on $[0,T]$ and that $h^{\mathrm{model}} \in C^{1,2}((0,T]\times\mathbb{R}^d)$; for the size of the gap we also assume the integrability condition stated below and that both h^{data} and h^{model} approach $\mathbf{1}_C$ along the model's paths.

Proof. Equation. By the Markov property, $h_t^{\mathrm{model}}(x) = \mathbb{E}\big[h_\varepsilon^{\mathrm{model}}(\hat{X}_{(\varepsilon)}) \mid \hat{X}_{(t)} = x\big]$ for $0 < \varepsilon < t$, so h^{model} evaluated along $\hat{X}$ is a martingale and, by Itô's formula, solves the Kolmogorov backward equation of $\hat{X}$ (Øksendal, 2003). In diffusion time this equation reads $\partial_t h^{\mathrm{model}} = \widehat{\mathcal{L}}_t^\theta h^{\mathrm{model}}$ with $\widehat{\mathcal{L}}_t^\theta = \frac{1}{2}g^2\Delta + (g^2 s_\theta - f)\cdot\nabla$, which is equation 3 with $\nabla \log p_t$ replaced by s_θ. The initial condition is attained along $\hat{X}$, by the argument of Theorem 1(b).

Equality. If $s_\theta = \nabla \log p_t$, the two Cauchy problems and the two reverse processes coincide: started from p_T, the model process has, by (A), the law of the time reversal of X, so attainment of the initial condition along $\hat{X}$ is attainment along X, and part (c) of Theorem 1 gives $h^{\mathrm{model}} = h^{\mathrm{data}}$ for every C. More generally, if the model's reverse process has the transition laws of the true reverse process, then $\Pr(\hat{X}_{(0)} \in C \mid \hat{X}_{(t)} = x) = \Pr(X_0 \in C \mid X_t = x)$ by definition.

Size of the gap. Let $e_t = \nabla \log p_t - s_\theta(\cdot,t)$ and $w_t = h_t^{\mathrm{data}} - h_t^{\mathrm{model}}$. Subtracting the two equations, $\partial_t w = \widehat{\mathcal{L}}_t^\theta w + g(t)^2\, e_t \cdot \nabla h_t^{\mathrm{data}}$. Applying Itô's formula to w along $\hat{X}$ from diffusion time t down

to ε, and assuming $\mathbb{E}\big[\int_\varepsilon^t g(\tau)^2\,|e_\tau \cdot \nabla h_\tau^{\mathrm{data}}|(\hat{X}_{(\tau)})\,\mathrm{d}\tau \mid \hat{X}_{(t)} = x\big] < \infty$ (with $|w| \le 1$ this also makes the stochastic integral a true martingale),

$$w_t(x) = \mathbb{E}\big[w_\varepsilon(\hat{X}_{(\varepsilon)}) \,\big|\, \hat{X}_{(t)} = x\big] + \mathbb{E}\Big[\int_\varepsilon^t g(\tau)^2\, e_\tau(\hat{X}_{(\tau)}) \cdot \nabla h_\tau^{\mathrm{data}}(\hat{X}_{(\tau)})\,\mathrm{d}\tau \;\Big|\; \hat{X}_{(t)} = x\Big].$$

The first term vanishes as $\varepsilon \downarrow 0$ when both functions approach $\mathbf{1}_C$ along the model's paths, and the gap is then an integral of the score error along the model's paths, weighted by the gradient of h^{data}: it vanishes when $e \equiv 0$ and does not depend on how either function is estimated. □

A.5 The closed form of the guidance field

For the variance-preserving kernel $q_t(x \mid x_0) = \mathcal{N}\big(x; \sqrt{\bar\alpha_t}\,x_0, (1-\bar\alpha_t)I\big)$, $\nabla_x q_t(x \mid x_0) = q_t(x \mid x_0)\,(\sqrt{\bar\alpha_t}\,x_0 - x)/(1-\bar\alpha_t)$. Differentiating m_t and p_t under the integral sign, which the Gaussian factor permits,

$$\nabla \log m_t(x) = \frac{\sqrt{\bar\alpha_t}\,\mathbb{E}[X_0 \mid X_t = x,\, X_0 \in C] - x}{1-\bar\alpha_t}, \quad \nabla \log p_t(x) = \frac{\sqrt{\bar\alpha_t}\,\mathbb{E}[X_0 \mid X_t = x] - x}{1-\bar\alpha_t},$$

the second being Tweedie's formula (Efron, 2011). Their difference is equation 4. The reference field of Tables 4 and 6 estimates both conditional means from the same reverse trajectories of the frozen model started at x_t. These trajectories follow the model's reverse process, so the reference is the right-hand side of equation 4 with the model's posterior in place of the data posterior. It equals $\nabla \log h_t^{\mathrm{model}}$ when the model's reverse kernel from level t to 0 has the form $K_t(\mathrm{d}x_0 \mid x) \propto \mu_t(\mathrm{d}x_0)\,q_t(x \mid x_0)$, as for an exact score, and otherwise differs from it by a term due to the model's error. Since h_θ estimates h_t^{data}, the reported cosines contain the estimand gap of Proposition 3 and this discrepancy.

A.6 The guided sampler and composition

In the DDPM parameterization $s_\theta = -\epsilon_\theta/\sqrt{1-\bar\alpha_t}$, so the replacement equation 7 is the same as replacing the score by $\tilde s = s_\theta + \gamma \nabla \log(h_\theta + \delta)$. With $\gamma = 1$, $\delta = 0$ and exact s_θ and h_θ, $\tilde s = \nabla \log p_t + \nabla \log h_t = \nabla \log p_t(\cdot \mid X_0 \in C)$, the score in the drift of Theorem 1(d); a sampler that discretizes the reverse-time dynamics with a score therefore discretizes the h-transformed dynamics when it is given $\tilde s$. DDIM discretizes the probability-flow ODE, whose velocity uses the score in the same way; with the conditional score it transports $p_T(\cdot \mid X_0 \in C)$ through the same marginals, which also solve the forward equation. On QM9 we add $\gamma \nabla \log(h_\theta + \delta)$ to the node-feature score only, so even at $\gamma = 1$ the correction is partial. On Stable Diffusion the correction is added to the classifier-free-guided noise prediction, so it conditions the guided sampler rather than a single score model.

If, for every $t \in (0, T]$, $\mathbf{1}_{C_1}(X_0)$ and $\mathbf{1}_{C_2}(X_0)$ are conditionally independent given X_t, then $\Pr(X_0 \in C_1 \cap C_2 \mid X_t = x) = h_t^{(1)}(x)\,h_t^{(2)}(x)$, the correction for the intersection is $\nabla \log h_t^{(1)} + \nabla \log h_t^{(2)}$, and Theorem 1(d) applied to $C_1 \cap C_2$ gives marginals $p_t(\cdot \mid X_0 \in C_1 \cap C_2)$. With weights (w_1, w_2) the drift at time t uses the score of the density proportional to $p_t\,(h_t^{(1)})^{w_1}(h_t^{(2)})^{w_2}$; for weights other than $(1,1)$ this family does not in general solve the forward equation, so it is not the marginal law of the guided process, and the weights act as a trade-off between the constraints.

