TRAVERSING THE SATISFACTION-DIVERSITY FRONTIER IN TEXT-TO-IMAGE DIFFUSION

Anonymous authors
Paper under double-blind review

ABSTRACT

Text-to-image generation enables users to explore several images generated from the same prompt. For these generated images to be useful, each one must reflect the user's preferences, measured by a learned reward, and differ visually from the others to maintain diversity. Existing methods are limited: they either address reward and diversity separately or combine them in one aggregate score, enabling high diversity to offset low rewards. In this paper, we address these limitations by formulating generation as *satisficing*: every image (candidate) must satisfy a reward floor and the batch of images must satisfy a diversity cutoff. The reward floor controls the balance between worst-candidate reward and batch diversity; we show that varying this floor defines a Pareto frontier. To traverse this frontier, we introduce SatisDive, a training-free inference-time method. SatisDive uses a batch-relative reward cutoff to distinguish lower- from higher-reward candidates, emphasizing reward improvement for candidates below the cutoff and diversity among candidates above it. On Pick-a-Pic, at matched DreamSim, SatisDive improves worst-candidate reward over FK steering by up to 0.43 with FLUX.1-dev as the base model and HPSv3 as the reward, and by up to 0.70 with SANA-1.6B as the base model and ImageReward as the reward. More broadly, across their overlapping DreamSim ranges, SatisDive's satisfaction–diversity curve Pareto-dominates FK steering's curve in each setting.

1 INTRODUCTION

An important goal of text-to-image generation is to give users a choice among several high-quality candidate images. For example, a user creating a poster illustration may compare images generated from the same prompt, choosing the lighting, background, or composition that best fits the poster. Recent inference-time methods increase diversity (Singh et al., 2024) to provide a wider range of options, maximize a learned reward (Singhal et al., 2025) to align generated images with user preferences, or select images by maximizing a linear combination of total image reward and batch diversity (Parmar et al., 2026).

However, these approaches can still limit the user's viable choices. Firstly, reward-steering methods can concentrate sampling on higher-reward images, making the returned images visually similar (Singhal et al., 2025; Kim et al., 2025b; Shekhar et al., 2026). Secondly, reward-free diversity methods make generated images more visually different, but do not explicitly account for learned reward (Singh et al., 2024; Kim et al., 2026; Corso et al., 2024; Vinograd et al., 2026). Thirdly, batch-level reward–diversity objectives (Parmar et al., 2026) allow greater diversity or high rewards for other images to compensate for one image's low reward.

These limitations motivate controlling worst-image reward through a reward floor, i.e., a minimum reward level required of each candidate. As the reward floor increases, the batch can become less diverse since fewer images satisfy the floor. We show that varying the reward floor defines a Pareto frontier between worst-image reward and batch diversity, which we call the satisfaction–diversity frontier. Along this frontier, we formulate image generation as *satisficing* (Simon, 1956): *every generated image must satisfy a reward floor, and the batch must satisfy a diversity cutoff.* Once a candidate clears the reward floor, further increases in reward do not improve satisfaction. To this end, we introduce SatisDive, a training-free inference-time method. As Figure 1 shows, SatisDive

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.

generates visually distinct images above the reward floor, while FK steering (Singhal et al., 2025), a strong reward-steering baseline, returns nearly identical images.

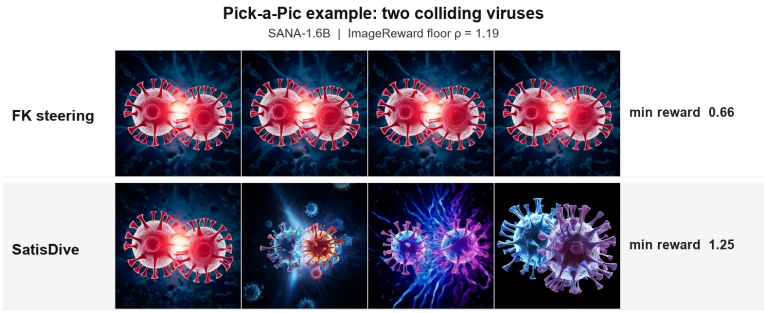

Figure 1: **SatisDive returns four visually distinct images above the reward floor, while FK steering returns four nearly identical images.** We use the SANA-1.6B (Xie et al., 2025) flow matching model with ImageReward (Xu et al., 2023) reward model. On a Pick-a-Pic (Kirstain et al., 2023) prompt depicting two colliding viruses, the minimum reward is 0.66 for FK steering and 1.25 for SatisDive, with reward floor ρ=1.19.

Our objective traverses the satisfaction–diversity frontier through a per-image reward penalty and a batch-level diversity penalty. Under the satisficing formulation, a visually different but low-reward image should contribute less to batch diversity. Thus, these penalties are complementary: images with rewards below a batch-relative reward cutoff incur larger reward penalties and contribute less to the diversity penalty, and images with rewards above the cutoff incur smaller reward penalties but contribute more to the diversity penalty. Because some candidates may remain low reward throughout denoising, we introduce late-stage latent replacement, which replaces their latents with those of higher-reward candidates. Subsequent stochastic denoising and the diversity penalty allow the resulting trajectories to produce distinct images.

Empirically, SatisDive improves worst-candidate reward without sacrificing batch diversity: across their overlapping DreamSim ranges, SatisDive's worst-candidate reward is consistently higher than FK steering's, thus our satisfaction–diversity curves Pareto-dominate FK steering's on both FLUX and SANA. Our contributions are as follows.

- We show that different reward floors define a Pareto frontier between worst-image reward and batch diversity. Under our *satisficing* formulation, every image must satisfy the selected reward floor, and the batch must satisfy a diversity cutoff.
- We introduce SatisDive, a training-free inference-time method whose objective provably traverses this satisfaction–diversity frontier. The objective uses a batch-relative reward cutoff to emphasize reward improvement for lower-reward candidates and diversity among higher-reward candidates.
- At matched DreamSim, SatisDive improves worst-image reward by up to 0.43 on FLUX and 0.70 on SANA. Our satisfaction–diversity curve Pareto-dominates the curves traced by FK steering on the FLUX.1-dev and SANA-1.6B base models.

2 Problem Formulation

A rectified-flow text-to-image model (Liu et al., 2023) generates an image from a prompt c using a learned velocity field $v_\theta(x_t, t, c)$. Generation starts from Gaussian noise $x_1 \sim \mathcal{N}(0, I)$ and integrates v_θ from $t = 1$ to $t = 0$ over T discretized steps, producing the image x_0. This formulation also covers other diffusion models such as DDPM (Ho et al., 2020), which differ in their noise schedule (Lipman et al., 2023). Table 4 in Appendix A summarizes the notation used throughout the paper.

Let $p_\theta(x_0 \mid c)$ denote the base model's distribution. We consider methods that maintain K candidates throughout a single sampling pass and generate a batch of K images (denoted $x_0^{1:K}$). Given a reward $r(x_0, c)$, we first consider reward-steering methods (e.g., FK steering (Singhal et al., 2025)) that yield high-reward images by targeting the tilted distribution p_λ:

$$p_\lambda(x_0 \mid c) \propto p_\theta(x_0 \mid c)\, \exp\big(\lambda\, r(x_0, c)\big), \qquad \lambda \geq 0, \tag{1}$$

where larger λ increases the relative probability of higher-reward images. These methods approximate Eq. 1 by resampling candidate latents at intermediate timesteps with selection probabilities determined by candidate rewards; these rewards are evaluated on the model's estimate $\hat{x}_0$ of the final image, computed as:

$$\hat{x}_0 = x_t - \sigma_t v_\theta(x_t, t, c), \tag{2}$$

where σ_t is the noise level at timestep t. Since rectified flow is deterministic, these steering methods use the equivalent marginal-preserving stochastic form (Mark et al., 2025).

Limitations of current methods. The reward-only tilt in Eq. 1 weights each candidate's selection probability during resampling according to its reward, without considering how the K images differ. Proposition 1 (proof in Appendix B) states what happens when this tilt is applied to the current candidates during resampling.

Proposition 1 (As $\lambda \to \infty$, resampling produces K copies of the highest-reward candidate). *At a resampling step with a unique highest-reward candidate, increasing λ increases its expected number of copies. In the limit as $\lambda \to \infty$, the probability that all K resampled candidates are copies of the highest-reward candidate approaches one.*

Consequently, a greater reward tilt can reduce the number of distinct candidates retained after resampling and thereby reduce the visual diversity among the generated images.

A second class of methods increases the *diversity* of generated images. NegToMe (Singh et al., 2024), for instance, increases the distance between the candidates' features. Being reward-agnostic, these methods do not constrain the sampling distribution toward higher rewards. Figure 2 visualizes the limitations of reward-steering and diversity methods in a toy 2D setting (details in Appendix C.1).

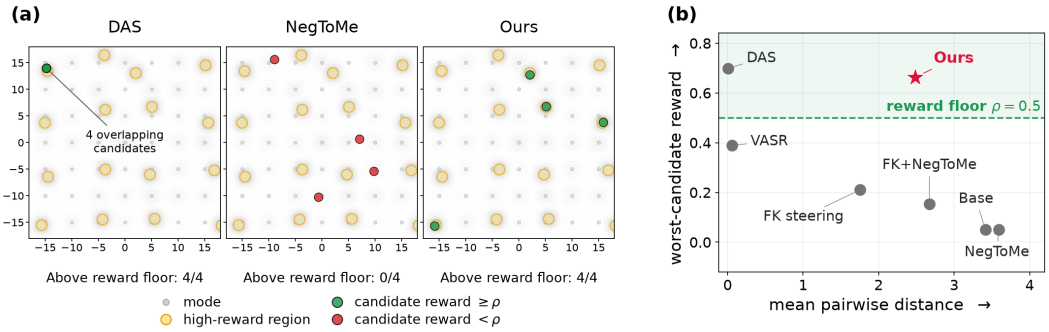

Figure 2: **In a toy setting, each baseline tends towards high diversity or high reward, not both.** Each method generates $K{=}4$ candidates from a frozen flow model trained on a mixture of Gaussians (details in Appendix C.1). Reward is bounded in $(0, 1]$, and the midpoint $\rho = 0.5$ is a reward floor that every candidate must satisfy. **(a)** To visualize clearly the coverage of high-reward regions, candidates are 2D points on a Cartesian x–y grid rather than images. We show one selected illustrative batch of $K{=}4$ candidates per method. DAS's candidates are all above ρ but are not diverse; they cover only a single high-reward region (yellow circles). Conversely, NegToMe's candidates are diverse yet are all below ρ (red circles). In these batches, only ours has all candidates above ρ in distinct high-reward regions. **(b)** We plot worst-candidate reward $\min_k r_k$ against diversity (mean pairwise distance) averaged over 200 runs for each method. Methods in the shaded green region have average worst-candidate rewards greater than ρ. DAS achieves the highest reward but lowest diversity; NegToMe achieves the highest diversity but lowest reward.

A natural third alternative treats diversity as a second reward and samples from a joint tilt:

$$\pi_{\text{joint}}(x_0^{1:K} \mid c) \propto \prod_{k=1}^{K} p_\theta(x_0^k \mid c)\, \exp\Big(\lambda \sum_{k=1}^{K} r(x_0^k, c) + \mu D(x_0^{1:K})\Big), \qquad \lambda, \mu \geq 0, \tag{3}$$

where D is a batch-level diversity term and μ weights it against reward r. Because reward and diversity are aggregated, the joint objective can be gamed: for $\lambda > 0$, one unit of D offsets a decrease of μ/λ in total reward. This exchange exists for every positive μ, so π_{joint} never requires a candidate to be high-reward.

3 METHOD

The approaches discussed in Section 2 do not jointly require a reward floor for every candidate and a minimum diversity for the whole batch. We therefore formulate generation as *satisficing* (Simon, 1956): every candidate must satisfy a reward floor ρ, while the batch must satisfy a diversity requirement. These requirements define the target distribution

$$\pi^\star(x_0^{1:K} \mid c) \;\propto\; \Big(\prod_{k=1}^{K} p_\theta(x_0^k \mid c)\,\mathbf{1}[r(x_0^k, c) \geq \rho]\Big)\,\mathbf{1}[D(x_0^{1:K}) \geq \delta], \tag{4}$$

where D measures batch diversity, δ is the required diversity level, and each indicator equals one when its condition holds and zero otherwise. Rejection sampling would draw K candidates independently from p_θ and reject the entire batch if either requirement is violated. Because the probability that all K candidates satisfy the reward floor decreases exponentially with K, rejection sampling can be prohibitively expensive.

We thus introduce SatisDive, which replaces both indicators with differentiable penalties. Our training-free approach rests on two mechanisms: (1) a satisficing objective that increases low rewards and diversifies the batch; and (2) late-stage latent replacement, which replaces the latent of a low-reward candidate with that of a higher-reward candidate. Section 3.1 details the objective and late-stage latent replacement and connects our objective to $\pi^\star$, while Section 3.2 proves that this objective traces the satisfaction–diversity Pareto frontier.

3.1 SATISFICING AT INFERENCE TIME

Our objective $\mathcal{J}$ consists of a reward penalty and a diversity penalty, differentiable counterparts to $\pi^\star$'s two indicators. For a differentiable reward $r(\cdot)$, we minimize $\mathcal{J}$ over each candidate's latent x_t^k. At each step, SatisDive compares each candidate's reward $r_k = r(\hat{x}_0^k, c)$ against a batch-relative reward cutoff τ:

$$\tau = \max_k r_k - \Delta, \qquad \Delta > 0, \tag{5}$$

where Δ is a fixed tolerance. Comparing against τ distinguishes low- from high-reward candidates in the batch. For candidate k, we use the sigmoid q_k as a smooth probability that $r_k \geq \tau$:

$$q_k = \mathrm{sigmoid}\big((r_k - \tau)/\beta_r\big), \tag{6}$$

where the temperature β_r is computed once per run as the standard deviation of the batch's rewards; $q_k \to 1$ for rewards above τ and $q_k \to 0$ below τ.

Reward Penalty. We define a penalty for each candidate k that increases with $\tau - r_k$; lower-reward candidates have larger penalties. Summing over the batch gives $\mathcal{J}_r$:

$$\mathcal{J}_r \triangleq -\sum_k \log q_k = \sum_k \mathrm{softplus}\big((\tau - r_k)/\beta_r\big). \tag{7}$$

Thus, $\mathcal{J}_r$ is the negative log-likelihood that every candidate in the batch exceeds τ.

