VASR: Variance-Aware Systematic Resampling for Reward-Guided Diffusion

Shivanshu Shekhar[1] **Sagnik Mukherjee**[1] **Jia Yi Zhang**[2] **Tong Zhang**[1]
[1] Siebel School of Computing and Data Science [2] Department of Statistics
University of Illinois Urbana-Champaign
{shekhar6, sagnikm3, jyzhang5, tozhang}@illinois.edu

Abstract

Sequential Monte Carlo (SMC) samplers for reward-guided diffusion models often suffer from rapid lineage collapse: a few high-reward particles dominate the population within a handful of resampling steps, destroying diversity and degrading sample quality. We propose a variance-decomposition framework for reward-guided diffusion SMC that separates continuation variance V_t^{cont} from residual variance V_t^{res}, revealing that high offspring-count variance under the commonly used multinomial resampling drives this collapse. This motivates VASR (Variance-Aware Systematic Resampling), which addresses both variance terms via variance-optimal mass allocation $m_t \propto w_t e^{r_t}$ (minimizing V_t^{cont}) and systematic resampling (controlling V_t^{res}). For latent diffusion models where intermediate rewards are noisy due to stochastic continuations, we propose VASR-MAX, a deliberately biased high-selection variant for variance-sensitive reward optimization. Both methods are training-free, fully parallelizable, and add only linear overhead. On MNIST and CIFAR-10, VASR achieves as high as 26% better FID than prior SMC methods while remaining $\sim 66\times$ faster than MCTS-based value methods at matched compute. On text-to-image generation, VASR-MAX consistently outperforms the strongest SMC baseline across compute budgets and matches MCTS-based methods within 2.5–3% reward at high budgets while being approximately $4\times$ faster.

1 Introduction

Diffusion models [1, 2] have become a dominant paradigm for generative modeling, achieving state-of-the-art performance across modalities including images [1, 3, 4], video [5, 6], and language [7, 8]. In practical settings, however, generation must satisfy objectives beyond reproducing the training distribution, such as generating images that aligns with human preferences. A common requirement is *reward alignment*, where generated samples should remain on the learned data manifold while maximizing a reward function $r : \mathcal{X} \to \mathbb{R}$ that captures human preferences, task objectives, or domain-specific constraints. Existing approaches to reward alignment broadly fall into two categories:

RL-based fine-tuning These approaches formulate alignment as a reinforcement learning objective applied to a pretrained generative model:

$$\mathcal{L}_{\text{RLHF}} = \mathbb{E}_{\tau \sim \pi_\theta}\left[R(\tau)\right] - \beta\, \text{KL}(\pi_\theta \| \pi_{\theta_{\text{ref}}}), \tag{1}$$

where $\pi_{\theta_{\text{ref}}}$ is a reference model and the KL term regularizes deviations from the pretrained distribution. Instantiations include policy gradient methods [9, 10], direct preference optimization [11, 12], and direct reward optimization [13]. While effective, these methods require costly fine-tuning and must be retrained whenever the reward function changes.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

Inference-time alignment An alternative direction avoids retraining by modifying the sampling procedure of a frozen diffusion model to target $\pi^*(x) \propto p_\theta(x) \exp(\lambda r(x))$, where p_θ is the pretrained model and $\lambda \geq 0$ controls alignment strength. These methods allow rapid adaptation to new reward functions without retraining or additional compute-heavy optimization, and hence are usually preferred over finetuning based methods.

Several families of inference-time methods have been proposed. Gradient-based approaches [14, 15] bias the denoising trajectory using reward gradients, which requires differentiable reward models that may destabilize the denoising process by pulling trajectories off the data manifold [16]. Search-based methods [17, 18] perform local reward-guided exploration, while value based methods [16] learn a value function via Monte Carlo rollouts and sample greedily, though both families are computationally impractical at scale. Particle-based methods sidestep both pitfalls: they scale tractably and do not require differentiable reward models. These methods [19–21] maintain a population of K particles during inference and, at each resampling step, draw a new population by sampling particles in proportion to their current reward estimates. However, existing SMC methods suffer from diversity collapse.

We trace this phenomenon of diversity collapse to the commonly used multinomial resampling in current SMC based methods, which introduces offspring-count fluctuations of order $\mathcal{O}(K)$ per particle, leading to rapid and often irreversible loss of diversity that severely degrades sample quality and distributional coverage. We address this through a **variance-decomposition framework** that separates the total estimator variance into two components: continuation variance V_t^{cont} from stochastic particle propagation, and residual variance V_t^{res} from random offspring counts. We minimize V_t^{cont} through variance-optimal mass allocation stratergy (§4) derived from the variance decomposition, and control V_t^{res} using systematic resampling (§3) which bound offspring deviations as $|N_t^{(k)} - m_t^{(k)}| \leq 1$ [22]. The resulting sampler, VASR (Variance-Aware Systematic Resampling), addresses both sources of variance in a principled manner. We also extend VASR to latent diffusion settings with VASR-MAX, where intermediate Tweedie proxy rewards are highly noisy and the objective is to obtain a single high-quality sample; in this regime, concentrating offspring on the highest-scoring particle yields more robust selection.

We validate VASR across a range of three models, four tasks, and five scales and find that it consistently improves over existing approaches. In particular, it yields higher sample quality, better diversity, and more stable inference compared to prior SMC-based and value-based baselines. Our findings demonstrate that principled variance control is key to improving diffusion-based inference, establishing VASR as a scalable and effective approach for inference-time alignment. Our main contributions are:

1. **Variance decomposition for reward-guided diffusion SMC.** We decompose per-step estimator variance into continuation variance V_t^{cont} (from particle propagation) and residual variance V_t^{res} (from offspring sampling). Both terms degrade sample quality under aggressive reward tilting, motivating paired control strategies.
2. **Paired variance control via reweighting and bounded resampling.** The continuation term V_t^{cont} is minimized through variance-optimal mass allocation $m_t^{(k)} \propto w_t^{(k)} \exp(\hat{r}_t^{(k)})$ derived from the decomposition. The residual term V_t^{res} is bounded via systematic resampling, which reduces per-particle variance from $\mathcal{O}(K)$ to $\mathcal{O}(1)$ [22]. This paired strategy yields the VASR sampler.
3. **Empirical validation of the variance mechanism.** Across MNIST, CIFAR-10, and Stable Diffusion, VASR improves distributional quality and lineage diversity over SMC baselines while remaining fully parallelizable. Ablations confirm the mechanism: jointly controlling both variance terms preserves lineages, improves FID, and yields better reward scaling under matched compute.

2 Related Work

Existing inference-time alignment approaches for diffusion models fall into three broad categories: gradient-based guidance, search/value-based methods, and particle-based sampling.

Gradient-based Guidance. Gradient-based methods steer the model towards the desired objective [14, 15, 23–25] using the gradient of the value or score function. This is closely related to

classifier guidance [26] and has been applied to incorporate semantic constraints, reward models, and preference signals during sampling. While effective, these methods are inapplicable to non-differentiable or discrete-state settings and introduces per-step gradient cost that scales with model size.

Search and Value-based Methods. Search-based methods generate multiple candidate denoising transitions and select higher-reward trajectories, either via greedy local search or by framing inference as a search problem over noise trajectories with verifier feedback [17, 18, 27]. While these approaches can improve sample quality, they typically rely on shallow look-ahead and may overfit to verifier biases under aggressive search [27]. Value function-based methods instead estimate soft values at intermediate states to guide sampling. Diffusion Tree Sampling (DTS) [16] formulates the reverse process as a finite-horizon tree and applies Monte Carlo Tree Search with soft Bellman backups, achieving strong asymptotic guarantees and improved compute efficiency. However, its sequential tree construction limits parallelism: at matched NFE budgets, DTS is 4-66$\times$ slower than fully parallel particle-based methods [16].

Particle-based Methods. Particle-based methods [28, 20, 21, 29] propagate multiple candidate trajectories (particles) simultaneously through the diffusion process, resampling them at intermediate steps according to potential functions that approximate the soft value. Sequential Monte Carlo (SMC) enables reward-guided generation with frozen model weights. SMC is theoretically guaranteed to recover the target distribution given exact potentials and infinite particles [30]. The core bottleneck for sample quality in particle-based diffusion alignment is that the repeated resampling step can dramatically reduce diversity, leading to particle degeneracy and lineage collapse when a few high-weight trajectories dominate the population.

Our approach is particle-based but distinguished by a variance-decomposition framework that separates continuation and residual variance. This decomposition reveals both terms as bottlenecks in reward-guided diffusion and motivates paired control: variance-optimal mass allocation for V_t^{cont} and systematic resampling for V_t^{res}.

3 Preliminaries

Diffusion Models. A diffusion model defines a forward noising process $q(x_t \mid x_0) = \mathcal{N}(x_t;\, \sqrt{\bar{\alpha}_t}\, x_0,\, (1-\bar{\alpha}_t)I)$, where $\bar{\alpha}_t = \prod_{s=1}^{t} \alpha_s$. A neural network $\epsilon_\theta(x_t, t)$ is trained to predict the injected noise, implicitly learning the score of the data distribution. Given a noisy sample x_t, the Tweedie estimate of the clean sample is

$$\hat{x}_0(x_t, t) = \frac{x_t - \sqrt{1-\bar{\alpha}_t}\, \epsilon_\theta(x_t, t)}{\sqrt{\bar{\alpha}_t}}, \tag{2}$$

which corresponds to the posterior mean $\mathbb{E}[x_0 \mid x_t]$ [31]. Sampling is typically performed using DDIM [31]. We write $\mathrm{DDIM}_\eta(x_t, t)$ for the DDIM update from x_t to x_{t-1} with stochasticity parameter $\eta \in [0, 1]$, given by

$$x_{t-1} = \sqrt{\bar{\alpha}_{t-1}}\, \hat{x}_0(x_t, t) + \sqrt{1 - \bar{\alpha}_{t-1} - \sigma_t^2(\eta)}\, \epsilon_\theta(x_t, t) + \sigma_t(\eta)\, \xi_t, \qquad \xi_t \sim \mathcal{N}(0, I), \tag{3}$$

where $\sigma_t^2(\eta) = \eta^2 \cdot \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\big(1 - \frac{\bar{\alpha}_t}{\bar{\alpha}_{t-1}}\big)$ controls the scheduler-scaled noise injection. Setting $\eta = 0$ recovers the deterministic update in which the only randomness arises from the initial noise $x_T \sim \mathcal{N}(0, I)$, while $\eta = 1$ recovers the DDPM [1] ancestral sampler.

