BFMT: ENHANCING SEARCH CAPABILITIES OF TREE SAMPLER VIA BOOTSTRAP FLOW-MAP TREE

Anonymous authors
Paper under double-blind review

ABSTRACT

The ability to efficiently navigate high-dimensional spaces to identify optimal candidates—whether designing targeted molecular structures or generating images with preferred attributes—stands as a central pillar of modern scientific progress. Recent tree-based samplers (Jain et al., 2025) enable highly scalable inference-time search by eliminating the reward-gradient bottleneck inherent to particle-based methods. Despite their potential, existing tree-based samplers are fundamentally constrained by the massive number of function evaluations (NFEs) required for node valuation, crippling their utility in exploration-heavy tasks. Furthermore, their reliance on small, uniform transitions at each depth precludes dynamic, adaptive search capabilities. To address this, we introduce **B**ootstrap **F**low-**M**ap-**T**ree (a.k.a BFMT), a computationally efficient sampling framework that enables full tree-path construction from any tree depth using a single function evaluation, drastically reducing computational overhead while providing critical foresight for sequential sampling. By enabling dynamic transition time-step scheduling, BFMT efficiently allocates its sampling budget, smoothly transitioning from broad global exploration to fine-grained local refinement of high-utility modes discovered through exploration. Extensive experiments and ablation studies across diverse domains—from high-dimensional images to large-scale molecules—demonstrate BFMT's superiority over baseline approaches in both search and alignment tasks.

1 INTRODUCTION

In many scientific and engineering applications, ranging from the discovery of life-saving therapeutics to the optimization of interactive visual recommendation systems, progress is often constrained by the high cost of evaluating candidate solutions. This creates a need for *feedback-efficient search*, which requires uncovering high-quality targets through a limited sequence of expensive queries. For example, in personalized medicine, finding an effective treatment for a rare disease may involve iteratively proposing drug candidates and refining future choices using patient or experimental feedback. Since each evaluation can require substantial time, resources, and wet-lab experimentation, the objective is to uncover optimal targets with as few feedback rounds as possible. Addressing it demands a framework that can efficiently explore complex, high-dimensional spaces—such as molecular design or image spaces—while strategically harnessing sequential feedback to optimize the search process under a strict evaluation budget.

Current approaches are fundamentally ill-equipped for this challenge. For instance, (Uehara et al., 2024) relies on online RL fine-tuning of diffusion models to approximately minimize the KL divergence to the target distribution. This inherently mode-seeking objective fails to capture diverse, high-utility regions of the target distribution. Moreover, these methods hinge on an online-trained reward model, whose early-phase bias can misdirect the fine-tuning process and degrade sample quality. Recent Sequential Monte Carlo–based inference-time scaling methods (Singhal et al., 2025; Skreta et al., 2025; Kim et al., 2025) circumvent direct model fine-tuning by operating on fixed generative priors. However, they exhibit severe weight degeneracy, which effectively collapses the proposals and suppresses sample diversity (Lee et al., 2025). Moreover, these approaches rely on reward values or gradients at every resampling step; in practice, these signals are unavailable and must be approximated, introducing systematic bias into the sampler and degrading sample quality.

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.

To circumvent these barriers, recent tree-based samplers (Guo et al., 2025; Jain et al., 2025) evaluate only terminal rewards, thereby obviating the need to compute the reward gradient at every intermediate denoising step. However, their practical utility remains severely hindered by the prohibitive number of function evaluations (NFEs) required for node valuation. Moreover, their reliance on uniform transitions at each tree depth stifles adaptive search, while their resource-blind exploration strategies make them unsuitable for budget-constrained applications. To overcome these bottlenecks, we propose a principled framework centered on the following question:

> *How can we develop a principled, computationally efficient, training-free tree sampler that maximizes search efficiency while remaining highly effective under strict sampling budgets?*