Diversity Penalty. We measure batch diversity D as the weighted mean of the pairwise cosine distances d_{ij} between the pooled backbone features of candidates i and j, each pair weighted by $q_i q_j$ (Eq. 6). Consequently, pairs of higher-reward candidates receive greater weight than pairs with lower-reward candidates. To account for the scale of pairwise distances in each run, we set δ to the median of the d_{ij}, computed once per run. At each step, the diversity penalty $\mathcal{J}_d$ is then:

$$\mathcal{J}_d \triangleq \mathrm{softplus}\big((\delta - D)/\beta_d\big), \quad \text{where} \quad D = \frac{\sum_{i<j} q_i q_j\, d_{ij}}{\sum_{i<j} q_i q_j}, \tag{8}$$

and β_d is the diversity temperature. Combining the two penalties yields our objective:

$$\mathcal{J} = \mathcal{J}_r + \mathcal{J}_d = \sum_k \mathrm{softplus}\big((\tau - r_k)/\beta_r\big) \;+\; \mathrm{softplus}\big((\delta - D)/\beta_d\big). \tag{9}$$

At intermediate denoising steps, we update the candidate latents using the gradients of $\mathcal{J}_d$ and $\mathcal{J}_r$ (scaling specified in Appendix C). The objective gives greater weight to increasing reward and

diversity below their cutoffs than above them. Proposition 2 (proof in Appendix B) connects this objective to the target $\pi^\star$ in Eq. 4.

Proposition 2 (The distribution induced by $\mathcal{J}$ converges to $\pi^\star$). *At any given timestep, let $M = \max_k r_k$ and set $\tau = \rho = M - \Delta$. Here, D in $\pi^\star$ is the unweighted mean pairwise distance, obtained from the weighted mean in Eq. 8 as all reward gates approach one. Suppose that independently sampling K candidates from the base model has a nonzero chance of producing a batch with $r_k \geq \rho$ for every candidate k and $D \geq \delta$ (Eq. 4). Suppose also that the event that $r_k = \rho$ for some k, and the event that $D = \delta$, each have probability zero. Then, as $\beta_r, \beta_d \to 0$, the distribution induced by $\mathcal{J}$ converges to $\pi^\star$.*

Late-Stage Latent Replacement. In practice, the gradient of $\mathcal{J}$ depends strongly on the structure of the latent space and the initial noise x_1^k, so candidates may remain in local reward optima throughout denoising. We thus propose late-stage latent replacement based on the intuition that, toward the end of denoising, it is preferable to copy from a known high-reward candidate than to continue updating a candidate that remains low reward. To see how reward may change over the remaining steps, we extrapolate each below-τ candidate's reward from the difference $r_k - r_k^{\text{prev}}$ since the previous update. If this candidate's extrapolated reward is less than τ, we replace its latent with that of a randomly chosen candidate above τ. We apply late-stage latent replacement only in the latter half of denoising, since an early $\hat{x}_0^k$ estimate is a poor predictor of a candidate's final reward. After replacement, the two candidates become visually distinct by the end of denoising due to $\mathcal{J}_d$ and stochastic sampling. Algorithm 1 details our full procedure.

Algorithm 1 SatisDive.

Require: base model p_θ, prompt c, reward r, batch size K, steps T, update steps $\mathcal{G}$, tolerance Δ

sample $x^1, \ldots, x^K \sim \mathcal{N}(0, I)$
for denoising step $s = 1, \ldots, T$ **do**
  if $s \in \mathcal{G}$ **then**
    for all candidates k **do**
      Predict clean image $\hat{x}_0^k$ ▷ *Eq.* (2)
      Compute reward $r_k = r(\hat{x}_0^k, c)$
    Compute reward cutoff $\tau = \max_k r_k - \Delta$ ▷ *Eq.* (5)
    Using τ, compute $\mathcal{J}_d$ and $\mathcal{J}_r$; update each x^k using their gradients ▷ *Eq.* (9)
    if $s > T/2$ and r_k^{prev} exists for every k **then**
      $m = |\{u \in \mathcal{G} : u > s\}|$ ▷ *m is the number of remaining update steps*
      $\tilde{r}_k = r_k + m \cdot \max(r_k - r_k^{\text{prev}}, 0)$, for every k ▷ *Get extrapolated reward $\tilde{r}_k$*
      for all k such that $r_k < \tau$ and $\tilde{r}_k < \tau$ **do**
        Sample $j \sim \mathcal{U}(\{i : r_i \geq \tau\})$
        Replace latent x^k with x^j ▷ *Late-stage latent replacement*
    $r_k^{\text{prev}} \leftarrow r_k$, for every k ▷ *Store rewards for the next update*
  Take next sampling step
return $x_0^{1:K}$

3.2 SatisDive's Objective Traces the Satisfaction–Diversity Pareto Frontier

By sweeping Δ, SatisDive's objective traces a Pareto frontier between reward and diversity. We now characterize how the reward constraint in $\pi^\star$ (Eq. 4) changes as the reward floor ρ varies. Let $A_\rho = \{x : r(x, c) \geq \rho\}$ be the set of images that satisfy the floor, and let p_ρ be the base output distribution restricted to A_ρ, preserving relative probabilities within this set. Because the reward constraint applies separately to each candidate, we rewrite $\pi^\star$ as a product of K reward-restricted distributions and the batch diversity indicator:

$$\pi^\star(x^{1:K} \mid c) \propto \Big(\prod_{k=1}^{K} p_\rho(x^k) \Big) \mathbf{1}[D(x^{1:K}) \geq \delta], \qquad p_\rho(x) \propto p_\theta(x \mid c)\, \mathbf{1}[x \in A_\rho]. \tag{10}$$

We denote $\{p_\rho\}$ as the family of distributions obtained by varying the reward floor ρ. For each floor, Lemma 5 (Appendix B) shows two properties of p_ρ. First, among all distributions that assign probability one to A_ρ, p_ρ uniquely minimizes the KL divergence to the base model. Second, the

support of every such distribution with finite KL is contained in the support of p_ρ. Here, satisfaction means the highest reward floor satisfied with probability one. Theorem 3 (proof in Appendix B) establishes $\{p_\rho\}$ as the satisfaction–diversity Pareto frontier.

Theorem 3 (The family $\{p_\rho\}$ forms the satisfaction–diversity Pareto frontier as ρ varies). *Consider reward floors ρ for which $p_\theta(A_\rho \mid c) > 0$, and suppose diversity is nonincreasing under support restriction, strictly so for proper restrictions. Every candidate drawn from p_ρ has reward at least ρ. For any two such reward floors ρ' and ρ with $\rho' \geq \rho$, $p_{\rho'}$ has diversity at most that of p_ρ. For each ρ, no distribution over A_ρ at finite KL to p_θ is more diverse than p_ρ. Therefore, $\{p_\rho\}$ forms the satisfaction–diversity Pareto frontier among distributions with finite guaranteed reward and finite KL to p_θ.*

The theorem applies to diversity measures satisfying the stated support-restriction assumption; our experiments evaluate the practical satisfaction–diversity tradeoff using mean pairwise DreamSim. Reward steering instead targets the tilted distributions p_λ. For finite λ and any reward floor ρ with $0 < p_\theta(A_\rho \mid c) < 1$, Corollary 4 (Appendix B) shows that p_λ assigns positive probability to images below ρ, unlike p_ρ. Thus, varying λ traces $\{p_\lambda\}$ rather than the satisfaction–diversity Pareto frontier established by $\{p_\rho\}$.

4 Experiments

Models, datasets, and baselines. We show results with three frozen base models: FLUX.1-dev (Labs, 2024), SANA-1.6B (Xie et al., 2025), and Stable Diffusion 1.5 (SD1.5) (Rombach et al., 2022). We use two reward models, HPSv3 (Ma et al., 2025) and ImageReward (Xu et al., 2023), and evaluate on the Pick-a-Pic (Kirstain et al., 2023) and HPDv2 (Wu et al., 2023b) datasets. Our main baselines fall into two categories: 1) reward-steering methods, namely Feynman–Kac (FK) steering (Singhal et al., 2025), VASR (Shekhar et al., 2026), and DAS (Kim et al., 2025b); and 2) diversity methods, namely Negative Token Merging (NegToMe) (Singh et al., 2024), which repels candidates in feature space. We also evaluate FK+NegToMe, which combines FK steering with NegToMe's diversity mechanism. All methods use the same seeds and return $K{=}4$ images (i.e., candidates) per prompt. We report SGI (Parmar et al., 2026) separately in Appendix D.9 because it prunes a much larger pool of 128 candidates. Baselines use their recommended settings; sampling details are provided in Appendix C.

Metrics. We evaluate images on two axes: reward and diversity. On the reward axis, every method is scored against a reward floor ρ. Rather than choosing ρ arbitrarily, we set it to the base model's median reward over all prompts. We report worst-candidate reward $\min_k r_k$, averaged over prompts, since a batch satisfies the floor ρ precisely when $\min_k r_k \geq \rho$; we also report mean reward. We define the satisfaction rate SAT_ρ as the fraction of generated candidates whose reward is at least ρ:

$$\mathrm{SAT}_\rho \;=\; \frac{1}{N \cdot K} \sum_{i=1}^{N \cdot K} \mathbf{1}[\, r_i \geq \rho \,], \tag{11}$$

where r_i is the reward of the i-th generated candidate and $\mathbf{1}[\cdot]$ is the indicator function. N is the number of prompts, with K candidates per prompt. Since ρ is the median reward of the base model's outputs, $\mathrm{SAT}_\rho = 0.5$ indicates parity with the base model. On the diversity axis, each batch is measured by the mean pairwise DreamSim (Fu et al., 2023) and LPIPS (Zhang et al., 2018) among its K candidates. We additionally report PickScore (Kirstain et al., 2023) as a held-out reward, which indicates whether reward gains reflect a broader improvement in preference.

4.1 SatisDive Traverses the Satisfaction–Diversity Frontier

Sweeping the tolerance Δ of our satisficing objective (Eq. 9) traces a frontier curve between worst-candidate reward and diversity. As shown in Figure 3 on FLUX and SANA, baseline methods tend to exhibit either high worst-candidate reward but low diversity (DAS, VASR), or low worst-candidate reward and high diversity (NegToMe). Although varying FK steering's reward-tilt parameter λ also traces a reward–diversity curve, SatisDive's curve Pareto-dominates (Hwang & Masud, 2012) FK steering's over their common DreamSim ranges. At matched DreamSim, SatisDive has higher worst-candidate reward at every overlapping point, with improvements of up to 0.43 on FLUX using HPSv3 and 0.70 on SANA using ImageReward.

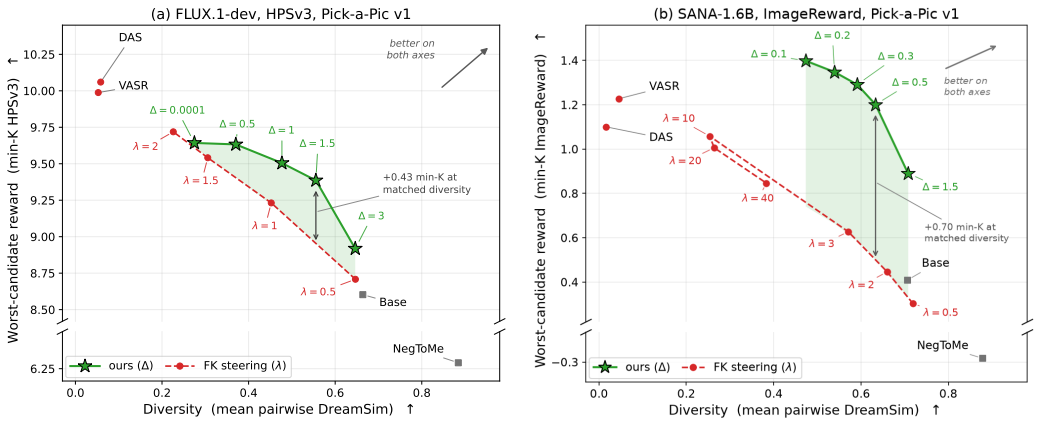

Figure 3: **SatisDive achieves better worst-candidate reward at matched diversity versus FK steering.** We plot mean worst-candidate reward $\min_k r_k$ against mean pairwise DreamSim, on Pick-a-Pic. We sweep SatisDive and FK steering on two settings. **(a)** FLUX.1-dev is the base model and HPSv3 is the reward, with Δ from 0.0001 to 3 and λ from 0.5 to 2. **(b)** SANA-1.6B is the base model and ImageReward is the reward, with Δ from 0.1 to 1.5 and λ from 0.5 to 40. The shaded band represents the worst-candidate reward gap between the two curves at the same diversity, marked at Δ=1.5 in (a) where it is $\approx$0.43 on FLUX HPSv3, and at Δ=0.5 in (b) where it is $\approx$0.70 on SANA ImageReward. We note that the top right region, representing high reward and high diversity, is ideal. DAS and VASR collapse to diversity below 0.06 in both settings, and NegToMe is the most diverse but its worst candidate on average falls below the base model's. SatisDive traces a better empirical frontier than FK steering between these extremes.

Table 1: **On Pick-a-Pic, SatisDive is competitive on satisfaction rate without collapsing diversity.** We use FLUX.1-dev with HPSv3 as the reward. DAS has the highest satisfaction rate at 0.612 but a mean pairwise DreamSim of only 0.059, while NegToMe is the most diverse at 0.885 and has the lowest satisfaction rate at 0.276. SatisDive attains a satisfaction rate of 0.552 at a DreamSim of 0.555, and matches base FLUX on both held-out rewards. We use ρ=9.66; $\pm$ denotes 95% CIs.

	Satisfaction–diversity			Additional metrics		Held-out	
Method	min ↑	DreamSim ↑	SAT_ρ ↑	mean ↑	LPIPS ↑	ImageReward	PickScore
Base	8.60	0.663 $\pm$0.013	0.500 $\pm$0.038	9.65	0.643 $\pm$0.008	1.02	22.28
NegToMe	6.30	0.885 $\pm$0.009	0.276 $\pm$0.027	8.17	0.788 $\pm$0.004	0.79	21.49
FK steering	9.54	0.306 $\pm$0.021	0.570 $\pm$0.041	10.07	0.291 $\pm$0.021	1.09	22.37
FK + NegToMe	7.69	0.360 $\pm$0.017	0.333 $\pm$0.034	8.70	0.310 $\pm$0.016	0.99	21.68
VASR	9.99	0.053 $\pm$0.004	0.598 $\pm$0.042	10.17	0.033 $\pm$0.005	1.10	22.40
DAS	10.06	0.059 $\pm$0.007	0.612 $\pm$0.041	10.26	0.040 $\pm$0.007	1.09	22.38
SatisDive (Ours)	9.39	0.555 $\pm$0.01	0.552 $\pm$0.039	10.03	0.531 $\pm$0.015	1.04	22.32

Table 2: **On HPDv2, SatisDive is competitive on satisfaction rate without collapsing diversity.** We use FLUX.1-dev with HPSv3 as the reward. SatisDive attains a satisfaction rate of 0.553 at a mean pairwise DreamSim of 0.567, against DAS's 0.580 at 0.082, and matches base FLUX on both held-out rewards. SAT_ρ uses floor ρ=13.93. $\pm$ denotes 95% confidence intervals.