Reward-Tilted Sampling via Particle Methods. Let $r : \mathcal{X} \to \mathbb{R}$ denote a reward function. Inference-time alignment aims to sample from the reward-tilted distribution $p^*(x_0) \propto p_\theta(x_0) \exp(\lambda r(x_0))$, where p_θ is the pretrained diffusion model and $\lambda \geq 0$ controls alignment strength. Since $r(x_0)$ is inaccessible at intermediate noisy states x_t, it is usually approximated via the Tweedie estimate $\hat{x}_0(x_t, t)$, making direct optimization tractable. Particle methods approximate p^* using K weighted particles $\{(x_t^{(i)}, w_t^{(i)})\}_{i=1}^K$ that alternate between selection and mutation. At each resampling step $t \in \mathcal{T}$, weights are updated using a potential $G_t(x_t^{(i)})$, typically defined as a function of rewards computed via the Tweedie proxy: $w_t^{(i)} \propto w_{t+1}^{(i)}\, G_t(x_t^{(i)})$. Particles are then propagated via DDIM, and the empirical measure $\hat{\pi}_t^K = K^{-1} \sum_{i=1}^K \delta_{x_t^{(i)}}$ (where δ_x denotes the Dirac delta at x)

converges to p^* as $K \to \infty$ [30]. To mitigate weight degeneracy, resampling periodically replaces low-weight particles with copies of high-weight ones. However, poorly designed resampling destroys particle diversity, rendering the approximation uninformative.

Resampling: Multinomial vs. Systematic. A resampling step replaces the current particle set by offspring counts $(N_t^{(1)}, \dots, N_t^{(K)})$ drawn with respect to positive masses $m_t^{(1)}, \dots, m_t^{(K)}$ with $\sum_k m_t^{(k)} = K$, where $m_t^{(k)}$ encodes how many offspring particle k should receive on average. The method is *unbiased* if $\mathbb{E}[N_t^{(k)}] = m_t^{(k)}$ and $\sum_k N_t^{(k)} = K$ almost surely, in which case assigning offspring weight $\tilde{w}_t^{(k)} = w_t^{(k)}/m_t^{(k)}$ preserves the weighted empirical measure in expectation (Proposition 5, Appendix B.2). *Multinomial resampling* draws

$$(N_t^{(1)}, \dots, N_t^{(K)}) \sim \mathrm{Multinomial}(K;\, m_t^{(1)}/K, \dots, m_t^{(K)}/K), \tag{4}$$

giving $\mathrm{Var}(N_t^{(k)}) = m_t^{(k)}(1 - m_t^{(k)}/K)$. Under aggressive reward tilting, a dominant particle with $m_t^{(k)} = \Theta(K)$ exhibits offspring-count variance $\mathcal{O}(K)$, directly inflating the variance of any downstream estimator [21, 32, 22]. *Systematic resampling* [32, 33, 22] realizes the same masses using a single shared random shift, reducing this to $\mathcal{O}(1)$ per particle — a property we exploit in VASR and derive in Section 4.

4 Variance-Aware Systematic Resampling

Algorithm 1 VASR / VASR-MAX: Variance-Aware Systematic Resampling

Require: Frozen model p_θ, reward r, particle count K, alignment strength λ, schedule $\mathcal{T} \subset \{1, \dots, T\}$, mode $\in \{\text{VASR}, \text{VASR-MAX}\}$

$x_T^{(i)} \sim \mathcal{N}(0, I)$, $w_T^{(i)} \leftarrow 1$ for $i = 1, \dots, K$
for $t = T, T-1, \dots, 1$ **do**
  if $t \in \mathcal{T}$ **then**
    $\tilde{r}_i \leftarrow r(\hat{x}_0(x_t^{(i)}, t))$; optionally z-score: $\tilde{r}_i \leftarrow (\tilde{r}_i - \bar{r}_t)/(\sigma_t + \varepsilon)$
    if mode $=$ VASR **then**
      $m_t^{(i)} \leftarrow K \cdot \dfrac{w_t^{(i)} \exp(\tilde{r}_i)}{\sum_j w_t^{(j)} \exp(\tilde{r}_j)}, \quad \tilde{w}_t^{(i)} \leftarrow w_t^{(i)}/m_t^{(i)}$
      Reorder by descending $w_t^{(i)} \exp(\tilde{r}_i)/m_t^{(i)}$; draw $U \sim \mathrm{Uniform}(0, 1)$; set $N_t^{(i)}$ via (6)
    else
      $m_t^{(i)} \leftarrow K \cdot \dfrac{w_t^{(i)} \exp(\tilde{r}_i)}{\sum_j w_t^{(j)} \exp(\tilde{r}_j)}, \quad \tilde{w}_t^{(i)} \leftarrow w_t^{(i)}/m_t^{(i)}$
      Reorder by descending $w_t^{(i)} \exp(\tilde{r}_i)/m_t^{(i)}$; draw $U \sim \mathrm{Uniform}(0, 1)$; set $N_t^{(i)}$ via (6)
      $i^\star \leftarrow \arg\max_j \; r(\hat{x}_0(x_t^{(j)}, t))$
      For each slot s: if s is an extra copy of its parent, redirect: indices$[s] \leftarrow i^\star$
      $w_t^{(i)} \leftarrow 1$ for all i
    end if
    $w_t^{(i)} \leftarrow \tilde{w}_t^{(i)}$ for surviving offspring
  end if
  $x_{t-1}^{(i)} \leftarrow \mathrm{DDIM}_{\eta=1}(x_t^{(i)}, t)$ for all i
end for
Draw N_{eval} samples from $\{x_0^{(i)}\}$ with probabilities $\propto w_0^{(i)} \exp(r(x_0^{(i)}))$
return Selected samples

Variance-Decomposition Principle. The design of VASR follows from separating the two variance sources introduced by a resampling step. Let $\mathcal{F}_t$ denote the current weighted particle set, let $m_t^{(i)}$ be the expected offspring count assigned to particle i, and let $\tilde{w}_t^{(i)} = w_t^{(i)}/m_t^{(i)}$ be the reweighted offspring weight. Write $S(x_t^{(i)})$ and $\sigma_t^2(x_t^{(i)})$ for the conditional mean and variance of the terminal reward score obtained by continuing particle i through the remaining denoising process.

Proposition 1 (Stagewise variance and plug-in allocation)**.** *Conditional on $\mathcal{F}_t$, the terminal estimator variance decomposes as*

$$\mathrm{Var}(\hat{Z}_T \mid \mathcal{F}_t) = V_t^{\mathrm{cont}} + V_t^{\mathrm{res}},$$

where the expected continuation term is

$$\mathbb{E}[V_t^{\mathrm{cont}} \mid \mathcal{F}_t] = \sum_{i=1}^{K} \frac{(w_t^{(i)})^2}{m_t^{(i)}} \sigma_t^2(x_t^{(i)}). \tag{5}$$

Here V_t^{res} is the variance contribution from the random offspring counts. Using the plug-in scale proxy $\sigma_t(x_t^{(i)}) \propto e^{\tilde{r}_t^{(i)}}$, the minimizer of (5) *over $\{m : \sum_i m_t^{(i)} = K\}$ satisfies $m_t^{(i)} \propto w_t^{(i)} e^{\tilde{r}_t^{(i)}}$.*

Proof. See Appendix B.2. □

VASR evolves K weighted DDIM trajectories ($\eta = 1$) and applies a systematic resampling step at a predefined schedule $\mathcal{T} \subset \{1, \dots, T\}$. At each $t \in \mathcal{T}$, given cumulative masses $M_0 = 0$, $M_k = \sum_{i=1}^{k} m_t^{(i)}$, a single $U \sim \mathrm{Uniform}(0, 1)$ determines all offspring counts via

$$N_t^{(k)} = \sum_{j=1}^{K} \mathbf{1}\{U + j - 1 \in (M_{k-1}, M_k]\}. \tag{6}$$

Proposition 2 (Variance Bounds for Systematic Resampling [22])**.** *Under the systematic resampling rule* (6)*, Douc et al. [22] establish that every particle satisfies*

$$|N_t^{(k)} - m_t^{(k)}| \le 1 \text{ a.s.}, \qquad \mathrm{Var}(N_t^{(k)}) \le \tfrac{1}{4}, \tag{7}$$

regardless of mass allocation. In contrast, multinomial resampling induces variance $\mathrm{Var}(N_t^{(k)}) = m_t^{(k)}(1 - m_t^{(k)}/K) = \mathcal{O}(K)$ for heavy-mass particles [22].

Proof. The bounds follow directly from Douc et al. [22]; we restate them here in our notation for completeness. See Appendix B.1 for the self-contained derivation. □

This represents a per-particle reduction from $\mathcal{O}(K)$ to $\mathcal{O}(1)$ that is strongest precisely when some masses are large, the regime induced by exponentially tilted rewards. Each offspring inherits weight $\tilde{w}_t^{(k)} = w_t^{(k)}/m_t^{(k)}$, and systematic resampling remains unbiased so this reweighting preserves the target weighted measure in expectation (Proposition 5). Together, Propositions 1 and 2 provide a complete variance control strategy: the variance-optimal allocation (8) minimizes V_t^{cont}, while systematic resampling bounds V_t^{res}. The procedure is summarised in Algorithm 1; its two key design choices are described below.