B Implementation details

B.1 Training objective

The heads of Sections 6.1, 6.6 and 6.7 minimize $\mathcal{L}_{\mathrm{bdry}} + \mathcal{L}_{\mathrm{sup}} + 0.1\,\mathcal{L}_{\mathrm{fc}}$ exactly as written in Section 4: t is drawn uniformly from $\{0, \dots, T-1\}$ for $\mathcal{L}_{\mathrm{sup}}$ and from $\{0, \dots, T-2\}$ for $\mathcal{L}_{\mathrm{fc}}$, and x_t and x'_{t+1} are noised from the same x_0 with independent noise. (Noise levels are indexed from 0 to $T-1$ in code, matching the samplers' timestep indices; Algorithm 1 draws one t for both terms, the code draws them independently, and the expected objective is the same.) The two regularizers play different roles. $\mathcal{L}_{\mathrm{bdry}}$ is minimized by $h(\cdot, 0) = \mathbf{1}_C$, the initial condition itself, so it adds

weight at $t = 0$ without changing the target. $\mathcal{L}_{\text{fc}}$ is a smoothing penalty and moves the target: when its stop-gradient target equals h_{t+1}, the minimizer of $\mathcal{L}_{\text{sup}} + \lambda_{\text{fc}}\mathcal{L}_{\text{fc}}$ at level t is, pointwise,

$$\frac{h_t(x) + \lambda_{\text{fc}}\,\mathbb{E}\big[h_{t+1}(X'_{t+1}) \mid X_t = x\big]}{1 + \lambda_{\text{fc}}},$$

a convex combination of h_t and the posterior mean, given $X_t = x$, of the label after an extra independent noising of X_0 to level $t + 1$. The second term need not be close to h_t even for adjacent levels. With $\lambda_{\text{fc}} = 0.1$ the bias is at most $1/11$ in absolute value, and Proposition 1 rests on $\mathcal{L}_{\text{sup}}$ alone.

The MNIST heads of Sections 3 and 6.5 and Appendix C.8 (digit 3, even digit, digit > 5) use the same three terms and weights with one difference in the consistency term: its pair is (x_t, x_{t-1}), noised from the same x_0 with a shared noise draw, and the stop-gradient is on the less noisy member $h_\theta(x_{t-1}, t - 1)$. The two members of this pair are close, so this term moves the target less than independent noisings do; the small-h evidence of Tables 4 and 6 comes from these heads.

B.2 Evaluation protocol

Satisfaction is the fraction of generated samples that satisfy the constraint; every run also records a Wilson 95% interval (Wilson, 1927). For the pixel predicates, the Stable Diffusion predicate and the molecular predicates, the function that labels the training data also scores the samples. For the MNIST digit constraints and the CIFAR-10 classes cat and ship, the training labels are the dataset labels, and a classifier trained separately on the same dataset judges the generated samples. Tables report means over seeds and, where shown, the standard deviation over seeds (normalized by the number of seeds).

FID (Heusel et al., 2017) is computed with `torch-fidelity` (Obukhov et al., 2020) between the samples of one seed and a reference set; the reference sets are listed per experiment below. The *CLIP score* is 100 times the cosine similarity between the image and prompt embeddings of the OpenAI CLIP ViT-L/14 model (Radford et al., 2021), averaged over images.

Molecular metrics. Validity is the fraction of generated graphs that decode to a valid RDKit molecule. Satisfaction and uniqueness are computed over valid molecules, and novelty is the fraction of unique valid molecules whose SMILES string is not among the 120,732 distinct SMILES of the 120,803 training molecules. QM9 is drawn from an exhaustive enumeration of small molecules, so a novel molecule here is one absent from the training set, not new chemistry.

Timing is wall-clock time per sample over a full run on one GPU. The MNIST, CIFAR-10 and QM9 runs used an NVIDIA RTX 4090, except the QM9 SVDD-PM run and the unguided reference it is compared with, which both ran on an NVIDIA T4, and the runs of Appendix C.4, Table 8 and the unguided Euler–Maruyama QM9 baseline, which used an NVIDIA A100. On Stable Diffusion, the unguided, guided and classifier-guided runs of Tables 19 and 20 shared one session, the SVDD-PM run used a second one, and the runs of Table 21 a third, on an NVIDIA A100.

B.3 MNIST

Data and base model. MNIST (LeCun et al., 1998), scaled to $[-1, 1]$: the 60,000 training images label and train the heads, and held-out quantities use the test set. The base model is a DDPM with a small U-Net that we trained on MNIST, with a linear schedule from $\beta_1 = 10^{-4}$ to $\beta_T = 0.02$, $T = 1000$, sampled with 1000 ancestral steps.

Heads. Every MNIST head, including the SVDD-MC value network, has 122,657 parameters: three 3×3 convolutions with 32, 64 and 128 channels (ReLU, two 2×2 max-pools, global average pooling), a 32-dimensional sinusoidal time embedding followed by a linear layer, and a head $160 \to 128 \to 64 \to 1$ with a sigmoid output. The heads of Sections 3 and 6.5 and Appendix C.8 were trained with Adam (learning rate 5×10^{-4}, batch 128) for 30 epochs (20 for the digit > 5 head). The ablation heads of Appendix C.7 were trained with AdamW (learning rate 3×10^{-4}, weight decay 10^{-4}, batch 128) for 30 epochs.

Diagnostics of Section 3. $\hat{\sigma}^2(t)$ is the sample variance of $h_\theta(X_t, t)$ over forward-noised test images, every 10 steps; the guidance norm is the mean ℓ_2 norm of $\nabla \log(h_\theta + \delta)$ over forward-noised test

images. For the Monte Carlo reference (Tables 4 and 6) we take, at each t, 50 test images, 25 of them of digit 3, noise each once, and run $K = 1{,}000$ reverse trajectories of the frozen model from each x_t in bf16 precision. Endpoints are judged by the classifier; $h_{\mathrm{MC}} = (n_{\mathrm{sat}} + 0.5)/(K + 1)$, and the reference field is equation 4 with both conditional means taken over the same trajectories. The reference costs $50Kt$ denoiser evaluations at level t, from 5×10^6 at $t = 100$ to 4.5×10^7 at $t = 900$. The δ sweep uses $\delta \in \{10^{-10}, 10^{-8}, 10^{-6}, 10^{-4}, 10^{-2}\}$ at $\gamma = 5$ with 3 seeds $\times$ 1,000 samples per value; the small-h census uses 5,000 held-out samples per noise level.