	Satisfaction–diversity			Additional metrics		Held-out	
Method	min ↑	DreamSim ↑	SAT_ρ ↑	mean ↑	LPIPS ↑	ImageReward	PickScore
Base	12.34	0.655 $\pm$0.004	0.500 $\pm$0.015	13.36	0.646 $\pm$0.003	1.11	22.73
NegToMe	9.65	0.865 $\pm$0.003	0.309 $\pm$0.011	12.02	0.781 $\pm$0.002	0.91	21.95
FK steering	13.44	0.197 $\pm$0.006	0.565 $\pm$0.016	13.77	0.182 $\pm$0.007	1.17	22.79
FK + NegToMe	11.53	0.312 $\pm$0.005	0.320 $\pm$0.013	12.50	0.277 $\pm$0.005	1.06	22.04
VASR	13.62	0.053 $\pm$0.002	0.569 $\pm$0.017	13.78	0.033 $\pm$0.002	1.17	22.79
DAS	13.64	0.082 $\pm$0.003	0.580 $\pm$0.017	13.82	0.066 $\pm$0.004	1.16	22.80
SatisDive (Ours)	13.07	0.567 $\pm$0.006	0.553 $\pm$0.015	13.71	0.559 $\pm$0.006	1.14	22.77

On Pick-a-Pic, Table 1 shows that SatisDive improves mean HPSv3 reward over base FLUX (10.03 against 9.65), indicating that our reward gains improve the whole batch and not just the worst candidate. Although DAS and VASR achieve slightly higher mean rewards (10.26 and 10.17 compared to SatisDive's 10.03), both are markedly less diverse than SatisDive. The same satisfaction–diversity pattern holds on HPDv2 in Table 2, where SatisDive achieves a competitive satisfaction rate and mean HPSv3 reward with notably higher diversity than the reward-steering methods. In particular, adding NegToMe to FK steering increases diversity but reduces the satisfaction rate below the base model's 0.5, whereas SatisDive maintains above-base satisfaction at near-base diversity. Appendix D evaluates SatisDive across additional settings and provides ablation and computational analyses.

Table 3: **SatisDive exceeds every reward-steering method on worst-candidate reward and diversity on SANA-1.6B.** We use Pick-a-Pic with ImageReward; the reward floor is $\rho = 1.19$. Our method is shown at three Δ settings, tracing its satisfaction–diversity frontier as reflected in Figure 3b. At $\Delta = 0.3$, SatisDive attains a worst-candidate reward of 1.29 and DreamSim of 0.591, exceeding every reward-steering method on both metrics. $\pm$ denotes 95% confidence intervals.

	Satisfaction–diversity			Additional metrics		Held-out	
Method	min ↑	DreamSim ↑	SAT_ρ ↑	mean ↑	LPIPS ↑	HPSv3	PickScore
Base	0.41	0.706 $\pm$0.012	0.500 $\pm$0.037	0.91	0.644 $\pm$0.010	7.53	21.93
NegToMe	−0.29	0.879 $\pm$0.011	0.380 $\pm$0.033	0.55	0.743 $\pm$0.009	5.98	21.29
FK steering	1.06	0.254 $\pm$0.022	0.625 $\pm$0.041	1.18	0.204 $\pm$0.021	7.98	22.06
FK + NegToMe	0.84	0.386 $\pm$0.025	0.585 $\pm$0.041	1.04	0.278 $\pm$0.022	7.07	21.63
VASR	1.23	0.046 $\pm$0.003	0.651 $\pm$0.041	1.25	0.023 $\pm$0.003	8.01	22.09
DAS	1.10	0.016 $\pm$0.001	0.599 $\pm$0.043	1.11	0.004 $\pm$0.001	7.72	21.93
SatisDive (Δ=0.3)	1.29	0.591 $\pm$0.017	0.735 $\pm$0.037	1.39	0.547 $\pm$0.016	7.74	21.97
SatisDive (Δ=0.5)	1.20	0.633 $\pm$0.015	0.696 $\pm$0.038	1.33	0.582 $\pm$0.014	7.67	21.95
SatisDive (Δ=0.75)	1.13	0.658 $\pm$0.015	0.674 $\pm$0.039	1.28	0.600 $\pm$0.013	7.63	21.92

Generalization to Other Base Models. To show SatisDive is not specific to one architecture or reward model, we extend to SANA-1.6B (flow matching, Table 3) and SD1.5 (UNet, Appendix D Table 11) with ImageReward (Xu et al., 2023). On SANA, SatisDive at $\Delta = 0.3$ achieves the highest worst-candidate reward and satisfaction rate of any evaluated method, with 1.29 worst-candidate reward versus VASR's 1.23 and 0.735 satisfaction rate versus VASR's 0.651. SatisDive also achieves a DreamSim of 0.591, versus VASR's 0.046.

We further test whether stopping reward maximization once a candidate satisfies the floor, without an explicit diversity term, is sufficient to preserve diversity. We therefore cap DAS's reward at ρ while leaving its other settings unchanged. This reward capping increases DAS's DreamSim from 0.016 to 0.066, but the resulting diversity remains far below SatisDive's 0.658; the change in satisfaction rate is not statistically significant. Thus, reward capping alone does not account for SatisDive's diversity (Appendix D.12).

4.2 Qualitative Results

Figure 4 shows individual batches from the SANA evaluation summarized in Figure 3. FK steering returns four images above the reward floor but repeats nearly the same composition, while Base and NegToMe produce more varied batches in which some images fall below the floor. On both prompts, SatisDive is the only method that returns four above-floor images with visibly different compositions. Additional qualitative results appear in Appendix E.



Figure 4: **On two Pick-a-Pic prompts, only SatisDive returns four visually distinct images that are all above the reward floor.** We use SANA-1.6B with ImageReward and matched starting seeds; the reward floor is ρ=1.19. The magenta border marks the worst-reward candidate in each row, whose reward is reported on the right. Base and NegToMe produce visually varied batches, but their worst-reward candidates do not satisfy the reward floor. NegToMe's worst-reward car image (on the left) is nearly black, and its worst-reward fairy image (on the right) does not clearly show an owl. FK steering's images exceed the reward floor, but its four images repeat nearly the same composition.

5 Related Work

Reward-driven sampling. Existing methods increase reward through resampling (Singhal et al., 2025; Wu et al., 2023a; Kim et al., 2025b; Shekhar et al., 2026), value-based lookahead (Li et al., 2025; Jain et al., 2025), stochastic transitions (Holderrieth et al., 2026), adaptive model evaluation (Kim et al., 2025a), or noise-space optimization and regularization (Hwang et al., 2026; Tang et al., 2025; Eyring et al., 2024; Harrington et al., 2026; Zhai et al., 2025). These methods do not explicitly optimize batch diversity and may collapse candidate lineages (Shekhar et al., 2026). CREPE (He et al., 2026) uses replica exchange to preserve diversity under reward tilting (comparison in Appendix D.8), while SGI (Parmar et al., 2026) selects a reward–diversity batch from a large candidate pool; neither imposes a reward floor on every candidate.

Reward-free diversity. NegToMe (Singh et al., 2024), CNO (Kim et al., 2026), Particle Guidance (Corso et al., 2024), and EDDY (Vinograd et al., 2026) repel candidates in feature or prediction space. Other methods vary conditioning, intermediate features, or initial noise (Sadat et al., 2024; Yadav et al., 2026; Li et al., 2026), while general guidance modifies model predictions, attention, or guidance schedules (Ho & Salimans, 2021; Ahn et al., 2024; Hong, 2024; Karras et al., 2024; Kynkäänniemi et al., 2024; Yu et al., 2023). These methods increase diversity without imposing a reward floor.

Pareto and satisficing objectives. AIG (Jena et al., 2025) studies average reward against distributional coverage; PROUD (Yao et al., 2024) defines Pareto objectives over per-image properties; and ParetoSlider (Golan et al., 2026), constrained diffusion alignment (Khalafi et al., 2025), and SITAlign (Chehade et al., 2025) optimize expected rewards. In contrast, our training-free formulation imposes a reward floor on every candidate and a diversity requirement on the batch, producing a frontier between worst-candidate reward and batch diversity.

6 Conclusion

SatisDive formulates image generation as satisficing: every candidate must satisfy a reward floor, and the batch must satisfy a diversity cutoff. Varying the floor provably traces a satisfaction–diversity Pareto frontier. On FLUX.1-dev with HPSv3 and SANA-1.6B with ImageReward, SatisDive's curves Pareto-dominate FK steering's over their overlapping DreamSim ranges, improving worst-candidate reward at matched DreamSim by up to 0.43 and 0.70, respectively. Extending the formulation to multiple reward floors remains future work.

AI USE STATEMENT

Generative AI tools were used to assist with manuscript editing, code implementation, developing and checking mathematical arguments, and planning and analyzing experiments. The authors made all scientific decisions, verified the incorporated outputs, and take responsibility for the final content.

ETHICS STATEMENT

We use prompts from publicly released text-to-image benchmarks and collect no new human-subject data. Because these benchmarks contain user-written prompts, the prompts and generated images may reflect biases or inappropriate content present in the source data and pretrained models.

REPRODUCIBILITY STATEMENT

Algorithm 1 specifies SatisDive. Appendix B provides the proofs, Appendix C reports the model, sampling, optimization, and baseline settings, and Appendix D describes the evaluation protocols and additional analyses. Anonymous code is available at `https://anonymous.4open.science/r/iclr2027-F772/`.

REFERENCES

Donghoon Ahn, Hyoungwon Cho, Jaewon Min, Wooseok Jang, Jungwoo Kim, SeonHwa Kim, Hyun Hee Park, Kyong Hwan Jin, and Seungryong Kim. Self-rectifying diffusion sampling with perturbed-attention guidance. In *European Conference on Computer Vision*, pp. 1–17. Springer, 2024.

Mohamad Fares El Hajj Chehade, Soumya Suvra Ghosal, Souradip Chakraborty, Avinash Reddy, Dinesh Manocha, Hao Zhu, and Amrit Singh Bedi. Bounded rationality for LLMs: Satisficing alignment at inference-time. In *Forty-second International Conference on Machine Learning*, 2025. URL `https://openreview.net/forum?id=cEhLObwvvu`.

Gabriele Corso, Yilun Xu, Valentin De Bortoli, Regina Barzilay, and Tommi Jaakkola. Particle guidance: Non-I.I.D. diverse sampling with diffusion models. In *International Conference on Learning Representations (ICLR)*, 2024. arXiv:2310.13102.

Luca Eyring, Shyamgopal Karthik, Karsten Roth, Alexey Dosovitskiy, and Zeynep Akata. Reno: Enhancing one-step text-to-image models through reward-based noise optimization. *Advances in Neural Information Processing Systems*, 37:125487–125519, 2024.

Stephanie Fu, Netanel Yakir Tamir, Shobhita Sundaram, Lucy Chai, Richard Zhang, Tali Dekel, and Phillip Isola. Dreamsim: Learning new dimensions of human visual similarity using synthetic data. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL `https://openreview.net/forum?id=DEiNSfh1k7`.

Shelly Golan, Michael Finkelson, Ariel Bereslavsky, Yotam Nitzan, and Or Patashnik. Paretoslider: Diffusion models post-training for continuous reward control. *arXiv preprint arXiv:2604.20816*, 2026.

Anne Harrington, A. Sophia Koepke, Shyamgopal Karthik, Trevor Darrell, and Alexei A. Efros. It's never too late: Noise optimization for collapse recovery in trained diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026. URL `https://arxiv.org/abs/2601.00090`. arXiv:2601.00090.

Jiajun He, Paul Jeha, Peter Potaptchik, Leo Zhang, José Miguel Hernández Lobato, Yuanqi Du, Saifuddin Syed, and Francisco Vargas. Crepe: Controlling diffusion with replica exchange. In *International Conference on Learning Representations*, volume 2026, pp. 99464–99494, 2026.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021. URL `https://openreview.net/forum?id=qw8AKxfYbI`.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pp. 6840–6851, 2020. arXiv:2006.11239.

Peter Holderrieth, Uriel Singer, Tommi Jaakkola, Ricky TQ Chen, Yaron Lipman, and Brian Karrer. Glass flows: Efficient inference for reward alignment of flow and diffusion models. In *The Fourteenth International Conference on Learning Representations*, 2026.

Susung Hong. Smoothed energy guidance: Guiding diffusion models with reduced energy curvature of attention. *Advances in Neural Information Processing Systems*, 37:66743–66772, 2024.

C-L Hwang and Abu Syed Md Masud. *Multiple objective decision making—methods and applications: a state-of-the-art survey*. Springer Science & Business Media, 2012.

Jisung Hwang, Yunhong Min, Jaihoon Kim, I-Chao Shen, and Minhyuk Sung. Noisetilt: Noise-tilted reverse kernels for diffusion reward alignment, 2026. URL https://arxiv.org/abs/2606.18066.

Vineet Jain, Kusha Sareen, Mohammad Pedramfar, and Siamak Ravanbakhsh. Diffusion tree sampling: Scalable inference-time alignment of diffusion models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025. URL https://openreview.net/forum?id=3D88hCO0Gd.

Sadeep Jayasumana, Srikumar Ramalingam, Andreas Veit, Daniel Glasner, Ayan Chakrabarti, and Sanjiv Kumar. Rethinking fid: Towards a better evaluation metric for image generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9307–9315, 2024.

Rohit Jena, Ali Taghibakhshi, Sahil Jain, Gerald Shen, Nima Tajbakhsh, and Arash Vahdat. Elucidating optimal reward-diversity tradeoffs in text-to-image diffusion models. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 232–242. IEEE, 2025.

Tero Karras, Miika Aittala, Tuomas Kynkäänniemi, Jaakko Lehtinen, Timo Aila, and Samuli Laine. Guiding a diffusion model with a bad version of itself. *Advances in Neural Information Processing Systems*, 37:52996–53021, 2024.

Shervin Khalafi, Ignacio Hounie, Dongsheng Ding, and Alejandro Ribeiro. Composition and alignment of diffusion models using constrained learning. *Advances in Neural Information Processing Systems*, 38:18629–18676, 2025.

Byungjun Kim, Soobin Um, and Jong Chul Ye. Diverse text-to-image generation via contrastive noise optimization. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=EVRMnAREc3.