Variance-Optimal Mass Allocation. We score each particle using the Tweedie reward proxy $\hat{r}_t^{(i)} = r(\hat{x}_0(x_t^{(i)}, t))$, optionally z-scored across particles for numerical stability when reward scales vary across steps. The masses are then set by the plug-in continuation-variance criterion in Proposition 1:

$$m_t^{(i)} \;\propto\; w_t^{(i)}\, e^{\tilde{r}_i}, \qquad \sum_i m_t^{(i)} = K, \tag{8}$$

where $\tilde{r}_i$ denotes the standardized score. This allocates offspring in proportion to each particle's weighted reward, concentrating compute on promising trajectories while keeping the total population fixed at K. In high-noise regimes where the goal is to obtain a single high-quality sample, such as text-to-image generation with pretrained reward models, stronger selection can improve robustness to noisy intermediate scores. In this regime we concentrate all offspring on the highest-scoring particle,

$$m_t^{(i)} = K \cdot \mathbf{1}\big\{i = \arg\max_j \; w_t^{(j)} e^{\tilde{r}_j}\big\}, \qquad \tilde{w}_t^{(i)} = 1, \tag{9}$$

making an explicit bias–variance trade-off in favor of stable high-reward continuation. We refer to this variant as VASR-MAX.

Table 1: **Quantitative comparison on MNIST and CIFAR-10.** We report mean ± standard deviation over 5 seeds after 10^6 NFEs. Metrics include FID, MMD, expected reward, and diversity (Div). Best values per column are **underlined**, and values within 5% of the best are highlighted. VASR consistently achieves the best or near-best performance across all settings.

	MNIST				MNIST even/odd				CIFAR-10			
Algorithm	FID↓	MMD↓	log r↑	Div↑	FID↓	MMD↓	log r↑	Div↑	FID↓	MMD↓	log r↑	Div↑
DPS	1.008 (4.173)	0.360 (0.236)	-0.167 (0.125)	0.487 (0.054)	1.471 (2.059)	0.449 (0.144)	-0.186 (0.059)	0.491 (0.007)	0.256 (0.060)	1.297 (0.504)	-0.273 (0.110)	0.531 (0.019)
FKD	0.037 (0.023)	0.163 (0.130)	-0.026 (0.013)	0.457 (0.050)	0.056 (0.022)	0.241 (0.112)	-0.032 (0.013)	0.456 (0.007)	0.241 (0.054)	0.923 (0.294)	-0.181 (0.158)	0.517 (0.025)
TDS	0.101 (0.057)	0.485 (0.390)	-0.013 (0.030)	0.429 (0.053)	0.143 (0.066)	0.785 (0.430)	-0.018 (0.032)	0.430 (0.010)	0.418 (0.175)	2.100 (1.221)	-0.043 (0.035)	0.486 (0.036)
DAS	0.030 (0.015)	0.120 (0.076)	-0.023 (0.012)	0.456 (0.052)	0.037 (0.010)	0.159 (0.072)	-0.027 (0.009)	0.457 (0.010)	0.205 (0.043)	0.738 (0.199)	-0.537 (0.236)	0.532 (0.021)
DTS	0.019 (0.002)	0.100 (0.016)	-0.020 (0.001)	0.494 (0.003)	0.014 (0.004)	0.075 (0.029)	-0.015 (0.004)	0.616 (0.053)	0.180 (0.036)	0.744 (0.183)	-0.301 (0.098)	0.540 (0.019)
VASR (ours)	0.015 (0.005)	0.087 (0.043)	-0.020 (0.002)	0.495 (0.092)	0.014 (0.001)	0.084 (0.010)	-0.016 (0.001)	0.616 (0.056)	0.132 (0.026)	0.598 (0.153)	-0.205 (0.099)	0.541 (0.019)

Ordering Heuristic. Before drawing the systematic grid we reorder particles in decreasing order of $w_t^{(i)} e^{\tilde{r}_i} / m_t^{(i)}$. Under (8) this ratio is approximately constant across particles, so neighboring cells in the cumulative-mass interval tend to have similar descendant scores, further reducing the residual variance V_t^{res} without affecting unbiasedness [22]. We verify this empirically in Appendix A.

5 Experiments

We present the main empirical results, evaluating VASR across two settings of increasing complexity. We first consider class-conditional posterior sampling on MNIST and CIFAR-10 using VASR, and then large-scale text-to-image generation with Stable Diffusion, where we use the VASR-MAX variant (Section 4). Unless stated otherwise, results are averaged over five random seeds, with $\lambda = 1.0$ for all methods. We conduct all experiments on 4 NVIDIA GH200 GPUs.

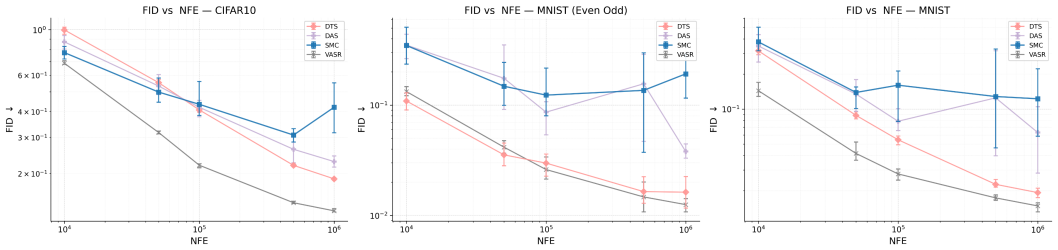

Figure 1: **Scaling behavior with increasing compute (NFEs).** FID vs NFEs on CIFAR-10 and MNIST. VASR outperforms FKD and DTS across all compute budgets, demonstrating favorable scaling with full parallelism.

Posterior Sampling under Class Conditioning. We evaluate class-conditional posterior sampling, where the goal is to sample from $p(\mathbf{x} \mid c) \propto p_\theta(\mathbf{x})\, p(c \mid \mathbf{x})$, with $p_\theta(\mathbf{x})$ an unconditional diffusion prior and $p(c \mid \mathbf{x})$ a pretrained classifier. Following [16], we consider MNIST and CIFAR-10 across all 10 classes, using the log-classifier likelihood as the reward, $r(\mathbf{x}) = \log p(c \mid \mathbf{x})$, and report results averaged over classes and seeds in Table 1. For MNIST, we also study a multimodal setting with even digits $\mathcal{C}_{\text{even}}$ and odd digits $\mathcal{C}_{\text{odd}}$, using $r(\mathbf{x}) = \log \max_{c \in \mathcal{C}} p(c \mid \mathbf{x})$, with results averaged over both groups.

We compare VASR against DTS [16], FKD [21], TDS [28], DAS [34], and DPS [15]. Table 1 reports mean and standard deviation over five seeds at 10^6 NFEs, with metrics computed on 5000 samples. For each method, we report the best performance under reward-weighted or uniform final sampling. Figure 1 shows FID versus NFEs to study scaling with compute; DPS and TDS are excluded due to high cross-seed variance that obscures comparison with stronger baselines.

Results. On single-class MNIST, VASR achieves the lowest FID and MMD while maintaining competitive reward, better diversity than FKD, and competitive diversity with DTS. This trend holds in the multimodal MNIST (even/odd) setting and extends to CIFAR-10, where VASR again attains lower FID and MMD with competitive reward and diversity. Although TDS achieves high rewards, Figure 9 shows this is due to mode collapse rather than true alignment with the target posterior.

Figure 1 shows that VASR scales consistently with compute, outperforming FKD and DTS at higher NFE budgets while remaining approximately $66\times$ faster than DTS. This is notable since particle-based methods typically exhibit noisy scaling behavior [16] (see Figures 1 and 2). We attribute this to variance-optimal mass allocation and systematic resampling, which bound offspring-count deviations by $|N_t^{(k)} - m_t^{(k)}| \leq 1$ and avoid the $\mathcal{O}(K)$ variance of multinomial resampling, improving coverage of the twisted distribution.

Qualitative analysis of CIFAR-10 results (class: car) in Figure 9 at 10^6 NFEs shows that FKD and TDS produce near-identical samples, indicating severe collapse, while VASR and DTS preserve substantially higher diversity. In contrast, DPS generates samples outside the base model support, degrading quality despite high diversity, consistent with prior work [16].

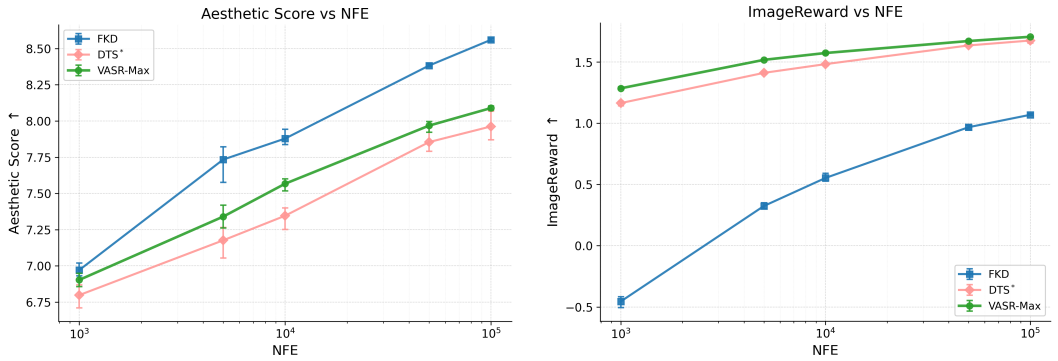

Figure 2: **Reward scaling with compute in text-to-image generation.** (Left) Aesthetic score on the simple animals benchmark; (right) ImageReward on DrawBench. Methods are evaluated under matched NFE budgets. VASR-MAX achieves competitive rewards while avoiding over-optimization and maintaining better efficiency than DTS.