Results of Section 3. The mean guidance norm is 2.32 at $t = 100$ and 0.225 at $t = 900$. The fraction of held-out states with $h_\theta < 0.01$ is 89% at $t = 50$, 21.6% at $t = 500$, 2.4% at $t = 600$ and 0% from $t = 700$ on. Over the five values of δ, satisfaction is $94.8 \pm 0.5\%$.

SVDD (Section 6.5). At every step each particle draws M candidates from the base model's transition (one shared denoiser call for the mean) and keeps one of them, either with probability proportional to its value or the one of highest value. SVDD-PM values a candidate by the classifier's probability of digit 3 on its posterior-mean (Tweedie) estimate, which costs one denoiser call per candidate. The SVDD-MC value network is trained by regression on 500,000 pairs (x_t, t), taken every 10 steps along 5,000 unguided trajectories of the base model and labelled by the classifier's verdict on each trajectory's endpoint (10.8% positive). Building these targets takes 5,000,000 per-sample denoiser evaluations (233.1 s), and the regression 391.5 s. Our head was retrained under the same harness (optimizer, epochs, batch size and steps per epoch) to time it: 340.3 s and no denoiser evaluations. The SVDD comparison uses 5 seeds $\times$ 5,000 samples per method.

B.4 CIFAR-10

Base model. The CIFAR-10 (Krizhevsky, 2009) DDPM of Ho et al. (2020) (`google/ddpm-cifar10-32`), sampled with 1000 ancestral steps.

Predicates. On images in $[0, 1]$, with luminance $Y = 0.299R + 0.587G + 0.114B$: *red-pixel* holds if more than 5% of pixels satisfy $R > G + 0.15$ and $R > B + 0.15$; *edge density* holds if the mean Sobel gradient magnitude of Y (3×3 kernels, zero padding) exceeds 0.8453; *low saturation* holds if the mean over pixels of $(\max - \min)/(\max + 10^{-8})$ across the colour channels is below 0.1712; *upper luminance* holds if the mean of Y over rows 0–15 minus its mean over rows 16–31 exceeds 0.1430. The last three thresholds are the 75th, 25th and 75th percentiles of the statistic on the first 10,000 training images. Positive rates on the 50,000 training images are 26.8%, 24.8%, 25.0% and 24.5%. For the class constraints of Table 9, labels are the CIFAR-10 classes.

Heads. h_θ has 675,969 parameters: 3×3 convolutions $3 \to 64 \to 64$, max-pool, $64 \to 128 \to 128$, max-pool, $128 \to 256$ (batch normalization and ReLU after each), global average pooling, a 64-dimensional sinusoidal time embedding with a linear layer, and a head $320 \to 256 \to 128 \to 1$ (ReLU, dropout 0.1 after the first layer, sigmoid output). It is trained on the full training set with AdamW (learning rate 3×10^{-4}, weight decay 10^{-4}, batch 128) for 50 epochs, in 435–448 s. The classifier-guidance baseline uses the same trunk and time embedding with a two-logit head (676,098 parameters), trained by cross-entropy on the same labels, at uniformly drawn t, with the same optimizer, batch size and number of epochs (301–322 s, plus about 4 s to label the training set). It guides with the standard rule $\epsilon \leftarrow \epsilon - \gamma\sqrt{1 - \bar{\alpha}_t}\, g$, $g = \nabla_{x_t} \log p_\phi(y{=}1 \mid x_t, t)$, or with the normalized rule, which divides g by $\|g\|$ per sample (CG-norm). A guided sample costs 0.404 s with either method, against 0.386 s unguided.

FID references. Each seed contributes 1,000 samples. The full-distribution reference is the CIFAR-10 test set (10,000 images). The constraint-conditional reference contains the test images that satisfy the predicate: 2,704 (red-pixel), 2,538 (edge density), 2,536 (low saturation) and 2,416 (upper luminance); for cat and ship, it contains the test images that the evaluation classifier assigns to the class.

B.5 Stable Diffusion 1.5

Base model and sampler. Stable Diffusion 1.5 (Rombach et al., 2022) (`runwayml/stable-diffusion-v1-5`) at 512×512, DDIM (Song et al., 2021a) with 50 steps and classifier-free guidance scale 7.5. The head operates on the $4 \times 64 \times 64$ latents (VAE posterior mean scaled by 0.18215) and is queried on the latent the U-Net sees at every step.

Predicate and data. An image is *grayscale* if its mean HSV saturation (PIL, scaled to $[0, 1]$) is below 0.10. The same function labels the training images, center-cropped to 512×512, and scores the decoded samples. Training uses a fixed subset of $30{,}000$ COCO train2017 images (Lin et al., 2014) (6.1% positive), split into $28{,}500$ for training and $1{,}500$ for probing.

Heads. h_θ has $1{,}672{,}833$ parameters: 3×3 convolutions $4 \to 64 \to 128 \to 256 \to 256$ with stride 2 and a final $256 \to 256$ convolution, each followed by group normalization (8 groups) and SiLU, global average pooling, a 64-dimensional time embedding, and the head of the CIFAR-10 network. It is trained on Stable Diffusion's 1000-step noise schedule with AdamW (learning rate 3×10^{-4}, weight decay 10^{-4}, batch 256) for 100 epochs, in 313 s. The classifier-guidance baseline has the same trunk with a two-logit head ($1{,}672{,}962$ parameters), trained by cross-entropy with the same optimizer, batch size and number of epochs (149 s). It guides with the normalized rule in Table 20, and with both rules in Table 21. On the probe split, the held-out accuracy of h_θ (thresholded at 0.5) at $t = 100, 300, 500, 700, 900$ is 98.4, 98.0, 97.5, 96.1 and 94.7%, and that of the classifier is 98.1, 97.6, 97.8, 95.1 and 94.5%.

Prompts and metrics. The prompts are the first captions of 200 randomly chosen COCO train2017 images. For each seed they are shuffled with the seed and cycled to 500 images, so all methods see the same prompts in the same order. The constraint-conditional FID reference is the set of $1{,}842$ grayscale images in the $30{,}000$-image subset; the full-distribution reference is $5{,}000$ random COCO val2017 images, center-cropped to 512×512. Table 21 uses a second prompt set, the first captions of 200 COCO train2017 images chosen with seed 0, with 1 seed $\times$ 500 images per run and the same full-distribution reference.

SVDD-PM. At every step, $M = 5$ stochastic DDIM candidates ($\eta = 1$) are drawn; each is valued by the predicate on its decoded posterior-mean estimate (from the text-conditional prediction), and the candidate of highest value is kept; 1 seed $\times$ 300 images.

B.6 QM9

Base model and samplers. The released GDSS checkpoint for QM9 (Jo et al., 2022): a variance-preserving SDE (β from 0.1 to 1.0) on node features and a variance-exploding SDE (σ from 0.2 to 1.0) on the adjacency, with $1{,}000$ steps and node counts drawn from the training graphs. All runs integrate the reverse SDE with the reverse-diffusion (Euler–Maruyama) predictor and the same score networks; guided runs add $\gamma \nabla \log(h_\theta + \delta)$ to the node-feature score, and the unguided baseline is the same sampler with $\gamma = 0$. For reference, GDSS's predictor–corrector sampler (reverse-diffusion predictor, one Langevin corrector step, signal-to-noise ratio 0.2, scale 0.9) gives 66.9% on the Lipinski-type filter. Each seed generates $1{,}000$ molecules in batches of 50.