Jaihoon Kim, TaeHoon Yoon, Jisung Hwang, and Minhyuk Sung. Inference-time scaling for flow models via stochastic generation and rollover budget forcing. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025a. URL https://openreview.net/forum?id=quY3zRPalR.

Sunwoo Kim, Minkyu Kim, and Dongmin Park. Test-time alignment of diffusion models without reward over-optimization. In *The Thirteenth International Conference on Learning Representations*, 2025b. URL https://openreview.net/forum?id=vi3DjUhFVm.

Yuval Kirstain, Adam Polyak, Uriel Singer, Shahbuland Matiana, Joe Penna, and Omer Levy. Pick-a-pic: An open dataset of user preferences for text-to-image generation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. URL https://arxiv.org/abs/2305.01569.

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. *Advances in Neural Information Processing Systems*, 37:122458–122483, 2024.

Black Forest Labs. Flux. https://github.com/black-forest-labs/flux, 2024.

Xiang Li, Dianbo Liu, and Kenji Kawaguchi. Initialization is half the battle: Generating diverse images from a guidance potential posterior. In *International Conference on Machine Learning (ICML)*, 2026. URL https://arxiv.org/abs/2606.02453. Spotlight; arXiv:2606.02453.

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 (NeurIPS)*, 2025. URL https://openreview.net/forum?id=6QbbaEGkO7.

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 *European conference on computer vision*, pp. 740–755. Springer, 2014.

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

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *International conference on learning representations (ICLR)*, 2023.

Yuhang Ma, Xiaoshi Wu, Keqiang Sun, and Hongsheng Li. Hpsv3: Towards wide-spectrum human preference score. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 15086–15095, October 2025.

Konstantin Mark, Leonard Galustian, Maximilian P-P Kovar, and Esther Heid. Feynman-kac-flow: Inference steering of conditional flow matching to an energy-tilted posterior. *arXiv preprint arXiv:2509.01543*, 2025.

Gaurav Parmar, Or Patashnik, Daniil Ostashev, Kuan-Chieh Wang, Kfir Aberman, Srinivasa Narasimhan, and Jun-Yan Zhu. Scaling group inference for diverse and high-quality generation. In *International Conference on Learning Representations*, volume 2026, pp. 89763–89790, 2026.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *2022 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 10674–10685. ieee, 2022.

Seyedmorteza Sadat, Jakob Buhmann, Derek Bradley, Otmar Hilliges, and Romann Weber. Cads: Unleashing the diversity of diffusion models through condition-annealed sampling. In *International Conference on Learning Representations*, volume 2024, pp. 23723–23755, 2024.

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.

Shivanshu Shekhar, Sagnik Mukherjee, Jia Yi Zhang, and Tong Zhang. Vasr: Variance-aware systematic resampling for reward-guided diffusion, 2026. URL https://arxiv.org/abs/2604.06779v2. arXiv:2604.06779v2 (distinct from FVD, the v1 of the same arXiv id).

Herbert A. Simon. Rational choice and the structure of the environment. *Psychological Review*, 63 (2):129–138, 1956.

Jaskirat Singh, Lindsey Li, Weijia Shi, Ranjay Krishna, Yejin Choi, Pang Wei Koh, Michael F Cohen, Stephen Gould, Liang Zheng, and Luke Zettlemoyer. Negative token merging: Image-based adversarial feature guidance. *arXiv preprint arXiv:2412.01339*, 2024.

Raghav Singhal, Zachary Horvitz, Ryan Teehan, Mengye Ren, Zhou Yu, Kathleen McKeown, and Rajesh Ranganath. A general framework for inference-time scaling and steering of diffusion models. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=Jp988ELppQ.

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

Zhiwei Tang, Jiangweizhi Peng, Jiasheng Tang, Mingyi Hong, Fan Wang, and Tsung-Hui Chang. Inference-time alignment of diffusion models with direct noise optimization. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=JpbqiD7n9r.

Gal Vinograd, Idan Achituve, and Ethan Fetaya. Diverse sampling in diffusion models with marginal preserving particle guidance, 2026. URL https://arxiv.org/abs/2605.06553.

Luhuan Wu, Brian L. Trippe, Christian A. Naesseth, David M. Blei, and John P. Cunningham. Practical and asymptotically exact conditional sampling in diffusion models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023a. URL https://openreview.net/forum?id=r9s3Gbxz7g.

Xiaoshi Wu, Yiming Hao, Keqiang Sun, Yixiong Chen, Feng Zhu, Rui Zhao, and Hongsheng Li. Human preference score v2: A solid benchmark for evaluating human preferences of text-to-image synthesis. *arXiv preprint arXiv:2306.09341*, 2023b.

Enze Xie, Junsong Chen, Junyu Chen, Han Cai, Haotian Tang, Yujun Lin, Zhekai Zhang, Muyang Li, Ligeng Zhu, Yao Lu, et al. Sana: Efficient high-resolution text-to-image synthesis with linear diffusion transformers. In *The Thirteenth International Conference on Learning Representations*, 2025.

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. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Ankit Yadav, Arpit Garg, Ta Duc Huy, and Lingqiao Liu. Stride: Training-free diversity guidance via pca-directed feature perturbation in single-step diffusion models. *arXiv preprint arXiv:2605.11494*, 2026.

Yinghua Yao, Yuangang Pan, Jing Li, Ivor Tsang, and Xin Yao. Proud: Pareto-guided diffusion model for multi-objective generation. *Machine Learning*, 113(9):6511–6538, 2024.

Jiwen Yu, Yinhuai Wang, Chen Zhao, Bernard Ghanem, and Jian Zhang. Freedom: Training-free energy-guided conditional diffusion model. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 23117–23127. IEEE, 2023.

Kevin Zhai, Utsav Singh, Anirudh Thatipelli, Souradip Chakraborty, Anit Kumar Sahu, Furong Huang, Amrit Singh Bedi, and Mubarak Shah. Mira: Towards mitigating reward hacking in inference-time alignment of t2i diffusion models. *arXiv preprint arXiv:2510.01549*, 2025.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *2018 IEEE/CVF conference on computer vision and pattern recognition*, pp. 586–595. IEEE, 2018.

APPENDIX CONTENTS

A Notation **15**

B Theoretical Results and Proofs **16**

B.1 Reward Tilting 16

B.2 Reward-Floor Frontier 17

B.3 Zero-Temperature Limit of SatisDive's Objective 19

C Experiment Settings **21**

C.1 2D Toy Setup 22

D Additional Experiments **23**

D.1 Effect of Increasing the Number of Candidates 23

D.2 Additional Results on DrawBench 23

D.3 Effect of a Higher Reward Floor 24

D.4 CMMD on MS-COCO 24

D.5 Component Ablations 24

D.6 FLUX.1-dev with ImageReward 25

D.7 Stable Diffusion 1.5 with ImageReward 25

D.8 Comparison with CREPE 26

D.9 Comparison with SGI 26

D.10 Inference-Time Computation 27

D.11 Reward-Temperature Robustness 28

D.12 Reward Capping without the Diversity Penalty 29

E Additional Qualitative Comparisons **29**

A NOTATION

We summarize the notation used throughout the paper.

Table 4: Summary of Notation.

Symbol	Description
c	Text prompt
K	Number of candidates in a batch
k	Candidate index
T	Number of sampling steps
s	Discrete denoising-step index, $s = 1, \dots, T$
t	Model time, decreasing from 1 (noise) to 0 (image)
x_t^k	Latent of candidate k at time t
$x_0^{1:K}$	Batch of K generated images
$\hat{x}_0^k$	Model's current estimate of candidate k's final image
$p_\theta(x_0 \mid c)$	Output distribution of the frozen base text-to-image model
$v_\theta(x_t, t, c)$	Velocity field of a flow-based text-to-image model
$r(x_0, c)$	Learned reward for image x_0 and prompt c
r_k	Reward of candidate k
ρ	Reward floor used to define satisfaction
τ	Batch-relative reward cutoff, $\tau = \max_k r_k - \Delta$
Δ	Tolerance controlling the gap between the largest reward and τ
q_k	Smooth gate indicating whether candidate k exceeds τ
d_{ij}	Cosine distance between the features of candidates i and j
D	Reward-gated mean pairwise distance; in the hard limit, this is the unweighted mean on admissible batches
δ	Diversity cutoff
β_r	Reward temperature
β_d	Diversity temperature
$\mathcal{J}_r$	Reward penalty
$\mathcal{J}_d$	Diversity penalty
$\mathcal{J}$	Sum of the reward and diversity penalties
$\mathcal{G}$	Sampling-step indices at which candidates are scored and updated using $\mathcal{J}$
r_k^{prev}	Reward of candidate k at the previous update step
m	Number of update steps remaining
$\tilde{r}_k$	Extrapolated reward of candidate k
p_λ	Reward-tilted output distribution with tilt strength λ
$\pi^\star$	Hard satisficing target distribution over batches
A_ρ	Images clearing reward floor ρ
p_ρ	Base output distribution restricted to A_ρ

B THEORETICAL RESULTS AND PROOFS

This section first analyzes reward tilting, then characterizes the reward-floor frontier, and finally proves the zero-temperature limit of SatisDive's objective.

Fix a prompt c. Let $p = p_\theta(\cdot \mid c)$ denote the base-model distribution, and write $R(x) = r(x, c)$ for the reward. Rewards are measurable and real-valued. All image distributions below have densities with respect to the same reference measure, denoted by dx in integrals. We use the same symbol for a distribution and its density. For a measurable set E, its probability is $p(E) = \int_E p(x)dx$, and E^c denotes its complement. An indicator $\mathbf{1}_E(x)$ is one on E and zero elsewhere.

We assume that the batch size $K \geq 2$ is a fixed finite integer and write $x^{1:K} = (x^1, \ldots, x^K)$. Candidate indices satisfy $k, \ell \in \{1, \ldots, K\}$, and pair indices satisfy $1 \leq i < j \leq K$. For a real reward floor ρ, let $A_\rho = \{x : R(x) \geq \rho\}$ be the set of images satisfying the floor. Its base-model probability is $Z_\rho = p(A_\rho)$. A floor is feasible when $Z_\rho > 0$. For such a floor, the base distribution restricted to A_ρ is

$$p_\rho(x) = \frac{p(x)\mathbf{1}[x \in A_\rho]}{Z_\rho}. \tag{12}$$

B.1 REWARD TILTING

Precise conditions for Proposition 1. At a given resampling step, fix K candidates $x^{1:K}$ with rewards $r_k = R(x^k)$, and let $j^\star$ be the index of the unique highest-reward candidate. Let $\lambda \geq 0$ be the reward-tilt strength and $a_k > 0$ a finite factor in candidate k's resampling mass that does not depend on λ; setting every $a_k = 1$ recovers the pure exponential reward weighting in Eq. 1. Define the normalized resampling mass of candidate k as

$$\omega_k(\lambda) = \frac{a_k e^{\lambda r_k}}{\sum_{\ell=1}^K a_\ell e^{\lambda r_\ell}}.$$

Let $N_k \in \{0, \ldots, K\}$ be the number of copies of candidate k after resampling. We assume that resampling returns K candidates from this fixed pool, so $\sum_k N_k = K$, and that $\mathbb{E}[N_k] = K\omega_k(\lambda)$ for every k. We do not assume that the selections are independent.

Proof of Proposition 1. All probabilities and expectations in this proof concern only the randomness of resampling from the fixed pool above. For any $\ell \neq j^\star$, the shared normalization cancels:

$$\frac{\omega_\ell(\lambda)}{\omega_{j^\star}(\lambda)} = \frac{a_\ell e^{\lambda r_\ell}}{a_{j^\star} e^{\lambda r_{j^\star}}} = \frac{a_\ell}{a_{j^\star}} e^{-\lambda(r_{j^\star} - r_\ell)} \longrightarrow 0. \tag{13}$$

The limit holds because $r_{j^\star} - r_\ell > 0$. Dividing the numerator and denominator of $\omega_{j^\star}$ by $a_{j^\star} e^{\lambda r_{j^\star}}$ gives

$$\omega_{j^\star}(\lambda) = \left(1 + \sum_{\ell \neq j^\star} \frac{a_\ell}{a_{j^\star}} e^{-\lambda(r_{j^\star} - r_\ell)}\right)^{-1} \longrightarrow 1.$$

Each summand decreases to zero, so the denominator decreases to one. Its reciprocal and the expected copy count $K\omega_{j^\star}(\lambda)$ therefore increase.

Because resampling returns exactly K candidates, $N_{j^\star} < K$ if and only if some other candidate survives. Also, $\mathbf{1}[N_\ell \geq 1] \leq N_\ell$, so taking expectations gives $\Pr[N_\ell \geq 1] \leq \mathbb{E}[N_\ell]$. The union bound and the expected copy counts therefore give

$$\begin{aligned}\Pr[N_{j^\star} < K] = \Pr\left[\bigcup_{\ell \neq j^\star} \{N_\ell \geq 1\}\right] &\leq \sum_{\ell \neq j^\star} \Pr[N_\ell \geq 1] \\ &\leq \sum_{\ell \neq j^\star} \mathbb{E}[N_\ell] = K \sum_{\ell \neq j^\star} \omega_\ell = K(1 - \omega_{j^\star}) \longrightarrow 0.\end{aligned}$$

The last equality uses $\sum_k \omega_k = 1$, and the limit uses $\omega_{j^\star} \to 1$. Taking the complementary event gives $\Pr[N_{j^\star} = K] = 1 - \Pr[N_{j^\star} < K] \to 1$.

Multinomial, Srinivasan Sampling Process (SSP), and systematic resampling satisfy the expectation condition. The conclusion also holds if VASR-MAX's redirection rule is applied after this systematic resampling: when resampling returns K copies of the highest-reward candidate, the redirection rule does not change the result. □

For the population-level comparison, assume R is bounded above and let $Z(\lambda) = \mathbb{E}_p[e^{\lambda R}]$. The normalized tilt in Eq. (1) is $p_\lambda(x) = p(x)e^{\lambda R(x)}/Z(\lambda)$.

Corollary 4 (A finite reward tilt does not enforce the reward floor)**.** *Fix ρ such that $0 < p(A_\rho) < 1$. Then $p_\lambda(A_\rho) < 1$ for every finite $\lambda \geq 0$.*

Proof. Let B be a finite upper bound for R. Since $0 < e^{\lambda R} \leq e^{\lambda B}$ and $\int p(x)dx = 1$, taking expectations gives $0 < Z(\lambda) \leq e^{\lambda B} < \infty$. The tilt therefore preserves positive probability below the floor:

$$p_\lambda(A_\rho^c) = \frac{\int_{A_\rho^c} e^{\lambda R(x)} p(x)dx}{Z(\lambda)} > 0, \qquad p_\lambda(A_\rho) = 1 - p_\lambda(A_\rho^c) < 1.$$

□

B.2 Reward-Floor Frontier

In this subsection, support means the set on which a density is positive, with sets of zero base-model probability ignored; $\mathrm{supp}(\nu)$ denotes the support of a distribution ν. A restriction is proper when it removes a set with positive base-model probability. The theorem's diversity assumption compares distributions by inclusion of these supports, including distributions that assign different probabilities to the same support. In particular, equal supports have equal diversity: each support is contained in the other, so both diversity inequalities hold. This distribution-level diversity is distinct from the per-batch statistic D used in the objective.