Text-to-Image Generation. In this setting, we study inference-time alignment for text-to-image generation, aiming to improve prompt adherence and perceptual quality without modifying the diffusion model. This is challenging due to the high-dimensional output space and mismatch between reward models and the data distribution. As discussed in Section 4, rewards are computed on the Tweedie proxy $\hat{x}_0(x_t, t)$ of noisy latents, and pretrained SD1.5 reward models produce substantially noisier signals than the log-classifier likelihoods used in pixel-space benchmarks. We therefore report results using VASR-MAX.

We evaluate VASR-MAX against FKD (strongest SMC baseline) and DTS* (value-based baseline) using Stable Diffusion v1.5 [4] as the generative prior $p_\theta(x \mid y)$. Following [16], we use two benchmarks: **(1) DrawBench** [35], evaluated with ImageReward [36] capturing prompt alignment and human preference; and **(2) Aesthetic Optimization** [9], using 45 animal-category prompts and the LAION Aesthetic Predictor [37] to measure visual quality independent of prompt correctness. All methods are evaluated under matched compute budgets with fixed NFEs per prompt.

Results. We report average reward across prompts and 3 seeds, selecting the highest-reward sample per prompt for each method, and analyze scaling in Figure 2. On the aesthetic benchmark [9] (Figure 2, left), FKD achieves the highest raw rewards but exhibits clear over-fitting, producing samples that deviate from the data distribution (see Figure 7). In contrast, VASR-MAX and DTS* obtain slightly lower rewards while preserving substantially better visual fidelity, with VASR-MAX yielding a better reward scaling. On DrawBench (Figure 2, right), VASR-MAX consistently outperforms FKD and DTS* across all compute budgets, achieving up to $1.6\times$ higher rewards than FKD while remaining competitive with DTS* despite being approximately $4\times$ faster. Qualitative results in Appendix Figures 7 and 8 confirm these trends: FKD produces artifact-prone or over-optimized samples, whereas VASR-MAX better preserves alignment with both the prompt and the base model distribution.

We attribute FKD's over-optimization and VASR-MAX's superior reward scaling to differences in how they handle noisy reward signals. Tweedie proxy rewards are inherently stochastic at high noise

levels, with intermediate scores $\hat{x}_0(x_t, t)$ exhibiting substantial noise even under a perfect reward model (§ A). VASR-MAX mitigates this by concentrating offspring on the highest-scoring particle, making selection robust to pointwise noise rather than propagating it through soft allocations, which prevents the accumulation of reward estimation errors.

6 Discussion

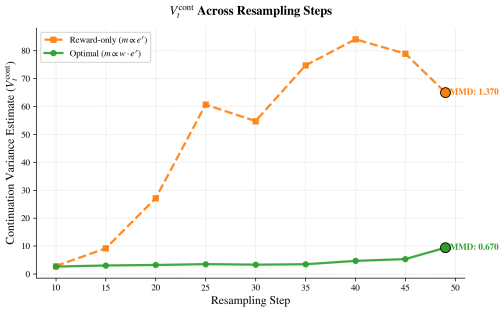

Figure 3: V_t^{cont} **under different mass allocations.** The variance-optimal allocation $m \propto w \cdot e^r$ maintains substantially lower V_t^{cont} than the baseline $m \propto e^r$ on CIFAR-10 (K=1000), yielding nearly 2× better terminal MMD.

Continuation Variance Control. To empirically validate the variance-optimal mass allocation from Proposition 1, we compare our allocation $m_t^{(k)} \propto w_t^{(k)} e^{\tilde{r}_k}$ with the baseline $m_t^{(k)} \propto e^{\tilde{r}_k}$ used in prior SMC methods [21]. Figure 3 shows results on CIFAR-10 with K=1000 particles. The variance-optimal allocation reduces the continuation variance proxy by approximately 8× and improves terminal MMD by nearly 2×, confirming that the allocation directly controls V_t^{cont} as predicted by the variance decomposition. We further see that lower continuation variance corresponds to better terminal sample quality, validating our prediction that controlling V_t^{cont} (via optimal allocation) is essential for high-quality reward-guided sampling.

Lineage Diversity and Sample Quality. To validate the role of lineage diversity in sample quality, we ablate the temperature parameter λ under systematic resampling. As shown in Figure 4, increasing λ sharpens selection by concentrating mass on higher-reward particles, improving exploitation but reducing lineage diversity and degrading distributional metrics. On CIFAR-10, increasing λ from 1.0 to 15.0 reduces mean lineage diversity from ≈ 1500 to ≈ 1000, while FID worsens from ≈ 0.40 to ≈ 1.65. Crucially, when lineage diversity under systematic resampling approaches that of multinomial resampling, FID degrades to a similar level, indicating that the performance gap is primarily driven by preserved lineage diversity. Thus, VASR's advantage stems from maintaining a more diverse set of ancestral trajectories, enabling better coverage of the reward-tilted distribution.

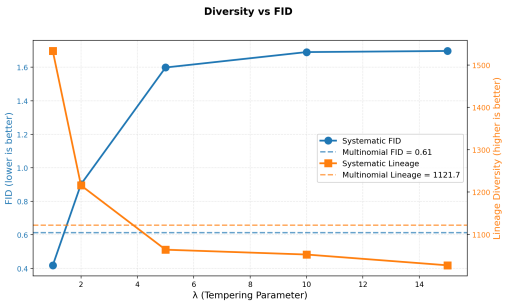

Figure 4: Temperature λ ablation. Higher λ reduces lineage diversity and degrades FID, approximately matching multinomial's performance at λ=4.0.

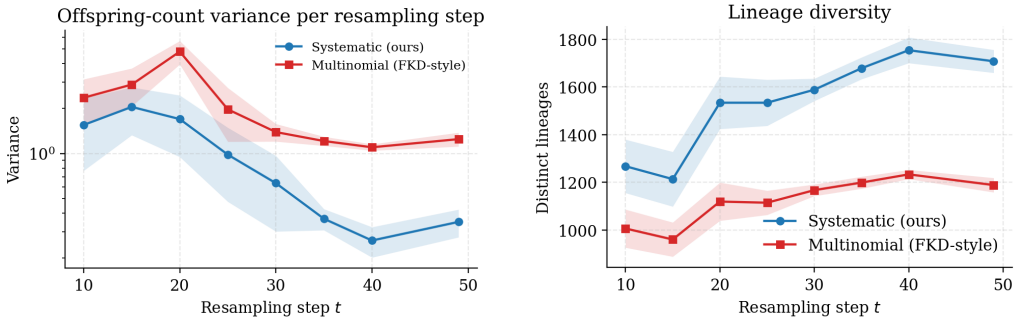

Figure 5: **Systematic vs. Multinomial resampling. Left:** Offspring-count variance. Systematic maintains low variance while multinomial exhibits large fluctuations. **Right:** Lineage diversity. Systematic retains more distinct lineages, avoiding multinomial's rapid collapse.

Offspring-Count Variance Analysis. A central design goal of VASR is controlling residual variance V_t^{res} from offspring sampling while maintaining selection pressure. To isolate the resampling rule's effect on V_t^{res}, we compare systematic vs. multinomial resampling under identical mass allocations $m_t^{(k)} \propto w_t^{(k)} e^{\tilde{r}_k}$, where multinomial draws $(N_t^{(1)}, \ldots, N_t^{(K)}) \sim \text{Multinomial}(K; m_t^{(1)}/K, \ldots, m_t^{(K)}/K)$ and systematic uses (6). Figure 5 reports offspring-count variance $\text{Var}(N_t^{(k)})$—the primary contributor to V_t^{res}—averaged across particles, and lineage diversity (the number of distinct particles from step $t{+}1$ that produced at least one offspring at step t). Systematic resampling achieves lower V_t^{res} and retains significantly more lineages than multinomial. This confirms Proposition 2: multinomial induces $\text{Var}(N_t^{(k)}) = m_t^{(k)}(1 - m_t^{(k)}/K) = \mathcal{O}(K)$ for heavy-mass particles, whereas systematic enforces $|N_t^{(k)} - m_t^{(k)}| \leq 1$ almost surely with $\text{Var}(N_t^{(k)}) \leq 1/4$ per particle regardless of mass [22]. The lineage gap is pronounced under aggressive tilting, where multinomial's large fluctuations in offspring counts increase V_t^{res}, causing only a few particles to reproduce and collapsing diversity. All results are reported for $K{=}2000$ and $\lambda{=}1.0$, averaged across CIFAR-10 classes over 3 random seeds.

Inference Efficiency. Inference-time efficiency is critical in practice, as runtime overhead can offset gains in sample quality. Value-based methods such as DTS [16] build a search tree over the denoising trajectory, which is inherently sequential since node expansions depend on prior value estimates, limiting parallelization and increasing runtime, especially at large NFE budgets. In contrast, VASR evolves K particles largely independently and is highly parallelizable. The systematic resampling step is $\mathcal{O}(K)$—requiring only a cumulative-mass computation and a single uniform draw—and is negligible compared to UNet evaluations, enabling efficient parallel execution.

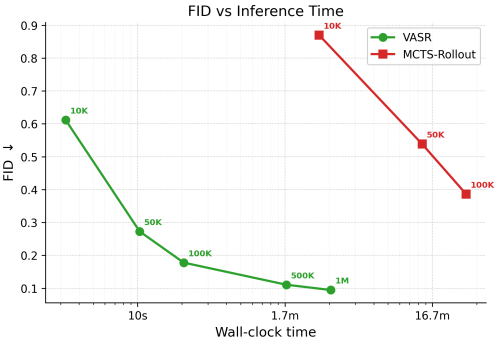

Figure 6: **FID vs. time on CIFAR-10.** VASR achieves lower FID at all budgets and is $\sim 66\times$ faster than DTS at matched NFEs; DTS fails to match even the worst VASR FID.