Data and predicates. The QM9 (Ramakrishnan et al., 2014) training split as preprocessed by GDSS: $120{,}803$ kekulized molecules with at most nine heavy atoms, all of which decode to valid molecules; the remaining $13{,}082$ molecules form the test split. Predicates are computed with RDKit on the decoded SMILES, identically for training labels and generated molecules. *Ring presence* holds if the molecule has at least one ring (89.5% of training molecules). The *Lipinski-type* filter (Lipinski et al., 1997) requires molecular weight below 125, at most one hydrogen-bond donor and at most two acceptors (35.3%); the thresholds were chosen by grid search for the positive rate closest to 35%, since QM9 molecules are far smaller than the drug-like molecules for which the original cutoffs were designed.

Heads. h_θ has $320{,}321$ parameters: the node features and the adjacency rows are each encoded by two linear layers of width 256 with ReLU and averaged over nodes, a 64-dimensional time embedding is appended, and a head $576 \to 256 \to 128 \to 1$ with sigmoid output follows. Training noises both the node features and the adjacency with their forward kernels and uses Adam (learning rate 3×10^{-4}, batch 32) for 150 epochs.

SVDD-PM. In the same Euler–Maruyama loop, each step draws $M = 5$ candidate transitions, decodes the posterior-mean estimate of each into a molecule, scores it with the predicate, and keeps the candidate of highest value; 1 seed $\times$ 500 molecules, timed on a T4 GPU, like the unguided reference it is compared with, GDSS's predictor–corrector sampler (8.24 against 1.12 s per molecule). The unguided Euler–Maruyama baseline took 0.60 s per molecule on an NVIDIA A100.

C ADDITIONAL RESULTS

C.1 TABLES AND ALGORITHM REFERENCED FROM THE MAIN TEXT

Table 4: The learned guidance field on MNIST digit 3. Mean cosine between $\nabla \log h_\theta$ and the rollout reference equation 4, over the eligible states among 50 per noise level (1,000 trajectories per state; Table 6), and the fraction of 5,000 held-out forward-noised samples with $h_\theta < 0.01$.

Noise level t	500	700	900
Cosine to reference	0.50	0.54	0.50
$h_\theta < 0.01$ (%)	21.6	0	0

Algorithm 1 Training and guided sampling

Train: repeat: draw x_0 with label $y = \mathbf{1}_C(x_0)$, draw t;
form x_t, x'_{t+1} by independent forward noising of x_0;
take a gradient step on $(h_\theta(x_t,t)-y)^2+(h_\theta(x_0,0)-y)^2+\lambda_{\mathrm{fc}}(h_\theta(x_t,t)-\mathrm{sg}[h_\theta(x'_{t+1},t+1)])^2$

Sample: $x_T \sim \mathcal{N}(0, I)$
for $t = T, \ldots, 1$ **do**
$g_t \leftarrow \nabla_{x_t} \log(h_\theta(x_t,t) + \delta)$
$\tilde{\epsilon} \leftarrow \epsilon_\theta(x_t,t) - \gamma\sqrt{1-\bar{\alpha}_t}\, g_t$
$x_{t-1} \leftarrow$ reverse step of the base sampler with $\tilde{\epsilon}$
end for
return x_0

Table 5: Estimators of h by training target and constraint type: SVDD (Li et al., 2025), Value Matching (Jensen et al., 2026), Guo et al. (2026), DOIT (Zhu et al., 2026), Doob's matching (Chang et al., 2026), StitchVM (Go et al., 2026), classifier guidance (CG) (Dhariwal & Nichol, 2021).

	Soft or bounded-below weights	Hard indicator $\mathbf{1}_C$
Simulation (model law)	SVDD, Value Matching	Guo et al., DOIT
Forward noising (data law)	Doob's matching, StitchVM	CG; **this work**

C.2 THE LEARNED GUIDANCE FIELD

Table 6 extends Table 4 to seven noise levels. From $t = 400$ on, where most states have enough satisfying endpoints to measure the reference, the cosine lies between 0.47 and 0.54, and the error of h_θ against h_{MC} falls from 0.17 to 0.03. At $t \leq 300$ the other half of the states have fewer than 10 satisfying endpoints, so the eligible states are those with h_{MC} near 1. There $\nabla \log h_t$ is small and its estimate in equation 4 rests on the few endpoints outside C, so we do not interpret the cosine at these levels.

Table 6: The learned guidance field on MNIST digit 3 against the Monte Carlo reference of Appendix B.3 (full version of Table 4). Eligible states have at least 10 satisfying endpoints out of $K = 1{,}000$; the cosine is averaged over eligible states, and the root-mean-square error of h_θ against h_{MC} and the mean of h_{MC} are computed over all 50 states.

t	Eligible states	Mean h_{MC}	RMSE of h_θ	Cosine to reference
100	25/50	0.500	0.190	-0.145
200	25/50	0.498	0.161	-0.067
300	25/50	0.456	0.141	+0.240
400	28/50	0.356	0.173	+0.467
500	42/50	0.234	0.170	+0.501
700	50/50	0.122	0.048	+0.544
900	50/50	0.111	0.031	+0.499

C.3 CIFAR-10

Table 7 gives the complete sweep of ours and of the normalized rule behind Table 1. At every guidance scale and under both protocols our samples are closer to the reference than those of normalized classifier guidance: the ratio of its FID to ours ranges from 1.39 to 3.21 for $\mathrm{FID_c}$ and from 1.39 to 2.55 for $\mathrm{FID_f}$. Table 8 gives the standard rule of Table 1 with $\mathrm{FID_f}$. On the class constraints of Table 9, satisfaction rises from 14.2% to 71.5% (cat) and from 7.2% to 87.5% (ship), and $\mathrm{FID_c}$ improves from 82.5 and 140.4 to at most 62.5 and 44.0 for $\gamma \geq 5$, while $\mathrm{FID_f}$ rises as the samples concentrate on one class.

Table 7: Full sweep of normalized classifier guidance (CG-norm) and ours behind Table 1 (the standard rule is in Table 8), on four CIFAR-10 predicates at every guidance scale, with the unguided model at $\gamma = 0$. Mean over 3 seeds $\times$ 1,000 samples; the seed standard deviation is at most 0.8 for guided satisfaction (2.3 for unguided) and 2.0 for FID.