We use $\mathrm{KL}(\nu\|p) = \int \nu(x) \log(\nu(x)/p(x))dx$, with zero-density contributions taken as zero. This divergence is $+\infty$ if ν assigns positive probability to any set of zero p probability. Thus finite divergence requires ν to be supported within p. All support comparisons and equalities between distributions below ignore sets of zero base-model probability.

The factorization in Eq. (10) follows by substituting Eq. (12) for each candidate:

$$\prod_{k=1}^{K} \big(p(x^k)\mathbf{1}[x^k \in A_\rho]\big) = \prod_{k=1}^{K} \big(Z_\rho p_\rho(x^k)\big) = Z_\rho^K \prod_{k=1}^{K} p_\rho(x^k).$$

Multiplying both sides by the batch diversity indicator in Eq. (4) preserves this equality. Since $Z_\rho^K > 0$ is constant with respect to $x^{1:K}$, it is absorbed into the batch normalizer. The diversity indicator remains in the target, exactly as in Eq. (10).

Lemma 5 (Among distributions over A_ρ, p_ρ is uniquely closest to the base model)**.** *For a feasible floor ρ, the distribution p_ρ uniquely minimizes $\mathrm{KL}(\cdot\|p)$ among distributions that assign probability one to A_ρ. Moreover, every such distribution ν with finite $\mathrm{KL}(\nu\|p)$ has support contained in $\mathrm{supp}(p_\rho)$.*

Proof. Wherever p_ρ is positive, $p_\rho/p = 1/Z_\rho$. Substituting this ratio and using $\int_{A_\rho} p_\rho(x)dx = 1$ gives

$$\mathrm{KL}(p_\rho\|p) = \int_{A_\rho} p_\rho(x) \log(1/Z_\rho)dx = -\log Z_\rho \int_{A_\rho} p_\rho(x)dx = -\log Z_\rho < \infty. \tag{14}$$

An infinite-KL distribution cannot improve this finite value. Consider any ν with finite KL to p and $\nu(A_\rho) = 1$. Finite KL requires its support to lie within that of p. The reward constraint also confines it to A_ρ. Its support therefore lies within their intersection, which is exactly the support of p_ρ.

The ratios below are consequently defined almost everywhere under ν. We insert $p_\rho/p = 1/Z_\rho$ into the density ratio and split the logarithm. The constant $\log Z_\rho$ can then be taken outside the integral.

Finally, $\int_{A_\rho} \nu(x)dx = 1$ gives

$$\begin{aligned}
\mathrm{KL}(\nu\|p) &= \int_{A_\rho} \nu(x) \log \frac{\nu(x)}{p(x)} dx \\
&= \int_{A_\rho} \nu(x) \log \left(\frac{\nu(x)}{p_\rho(x)} \frac{1}{Z_\rho} \right) dx \\
&= \int_{A_\rho} \nu(x) \log \frac{\nu(x)}{p_\rho(x)} dx - \log Z_\rho \int_{A_\rho} \nu(x)dx \\
&= \mathrm{KL}(\nu\|p_\rho) - \log Z_\rho. \qquad (15)
\end{aligned}$$

KL divergence is nonnegative, with equality only for identical distributions. Thus $\mathrm{KL}(\nu\|p) \geq -\log Z_\rho = \mathrm{KL}(p_\rho\|p)$, with equality only for $\nu = p_\rho$. This proves the unique minimum; support containment was established above. □

Proof of Theorem 3. We first compare diversity at different reward floors. We then show that no admissible distribution dominates p_ρ, and that the family represents every frontier point in the stated comparison class.

Diversity at each floor. By Eq. (12), p_ρ assigns probability one to A_ρ and is positive exactly where p is positive within A_ρ. Let $\rho' \geq \rho$ be another feasible floor. Since $R(x) \geq \rho'$ implies $R(x) \geq \rho$, we have $A_{\rho'} \subseteq A_\rho$. Intersecting these sets with the support of p gives

$$\mathrm{supp}(p_{\rho'}) = \mathrm{supp}(p) \cap A_{\rho'} \subseteq \mathrm{supp}(p) \cap A_\rho = \mathrm{supp}(p_\rho). \qquad (16)$$

By assumption, restricting support cannot increase diversity, and a proper restriction strictly decreases it. Equation (16) therefore implies that $p_{\rho'}$ is no more diverse than p_ρ. The inequality is strict when $p(A_\rho \setminus A_{\rho'}) > 0$. Otherwise, the two conditioning sets differ only on a set of zero base-model probability, so $p_{\rho'} = p_\rho$ almost everywhere.

Now consider any distribution that assigns probability one to A_ρ and has finite divergence to p. Lemma 5 places its support within $\mathrm{supp}(p_\rho)$. Its diversity therefore cannot exceed the diversity of p_ρ. Thus, p_ρ has maximal diversity at each floor.

The highest guaranteed reward. For Pareto optimality, satisfaction is the highest reward floor satisfied with probability one. The specified ρ need not be the highest floor that p_ρ satisfies. We therefore write $g = \sup\{a \in \mathbb{R} : p_\rho(R \geq a) = 1\}$ for the supremum of the guaranteed floors. Since ρ itself is guaranteed, $g \geq \rho$.

First, g is finite. If it were infinite, every event $\{R \geq n\}$, for positive integers n, would have probability one. These events decrease to the empty set because rewards are real-valued. Continuity of probability would then give probability one to the empty set, a contradiction.

Next, the floor g is itself guaranteed. For each positive integer n, the supremum definition supplies a guaranteed floor greater than $g - 1/n$. This implies $p_\rho(R \geq g - 1/n) = 1$. These events decrease to $\{R \geq g\}$, so continuity of probability gives

$$p_\rho(R \geq g) = \lim_{n\to\infty} p_\rho(R \geq g - 1/n) = 1.$$

The same decreasing-event argument applies to any distribution with finite guaranteed reward.

No competitor dominates p_ρ. A distribution dominates another if neither satisfaction nor diversity is lower and at least one is higher. Consider a distribution ν with $\mathrm{KL}(\nu\|p) < \infty$ that guarantees reward at least g. Since $g \geq \rho$, it satisfies $\nu(A_\rho) = 1$. Lemma 5 therefore places its support within that of p_ρ, so ν cannot have greater diversity.

It remains to rule out higher satisfaction with the same diversity. Suppose ν guarantees a strictly higher floor $h > g$. Then $p_\rho(R < h) > 0$: otherwise h would also be guaranteed under p_ρ, contradicting the definition of g. In contrast, $\nu(R < h) = 0$. Since p_ρ assigns probability one to A_ρ, substituting its density from Eq. (12) gives

$$p(A_\rho \setminus A_h) = Z_\rho p_\rho(R < h) > 0, \qquad \nu(A_\rho \setminus A_h) \leq \nu(R < h) = 0.$$

The competitor therefore excludes a set of positive base-model probability from the support of p_ρ. By strict support monotonicity, ν has lower diversity. No competitor can improve either guaranteed

reward or diversity without reducing the other, so p_ρ is Pareto-optimal. Floors producing the same distribution give the same frontier point.

Every frontier point is represented. Take any ν with finite divergence to p and finite guaranteed reward $h = \sup\{a \in \mathbb{R} : \nu(R \geq a) = 1\}$. The decreasing-event argument above gives $\nu(A_h) = 1$. We must have $Z_h = p(A_h) > 0$: otherwise ν would assign probability one to a set of zero p probability, contradicting finite KL. Thus p_h exists. Lemma 5 gives $\mathrm{supp}(\nu) \subseteq \mathrm{supp}(p_h)$, so p_h has at least as much diversity as ν.

We also need to check that p_h has the same satisfaction as ν. It guarantees reward at least h by construction. Suppose it guaranteed some floor $b > h$. Its conditional density would give $p(A_h \setminus A_b) = Z_h p_h(R < b) = 0$. Finite KL would then imply $\nu(A_h \setminus A_b) = 0$. Together with $\nu(A_h) = 1$, this gives $\nu(A_b) = 1$, contradicting the definition of h. Therefore, the highest guaranteed reward of p_h is exactly h.

We have shown that p_h has the same satisfaction as ν and at least as much diversity. A proper support inclusion gives strictly greater diversity, so such a ν is dominated by p_h. Equal supports give the same diversity and hence the same satisfaction–diversity pair. Every undominated pair of finite satisfaction and diversity is therefore represented by the family, although more than one distribution can represent the same pair.

□

B.3 Zero-Temperature Limit of SatisDive's Objective

Proof of Proposition 2. We first show that the objective's weight approaches the indicator of a feasible batch. We then show that its normalizing constant converges to the probability of that event. Finally, we show that the normalized distribution converges to $\pi^\star$.

Setup. Fix a current candidate batch and let M be its maximum reward. Fix the tolerance $\Delta > 0$ and set $\tau = \rho = M - \Delta$. Also fix the diversity cutoff $\delta \in \mathbb{R}$ for this objective evaluation. These values remain fixed as we integrate over batches and take temperature limits. In particular, M is not recomputed for each integration variable.

For a generic batch $x^{1:K}$, write $r_k = R(x^k)$ for candidate k's reward. The density of a batch of independent base-model samples is $P_c(x^{1:K}) = \prod_{k=1}^K p(x^k)$. Integrals over batches use the product reference measure $dx^1 \cdots dx^K$.

Let $d_{ij} = d_{ij}(x^{1:K})$ be the measurable cosine distances between candidates' features. We hold the feature map fixed and assume these distances are well defined. They lie in $[0, 2]$ and do not depend on the temperatures. There are $\binom{K}{2} = K(K-1)/2$ candidate pairs. For a reward temperature $\beta_r > 0$, write the reward gate and weighted diversity as

$$q_k = \frac{1}{1 + e^{(\rho - r_k)/\beta_r}}, \qquad D_{\beta_r} = \frac{\sum_{i<j} q_i q_j d_{ij}}{\sum_{i<j} q_i q_j}.$$

Here D_{β_r} is the D in Eq. (8). The subscript makes its dependence on the reward temperature explicit. The hard target instead uses the unweighted mean $D_0 = \binom{K}{2}^{-1} \sum_{i<j} d_{ij}$. We will show below that $D_{\beta_r} \to D_0$ whenever all rewards exceed the floor.

Define the feasible batch set $C = \{x^{1:K} : r_k \geq \rho \text{ for every } k,\ D_0 \geq \delta\}$. The proposition assumes $P_c(C) > 0$, so conditioning on this set is possible. It also assumes $P_c(r_k = \rho) = 0$ for every k and $P_c(D_0 = \delta) = 0$. Since there are finitely many candidates, the union of these boundary events also has probability zero. We may therefore establish the limit outside these events.

For temperatures $\beta_r, \beta_d > 0$, define the weight $w_{\beta_r,\beta_d} = \exp(-\mathcal{J})$ using Eq. (9). Its normalizing constant is $Z_{\beta_r,\beta_d} = \mathbb{E}_{P_c}[w_{\beta_r,\beta_d}]$.

These quantities are well defined at every positive temperature. Every q_k lies in $(0, 1)$, and $K \geq 2$ ensures that there is at least one pair. Thus the denominator of D_{β_r} is positive, and its weighted average is finite. Both penalties are consequently finite and nonnegative, giving $0 < w_{\beta_r,\beta_d} \leq 1$.

Taking expectations gives $0 < Z_{\beta_r,\beta_d} \leq 1$. The induced distribution is therefore

$$\widetilde{\pi}_{\beta_r,\beta_d}(x^{1:K}) = \frac{P_c(x^{1:K}) w_{\beta_r,\beta_d}(x^{1:K})}{Z_{\beta_r,\beta_d}}. \tag{17}$$

1. Convergence of the weights. Equation (7) gives $\mathcal{J}_r = -\sum_k \log q_k$. We substitute this and $\mathrm{softplus}(z) = \log(1 + e^z)$ into $\exp(-\mathcal{J}_r - \mathcal{J}_d)$. Exponentiating the logarithms gives the quotient below. Rewriting its denominator using $\mathrm{sigmoid}(z) = 1/(1 + e^{-z})$ gives a product of reward gates and a diversity gate:

$$\begin{aligned} w_{\beta_r,\beta_d} &= e^{\sum_k \log q_k}\, e^{-\log(1+e^{(\delta - D_{\beta_r})/\beta_d})} \\ &= \frac{\prod_{k=1}^K q_k}{1 + e^{(\delta - D_{\beta_r})/\beta_d}} = \left(\prod_{k=1}^K q_k\right) \mathrm{sigmoid}\left(\frac{D_{\beta_r} - \delta}{\beta_d}\right). \end{aligned} \tag{18}$$

Take $\beta_r, \beta_d \to 0$ through positive values. Outside the boundary events, there are two reward cases. If any candidate has $r_k < \rho$, then $e^{(\rho - r_k)/\beta_r} \to \infty$, so $q_k \to 0$. All factors in Eq. (18) lie between zero and one. Hence $0 \leq w_{\beta_r,\beta_d} \leq q_k \to 0$: one failed reward requirement makes the batch weight vanish, regardless of diversity.

Otherwise every $r_k > \rho$, so $e^{(\rho - r_k)/\beta_r} \to 0$ and every $q_k \to 1$. Each pair weight $q_i q_j$ then tends to one. Taking limits in the finite sums, whose denominator tends to $\binom{K}{2} > 0$, gives

$$\lim_{\beta_r \to 0} D_{\beta_r} = \frac{\sum_{i<j} 1 \cdot 1 \cdot d_{ij}}{\sum_{i<j} 1 \cdot 1} = \frac{\sum_{i<j} d_{ij}}{\binom{K}{2}} = D_0. \tag{19}$$

We must now check the diversity gate, whose numerator and temperature both change. Since $D_0 \neq \delta$, the limiting diversity has a nonzero gap from the cutoff. Convergence to D_0 ensures that, for sufficiently small β_r, $|D_{\beta_r} - D_0| < |D_0 - \delta|/2$. Thus D_{β_r} stays on the same side of δ as D_0. The triangle inequality also bounds its distance from the cutoff:

$$|D_{\beta_r} - \delta| \geq |D_0 - \delta| - |D_{\beta_r} - D_0| > \frac{|D_0 - \delta|}{2} > 0.$$

The positive lower bound does not depend on either temperature. Dividing by $\beta_d \to 0$ therefore makes the magnitude of the sigmoid argument diverge. When $D_0 > \delta$, its sign is positive and the diversity gate tends to one. When $D_0 < \delta$, its sign is negative and the gate tends to zero. This holds regardless of the relative temperature rates.