Figure 6 shows FID versus wall-clock time on CIFAR-10 (class: car). VASR consistently achieves lower FID than DTS across all budgets and is approximately $66\times$ faster at matched NFE. Even at its maximum runtime, DTS fails to match the best FID of VASR, highlighting its higher computational cost despite amortization via tree caching. This underscores the efficiency and parallelization advantages of particle-based inference.

7 Conclusions & Limitations

We introduce a variance-decomposition framework for reward-guided diffusion SMC that separates estimator variance into continuation variance V_t^{cont} (from particle propagation) and residual variance V_t^{res} (from offspring sampling), motivating VASR. It jointly controls both via a variance-optimal mass allocation $m_t^{(k)} \propto w_t^{(k)} e^{\tilde{r}_k}$ that minimizes V_t^{cont}, and systematic resampling that enforces $|N_t^{(k)} - m_t^{(k)}| \leq 1$ to bound V_t^{res}. Ablations show this reduces continuation variance by $\approx 8\times$, improves terminal MMD by nearly $2\times$, and preserves $\approx 1.5\times$ more ancestral lineages than multinomial resampling. Across MNIST, CIFAR-10, and Stable Diffusion, VASR achieves stronger distributional quality while being $\sim$ 4–$66\times$ faster than tree-search methods at matched compute. In high-noise latent diffusion settings, intermediate Tweedie reward estimates are inherently noisy, motivating VASR-MAX, which concentrates offspring on the highest-scoring particle at each step, trading unbiasedness for robustness. While effective, this sacrifices theoretical guarantees, suggesting that better denoising of intermediate rewards could enable fully unbiased variance-optimal allocation. Additional limitations include lack of evaluation on flow matching and consistency models, restriction to single-objective alignment, and absence of multi-objective and multi-modal reward analysis, which remain promising directions for future work.

Acknowledgments and Disclosure of Funding

This work was supported in part by NSF grants 2416897, 2505932, and 2601666 and ONR grant N000142512318. Computing resources were provided by the National Center for Supercomputing Applications (NCSA) through Delta (NSF OAC-2005572) and DeltaAI (NSF OAC-2320345), as well as Illinois Computes and the NAIRR Pilot (NAIRR250157).

References

[1] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models, 2020. URL https://arxiv.org/abs/2006.11239.

[2] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution, 2020. URL https://arxiv.org/abs/1907.05600.

[3] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis, 2023. URL https://arxiv.org/abs/2307.01952.

[4] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2022. URL https://arxiv.org/abs/2112.10752.

[5] Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, Da Yin, Yuxuan Zhang, Weihan Wang, Yean Cheng, Bin Xu, Xiaotao Gu, Yuxiao Dong, and Jie Tang. Cogvideox: Text-to-video diffusion models with an expert transformer, 2025. URL https://arxiv.org/abs/2408.06072.

[6] Google DeepMind. Veo 3. https://aistudio.google.com/models/veo-3, 2025. Accessed: 2026-03-15.

[7] Tiwei Bie, Maosong Cao, Kun Chen, Lun Du, Mingliang Gong, Zhuochen Gong, Yanmei Gu, Jiaqi Hu, Zenan Huang, Zhenzhong Lan, Chengxi Li, Chongxuan Li, Jianguo Li, Zehuan Li, Huabin Liu, Lin Liu, Guoshan Lu, Xiaocheng Lu, Yuxin Ma, Jianfeng Tan, Lanning Wei, Ji-Rong Wen, Yipeng Xing, Xiaolu Zhang, Junbo Zhao, Da Zheng, Jun Zhou, Junlin Zhou, Zhanchao Zhou, Liwang Zhu, and Yihong Zhuang. Llada2.0: Scaling up diffusion language models to 100b, 2025. URL https://arxiv.org/abs/2512.15745.

[8] Subham Sekhar Sahoo, Marianne Arriola, Yair Schiff, Aaron Gokaslan, Edgar Marroquin, Justin T Chiu, Alexander Rush, and Volodymyr Kuleshov. Simple and effective masked diffusion language models, 2024. URL https://arxiv.org/abs/2406.07524.

[9] Kevin Black, Michael Janner, Yilun Du, Ilya Kostrikov, and Sergey Levine. Training diffusion models with reinforcement learning, 2024. URL https://arxiv.org/abs/2305.13301.

[10] Ying Fan, Olivia Watkins, Yuqing Du, Hao Liu, Moonkyung Ryu, Craig Boutilier, Pieter Abbeel, Mohammad Ghavamzadeh, Kangwook Lee, and Kimin Lee. Dpok: Reinforcement learning for fine-tuning text-to-image diffusion models, 2023. URL https://arxiv.org/abs/2305.16381.

[11] Kai Yang, Jian Tao, Jiafei Lyu, Chunjiang Ge, Jiaxin Chen, Qimai Li, Weihan Shen, Xiaolong Zhu, and Xiu Li. Using human feedback to fine-tune diffusion models without any reward model, 2024. URL https://arxiv.org/abs/2311.13231.

[12] Bram Wallace, Meihua Dang, Rafael Rafailov, Linqi Zhou, Aaron Lou, Senthil Purushwalkam, Stefano Ermon, Caiming Xiong, Shafiq Joty, and Nikhil Naik. Diffusion model alignment using direct preference optimization, 2023. URL https://arxiv.org/abs/2311.12908.

[13] Kevin Clark, Paul Vicol, Kevin Swersky, and David J Fleet. Directly fine-tuning diffusion models on differentiable rewards, 2024. URL https://arxiv.org/abs/2309.17400.

[14] Arpit Bansal, Hong-Min Chu, Avi Schwarzschild, Soumyadip Sengupta, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Universal guidance for diffusion models, 2023. URL https://arxiv.org/abs/2302.07121.

[15] Hyungjin Chung, Jeongsol Kim, Michael T. Mccann, Marc L. Klasky, and Jong Chul Ye. Diffusion posterior sampling for general noisy inverse problems, 2024. URL https://arxiv.org/abs/2209.14687.

[16] Vineet Jain, Kusha Sareen, Mohammad Pedramfar, and Siamak Ravanbakhsh. Diffusion tree sampling: Scalable inference-time alignment of diffusion models, 2025. URL https://arxiv.org/abs/2506.20701.

[17] 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, 2024. URL https://arxiv.org/abs/2408.08252.

[18] Xiner Li, Masatoshi Uehara, Xingyu Su, Gabriele Scalia, Tommaso Biancalani, Aviv Regev, Sergey Levine, and Shuiwang Ji. Dynamic search for inference-time alignment in diffusion models, 2025. URL https://arxiv.org/abs/2503.02039.

[19] Gabriel Cardoso, Yazid Janati El Idrissi, Sylvain Le Corff, and Eric Moulines. Monte carlo guided diffusion for bayesian linear inverse problems, 2023. URL https://arxiv.org/abs/2308.07983.

[20] Zehao Dou and Yang Song. Diffusion posterior sampling for linear inverse problem solving: A filtering perspective. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=tplXNcHZs1.

[21] Raghav Singhal, Zachary Horvitz, Ryan Teehan, Mengye Ren, Zhou Yu, Kathleen McKeown, and Rajesh Ranganath. A general framework for inference-time scaling and steering of diffusion models, 2025. URL https://arxiv.org/abs/2501.06848.

[22] Randal Douc, Olivier Cappé, and Eric Moulines. Comparison of resampling schemes for particle filtering, 2005. URL https://arxiv.org/abs/cs/0507025.

[23] Yutong He, Naoki Murata, Chieh-Hsin Lai, Yuhta Takida, Toshimitsu Uesaka, Dongjun Kim, Wei-Hsiang Liao, Yuki Mitsufuji, J. Zico Kolter, Ruslan Salakhutdinov, and Stefano Ermon. Manifold preserving guided diffusion, 2023. URL https://arxiv.org/abs/2311.16424.

[24] Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations, 2021. URL https://arxiv.org/abs/2011.13456.

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

[26] Prafulla Dhariwal and Alex Nichol. Diffusion models beat gans on image synthesis, 2021. URL https://arxiv.org/abs/2105.05233.

[27] Nanye Ma, Shangyuan Tong, Haolin Jia, Hexiang Hu, Yu-Chuan Su, Mingda Zhang, Xuan Yang, Yandong Li, Tommi Jaakkola, Xuhui Jia, and Saining Xie. Inference-time scaling for diffusion models beyond scaling denoising steps, 2025. URL https://arxiv.org/abs/2501.09732.

[28] Luhuan Wu, Brian Trippe, Christian Naesseth, David Blei, and John P Cunningham. Practical and asymptotically exact conditional sampling in diffusion models. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 31372–31403. Curran Associates, Inc., 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/63e8bc7bbf1cfea36d1d1b6538aecce5-Paper-Conference.pdf.

[29] Brian L. Trippe, Jason Yim, Doug Tischer, David Baker, Tamara Broderick, Regina Barzilay, and Tommi Jaakkola. Diffusion probabilistic modeling of protein backbones in 3d for the motif-scaffolding problem, 2023. URL https://arxiv.org/abs/2206.04119.

[30] Pierre Del Moral. *Feynman-Kac Formulae: Genealogical and Interacting Particle Systems With Applications*, volume 100. 05 2004. ISBN 0387202684. doi: 10.1007/978-1-4684-9393-1.

[31] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models, 2022. URL https://arxiv.org/abs/2010.02502.