		Satisfaction (%)		$\mathrm{FID_c}$		$\mathrm{FID_f}$	
Predicate	γ	CG-norm	Ours	CG-norm	Ours	CG-norm	Ours
Red-pixel	0	31.3		54.0		42.5	
	2	100.0	97.2	107.3	48.8	128.6	59.0
	5	100.0	100.0	187.0	58.2	198.5	77.7
	10	100.0	100.0	209.4	75.1	218.4	98.0
	15	100.0	100.0	212.3	93.1	220.9	115.4
Upper luminance	0	23.1		69.9		42.5	
	2	99.9	99.5	61.7	39.2	99.6	57.2
	5	100.0	100.0	81.4	43.9	119.2	76.0
	10	100.0	100.0	86.1	50.0	124.2	86.3
	15	100.0	100.0	87.7	52.7	125.8	90.0
Edge density	0	16.5		61.4		42.5	
	2	99.7	99.3	68.6	44.6	90.7	52.3
	5	100.0	100.0	96.9	59.3	122.5	77.0
	10	100.0	100.0	106.3	73.3	133.1	92.9
	15	100.0	100.0	109.5	78.8	137.2	98.7
Low saturation	0	16.8		63.0		42.5	
	2	99.8	75.2	68.7	43.4	90.2	47.2
	5	100.0	95.9	101.6	45.9	120.6	58.9
	10	100.0	99.5	111.7	61.1	129.6	76.9
	15	100.0	99.9	116.8	73.2	134.9	88.7

Table 8: Classifier guidance with the standard rule on the four CIFAR-10 predicates of Table 1. Satisfaction (%), $\mathrm{FID_c}$ and $\mathrm{FID_f}$; mean over 3 seeds $\times$ 1,000 samples; the seed standard deviation is at most 1.3 for $\mathrm{FID_c}$ and 0.2 for satisfaction.

	$\gamma = 5$			$\gamma = 15$		
Predicate	Sat.	$\mathrm{FID_c}$	$\mathrm{FID_f}$	Sat.	$\mathrm{FID_c}$	$\mathrm{FID_f}$
Red-pixel	100.0	62.8	82.6	100.0	98.2	120.1
Upper luminance	100.0	43.4	75.0	100.0	50.1	86.6
Edge density	99.7	52.5	67.5	100.0	72.7	90.4
Low saturation	99.5	48.6	62.4	100.0	71.2	85.6

Table 9: Our guidance on two CIFAR-10 class constraints. A separately trained classifier judges the samples and selects the constraint-conditional FID reference from the test set. Mean over 3 seeds $\times$ 1,000 samples; the seed standard deviation is at most 2.0.

Class	γ	Satisfaction (%)	$\mathrm{FID_c}$	$\mathrm{FID_f}$
Cat	0	14.2	82.5	42.5
	2	32.7	66.6	54.5
	5	51.2	60.8	70.5
	10	66.2	60.0	86.2
	15	71.5	62.5	95.8
Ship	0	7.2	140.4	42.5
	2	50.2	52.7	77.3
	5	75.9	41.5	114.7
	10	84.6	42.6	132.8
	15	87.5	44.0	139.5

C.4 Allocation, estimand and rejection experiments

These runs use the base model and heads of Appendix B.4, 1,000 ancestral steps unless stated, 1,000 samples per run, and an NVIDIA A100; the classifier-guidance arms use the matched classifiers of Section 6.1 with the standard rule (and, in Table 2, two further classifiers trained with the same recipe). Unguided pass rates differ slightly between the sample sets of Tables 1, 12 and the rejection runs below (30.8 to 31.5% on red-pixel, 21.8 to 23.1% on upper luminance), as expected at these sample sizes.

Step profile. Along 250 standard-rule trajectories at $\gamma = 2$, $\bar{m}(t)$ is the mean over samples of $\sqrt{1-\bar{\alpha}_t}\,\|g\|$ (Table 10). The rule with an equal step gives every sample $\gamma\,\bar{m}(t)\,g/\|g\|$; the $1/\hat{h}$ rule gives sample i the step $\gamma\,\bar{m}(t)\,w_i\,g_i/\|g_i\|$ with $w_i = (1/\hat{h}_i)/\mathrm{mean}_j(1/\hat{h}_j)$ over the batch of 250 and $\hat{h}_i = p_\phi(y{=}1 \mid x_t)$ floored at 10^{-3}. Classifiers 2 and 3 of Table 2 are trained with training seeds 1 and 2 and the budget of Appendix B.4; for each, $\bar{m}(t)$ is measured along its own 1,000 standard-rule trajectories (sampling seed 0).

Table 10: Mean step per unit γ, $\sqrt{1-\bar{\alpha}_t}\,\|\nabla \log \hat{h}\|$ averaged over 250 trajectories at $\gamma = 2$: "ours" rows along our guided trajectories, CG rows along standard-rule trajectories of the matched classifier. The normalized rule's step is $\sqrt{1-\bar{\alpha}_t}$.

t	999	899	799	699	599	499	399	299	199	99
Red-pixel, ours	0.067	0.066	0.118	0.209	0.276	0.350	0.376	0.301	0.232	0.143
Red-pixel, CG	0.091	0.084	0.147	0.245	0.356	0.401	0.317	0.232	0.131	0.069
Upper lum., ours	0.163	0.181	0.358	0.489	0.372	0.186	0.090	0.044	0.023	0.012
Upper lum., CG	0.141	0.205	0.432	0.526	0.266	0.099	0.046	0.018	0.012	0.006
Normalized	1.000	1.000	0.999	0.997	0.987	0.960	0.897	0.777	0.584	0.321

Step and $\hat{h}$ across samples. Along 250 standard-rule trajectories at $\gamma = 2$ (sampling seed 600), Table 11 gives the Spearman correlation between each sample's step and $\hat{h}(x_t)$; every correlation has $p < 10^{-24}$.

Table 11: Spearman correlation between the per-sample step $\sqrt{1-\bar{\alpha}_t}\,\|g_i\|$ and $\hat{h}_i = p_\phi(y{=}1 \mid x_t)$ across 250 standard-rule trajectories at $\gamma = 2$.

t	799	599	399	199
Red-pixel	-0.688	-0.598	-0.969	-0.998
Upper luminance	-0.791	-0.993	-0.998	-0.969

Estimands. Calibration is the expected calibration error with ten equal-width bins, averaged over ten noise levels spaced evenly along the sampler's schedule; model-law scores use 2,000 unguided trajectories (seed 300) and their outcomes, data-law scores the 10,000 forward-noised CIFAR-10 test images. With 100 steps the sampler uses the base model's scheduler with 100 strided timesteps (Table 12). Guided runs with 100 steps (seed 0), satisfaction (%) and FID_c: red-pixel unguided 45.7 / 118.1, ours $\gamma = 2$ 99.5 / 103.9, CG $\gamma = 2$ 99.7 / 107.6; upper luminance unguided 22.3 / 140.1, ours $\gamma = 2$ 99.5 / 104.3, CG $\gamma = 2$ 99.6 / 104.3.

Table 12: Calibration of h_θ under the model's law and the data law (mean expected calibration error over ten noise levels), with the model's unguided pass rate over the 2,000 trajectories.