To this end, we introduce the Bootstrapped Flow-Map Tree (BFMT), a novel sampling framework that synergizes history-aware tree search with highly efficient Flow Map dynamics. While standard tree search requires computationally expensive rollouts to estimate intermediate node values, BFMT utilizes Flow Maps to collapse this evaluation into a single NFE. Although the deterministic ODE nature of Flow Maps typically bottlenecks the stochastic, DDPM-like transitions required for child-node sampling, we circumvent this by deploying a *bootstrapped sufficient-statistic* scheme. This mechanism enables the construction of the *entire stochastic tree path without incurring any additional NFEs*. Furthermore, because Flow Maps permit state estimation across arbitrary time intervals, BFMT accommodates non-uniform transition steps. Breaking from rigid diffusion schedules unlocks dynamic, granular control over the search process and the exploration-exploitation trade-off. Inherently, BFMT executes a smooth hierarchical search: early stages near the root prioritize broad global exploration to uncover diverse modes, while deeper layers progressively exploit local neighborhoods of discovered high-utility regions—guided by sequential feedback and Bellman backups—to maximize the overall search efficiency. Finally, we augment BFMT with a novel budget-aware node selection strategy that outperforms standard UCT in resource-constrained environments. We provide a conceptual depiction of the BFMT framework in Figure 1. We summarize our primary contributions as follows:

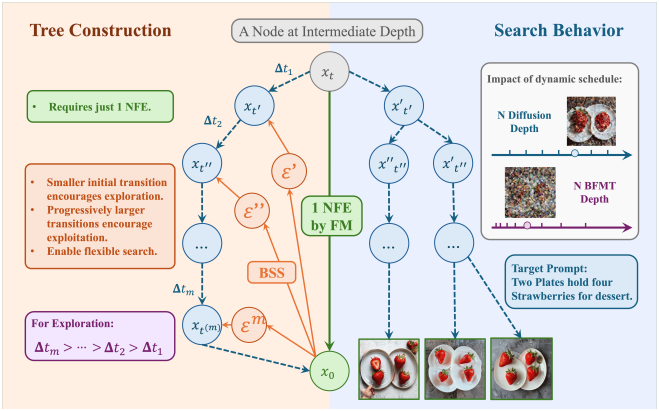


Figure 1: Overview of BFMT.

Contributions

- **Flow-Based Tree Sampler**: We introduce the Bootstrapped Flow-Map Tree, a principled, tree-based sampling framework for feedback-efficient search.
- **Stochastic Path Construction with single NFE**: We formulate a bootstrapped sufficient-statistics scheme that synthesizes complete, DDPM-like stochastic tree trajectories from deterministic ODEs using only a single NFE.
- **Dynamic Hierarchical Search**: By leveraging Flow Maps for non-uniform time transitions, BFMT unlocks adaptive search capabilities, seamlessly shifting from broad global mode exploration to fine-grained local refinement.
- **Budget-Aware Node Selection**: We propose a custom budget-aware tree search strategy designed specifically for resource-constrained environments.
- **Rigorous Empirical Validation across Diverse Domains**: Comprehensive quantitative and qualitative ablations validate each component of BFMT across challenging tasks, from high-dimensional image synthesis to large-scale molecular design.

2 METHODOLOGY: BFMT FRAMEWORK

Background: Feedback-efficient search requires strategic exploration of the target landscape. This can be achieved by a policy that efficiently samples the target distribution:

$$\pi^*(x) = \frac{1}{Z} p_\theta^{\text{pre}}(x) \exp(\beta\, r(x)) \tag{1}$$

In this formulation, $p_\theta^{\text{pre}}(x)$ denotes the pretrained generative prior, $r(x)$ represents the reward function, and β is the inverse temperature parameter, with Z serving as the normalizing constant. The target distribution can be seen as the optimal policy for the following objective (Uehara et al., 2025):

$$\pi^*(x) := \arg\max_\pi \mathbb{E}_{x\sim\pi(\cdot)}[r(x)] - \frac{1}{\beta} D_{\text{KL}}(\pi \parallel p_\theta^{\text{pre}}) \tag{2}$$

Efficiently optimizing $\pi^*(x)$ requires the estimation of the soft value function at time t (Jain et al., 2025), defined as:

$$V_t(x_t) := \frac{1}{\beta} \log \mathbb{E}_{p_\theta^{\text{pre}}(x_{0:t-1}|x_t)} \left[\exp\left(\beta r(x_0)\right)\right] \tag{3}$$