[32] *Systematic Sampling*, chapter 4, pages 83–120. John Wiley & Sons, Ltd, 2008. ISBN 9780470374597. doi: https://doi.org/10.1002/9780470374597.ch4. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/9780470374597.ch4.

[33] Genshiro Kitagawa. Monte carlo filter and smoother for non-gaussian nonlinear state space models. *Journal of Computational and Graphical Statistics*, 5:1–25, 1996. URL https://api.semanticscholar.org/CorpusID:123545380.

[34] Sunwoo Kim, Minkyu Kim, and Dongmin Park. Test-time alignment of diffusion models without reward over-optimization, 2025. URL https://arxiv.org/abs/2501.05803.

[35] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems*, 35:36479–36494, 2022.

[36] Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. Imagereward: Learning and evaluating human preferences for text-to-image generation, 2023. URL https://arxiv.org/abs/2304.05977.

[37] Zijie J. Wang, Evan Montoya, David Munechika, Haoyang Yang, Benjamin Hoover, and Duen Horng Chau. Large-scale prompt gallery dataset for text-to-image generative models. *arXiv:2210.14896 [cs]*, 2022. URL https://arxiv.org/abs/2210.14896.

[38] Shivanshu Shekhar, Shreyas Singh, and Tong Zhang. See-dpo: Self entropy enhanced direct preference optimization, 2025. URL https://arxiv.org/abs/2411.04712.

A Additional Experiments & Details

For all VASR & VASR-MAX experiments, we use variance-optimal mass allocation ($m_t \propto w_t \cdot e^{r_t}$) with λ=1.0 and the ordering heuristic enabled. Resampling parameters are task-specific: on MNIST and CIFAR-10, we resample every 5 steps from t=10 to t=40; for Stable Diffusion aesthetic score optimization (simple animal prompts), we use frequency 5 from t=20 to t=45; for ImageReward on DrawBench, we resample every 5 steps from t=10 to t=40. For SMC baselines (FKD, TDS, DAS), we found resampling frequency 10 with start and end timesteps $(10, 40)$ to work best across all settings. For MCTS-based methods (DTS), we use default parameters from the original implementation [16], as these yielded the strongest performance in our experiments.

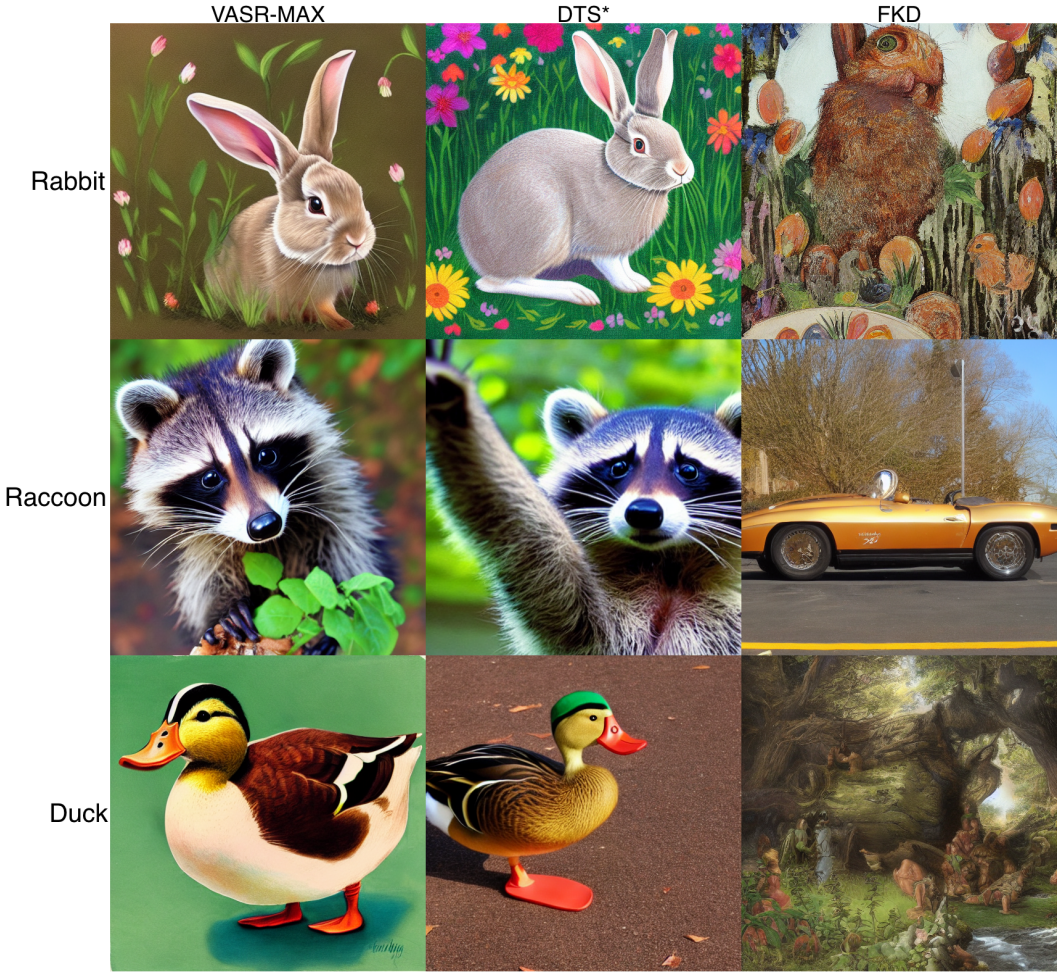

Figure 7: **Qualitative comparison on aesthetic optimization.** Samples from each method for a 100K NFE budget. As emphasized earlier, FKD achieves a high aesthetic score but produces samples that have overfit to the reward model and do not make sense [38], a symptom of over-fitting. VASR-MAX generates visually appealing images that remain faithful to the prompt.

Stochastic Reward Noise at Intermediate Timesteps. A key assumption underlying VASR-MAX is that intermediate reward signals are inherently noisy due to the stochasticity of the decoding process. To verify this, we run K=40 independent particles through the pipeline, and at each resampling timestep t we draw M=10 stochastic rollouts (η=1) from each particle and score every terminal image with ImageReward. Figure 10 plots the per-particle variance $\widehat{\mathrm{Var}}_k$ against mean reward $\bar{r}_k$ at each diagnostic step. Despite mean rewards spanning $[-1.0, -0.2]$, the within-particle variance remains roughly constant at 0.02–0.04 with no systematic dependence on reward rank. This confirms that score differences between particles at intermediate timesteps are comparable in magnitude to



Figure 8: **Qualitative comparison on DrawBench (ImageReward).** Best samples generated by each method for each prompt at a 100K NFE budget. VASR-MAX produces samples with higher prompt alignment and perceptual quality compared to DTS and FKD, consistent with the quantitative results in Figure 2.



Figure 9: **Qualitative comparison on CIFAR-10 (class: car).** Samples generated with 10^6 NFEs, selected via reward-weighted sampling. FKD and TDS exhibit mode collapse with visually similar outputs, while DPS produces out-of-distribution images. VASR and DTS maintain distribution alignment with higher sample diversity.

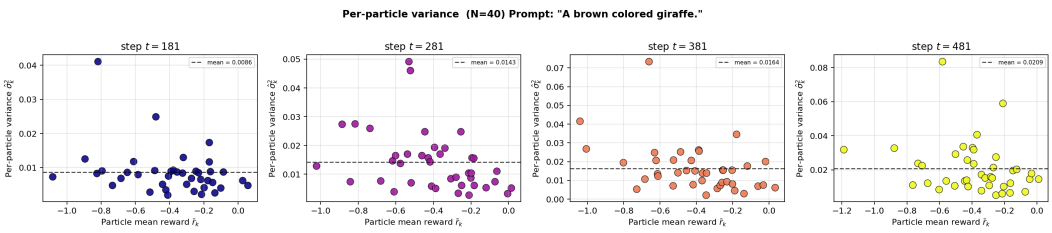

Figure 10: **Per-particle reward variance vs. mean reward** (prompt: "A brown colored giraffe."). Each point is a particle, with the dashed line showing mean variance. Despite variation in mean rewards, within-particle variance remains nearly constant (0.02–0.04), independent of reward rank.

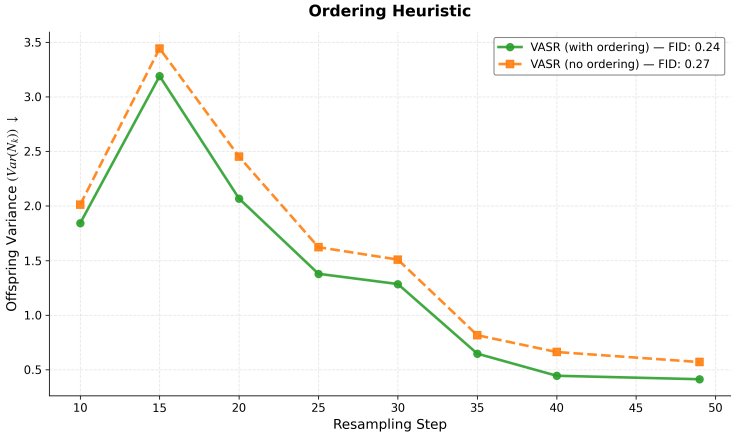

Figure 11: **Effect of ordering heuristic on offspring variance.** Ordering particles before systematic resampling reduces offspring-count variance at each step and improves terminal FID.

the inherent rollout noise, making a soft proportional allocation unreliable. VASR-MAX side-steps this entirely by committing to the argmax particle, whose selection is robust to small additive noise regardless of the absolute score scale.