	1000 steps			100 steps		
Predicate	Pass (%)	Model law	Data law	Pass (%)	Model law	Data law
Red-pixel	31.5	0.051	0.051	46.0	0.155	0.049
Upper luminance	21.8	0.021	0.024	21.8	0.023	0.024

Rejection sampling and the rare constraint. Rejection runs draw 4,000 (red-pixel) and 4,500 (upper luminance) unguided samples with seed 100 and keep the first 1,000 that pass: 1,230 of 4,000 (30.8%) and 1,014 of 4,500 (22.5%) pass, and the first 1,000 accepted samples have FID_c 48.4 and 42.4 and FID_f 58.2 and 51.6. The rare constraint holds if the red-pixel fraction of the image exceeds 0.6104, the 99th percentile over the training set: 498 of 50,000 training and 87 of 10,000 test images pass, and its FID_c reference is all 585. Both estimators are retrained on these labels with the budget of Appendix B.4, with two training seeds (Table 13). Unguided and guided samples take 0.326 and 0.345 s.

Table 13: The rare constraint (1.00% of training images; the model passes 1.7% of 2,000 unguided samples). Satisfaction (%) and FID_c, sampling seed 0, 1,000 samples per run, for heads trained with seed 0 and seed 1; CG uses the standard rule; –: not run.

Method	γ	Sat. (seed 0)	FID_c (seed 0)	Sat. (seed 1)	FID_c (seed 1)
Ours	2	99.0	80.2	95.5	79.8
	5	99.9	104.9	–	–
	10	100.0	131.9	–	–
CG	2	95.4	78.6	96.0	80.8
	5	100.0	109.0	–	–
	10	100.0	149.6	–	–

C.5 Classifier accuracy across noise levels

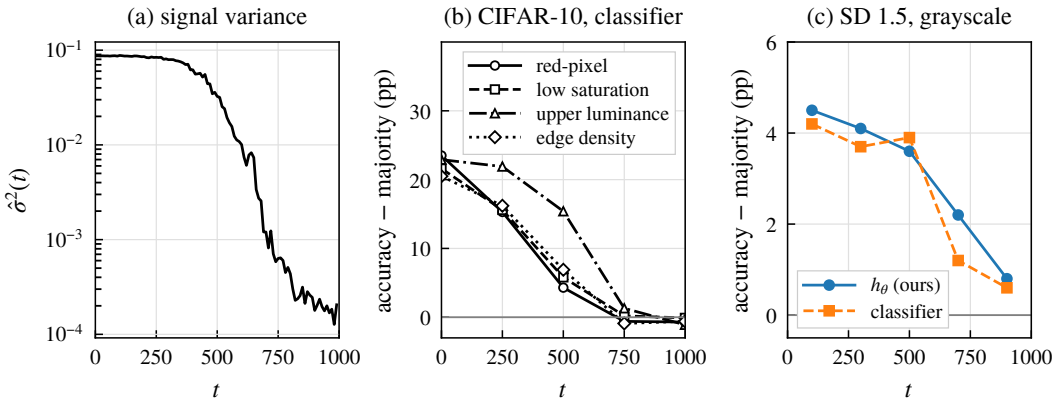

Figure 3: **Signal contraction in three settings.** (a) The constraint-signal variance $\hat{\sigma}^2(t)$ on MNIST falls 492× between $t = 100$ and $t = 900$ (Proposition 2). (b) On CIFAR-10, the held-out accuracy of the matched noisy classifier above its majority baseline decays to within 1.3 points of that baseline by $t = 750$ for all four predicates. (c) On Stable Diffusion 1.5, the held-out accuracies of h_θ and the matched classifier are within 1.0 point of each other at every probed noise level.

By $t = 750$ every classifier is within 1.3 points of its majority baseline (Table 14, Figure 3b). On red-pixel, our head loses its accuracy at the same rate as the classifier. Both estimate the same function (Section 5); what differs between the two methods is the drift applied once the signal is gone.

Table 14: Held-out accuracy (%) across noise levels. Classifier rows: the matched noisy-state classifiers of Section 6.1 on the first 2,000 CIFAR-10 test images (Figure 3b). h_θ row: our head for red-pixel, thresholded at 0.5, on the 4,000 test images of Figure 1, of which 27.1% are positive. The majority baseline is 100 minus the training positive rate.

Predicate	Estimator	$t = 0$	250	500	750	999	Majority
Red-pixel	Classifier	96.7	88.5	77.5	72.6	72.5	73.2
	h_θ	94.3	87.2	77.6	72.8	72.9	73.2
Upper luminance	Classifier	98.4	97.4	90.9	76.8	74.4	75.5
Edge density	Classifier	95.7	91.4	82.1	74.3	74.6	75.2
Low saturation	Classifier	96.6	90.6	80.8	75.2	74.9	75.0

C.6 Comparison with SVDD on MNIST

With proportional selection SVDD-PM needs $M = 20$ candidates, at $21\times$ our time per sample, to match our satisfaction; keeping the best of $M = 5$ candidates reaches 100.0% at $5.7\times$ our time per sample (Table 15). Every SVDD-PM step also evaluates the checker on the posterior-mean estimates of M candidates, whereas our sampler never calls it.

Table 15: Guidance on MNIST digit 3 against SVDD-PM, 5 seeds $\times$ 5,000 samples, our head and the SVDD-MC value network have 122,657 parameters each. Time is per sample on one RTX 4090. Training the SVDD-MC value network took 624.6 s, of which 233.1 s built the targets from 5,000,000 denoiser evaluations; training our head took 340.3 s and no denoiser evaluations.

Method	M	Candidate selection	Satisfaction (%)	s/sample
Ours ($\gamma = 5$)	–	–	95.1 ± 0.3	0.050
SVDD-PM	5	proportional to value	92.5 ± 0.3	0.294
SVDD-PM	20	proportional to value	95.2 ± 0.3	1.048
SVDD-PM	5	highest value	100.0 ± 0.0	0.283

C.7 Objective ablations

The method needs only $\mathcal{L}_{\text{sup}}$ (Table 16): with three training seeds per objective, $\mathcal{L}_{\text{sup}}$ alone reaches $86.7 \pm 3.3\%$, and removing either regularizer, or both, does not lower satisfaction. The four objectives reach 81.3 to 86.7%, and the spread across training seeds (standard deviation 1.5 to 3.3 points) is several times larger than the spread across sampling seeds (0.14 to 0.44 points). This is what Proposition 1 predicts: $\mathcal{L}_{\text{sup}}$ alone identifies h. The heads of Section 6 were trained with the full objective before this ablation and are reported unchanged; we attribute no result to the regularizers. Among constructions of the consistency pair (Table 17), independent noisings of the same x_0, the construction of Section 4, are best at $\gamma = 2$ and within 0.4 points of the best at $\gamma \geq 5$.