Estimating the value of x_t via Equation 3 introduces a computational bottleneck, as it necessitates simulating the entire denoising trajectory from x_t. Because these stochastic transition steps must be infinitesimal to minimize discretization error, they demand a prohibitively large number of transition steps and a correspondingly large NFE cost. Standard Tweedie-based posterior estimators are inadequate, as their pronounced inaccuracy at high noise levels critically undermines accurate value estimation. Moreover, a fundamental tension exists: while ODE sampling is highly efficient, enabling search strictly requires stochastic transitions. To resolve this conflict, (Holderrieth et al., 2026) introduce an ODE-based flow model that enables sampling from $x_0 \sim p(x_0|x_t)$ in a single step. Theoretically, given x_t, there exists an ODE that transports the prior p_1 to the conditional posterior $p_{0|t}(\cdot|x_t)$. The associated drift $b_s(\cdot; x_t)$ for this flow can be defined as the solution to the standard flow matching (FM) problem (Lipman et al., 2022), targeting the conditional posterior $p_{0|t}(\cdot|x_t)$ rather than the marginal data distribution p_0:

$$b_s(x; x_t) = \mathbb{E}\left[\dot{\alpha}_s x_1 + \dot{\beta}_s x_0 \mid x_s = x\right], \quad x_s = \alpha_s x_1 + \beta_s x_0, \quad x_1 \sim \mathcal{N}(0, I_d), \quad x_0 \sim p_{0|t}(\cdot|x_t) \tag{4}$$

Consequently, the probability flow associated with $b_s(\cdot; x_t)$ satisfies:

$$\frac{d}{ds} x_s = b_s(x_s; x_t), \quad x_1 \sim \mathcal{N}(0, I_d) \implies \text{Law}(x_0) = p_{0|t}(\cdot|x_t) \tag{5}$$

Here α_t, β_t are time-dependent coefficients satisfying $\alpha_0 = \beta_1 = 0$ and $\alpha_1 = \beta_0 = 1$. Assuming a Gaussian prior p_1, the conditional drift b_s can be derived analytically (Holderrieth et al., 2025). This is achieved through a reparameterization of the unconditional drift b_t of the pretrained flow model:

$$b_s(x_s; x_t) = w_1 x_s + w_2 b_{t^*}(S(x_s, x_t)) + w_3 x_t$$

Where, w_1, w_2, and w_3 represent scalar weights. S serves as a linear *sufficient statistic*, and t^* denotes a reparametrized time parameter:

$$S_{s,t}(x_s, x_t) = \frac{\alpha_s \sigma_t^2 x_s + \alpha_t \sigma_s^2 x_t}{\sigma_t^2 \alpha_s^2 + \alpha_t^2 \sigma_s^2}, \qquad t^*(s,t) = g^{-1}\left(\frac{\sigma_t^2 \sigma_s^2}{\sigma_t^2 \alpha_s^2 + \alpha_t^2 \sigma_s^2}\right), \quad g(t) = \frac{\sigma_t^2}{\alpha_t^2}. \tag{6}$$

Here σ_t represents the diffusion schedule. Although this reparameterization ensures the accessibility of b_s, the process of generating posterior samples from $p_{0|t}(\cdot|x_t)$ via the integration of ODE trajectories remains computationally demanding. A standard approach to overcome this is teacher-distillation (Boffi et al., 2026), in which the exact analytical field $b_s(x_s; x_t)$ serves as the teacher for the distillation process. The student model's velocity $\hat{v}$ is optimized using a combined objective, where the instantaneous loss directly minimizes the mean squared error between the student's instantaneous velocity, $\hat{v}_{s,s}$, and the analytical teacher field:

$$\mathcal{L}_{\text{inst}}^{\text{distill}}(\hat{v}) := \int_0^1 \int_0^1 \mathbb{E}||\hat{v}_{s,s}(x; x_t) - b_s(x; x_t)||^2 ds dt \tag{7}$$