Combining these diversity cases with the reward cases gives

$$\lim_{\beta_r,\beta_d \to 0} w_{\beta_r,\beta_d} = \left(\prod_k \mathbf{1}[r_k \geq \rho]\right) \mathbf{1}[D_0 \geq \delta] = \mathbf{1}_C \quad \text{almost everywhere under } P_c. \tag{20}$$

2. Convergence of the normalizing constant. To normalize the limiting weights, we also need the limit of their expectation. Since $0 \leq w_{\beta_r,\beta_d} \leq 1$ and P_c is a probability distribution, the constant one is an integrable bound. Dominated convergence therefore allows us to move the limit inside the expectation in Eq. (20):

$$\lim_{\beta_r,\beta_d \to 0} Z_{\beta_r,\beta_d} = \lim_{\beta_r,\beta_d \to 0} \mathbb{E}_{P_c}[w_{\beta_r,\beta_d}] = \mathbb{E}_{P_c}[\mathbf{1}_C] = P_c(C) > 0. \tag{21}$$

3. Convergence of the normalized distribution. The target in Eq. (4) is the base-batch distribution conditioned on C:

$$\pi^\star(x^{1:K}) = \frac{P_c(x^{1:K}) \mathbf{1}_C(x^{1:K})}{P_c(C)}. \tag{22}$$

Equations (20) and (21) give $w_{\beta_r,\beta_d}/Z_{\beta_r,\beta_d} \to \mathbf{1}_C/P_c(C)$ almost everywhere under P_c. Division is valid because the limiting normalizer $P_c(C)$ is positive.

To conclude convergence in total variation, we need to integrate the absolute difference of these ratios. For sufficiently small temperatures, convergence of the normalizer ensures $Z_{\beta_r,\beta_d} \geq P_c(C)/2$. The triangle inequality and $0 \leq w_{\beta_r,\beta_d}, \mathbf{1}_C \leq 1$ then give

$$\left|\frac{w_{\beta_r,\beta_d}}{Z_{\beta_r,\beta_d}} - \frac{\mathbf{1}_C}{P_c(C)}\right| \leq \frac{w_{\beta_r,\beta_d}}{Z_{\beta_r,\beta_d}} + \frac{\mathbf{1}_C}{P_c(C)} \leq \frac{2}{P_c(C)} + \frac{1}{P_c(C)} = \frac{3}{P_c(C)}.$$

This finite constant is integrable under the probability distribution P_c, so dominated convergence applies again. Total variation is half the integral of the absolute density difference. We substitute the densities from Eqs. (17) and (22) into this integral. Factoring out the common density P_c turns the integral into an expectation:

$$2\left\|\widetilde{\pi}_{\beta_r,\beta_d}-\pi^\star\right\|_{\mathrm{TV}}=\int\left|\widetilde{\pi}_{\beta_r,\beta_d}(x^{1:K})-\pi^\star(x^{1:K})\right|dx^1\cdots dx^K$$
$$=\mathbb{E}_{P_c}\left[\left|\frac{w_{\beta_r,\beta_d}}{Z_{\beta_r,\beta_d}}-\frac{\mathbf{1}_C}{P_c(C)}\right|\right]\longrightarrow 0. \tag{23}$$

The integrand tends to zero by the ratio limit above. Dividing by two proves convergence to $\pi^\star$ in total variation as $\beta_r,\beta_d\to 0$. □

Pareto optimality of the batch targets. We now apply the support argument to distributions over whole batches, retaining the fixed diversity requirement. Fix P_c, D_0, and δ as above. Write $S(x^{1:K})=\min_k R(x^k)$ for worst-candidate reward. The feasible set is then $C=\{S\geq\rho,\ D_0\geq\delta\}$, and $\pi^\star=P_c\mathbf{1}_C/P_c(C)$, with $P_c(C)>0$.

Assume distribution-level diversity is nonincreasing under support restriction. Assume also that it strictly decreases when a set of positive P_c probability is removed. This diversity compares distributions over batches; it is distinct from the per-batch statistic D_0.

The conditioning calculation in Lemma 5 uses only the positive probability of the conditioning event. We can therefore substitute P_c, C, and $\pi^\star$ for p, A_ρ, and p_ρ. This gives $\mathrm{KL}(\pi^\star\|P_c)=-\log P_c(C)<\infty$. It also shows that every finite-KL distribution assigning probability one to C has support contained in that of $\pi^\star$.

Let g be the highest guaranteed value of S under $\pi^\star$. Here S is real-valued and is at least ρ with probability one under $\pi^\star$. The attained-floor argument in Theorem 3's proof therefore applies: g is finite, $g\geq\rho$, and $\pi^\star(S\geq g)=1$.

Consider a competitor ν with finite KL to P_c that satisfies $D_0\geq\delta$ almost surely and guarantees reward at least g. Because $g\geq\rho$, this competitor also satisfies the reward requirement defining C. The union bound makes both requirements explicit:

$$\nu(C^c)\leq\nu(S<\rho)+\nu(D_0<\delta)=0+0=0,\qquad\nu(C)=1.$$

The lemma therefore places ν's support within that of $\pi^\star$, so ν cannot have greater diversity.

If ν guarantees a strictly higher floor $h>g$, then $\pi^\star(S<h)>0$. Otherwise h would also be guaranteed under $\pi^\star$, contradicting the definition of g. In contrast, $\nu(S<h)=0$. Substituting the target density gives

$$P_c(C\cap\{S<h\})=P_c(C)\pi^\star(S<h)>0,\qquad\nu(C\cap\{S<h\})=0.$$

This is a proper support restriction, so ν has strictly lower diversity. Thus $\pi^\star$ is Pareto-optimal between guaranteed worst-candidate reward and support-based diversity within the stated class of batch distributions.

At a fixed objective evaluation, $\rho=M-\Delta$. Varying $\Delta>0$ therefore traverses this target family over the feasible floors below M.

C Experiment Settings

Sampling and update schedules. The main comparisons use 499 Pick-a-Pic prompts and 3200 HPDv2 prompts; the 200 DrawBench prompts are evaluated separately in Appendix D. We compute 95% percentile bootstrap confidence intervals by resampling prompts with replacement, keeping each prompt's four candidates together and holding ρ fixed. The displayed $\pm$ values are the interval half-widths. For the reward gaps at matched DreamSim in Figure 3, we linearly interpolate FK steering's worst-candidate reward between adjacent evaluated points, without extrapolating beyond its sampled DreamSim range.

We sample FLUX at 1024×1024 with 28 Euler steps and guidance 3.5, SANA at 1024×1024 with 20 Euler steps and guidance 4.5, and SD1.5 at 512×512 with 100 DDIM (Song et al., 2021) steps,

$\eta = 1$, and guidance 7.5. For FLUX and SANA, every method uses the marginal-preserving SDE of Mark et al. (2025) with noise amplitude $a = 0.3$. At steps where reward is evaluated, we estimate $\hat{x}_0$ by averaging the velocity predictions from the current and previous steps. Because the SDE correction is applied through a first-order Euler update, we replace SANA's shipped second-order solver with Euler for each method. For methods with scheduled scoring or resampling, the common steps are $4, 8, \ldots, 24$ on FLUX and $3, 6, \ldots, 18$ on SANA. On SD1.5, the baselines use their published configurations, and SatisDive follows FK steering's schedule, updating latents at steps $20, 40, 60, 80$ and 99. We used an NVIDIA A100 GPU for the original experiments and NVIDIA H100 GPUs for runtime and memory profiling. Unless stated otherwise, each method returns four candidates per prompt using matched starting seeds.

Baseline settings. DAS uses Srinivasan Sampling Process (SSP) resampling, while FK steering uses multinomial resampling; both resample when the effective sample size is below $0.5K$. For DAS, the annealing rate γ is determined by the number of sampling steps, and we set α using its recommended range for the ratio of the reward-guidance norm to the model-drift norm. FK steering applies $e^{\lambda r}$ to the unstandardized reward, so we select λ according to the reward scale. We use the published NegToMe settings.

SatisDive settings. We use a fixed step-size multiplier of 1 for the latent updates. We set δ to the median of the $\binom{K}{2}$ pairwise distances at the first update step and set $\beta_d = \frac{1}{2}\delta$. At each update, the diversity D in Eq. 8 is the mean of the current pairwise distances weighted by $q_i q_j$. Each d_{ij} is the cosine distance between pooled features from FLUX joint block 12, SANA transformer block 7, or the SD1.5 UNet bottleneck. We apply late-stage latent replacement only during the second half of denoising ($s > T/2$) for every reward and base-model setting. We recompute τ at each update step but stop gradients through $\max_k r_k$; otherwise, increasing the current maximum reward would raise τ and increase the penalties on the other candidates, producing a gradient that reduces the maximum reward. The only SatisDive setting swept in the main experiments is the tolerance Δ. For each reward and base model, the selected Δ is approximately three-quarters of the median per-prompt difference between the largest and smallest candidate rewards. We also report sweeps over Δ.

Numerical latent updates. For each candidate, we normalize the reward and diversity gradients separately. We weight the normalized reward gradient by $(1 - q_k)/\max_j(1 - q_j)$, then clip each gradient separately to a maximum norm of 0.1 times the mean candidate latent norm at the first update. After clipping, we multiply the reward gradient by $0.5 + 1.5(s-1)/(T-1)$ and subtract the sum of the two gradients from the latent. We hold the pair weights $q_i q_j$ fixed when differentiating the diversity penalty.

Settings across rewards and base models. For each reward and base model, ρ is the median reward of the base model. With HPSv3, FLUX uses $\rho = 9.66$, $\Delta = 1.5$, DAS $\alpha = 0.1$, and FK $\lambda = 1.5$. With ImageReward, FLUX uses $\rho = 1.23$, $\Delta = 0.5$, DAS $\alpha = 0.005$, and FK $\lambda = 10$; SANA uses $\rho = 1.19$, $\Delta = 0.5$, DAS $\alpha = 0.005$, and FK $\lambda = 10$; and SD1.5 uses $\rho = 0.18$, $\Delta = 0.75$, DAS $\alpha = 0.001$, and FK $\lambda = 10$. DAS uses $\gamma = 0.024$ on FLUX, $\gamma = 0.035$ on SANA, and $\gamma = 0.008$ on SD1.5, as determined by their respective sampling-step counts.

The SANA and SD1.5 qualitative examples with ImageReward use the baseline parameters listed above and $\Delta = 0.5$ for SatisDive. Each method returns four candidates from matched starting seeds.

C.1 2D Toy Setup

The base model is a rectified-flow velocity network trained on a mixture of 49 Gaussians on a 7×7 grid with spacing 5, using the same interpolant as FLUX, and is frozen for all runs. The reward is the maximum of 49 Gaussian bumps, one per mode. The 16 bumps on every other row and column have height 1, and the other 33 have heights drawn in $[0.15, 0.45]$. We use the floor $\rho = 0.5$, so only the 16 high-reward modes contain points that satisfy the floor. The centers of the 16 high-reward bumps are displaced from the grid by Gaussian noise with standard deviation 1.5. For the displayed SatisDive operating point, we use $\Delta = 0.325$, $\beta_r = 0.2$, $\beta_d = 0.5$, diversity cutoff $\delta = 20$, and latent-update multiplier 0.7, with scoring and updates at steps $3, 6, \ldots, 24$ of the 28 sampling steps. These toy-specific settings were selected from a hyperparameter sweep; β_r, β_d, and δ are fixed rather than computed from each batch. Here β_d is independent of δ, unlike the $\frac{1}{2}\delta$ used for the image models.

The plotted SatisDive metrics average all 200 runs with seeds 42–241. The candidate illustration uses seed 88, selected to show four candidates above the floor in distinct high-reward regions; it is not an average batch. The base model, reward landscape, and baseline configurations are unchanged.

D Additional Experiments

Unless stated otherwise, the experiments below use Pick-a-Pic prompts, with ρ set to the base model's median reward and the metrics defined in Section 4.

D.1 Effect of Increasing the Number of Candidates

We test whether SatisDive remains more diverse than reward-steering baselines when the number of candidates increases from $K{=}4$ to $K{=}16$. As Table 5 shows with the SD1.5 base model and ImageReward, SatisDive achieves a much higher DreamSim of 0.77 than DAS (0.04) or VASR (0.47).

Table 5: **At $K{=}16$, SatisDive maintains a competitive satisfaction rate and higher diversity than the reward-steering baselines.** We use SD1.5 with ImageReward, $\Delta{=}0.75$, and $\rho{=}0.16$, the median reward of the base model's $K{=}16$ outputs.

	Satisfaction–diversity			Additional metric	Held-out	
Method	min ↑	DreamSim ↑	SAT_ρ ↑	LPIPS ↑	HPSv3	PickScore
Base	−1.19	0.89	0.50	0.76	5.74	20.49
NegToMe	−1.82	1.01	0.30	0.83	2.00	19.42
FK steering	0.69	0.45	0.88	0.37	7.25	20.91
FK + NegToMe	−1.03	0.64	0.46	0.53	0.43	19.26
VASR	0.74	0.47	0.88	0.38	7.55	20.96
DAS	1.20	0.04	0.86	0.01	−3.48	18.37
Ours ($\Delta{=}0.75$)	0.41	0.77	0.83	0.69	6.75	20.76

D.2 Additional Results on DrawBench

We use DrawBench (Saharia et al., 2022) to evaluate whether our improvements extend beyond the Pick-a-Pic and HPDv2 prompts used in the main text. Its prompts cover counting, spatial relations, rare combinations, and long descriptions. As Table 6 shows, SatisDive attains a satisfaction rate of 0.63 and DreamSim of 0.48, compared with 0.60 and 0.05 for DAS and 0.57 and 0.18 for FK steering. Although NegToMe produces greater diversity, it has a substantially lower satisfaction rate of 0.34.

Table 6: **SatisDive has the highest observed satisfaction rate on DrawBench and higher observed diversity than every reward-steering method.** We use FLUX.1-dev with HPSv3 on the 200 DrawBench prompts and set $\rho{=}11.60$.