Table 16: Loss ablation on MNIST digit 3 at $\gamma = 3$ (strict fp32). Three heads per objective (training seeds 0–2), each evaluated over 5 sampling seeds $\times$ 5,000 samples; mean and standard deviation across training seeds. For every head, the standard deviation across sampling seeds is 0.14 to 0.44 points.

Objective	Per training seed (%)	Mean (%)
$\mathcal{L}_{\text{bdry}} + \mathcal{L}_{\text{sup}} + 0.1\,\mathcal{L}_{\text{fc}}$ (full)	79.2 / 79.9 / 84.9	81.3 ± 2.5
$\mathcal{L}_{\text{bdry}} + \mathcal{L}_{\text{sup}}$	85.3 / 83.9 / 89.6	86.3 ± 2.4
$\mathcal{L}_{\text{sup}} + 0.1\,\mathcal{L}_{\text{fc}}$	87.4 / 83.9 / 86.7	86.0 ± 1.5
$\mathcal{L}_{\text{sup}}$ only	88.2 / 82.2 / 89.8	86.7 ± 3.3

Table 17: Construction of the consistency pair in $\mathcal{L}_{\text{fc}}$ on MNIST digit 3: one head per construction (training seed 0, default GPU precision), evaluated over 5 seeds $\times$ 5,000 samples. Satisfaction (%), mean $\pm$ standard deviation over seeds.

Pair (x_t, x'_{t+1})	$\gamma = 2$	$\gamma = 5$	$\gamma = 10$
Independent noisings of x_0 (Section 4)	76.7 ± 0.6	94.0 ± 0.1	98.3 ± 0.1
Shared noise draw	72.8 ± 0.6	94.1 ± 0.2	98.7 ± 0.1
x'_{t+1} drawn from $q(x_{t+1} \mid x_t)$	63.2 ± 0.5	87.8 ± 0.3	96.1 ± 0.2

C.8 Composition on MNIST

Guiding with one head raises its own constraint but leaves the joint rate between 25.7% and 31.8%. Adding the two log-gradients raises the joint rate from 16.1% to 67.3% at $\gamma = 10$ with equal weights, and to 74.3% when the even head is weighted twice; the weights move satisfaction between the two constraints (Table 18).

Table 18: Composition of two separately trained MNIST heads, digit even and digit > 5, whose intersection is $\{6, 8\}$. Entries: joint satisfaction (%), followed by the two marginal rates (even, > 5) in parentheses; mean over 5 seeds $\times$ 5,000 samples (seed standard deviation at most 1.0). Weights (w_1, w_2) multiply $\log h_{\text{even}}$ and $\log h_{>5}$; $(1, 0.5)$ at $\gamma = 10$ applies the same guidance as $(2, 1)$ at $\gamma = 5$, as does $(0.5, 1)$ at $\gamma = 10$ with $(1, 2)$ at $\gamma = 5$.

Guidance	$\gamma = 2$	$\gamma = 5$	$\gamma = 10$
Unguided		16.1 (53.7, 33.3)	
Even only	25.7 (84.9, 31.0)	26.9 (92.5, 29.6)	26.8 (95.7, 28.5)
> 5 only	29.5 (51.4, 63.2)	31.8 (45.7, 76.8)	29.9 (39.4, 84.6)
Composed, $(1, 1)$	43.2 (81.6, 56.5)	56.0 (87.1, 66.7)	67.3 (91.3, 74.8)
Composed, $(2, 1)$	46.4 (88.1, 54.9)	60.4 (92.1, 66.9)	74.3 (95.5, 78.2)
Composed, $(1, 2)$	48.0 (78.3, 65.7)	61.0 (85.2, 74.1)	71.4 (90.3, 80.1)
Composed, $(0.5, 1)$	39.1 (72.5, 58.9)	50.9 (79.8, 67.7)	60.9 (85.2, 74.0)
Composed, $(1, 0.5)$	37.3 (83.8, 46.9)	49.6 (89.1, 57.8)	60.3 (92.0, 66.9)

C.9 Stable Diffusion 1.5

Table 19: Stable Diffusion 1.5, grayscale predicate, 3 seeds $\times$ 500 images (SVDD-PM: 1 seed $\times$ 300, $M = 5$; Wilson 95% interval 6.3–12.8%). CLIP score of Radford et al. (2021); FID_c and FID_f against the references of Appendix B.5.

Method	γ / M	Sat. (%)	CLIP	FID_c	FID_f	s/sample
Unguided	0	7.2	26.19	109.9	88.9	0.81
Ours	10	33.1	26.23	102.6	89.9	1.03
Ours	35	51.1	26.08	99.6	92.0	1.03
Ours	75	60.0	25.96	98.2	93.5	1.03
SVDD-PM	$M = 5$	9.0	26.47	133.9	117.6	16.65

Our satisfaction increases with γ at every step of the sweep, from 33.1% at $\gamma = 10$ to 60.0% at $\gamma = 75$, while the CLIP score stays within 0.23 of the unguided value and FID_c improves from 109.9 to between 98.2 and 102.6. Normalized classifier guidance is highest at $\gamma = 10$, the smallest scale tested (65.1%), and declines as γ grows, to 24.7% at $\gamma = 75$ (Table 20).

Table 20: Full sweep behind Table 19: grayscale predicate on Stable Diffusion 1.5, 3 seeds $\times$ 500 images; CG-norm is the matched classifier with the normalized rule. Satisfaction is the mean $\pm$ standard deviation over seeds; CLIP score and FID are means.

Method	γ	Satisfaction (%)	CLIP	FID_c	FID_f
Unguided	0	7.2 ± 1.1	26.19	109.9	88.9
Ours	10	33.1 ± 1.1	26.23	102.6	89.9
Ours	20	44.2 ± 0.6	26.11	100.4	91.0
Ours	35	51.1 ± 1.4	26.08	99.6	92.0
Ours	50	54.5 ± 1.4	26.01	100.0	93.6
Ours	75	60.0 ± 0.5	25.96	98.2	93.5
CG-norm	10	65.1 ± 0.9	26.09	99.3	92.5
CG-norm	20	56.2 ± 1.6	26.00	101.1	92.7
CG-norm	35	43.1 ± 2.9	25.96	101.1	92.3
CG-norm	50	34.1 ± 2.6	25.87	101.1	92.0
CG-norm	75	24.7 ± 2.3	25.62	103.5	94.0

Table 21: Stable Diffusion 1.5 on the second prompt set of Appendix B.5 (1 seed $\times$ 500 images): ours, the matched classifier with the standard rule (CG) and with the normalized rule (CG-norm). Satisfaction (%), CLIP score and FID_f.