Coupled with a consistency loss ($\mathcal{L}_{\text{cons}}^{\text{distill}}$) (Boffi et al., 2025) that ensures the student accurately integrates this velocity over varying time intervals. $\mathcal{L}_{\text{cons}}^{\text{distill}}(\hat{v})$ is defined as follows:

$$\int_0^1 \int_0^u \mathbb{E}\left\| \hat{v}_{s,u}(x_s; x_t') - \text{sg}\left(b_u(\hat{X}_{s,u}(x_s; x_t')); x_t'\right) - (u-s)\partial_u \hat{v}_{s,u}(x_s; x_t') \right\|^2 ds\, du$$

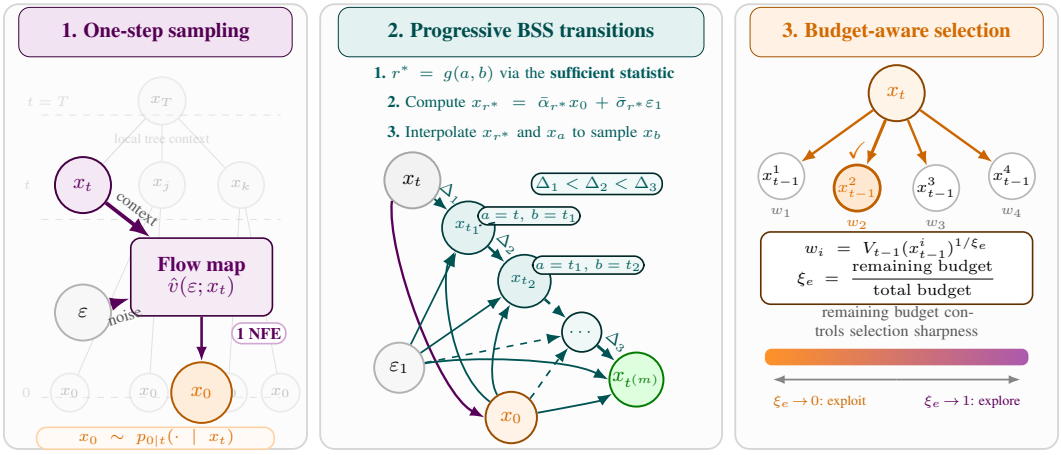


Figure 2: Key components of BFMT.

Here, $\hat{X}_{s,u}(\cdot\,; x_t) : \mathbb{R}^d \to \mathbb{R}^d$ acts as the solution operators for the context-dependent ODEs as in 5. The distillation compresses the iterative multi-step ODE into a single amortized map, *enabling computationally efficient, one-step sampling directly from the conditional posterior* $p_{0|t}(\cdot|x_t)$.

Proposed Approach: Armed with the distilled flow-map model $\hat{v}(x; x_t)$, our next challenge is:

> How can we exploit $\hat{v}(x; x_t)$ to orchestrate an entire sequence of Markovian stochastic transitions that synthesizes the full tree trajectory from x_t to x_0 with just a single NFE?

As a first step, we directly generate x_0 from x_t by sampling noise $\epsilon \sim \mathcal{N}(0, I)$ and passing it through the distilled flow map $\hat{v}(\epsilon; x_t)$. By definition, this operation yields exact samples from the true posterior, $x_0 \sim p_{0|t}(\cdot|x_t)$. Having anchored the trajectory's endpoints (i.e., ϵ and x_0), we realize the complete DDPM-like stochastic trajectory from x_t using only a single NFE by our proposed Bootstrap sufficient statistics (BSS) approach. Concretely, given the initial state x_t and the terminal prediction x_0 computed via $\hat{v}(\epsilon; x_t)$, we construct the trajectory recursively as follows:

$$\left.\begin{aligned} &\epsilon \sim N(0, I) \to x_{r^*} = \bar{\alpha}_{r^*} x_0 + \bar{\sigma}_{r^*} \varepsilon \to x_{t'} = \alpha_{t'} \frac{\bar{\alpha}_{r^*} \sigma_t^2 x_{r^*} + \alpha_t \bar{\sigma}_{r^*}^2 x_t}{\bar{\alpha}_{r^*}^2 \sigma_t^2 + \alpha_t^2 \bar{\sigma}_{r^*}^2}, \\ &r^*(t, t') = g^{-1}\left(\frac{g(t) g(t')}{g(t) - g(t')}\right), \quad g(t) = \frac{\sigma_t^2}{\alpha_t^2} \end{aligned}\right\} \begin{array}{l}\textbf{Bootstrap Loop} \\ \text{Update } t \leftarrow t' \\ \text{until } t' = 0\end{array} \tag{8}$$