	Satisfaction–diversity		Additional metric
Method	DreamSim ↑	SAT_ρ ↑	LPIPS ↑
Base	0.70	0.50	0.64
NegToMe	0.90	0.34	0.78
FK steering	0.18	0.57	0.16
FK + NegToMe	0.31	0.37	0.27
VASR	0.05	0.56	0.03
DAS	0.05	0.60	0.03
Ours	0.48	0.63	0.45

D.3 Effect of a Higher Reward Floor

We repeat the SANA comparison with ρ set to the 75th percentile rather than the median of the base model's ImageReward distribution. As Figure 5 shows, SatisDive has a higher observed satisfaction rate and DreamSim than every reward-steering baseline at $\Delta \in \{0.1, 0.3, 0.5, 0.75\}$.

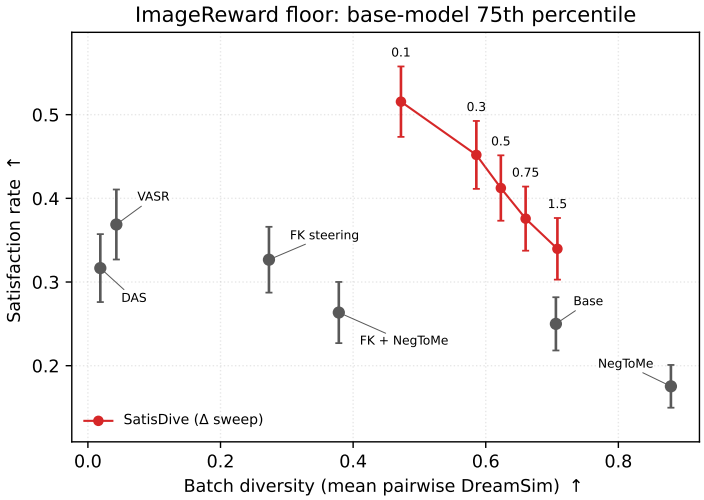

Figure 5: **At $\Delta \in \{0.1, 0.3, 0.5, 0.75\}$, SatisDive has a higher observed satisfaction rate and DreamSim than every reward-steering baseline under the stricter floor.** We use SANA-1.6B with ImageReward and set ρ=1.69, the 75th percentile of the base model's candidate rewards. Error bars are 95% bootstrap confidence intervals over prompts. For legibility, we show five of the seven Δ settings.

D.4 CMMD on MS-COCO

CMMD (Jayasumana et al., 2024) measures the distance between distributions of generated and real images. We compute CMMD between each method's generated images and the MS-COCO (Lin et al., 2014) val2017 images to test whether SatisDive's diversity advantage over reward-steering methods coincides with greater distance from the real-image distribution. SatisDive obtains a CMMD of 0.81, compared with 0.83–0.88 for the baselines.

Table 7: **SatisDive has the lowest observed CMMD among the evaluated methods.** We report CMMD ($\times 10^3$; lower is better) using 2000 images generated from 500 MS-COCO captions per method and all 5000 MS-COCO val2017 images as the reference distribution. We use FLUX.1-dev with HPSv3, Δ=1.5 for SatisDive, λ=1.5 for FK steering, α=0.1 and γ=0.024 for DAS, and the published NegToMe settings.

Method	CMMD ↓
Base	0.87
NegToMe	0.83
FK steering	0.88
DAS	0.87
Ours	**0.81**

D.5 Component Ablations

Table 8 removes late-stage latent replacement and the diversity penalty separately from SatisDive to measure how each component changes satisfaction and diversity. Removing late-stage latent

replacement reduces worst-candidate HPSv3 from 9.39 to 8.58 and satisfaction from 0.55 to 0.50, while removing the diversity penalty reduces DreamSim from 0.56 to 0.54.

Table 8: **Removing late-stage latent replacement lowers worst-candidate reward, while removing the diversity penalty lowers DreamSim.** We remove one component at a time from SatisDive using FLUX.1-dev with HPSv3, Δ=1.5, and ρ=9.66. The 95% bootstrap confidence intervals computed from per-prompt differences exclude zero for the changes in worst-candidate and mean HPSv3 after removing late-stage latent replacement and for the changes in DreamSim and mean HPSv3 after removing the diversity penalty.

	Satisfaction–diversity			Additional metric	Held-out	
Variant	min ↑	DreamSim ↑	SAT_ρ ↑	mean ↑	ImageReward	PickScore
Full method	9.39	0.56	0.55	10.03	1.04	22.32
w/o replacement	8.58	0.68	0.50	9.65	1.01	22.27
w/o diversity penalty	9.43	0.54	0.56	10.07	1.07	22.35

We additionally ablate late-stage latent replacement on Stable Diffusion 1.5 with ImageReward. As Table 9 shows, removing latent replacement reduces worst-candidate reward from 0.54 to −0.07 and satisfaction from 0.77 to 0.64. DreamSim increases from 0.72 to 0.88 because the batch retains a low-reward candidate that the full method would replace.

Table 9: **late-stage latent replacement also improves worst-candidate reward and satisfaction on Stable Diffusion 1.5.** We compare the full method with a variant without latent replacement using Stable Diffusion 1.5 with ImageReward on Pick-a-Pic, Δ=0.75, and ρ=0.18. The ± value is the half-width of a 95% prompt-bootstrap confidence interval, where available.

	Satisfaction–diversity			Held-out	
Variant	min ↑	DreamSim	SAT_ρ ↑	HPSv3	PickScore
Full method	$0.54_{\pm 0.08}$	0.72	0.77	6.40	20.69
w/o replacement	−0.07	0.88	0.64	5.94	20.57

We also test whether the replaced candidate and the selected higher-reward candidate, whose latents are identical immediately after replacement, remain identical after subsequent denoising. The full FLUX.1-dev run in Table 8 records 521 replacement events on 314 Pick-a-Pic prompts, all during the final two update steps. Immediately before replacement, the replaced candidate has a mean reward of 8.49, compared with 10.87 for the selected candidate. Across 503 distinct pairs of replaced and selected candidates from these events, the corresponding final images have a mean DreamSim of 0.195. Thus, identical latents do not necessarily remain identical after subsequent denoising.

D.6 FLUX.1-dev with ImageReward

Table 10 repeats the FLUX.1-dev comparison with ImageReward instead of HPSv3 to test whether SatisDive remains more diverse than reward-steering methods at comparable satisfaction. SatisDive and FK steering both attain a satisfaction rate of 0.61, while their DreamSim values are 0.57 and 0.23, respectively.

D.7 Stable Diffusion 1.5 with ImageReward

Table 11 tests whether the satisfaction–diversity comparison holds on SD1.5 with ImageReward. At Δ=0.5, SatisDive attains a satisfaction rate of 0.80 and DreamSim of 0.65, compared with 0.72 and 0.49 for FK steering and 0.77 and 0.36 for VASR. DAS attains a higher satisfaction rate of 0.84 but a DreamSim of 0.04. VASR (soft) uses reward-weighted resampling without redirecting duplicate selections to the highest-reward candidate.

Table 10: **With ImageReward, SatisDive matches FK steering's satisfaction while maintaining higher diversity.** We repeat the FLUX.1-dev comparison using ImageReward, Δ=0.5, and ρ=1.23. The $\pm$ values are half-widths of 95% percentile bootstrap confidence intervals over prompts.

	Satisfaction–diversity			Additional metric	Held-out	
Method	min ↑	DreamSim ↑	SAT_ρ ↑	LPIPS ↑	HPSv3	PickScore
Base	0.68	0.66 ±0.01	0.50 ±0.04	0.64	9.65	22.28
NegToMe	0.20	0.88 ±0.01	0.39 ±0.03	0.79	8.17	21.49
FK steering	1.16	0.23 ±0.02	0.61 ±0.04	0.20	9.74	22.38
FK + NegToMe	0.97	0.45 ±0.02	0.56 ±0.04	0.38	7.45	21.33
VASR	1.24	0.05 ±0.00	0.62 ±0.04	0.03	9.79	22.40
DAS	1.18	0.05 ±0.01	0.58 ±0.04	0.03	9.75	22.35
Ours (Δ=0.5)	1.10	0.57 ±0.02	0.61 ±0.04	0.55	9.72	22.34

Table 11: **SatisDive achieves higher satisfaction and diversity than FK steering on Stable Diffusion 1.5.** We use ImageReward with ρ=0.18 and report SatisDive at three values of Δ. The $\pm$ values are half-widths of 95% percentile bootstrap confidence intervals over prompts.

	Satisfaction–diversity			Additional metric	Held-out	
Method	min ↑	DreamSim ↑	SAT_ρ ↑	LPIPS ↑	HPSv3	PickScore
Base	−0.62	0.89 ±0.01	0.50 ±0.03	0.77	5.77	20.48
NegToMe	−1.24	1.03 ±0.01	0.32 ±0.03	0.84	2.79	19.63
FK steering	0.35	0.49 ±0.03	0.72 ±0.04	0.40	6.71	20.78
FK + NegToMe	−0.63	0.71 ±0.03	0.45 ±0.04	0.56	1.63	19.53
VASR	0.62	0.36 ±0.01	0.77 ±0.04	0.26	7.00	20.81
VASR (soft)	0.16	0.54 ±0.03	0.69 ±0.04	0.44	6.61	20.71
DAS	1.14	0.04 ±0.01	0.84 ±0.03	0.01	−4.01	18.29
Ours (Δ=0.5)	0.70 ±0.08	0.65 ±0.02	0.80 ±0.03	0.57	6.57	20.72
Ours (Δ=1.0)	0.44 ±0.08	0.76 ±0.02	0.74 ±0.03	0.68	6.32	20.66
Ours (Δ=1.25)	0.32 ±0.09	0.80 ±0.01	0.71 ±0.04	0.71	6.24	20.64

We use DAS's published SD1.5 settings. DAS attains the highest ImageReward satisfaction rate, 0.84, but its mean held-out HPSv3 is −4.01, compared with 5.77 for the base model and 6.24–6.57 for SatisDive.

D.8 Comparison with CREPE

Table 12 compares SatisDive with CREPE (He et al., 2026) at reward scales 10 and 100 on the same Pick-a-Pic prompts using SD1.5 and ImageReward. At Δ=1.0, SatisDive achieves higher worst-candidate reward at similar DreamSim than both CREPE settings; at Δ=1.25, it has higher observed means for both metrics.

We adapt the released CREPE sampler to SD1.5, generating 512 × 512 images with guidance 7.5, 64 noise levels, and 200 replica-exchange iterations. We evaluate effective terminal reward scales 10 and 100; scale 100 corresponds to the released reward maximum, with all other settings unchanged between runs. Four outputs are retained at fixed late iterations and denoised to completion, without reward-based selection. SatisDive uses the SD1.5 configuration in Appendix C. We evaluate independently generated batches on the same prompts using common ImageReward and DreamSim implementations.

D.9 Comparison with SGI

We compare with SGI (Parmar et al., 2026), which begins with M=128 candidates and progressively prunes them to return K=4. Its generation budget is therefore not matched to methods that maintain four candidates throughout sampling. We use HPSv3 to score individual candidates and sweep SGI's

Table 12: **SatisDive improves worst-candidate reward at comparable diversity to both evaluated CREPE settings.** Both methods return $K=4$ images per prompt using SD1.5 with ImageReward. We use $\rho=0.18$ and report means on Pick-a-Pic; $\pm$ values are half-widths of 95% percentile confidence intervals from $200{,}000$ prompt-bootstrap resamples, keeping each prompt's four candidates together and ρ fixed.

	Satisfaction–diversity			Additional metric
Method	min ↑	DreamSim ↑	SAT_ρ ↑	mean ↑
CREPE (scale 10)	-0.33 ± 0.09	0.761 ± 0.011	0.528 ± 0.036	0.20 ± 0.08
CREPE (scale 100)	0.07 ± 0.08	0.753 ± 0.011	0.660 ± 0.034	0.52 ± 0.07
Ours ($\Delta=0.5$)	0.70 ± 0.08	0.650 ± 0.021	0.802 ± 0.033	0.88 ± 0.08
Ours ($\Delta=1.0$)	0.44 ± 0.08	0.762 ± 0.016	0.740 ± 0.034	0.71 ± 0.08
Ours ($\Delta=1.25$)	0.32 ± 0.09	0.796 ± 0.015	0.708 ± 0.035	0.63 ± 0.08

diversity weight, denoted λ_{SGI} to distinguish it from FK steering's λ. As λ_{SGI} increases, DreamSim increases from 0.647 to 0.823, while satisfaction rate decreases from 0.592 to 0.507.

Table 13: **Increasing SGI's diversity weight increases DreamSim but decreases satisfaction rate.** We evaluate SGI on Pick-a-Pic using FLUX.1-dev with HPSv3. SGI prunes $M=128$ candidates to $K=4$, and λ_{SGI} weights its DINO diversity score. We use $\rho=9.66$; the $\pm$ values are half-widths of 95% bootstrap confidence intervals over prompts.

	Satisfaction–diversity			Additional metric
λ_{SGI}	min ↑	DreamSim ↑	SAT_ρ ↑	mean ↑
0	9.32 ± 0.20	0.647 ± 0.012	0.592 ± 0.037	10.20
1.5	9.17 ± 0.20	0.693 ± 0.012	0.577 ± 0.037	10.15
6	8.89 ± 0.20	0.744 ± 0.013	0.563 ± 0.036	10.04
24	8.41 ± 0.22	0.803 ± 0.013	0.529 ± 0.035	9.82
96	8.12 ± 0.23	0.823 ± 0.013	0.507 ± 0.035	9.65

D.10 Inference-Time Computation

Figure 6 reports diffusion-model passes separately from forward evaluations of auxiliary scoring models. Under the FLUX.1-dev configuration, SatisDive uses more diffusion-model passes than Base, NegToMe, FK steering, FK + NegToMe, and VASR, but fewer diffusion-model passes and reward-model forward evaluations than DAS even when every partial SatisDive backward pass is counted as a full pass. SGI uses more diffusion-model passes and repeatedly applies HPSv3 and DINO while pruning $M=128$ candidates to $K=4$. These are operation counts rather than wall-clock measurements.

For SatisDive, the 152 diffusion-model forward passes comprise 112 denoising passes, 24 additional gradient-evaluation passes, 12 replacement-check passes, and 4 final feature-extraction passes. Adding 24 full and 24 partial backward passes gives the reported upper bound of 200. The 40 reward evaluations comprise 24 update evaluations, 12 replacement checks, and 4 final evaluations. Panel (b) reports auxiliary-model forward evaluations.