Ordering Heuristic. Before drawing the systematic grid we reorder particles in decreasing order of $w_t^{(i)} e^{\tilde{r}_i} / m_t^{(i)}$. Under the variance-optimal mass allocation (8), this ratio is approximately constant across particles, meaning that neighboring cells in the cumulative-mass interval correspond to particles with similar weighted reward scores. This spatial correlation tends to reduce fluctuations from the random grid placement beyond the $\mathcal{O}(1)$ guarantee of systematic resampling itself. Importantly, because reordering is a deterministic permutation applied before the random grid is drawn, it preserves the unbiasedness of the systematic rule.

Figure 11 empirically confirms this variance reduction on CIFAR-10: ordering substantially reduces mean offspring-count variance at each resampling step while maintaining the bounded variance guarantee. This improved variance control translates directly to better sample quality, with terminal FID improving from 0.27 (no ordering) to 0.24 (with ordering). These results demonstrate that the ordering heuristic is a simple yet effective mechanism for enhancing both the statistical efficiency and distributional accuracy of variance-aware systematic resampling.

B Theoretical Results

This section collects the theoretical results referenced from the main text. Section B.1 treats the offspring-count variance properties of multinomial and systematic resampling (referenced from Section 3). Section B.2 derives the variance-optimal mass allocation, the unbiasedness guarantee, and the stagewise variance decomposition that motivate the design of VASR (referenced from Section 4).

B.1 Resampling Variance Bounds

Throughout this section we work with positive masses $m_t^{(1)}, \dots, m_t^{(K)}$ summing to K, and offspring counts $(N_t^{(1)}, \dots, N_t^{(K)})$ drawn under either the multinomial scheme of (4) or the systematic scheme of (6).

B.1.1 Multinomial Offspring-Count Variance

Proposition 3 (Multinomial offspring-count variance [22])**.** *Let* $(N_t^{(1)}, \dots, N_t^{(K)}) \sim \mathrm{Multinomial}(K; p_1, \dots, p_K)$ *with* $p_k = m_t^{(k)}/K$ *and* $\sum_k m_t^{(k)} = K$*. Then for every* $k \in \{1, \dots, K\}$*,*

$$\mathbb{E}[N_t^{(k)}] = m_t^{(k)}, \qquad \mathrm{Var}(N_t^{(k)}) = m_t^{(k)} \left(1 - \frac{m_t^{(k)}}{K}\right). \tag{10}$$

In particular, if $m_t^{(k)} = cK$ *for some constant* $c \in (0, 1)$*, then* $\mathrm{Var}(N_t^{(k)}) = cK(1-c) = \Theta(K)$*.*

Proof. Each marginal $N_t^{(k)}$ is the number of successes in K i.i.d. Bernoulli trials with success probability $p_k = m_t^{(k)}/K$, and is therefore distributed as $\mathrm{Binomial}(K, p_k)$. The mean and variance of a binomial are

$$\mathbb{E}[N_t^{(k)}] = Kp_k = m_t^{(k)}, \qquad \mathrm{Var}(N_t^{(k)}) = Kp_k(1-p_k) = m_t^{(k)}\big(1 - m_t^{(k)}/K\big).$$

The scaling claim follows by substituting $m_t^{(k)} = cK$. □

B.1.2 Systematic Offspring-Count Bounds

Proposition 4 (Systematic resampling: deviation and variance [22])**.** *Let* $m_t^{(1)}, \dots, m_t^{(K)}$ *be positive masses summing to* K*, with cumulative masses* $M_k = \sum_{i=1}^{k} m_t^{(i)}$ *and* $M_0 = 0$*. Let* $U \sim \mathrm{Uniform}(0, 1)$ *and define the systematic offspring counts*

$$N_t^{(k)} = \sum_{j=1}^{K} \mathbf{1}\big\{U + j - 1 \in (M_{k-1}, M_k]\big\}, \qquad k = 1, \dots, K.$$

Then:

- *(i) (Unbiasedness)* $\mathbb{E}[N_t^{(k)}] = m_t^{(k)}$ *and* $\sum_{k=1}^{K} N_t^{(k)} = K$ *almost surely.*
- *(ii) (Bounded deviation)* $|N_t^{(k)} - m_t^{(k)}| \le 1$ *almost surely.*
- *(iii) (Variance bound)* $\mathrm{Var}(N_t^{(k)}) = \{m_t^{(k)}\}\big(1 - \{m_t^{(k)}\}\big) \le \frac{1}{4}$*, where* $\{x\} = x - \lfloor x \rfloor$ *denotes the fractional part.*

Proof. The bounds in (i) and (ii) follow from Douc et al. [22]; we restate them here in our notation for completeness, and extend to the explicit variance formula in (iii).

Let $S_k = (M_{k-1}, M_k]$ denote the interval of length $m_t^{(k)}$ allocated to particle k on the line $(0, K]$. The point pattern $\{U + j - 1 : j = 1, \dots, K\}$ consists of K equally spaced points with unit gap, with the first point uniformly distributed in $(0, 1]$.

(i) Unbiasedness. Conditional on U, the indicator $\mathbf{1}\{U + j - 1 \in S_k\}$ is deterministic, and

$$\mathbb{E}[N_t^{(k)}] = \int_0^1 \sum_{j=1}^{K} \mathbf{1}\{u + j - 1 \in S_k\}\, du = \int_0^K \mathbf{1}\{v \in S_k\}\, dv = |S_k| = m_t^{(k)},$$

where we changed variables $v = u + j - 1$ and used the fact that the K unit-length intervals $(j-1, j]$ for $j = 1, \dots, K$ partition $(0, K]$. The almost-sure conservation $\sum_k N_t^{(k)} = K$ holds because every one of the K points falls in exactly one S_k.

(ii) Bounded deviation. Since the K points are equally spaced with gap one, the number of points falling in any half-open interval S_k of length $m_t^{(k)}$ is either $\lfloor m_t^{(k)} \rfloor$ or $\lceil m_t^{(k)} \rceil$:

- If $m_t^{(k)}$ is an integer, then exactly $m_t^{(k)}$ points fall in S_k for every value of U, so $N_t^{(k)} = m_t^{(k)}$ almost surely.
- If $m_t^{(k)}$ is not an integer, write $m_t^{(k)} = \lfloor m_t^{(k)} \rfloor + \{m_t^{(k)}\}$. The number of unit-length sub-intervals that fit fully inside S_k is $\lfloor m_t^{(k)} \rfloor$, and the residual sub-interval of length $\{m_t^{(k)}\}$ may or may not contain a point depending on the offset U. So $N_t^{(k)} \in \{\lfloor m_t^{(k)} \rfloor, \lceil m_t^{(k)} \rceil\}$.

In both cases $|N_t^{(k)} - m_t^{(k)}| \leq 1$ almost surely.

(iii) Variance bound. If $m_t^{(k)}$ is an integer, $N_t^{(k)} = m_t^{(k)}$ almost surely and the variance is zero. Otherwise let $\phi = \{m_t^{(k)}\} \in (0, 1)$. From (ii), $N_t^{(k)}$ takes only two values, $\lfloor m_t^{(k)} \rfloor$ and $\lfloor m_t^{(k)} \rfloor + 1$. By (i), $\mathbb{E}[N_t^{(k)}] = m_t^{(k)} = \lfloor m_t^{(k)} \rfloor + \phi$, so

$$\Pr[N_t^{(k)} = \lceil m_t^{(k)} \rceil] = \phi, \qquad \Pr[N_t^{(k)} = \lfloor m_t^{(k)} \rfloor] = 1 - \phi.$$

Therefore $N_t^{(k)} - \lfloor m_t^{(k)} \rfloor \sim \text{Bernoulli}(\phi)$, and

$$\text{Var}(N_t^{(k)}) = \phi(1 - \phi) = \{m_t^{(k)}\}\big(1 - \{m_t^{(k)}\}\big) \leq \tfrac{1}{4},$$

with equality at $\phi = 1/2$. □

Remark 1. *The variance bound in Proposition 4(iii) is independent of K and of $m_t^{(k)}$ itself — it depends only on the fractional part $\{m_t^{(k)}\}$. Combined with the multinomial variance in Proposition 3, this gives the per-particle improvement from $\mathcal{O}(K)$ to $\mathcal{O}(1)$ stated in the main text [22].*

B.2 VASR Mass Allocation and Variance Decomposition

This section derives the unbiasedness guarantee, the variance-optimal mass allocation, and the stagewise variance decomposition that justify the design of VASR. Throughout we write $\tilde{r}_t^{(i)} := r(\hat{x}_0(x_t^{(i)}, t))$ for the standardized Tweedie reward proxy evaluated at particle i, and $\mathcal{F}_t$ for the sigma-algebra generated by the particle history up to time t.