$\bar{\alpha}$ and $\bar{\sigma}^2$ are defined in the App. B. Crucially, *Equation 8 (i.e., BSS) imposes no restrictions on t', granting the flexibility to use dynamic transition steps for adaptive search.* In the following proposition, we show that this realized stochastic trajectory perfectly matches trajectories with DDPM-like transitions, theoretically guaranteeing convergence to the target distribution at the terminal step.

Proposition 4.1 (Bootstrap Sufficient Statistic Realizes DDPM-Like Stochastic Trajectory)

Let $0 < t'' < t' < t$ be any two consecutive transition time steps in a trajectory, and $p^{\text{DDPM}}_{t''|t'}$ be the DDPM transition kernel. Then for $x_{t''}$ obtained via 8, starting from x_t, satisfies:

$$x_{t''} \sim p^{\text{DDPM}}_{t''|t'}(\cdot \mid x_t) \tag{9}$$

Equipped with the distilled flow map, the proposed bootstrap sufficient statistic mechanism synthesizes a complete DDPM-like stochastic trajectory in a single NFE. Furthermore, the Markovian structure of this stochastic chain naturally induces a finite-horizon search tree over the d-dimensional diffusion space. In this search tree, the nodes at depth t represent noisy states x_t, and the edges

represent a denoising step. Each node x_t can be stochastically denoised into multiple children $x_{t-1} \sim \mathrm{BSS}^1\left(x_0 \sim p_{0|t}(\cdot \mid x_t)\right)$. BSS^1 represents a single iteration of bootstrap sufficient statistic loop defined in 8. Consistent with standard tree samplers (Jain et al., 2025), each node maintains its state and timestep (x_t, t), visit count $N(x_t)$, and a Monte Carlo estimate of the soft value function $V_t(x_t)$. Analogous to the soft Bellman equation, $V_t(x_t)$ satisfies the following recursive relation:

$$V_t(x_t) = \frac{1}{\beta} \log \mathbb{E}_{x_{t-1} \sim \mathrm{BSS}^1\left(x_0 \sim p_{0|t}(\cdot|x_t)\right)} \left[\exp(\beta \;\; V_{t-1}(x_{t-1}))\right]. \tag{10}$$

Moreover, $V_t(x_t)$ characterizes the optimal policy $\pi^* := \mathrm{argmax}_\pi \, \mathbb{E}_{x \sim \pi(\cdot)}[r(x)] - \frac{1}{\beta} D_{\mathrm{KL}}(\pi \parallel p_\theta)$:

$$\pi_t^*(x_{t-1} \mid x_t) = \frac{\mathrm{BSS}^1\left(x_0 \sim p_{0|t}(\cdot \mid x_t)\right) \exp(\beta V_{t-1}(x_{t-1}))}{\int \mathrm{BSS}^1\left(x_0 \sim p_{0|t}(\cdot \mid x_t)\right) \exp(\beta V_{t-1}(x_{t-1}))\, dx_{t-1}}. \tag{11}$$

Here, $V_0(x_0) = r(x_0)$ is the terminal clean ground-truth reward feedback. We present the derivation of equation 11 in the Appendix C.

Budget-Aware Node Selection However, choosing children solely by their soft value estimates starves the search of trajectories that reach diverse modes. Prior approaches (Jain et al., 2025), relying on Upper Confidence Tree (Couëtoux et al., 2011), are well-suited for asymptotic exploration but fundamentally assume unbounded horizons, failing to adapt to strict, depleting resource constraints. To address this and prevent catastrophic budget exhaustion on unpromising paths, we introduce a *Budget-Aware Stochastic Exploration* mechanism (BASE) and sample each child (we use $\mathcal{C}(x_t)$ to denote the set of children of node x_t) according to the following normalized probability distribution:

$$\mathbf{p}^{\mathrm{BASE}}(x_{t-1}) := \frac{w_{x_{t-1}}}{\sum_{x_j \in \mathcal{C}(x_t)} w_{x_j}}, \quad w_{x_{t-1}} = V_{t-1}(x_{t-1})^{\frac{1}{\zeta_t}}, \quad \zeta_t = \left(\frac{\text{Remaining Budget}}{\text{Total Budget}}\right) \tag{12}$$