Method	γ	Satisfaction (%)	CLIP	FID_f
Unguided	0	6.4	26.77	87.25
Ours	10	30.0	26.62	89.65
Ours	35	48.8	26.52	90.89
Ours	75	58.0	26.37	92.96
CG	10	28.8	26.60	89.72
CG	35	46.6	26.50	92.35
CG	75	54.4	26.26	91.92
CG-norm	10	61.8	26.40	91.66
CG-norm	75	25.4	26.06	93.78

C.10 QM9

On the Lipinski-type filter satisfaction rises with γ from 88.7% to 99.5% while uniqueness falls from 92.4% to 63.2%, and novelty stays at or above 92.8%, against 54.3% for the unguided model with the same sampler (Table 22). With composed guidance the joint rate is within 0.7 points of the product of the guided marginals at every $\gamma \geq 5$ (Table 23).

Table 22: Full QM9 results for the Lipinski-type filter, 3 seeds $\times$ 1,000 molecules, all with the Euler–Maruyama sampler (satisfaction: mean $\pm$ standard deviation over seeds). †SVDD-PM: 1 seed $\times$ 500 molecules on a T4 GPU (Appendix B.6).

Method	γ / M	Satisfaction (%)	Valid (%)	Unique (%)	Novel (%)
Lipinski-type filter					
Unguided	0	54.3 ± 0.9	100.0	98.0	98.3
Ours	2	88.7 ± 0.4	100.0	92.4	96.2
Ours	5	94.1 ± 0.4	100.0	88.8	95.2
Ours	10	96.8 ± 0.2	100.0	83.2	94.5
Ours	20	98.1 ± 0.2	100.0	76.0	94.4
Ours	50	99.5 ± 0.4	100.0	63.2	92.8
SVDD-PM†	M=5	96.0	100.0	97.8	97.1

Table 23: Composition on QM9: separately trained Lipinski-type and ring-presence heads, guided by $\log h_{\text{Lip}} + \log h_{\text{ring}}$; $\gamma = 0$ is the unguided model with the same sampler, the run of Table 22. Joint satisfaction (mean $\pm$ standard deviation over 3 seeds $\times$ 1,000 molecules), the two marginal rates, and the product of the two rates shown.

γ	Joint (%)	Lipinski (%)	Ring (%)	Product (%)	Unique (%)	Novel (%)
0	39.6 ± 0.4	54.3	76.7	41.6	98.0	98.3
5	72.4 ± 0.4	92.9	78.6	73.0	92.3	96.0
10	75.5 ± 0.6	95.1	79.8	75.9	91.5	95.6
20	78.3 ± 0.5	96.3	81.6	78.6	90.8	95.9

C.11 Additional figures

Figure 4 computes h_t exactly on a two-dimensional example, Figure 5 shows paired Stable Diffusion samples (Section 6.7), and Figure 6 plots the QM9 composition of Table 23.

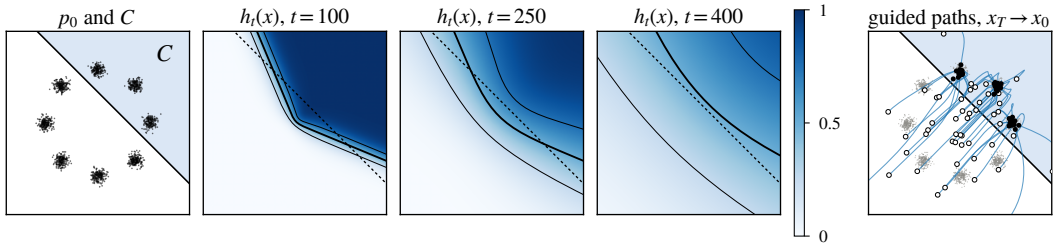

Figure 4: **The constraint posterior h_t on a two-dimensional example, computed exactly.** First panel: p_0 is a mixture of eight Gaussians and C is a half-plane (shaded), with $\Pr(X_0 \in C) = 0.375$. Next three panels: $h_t(x) = \Pr(X_0 \in C \mid X_t = x)$ at three noise levels; black curves are the 0.25, 0.5 and 0.75 level sets and the dashed line is the boundary of C. As t grows, h_t flattens toward $\Pr(X_0 \in C)$ (Proposition 2). Last panel: deterministic reverse paths guided by $\nabla \log h_t$ with $\gamma = 1$, from x_T (open circles) to x_0 (filled). All 2,000 guided samples land in C, against 38.6% without guidance.

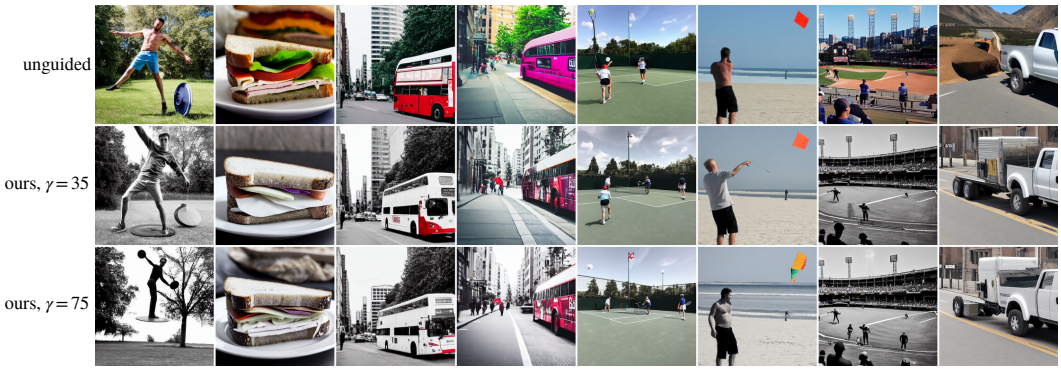

Figure 5: **Stable Diffusion 1.5, grayscale constraint.** The first eight COCO captions in our sample (columns) and, in rows, the unguided sample and ours at $\gamma = 35$ and $\gamma = 75$ from the same seed (DDIM, 50 steps). Guidance with h_θ reduces colour while largely keeping the layout of the unguided image.

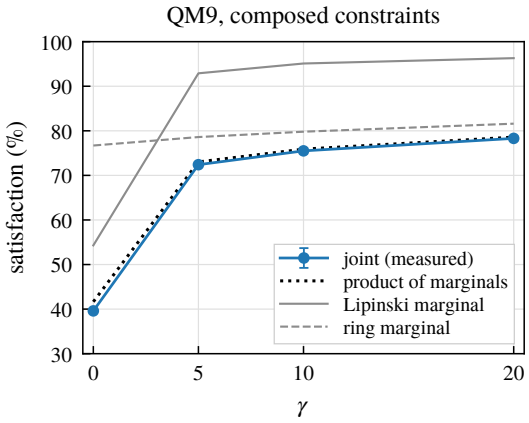

Figure 6: **QM9 composition** with separately trained Lipinski-type and ring-presence heads, guided by $\log h_{\mathrm{Lip}} + \log h_{\mathrm{ring}}$. Joint satisfaction (mean $\pm$ seed standard deviation, 3 seeds $\times$ 1,000 molecules) against the product of the two guided marginal rates, with the marginals shown for reference.