B.2.1 Unbiasedness of the Reweighted Empirical Measure

Proposition 5 (VASR preserves the weighted empirical measure)**.** *Let the particle set $\{(x_t^{(i)}, w_t^{(i)})\}_{i=1}^K$ be propagated through one resampling step at time $t \in \mathcal{T}$ with masses $m_t^{(i)}$ and offspring counts $N_t^{(i)}$ satisfying $\mathbb{E}[N_t^{(i)}] = m_t^{(i)}$ and $\sum_i N_t^{(i)} = K$ almost surely. If each offspring of particle i is assigned weight $\tilde{w}_t^{(i)} = w_t^{(i)}/m_t^{(i)}$, then for every bounded measurable test function f on $\mathcal{X}$,*

$$\mathbb{E}\left[\sum_{i=1}^K N_t^{(i)} \tilde{w}_t^{(i)} f(x_t^{(i)}) \,\middle|\, \mathcal{F}_t\right] = \sum_{i=1}^K w_t^{(i)} f(x_t^{(i)}).$$

Proof. Conditional on $\mathcal{F}_t$, the particles $x_t^{(i)}$ and masses $m_t^{(i)}$ are deterministic. By linearity of expectation and the unbiasedness condition $\mathbb{E}[N_t^{(i)} \mid \mathcal{F}_t] = m_t^{(i)}$,

$$\mathbb{E}\left[\sum_{i=1}^K N_t^{(i)} \tilde{w}_t^{(i)} f(x_t^{(i)}) \,\middle|\, \mathcal{F}_t\right] = \sum_{i=1}^K \frac{w_t^{(i)}}{m_t^{(i)}} f(x_t^{(i)})\, m_t^{(i)} = \sum_{i=1}^K w_t^{(i)} f(x_t^{(i)}).$$ □

Remark 2. *Proposition 5 holds for any unbiased resampling scheme. It is the foundation of* VASR*: the choice of resampling rule affects only the variance of the estimator, not its expectation.* VASR-MAX *uses the spike allocation* (9)*, which assigns zero mass to all but one particle and therefore violates the positivity assumption $m_t^{(i)} > 0$; the unbiasedness guarantee does not apply, and* VASR-MAX *is a deliberately biased estimator chosen for robustness to noisy reward signals.*

B.2.2 Variance-Optimal Mass Allocation

At each $t \in \mathcal{T}$, VASR sets masses as

$$m_t^{(i)} = K \cdot \frac{w_t^{(i)}\, e^{\tilde{r}_t^{(i)}}}{\sum_j w_t^{(j)}\, e^{\tilde{r}_t^{(j)}}}, \tag{11}$$

where $\tilde{r}_t^{(i)}$ is the optionally standardized Tweedie reward proxy. We derive this allocation from a plug-in continuation-variance criterion in which the proxy captures the scale of each particle's future terminal score.

Let $S(x_t^{(i)}) := \mathbb{E}[e^{r(X_T)} \mid x_t^{(i)}]$ denote the conditional terminal score of particle i, and $\sigma_t^2(x_t^{(i)}) := \mathrm{Var}(e^{r(X_T)} \mid x_t^{(i)})$ its conditional variance under stochastic DDIM propagation (η=1). After resampling, each of the $N_t^{(i)}$ offspring of particle i is propagated independently with weight $\tilde{w}_t^{(i)} = w_t^{(i)}/m_t^{(i)}$. Taking the expectation of the continuation variance with respect to the unbiased offspring counts gives

$$\mathbb{E}[V_t^{\mathrm{cont}} \mid \mathcal{F}_t] = \sum_{i=1}^{K} \frac{(w_t^{(i)})^2}{m_t^{(i)}}\, \sigma_t^2(x_t^{(i)}). \tag{12}$$

Proposition 6 (Variance-optimal mass allocation)**.** *Let $a_i > 0$ denote a proxy for the continuation standard deviation $\sigma_t(x_t^{(i)})$, and suppose $a_i \propto e^{\tilde{r}_t^{(i)}}$. Then the minimizer of the surrogate*

$$\sum_{i=1}^{K} \frac{(w_t^{(i)})^2 a_i^2}{m_t^{(i)}}$$

over the simplex $\{m \in \mathbb{R}^K_{>0} : \sum_i m_t^{(i)} = K\}$ is

$$m_t^{(i),\star} \propto w_t^{(i)}\, e^{\tilde{r}_t^{(i)}},$$

which is exactly the allocation used by VASR (8)*.*

Proof. Define $b_i := w_t^{(i)} a_i > 0$ and minimize $\sum_i b_i^2/m_i$ subject to $\sum_i m_i = K$ via Lagrange multipliers:

$$\mathcal{L}(m, \mu) = \sum_i \frac{b_i^2}{m_i} + \mu \left(\sum_i m_i - K \right).$$

The stationarity condition $\partial \mathcal{L}/\partial m_i = 0$ gives

$$-\frac{b_i^2}{m_i^2} + \mu = 0, \qquad m_i^\star = \frac{b_i}{\sqrt{\mu}}.$$

Thus $m_i^\star \propto b_i = w_t^{(i)} a_i$. If $a_i \propto e^{\tilde{r}_t^{(i)}}$, then $m_i^\star \propto w_t^{(i)} e^{\tilde{r}_t^{(i)}}$. The objective is strictly convex in m on the positive orthant, so this stationary point is the unique global minimum on the simplex. □

Remark 3. *The reward-weighted allocation* (8) *uses $e^{\tilde{r}_t^{(i)}}$ as a tractable plug-in estimate of future terminal-score scale. If the absolute continuation variance is locally constant, $\sigma_t^2(x_t^{(i)}) \approx \tau_t^2$, the exact minimizer of* (12) *reduces to the weight-only allocation $m_i^\star \propto w_t^{(i)}$. In high-noise regimes where the plug-in score scale is less reliable,* VASR-MAX *provides the complementary robust selection rule used in our text-to-image experiments.*

B.2.3 Stagewise Variance Decomposition

Proposition 7 (Stagewise variance decomposition)**.** *The conditional variance of the terminal estimator at step t admits the decomposition*

$$\mathrm{Var}(\hat{Z}_T \mid \mathcal{F}_t) = V_t^{\mathrm{cont}} + V_t^{\mathrm{res}},$$

where V_t^{cont} is defined in (12) *and the resampling term is the full quadratic form*

$$V_t^{\mathrm{res}} = \sum_{i,j=1}^{K} \tilde{w}_t^{(i)} \tilde{w}_t^{(j)} S(x_t^{(i)}) S(x_t^{(j)}) \mathrm{Cov}(N_t^{(i)}, N_t^{(j)} \mid \mathcal{F}_t).$$

Its diagonal contribution satisfies

$$D_t^{\mathrm{res}} := \sum_{i=1}^{K} (\tilde{w}_t^{(i)})^2 S(x_t^{(i)})^2 \mathrm{Var}(N_t^{(i)} \mid \mathcal{F}_t) \le \frac{1}{4} \sum_{i=1}^{K} (\tilde{w}_t^{(i)})^2 S(x_t^{(i)})^2 \tag{13}$$

under systematic resampling.

Proof. Apply the law of total variance to $\hat{Z}_T = \sum_i N_t^{(i)} \tilde{w}_t^{(i)} S_T^{(i)}$ with respect to the offspring counts N:

$$\mathrm{Var}(\hat{Z}_T \mid \mathcal{F}_t) = \underbrace{\mathbb{E}[\mathrm{Var}(\hat{Z}_T \mid \mathcal{F}_t, N) \mid \mathcal{F}_t]}_{V_t^{\mathrm{cont}}} + \underbrace{\mathrm{Var}(\mathbb{E}[\hat{Z}_T \mid \mathcal{F}_t, N] \mid \mathcal{F}_t)}_{V_t^{\mathrm{res}}}.$$

For the first term, conditional on N and $\mathcal{F}_t$, the descendants of distinct particles propagate independently under DDIM (η=1), so $\mathrm{Var}(\hat{Z}_T \mid \mathcal{F}_t, N) = \sum_i N_t^{(i)} (\tilde{w}_t^{(i)})^2 \sigma_t^2(x_t^{(i)})$. Taking expectation over N with $\mathbb{E}[N_t^{(i)}] = m_t^{(i)}$ recovers (12). For the second term, $\mathbb{E}[\hat{Z}_T \mid \mathcal{F}_t, N] = \sum_i N_t^{(i)} \tilde{w}_t^{(i)} S(x_t^{(i)})$, so taking the variance over N gives the covariance quadratic form in the statement. The diagonal bound (13) follows by retaining the $i = j$ terms and applying $\mathrm{Var}(N_t^{(i)} \mid \mathcal{F}_t) \le 1/4$ from Proposition 4. □

Remark 4. *Proposition 7 separates per-step variance into a continuation term V_t^{cont} and a resampling term V_t^{res}. The allocation* (11) *targets the continuation term through an importance-weighted score-scale surrogate, while systematic resampling controls the offspring-count variances that appear in the diagonal part of V_t^{res} [22]. The ordering heuristic is not needed for unbiasedness; it is used to reduce the empirical effect of the off-diagonal covariance terms by placing particles with similar weighted continuation scores next to each other on the systematic grid.*

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: The abstract and introduction state all three contributions explicitly with each claim supported by the corresponding theoretical result in Appendix B or empirical result in Table 1 and Figures 1-5

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: Limitations are discussed in section 7.

   Guidelines:

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

3. **Theory assumptions and proofs**

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

Answer: [Yes]

Justification: Full proofs for all theoretical results are provided in Appendix B, with assumptions explicitly stated in each proposition.

Guidelines:

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

4. **Experimental result reproducibility**

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

   Answer: [Yes]

   Justification: All hyperparameter, model choices, resampling schedules and evaluation protocols are specified in section 5 and the appendix.

   Guidelines:

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

5. **Open access to data and code**

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

Answer: [No]

Justification: All datasets used are publically available. Code will be released upon acceptance.

Guidelines:

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

6. **Experimental setting/details**

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

Answer: [Yes]

Justification: Hyperparameters including particle count K, alignment strength λ, resampling schedule $\mathcal{T}$ and NFE budgets are reported in section 5 for all experiments.

Guidelines:

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

7. **Experiment statistical significance**

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

Answer: [Yes]

Justification: Table 1 reports mean and standard deviation over 5 random seeds for all methods across metrics.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

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

   Answer: [Yes]

   Justification: Experiments were run on 4 NVIDIA GH200 GPUs.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: This work introduces a sampling algorithm for pretrained diffusion models and does not involve data collection, human subjects or deployment.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

    Justification: This paper proposes a foundational sampling algorithm, while it could improve generation quality, it does not introduce new generative capabilities beyond existing models.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: This paper does not release new models or datasets.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All datasets and pretrained models used are properly cited and used in accordance with their respective licenses.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [No]

    Justification: Code will be released upon acceptance.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

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

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

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.

- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

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

    Answer: [N/A]

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

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.