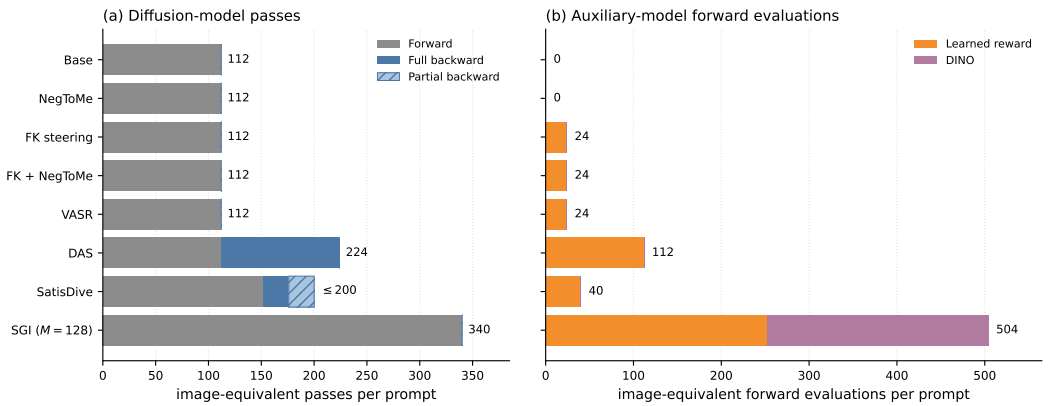

Figure 6: **Inference-time operation counts for the evaluated FLUX.1-dev methods.** One image-equivalent pass processes one candidate once through the indicated model; processing a batch of b candidates therefore counts as b passes. We count operations per prompt for $K{=}4$ returned candidates and 28 denoising steps. **(a)** SatisDive uses at most 200 diffusion-model passes, compared with 112 for Base, NegToMe, FK steering, FK + NegToMe, and VASR, 224 for DAS, and 340 for SGI. The SatisDive count is an upper bound because 24 of its 48 backward passes stop at the intermediate feature capture. DAS computes a reward gradient at every denoising step; its SSP resampling condition is checked at steps $4, 8, \ldots, 24$. **(b)** We count auxiliary-model forward evaluations. SGI uses 252 HPSv3 evaluations and 252 DINO evaluations while pruning $M{=}128$ candidates to $K{=}4$. The counts exclude VAE decoding and SGI's candidate-selection optimization.

Table 14 separately reports measured generation time and GPU memory for the FLUX.1-dev implementations. We use the same three prompts and starting seeds for all methods, following one excluded warmup prompt.

Table 14: **Measured runtime and GPU memory for FLUX.1-dev.** Each prompt returns $K{=}4$ images at 1024×1024 resolution using 28 denoising steps and HPSv3 for reward-guided methods. Time is the mean over three prompts, excluding model loading and image-file writes. Memory is the largest sampled resident GPU memory, summed across GPUs over generation and reward-model processes, rather than an exact peak. All runs use H100 80 GB GPUs; two-GPU methods place the generator and reward model on separate devices. Figure 3 reports the corresponding worst-candidate reward improvements at matched diversity.

Method	Time (s/prompt)	Total GPU memory (GiB)	GPUs used
Base	22.5	42.2	1
NegToMe	25.9	42.2	1
FK steering	37.7	58.5	2
FK + NegToMe	40.5	58.5	2
VASR	37.6	58.5	2
DAS	117.2	78.2	2
SatisDive	60.8	90.3	2

D.11 Reward-Temperature Robustness

Because β_r is computed from the standard deviation of the candidate rewards, it depends on the initial-noise seed. We repeat the Stable Diffusion 1.5 experiment with five seeds. Across these seeds, worst-candidate reward ranges from 0.54 to 0.59, satisfaction from 0.75 to 0.78, and DreamSim from 0.71 to 0.73. None of the pairwise differences between seeds is statistically significant for any reported metric at the 5% level.

Table 15: **SatisDive's satisfaction and diversity remain stable across five initial-noise seeds.** We use Stable Diffusion 1.5 with ImageReward on the same Pick-a-Pic prompts, set Δ=0.75, and compute β_r separately for each prompt and seed. The $\pm$ values are half-widths of 95% bootstrap confidence intervals over prompts.

	Satisfaction–diversity			Additional metric	Held-out	
Run	min ↑	DreamSim ↑	SAT_ρ ↑	LPIPS ↑	HPSv3	PickScore
Seed 0	0.54 ± 0.08	0.72 ± 0.02	0.77 ± 0.03	0.64	6.40	20.69
Seed 1	0.59 ± 0.08	0.72 ± 0.02	0.76 ± 0.03	0.64	6.51	20.71
Seed 2	0.59 ± 0.08	0.71 ± 0.02	0.76 ± 0.03	0.63	6.52	20.69
Seed 3	0.58 ± 0.08	0.71 ± 0.02	0.78 ± 0.03	0.62	6.63	20.72
Seed 4	0.54 ± 0.08	0.73 ± 0.02	0.75 ± 0.03	0.65	6.42	20.69

D.12 Reward Capping without the Diversity Penalty

To test whether stopping further reward optimization at ρ is sufficient to preserve diversity, we modify DAS by replacing r_k with $\min(r_k, \rho)$ in its guidance gradient and resampling weights. This makes rewards at or above ρ equivalent in both computations without adding SatisDive's diversity penalty. All other DAS settings remain unchanged. Reward capping increases DreamSim from 0.016 to 0.066, while SatisDive reaches 0.658 on the same prompts and reward floor. The change in DAS's satisfaction rate is not statistically significant.

Table 16: **Reward-capped DAS remains less diverse than SatisDive.** We use SANA-1.6B with ImageReward on Pick-a-Pic and set ρ=1.19. The reward-capped variant replaces r_k with $\min(r_k, \rho)$ in the guidance gradient and resampling weights. The $\pm$ values are half-widths of 95% bootstrap confidence intervals over prompts.

	Satisfaction–diversity			Additional metric	Held-out
Method	min ↑	DreamSim ↑	SAT_ρ ↑	mean ↑	HPSv3
DAS	1.10 ± 0.07	0.016 ± 0.001	0.599 ± 0.043	1.11	7.72
DAS (reward capped)	1.06 ± 0.07	0.066 ± 0.008	0.577 ± 0.042	1.09	7.65
Ours (Δ=0.75)	1.13 ± 0.07	0.658 ± 0.015	0.674 ± 0.039	1.28	7.63

E Additional Qualitative Comparisons

Figures 7–9 extend the SANA-1.6B comparison to all seven methods. Each row contains four candidates generated from matched starting seeds and reports their minimum ImageReward and mean pairwise DreamSim; the pink outline marks the candidate with the lowest ImageReward in that row. The reward floor is ρ_{IR}=1.19. Figure 10 provides a corresponding comparison with Stable Diffusion 1.5 at its reward floor ρ_{IR}=0.18.

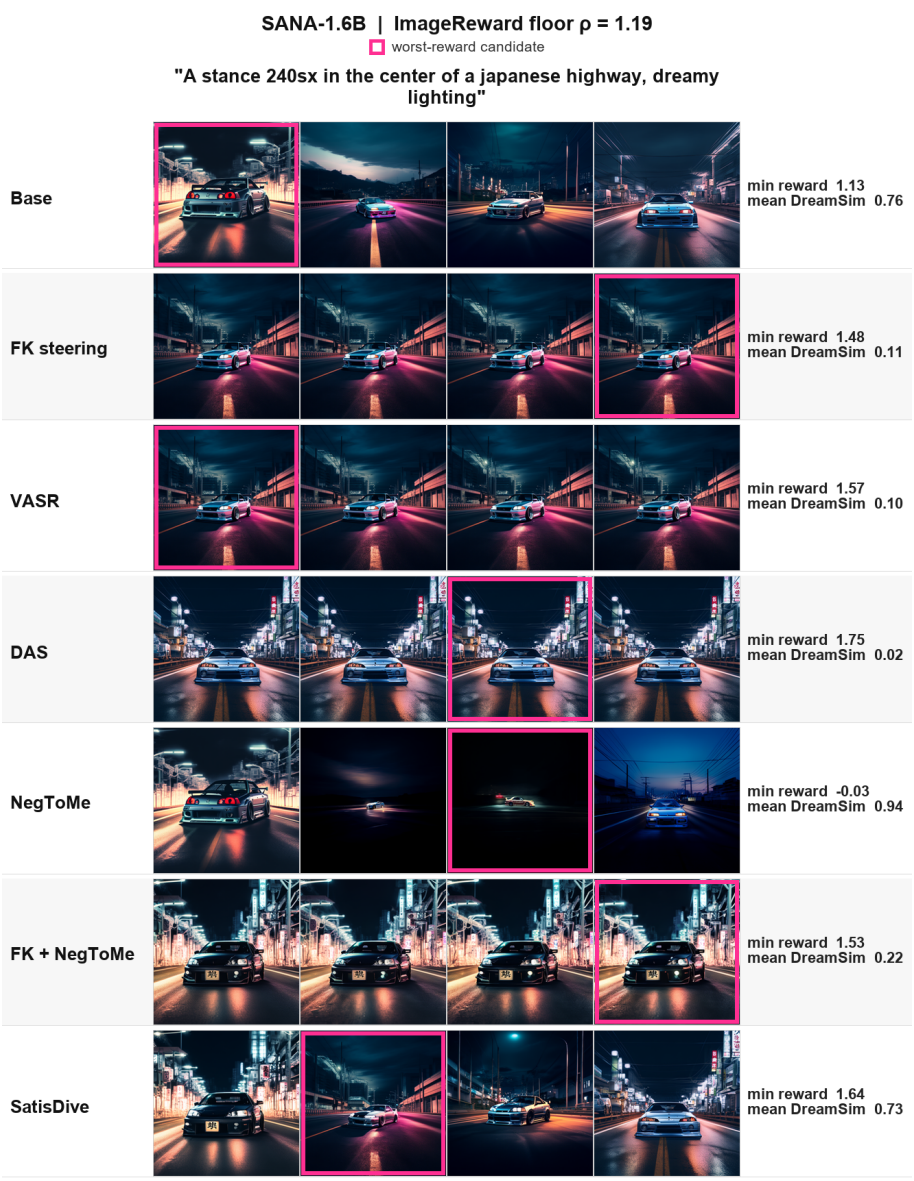

Figure 7: **On the Japanese-highway car prompt, SatisDive returns four visually distinct candidates above the reward floor.** SatisDive attains a minimum ImageReward of 1.64 and a DreamSim of 0.73. FK steering, VASR, DAS, and FK + NegToMe also satisfy the reward floor, but their DreamSim values are 0.11, 0.10, 0.02, and 0.22. Base and NegToMe remain diverse, but their minimum rewards of 1.13 and -0.03 fall below the reward floor.



Figure 8: **On the fairy-and-owl prompt, SatisDive has the highest minimum ImageReward,** 1.73**, and a DreamSim of** 0.68**.** FK steering, VASR, DAS, and FK + NegToMe have DreamSim values between 0.02 and 0.23. Base and NegToMe remain diverse, but their minimum rewards of 0.69 and -1.44 are below the reward floor.



Figure 9: **On the dog-driving-a-bus prompt, SatisDive attains a minimum ImageReward of** 1.46 **and a DreamSim of** 0.84**.** VASR and DAS satisfy the reward floor but return visually similar candidates. Base and NegToMe have comparable diversity to SatisDive, but each includes candidates below the reward floor.

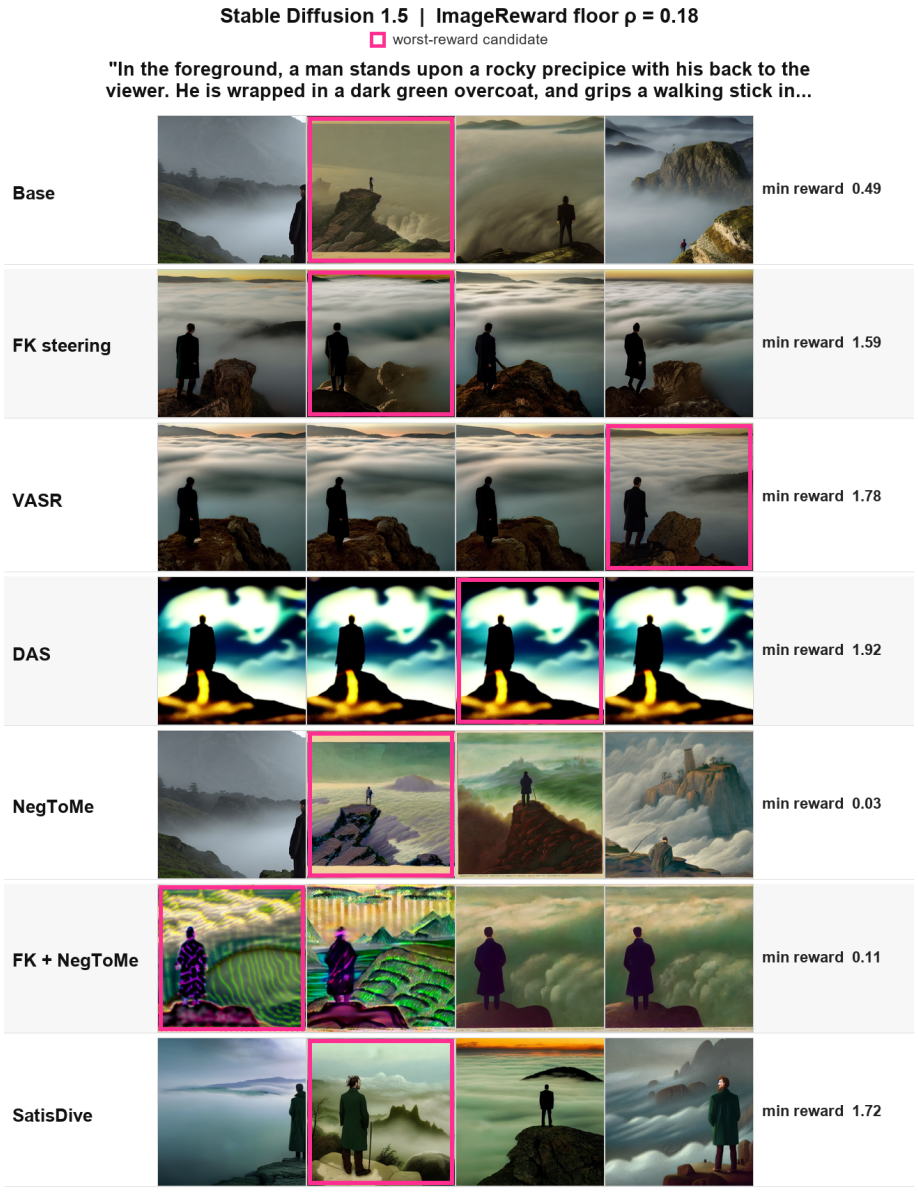

Figure 10: **For Stable Diffusion 1.5, SatisDive returns four visually distinct images with a minimum ImageReward of** 1.72**.** The prompt describes a man on a rocky precipice above clouds. Base and NegToMe have minimum rewards of 0.49 and 0.03, while the candidates from FK steering, VASR, and DAS are visually similar within each row. All methods use matched starting seeds.