Specifically, Starting from the root x_1, sample a child $x_{t-1} \in \mathcal{C}(x_t)$ following the $\mathbf{p}^{\mathrm{BASE}}(\cdot)$ distribution recursively until either an unexpanded node is reached or $t = 0$. BASE approach shifts the search behavior based on the remaining budget. When resources are plentiful ($\zeta_t \approx 1$), the scaling exponent $\alpha_t = \frac{1}{\zeta_t}$ remains near 1, creating a sampling distribution that follows the raw node values and encourages broad exploration. However, as the budget is depleted ($\zeta_t \to 0$), α_t grows, amplifying the differences between node values and focusing the probability on the most promising paths. Consequently, the agent transitions from an exploratory strategy to a strictly exploitative one as it nears its resource limit.

During each online interaction step, we traverse from the root to a terminal node by recursively applying the selection rule (Eqn. 12). Upon sampling this terminal node, we evaluate it via $V_0(x_0) = r(x_0)$, we recursively propagate this signal up the tree using a Monte-Carlo soft Bellman backup (Equation 10) and increment the visit counts for all nodes along the path. During tree traversal, if an intermediate node x_t ($t > 0$) has not yet reached its branching capacity (i.e., $|\mathcal{C}(x_t)| < B(x_t) = \lceil C \cdot N(x_t)^\zeta \rceil, \quad C > 0, \zeta \in (0, 1)$), we actively expand it. Following (Couëtoux et al., 2011; Jain et al., 2025), to systematically exploit the most promising trajectories, we prioritize the expansion of highly visited nodes. We instantiate a new child $x_{t-1} \sim \mathrm{BSS}^1\left(x_0 \sim p_{0|t}(\cdot \mid x_t)\right)$ and initialize its statistics to $V_{t-1}(x_{t-1}) = 0$ and $N(x_{t-1}) = 1$. From each newly instantiated node, we execute a rollout to a terminal state x_0 via the recursive BSS loop (Eqn. 8), seamlessly integrating this new path into the search tree. We term this framework BFMT, with its key features visualized in Fig. 2.

As feedback accumulates, the generated samples via the BFMT approach strictly converge to the true target distribution, mirroring the behavior of the prior tree sampler. We also show that the sampling error decays polynomially with respect to the total sampling budget. Crucially, ***BFMT's efficiency hinges on its quadratic dependence of the convergence rate on the tree horizon: by realizing a long horizon with a single NFE or a single step, BFMT can substantially accelerate convergence.***

Proposition 4.3 (BFMT Convergence Rate) (Informal)

Let $r(x_0)$ be bounded and L-Lipschitz continuous, M be the feedback budget, and T be the maximum tree depth. Under standard regularity assumptions, the Total Variation distance between the BFMT sampling policy $\hat{q}_M(x_0)$ and the target optimal policy $\pi^*(x_0)$ is bounded:

$$D_{TV}(\hat{q}_M \parallel \pi^*) \leq \mathcal{O}\left(\beta T^2 M^{-1/4}\right)$$

Importantly, BFMT seamlessly generalizes to domains lacking an off-the-shelf stochastic flow map by integrating different variants of distilled models, including standard deterministic flow maps—such as consistency models (detailed derivations in App. J). Extensive experiments with state-of-the-art consistency models confirm BFMT's superior performance and broad applicability (see Results).

Generalizability to Discrete Domain The key motivation for extending inference-time search with BFMT samplers to the discrete domain is the elegant continuous relaxation introduced by Categorical Flow Maps (Roos et al., 2026) (CFMs). Rather than navigating the complexities of discrete Markov chains, CFMs model the generative process as a smooth, continuous probability path constrained directly on the probability simplex Δ^K. The core "trick" relies on using a simplex-constrained endpoint parameterization. Because the entire flow trajectory unfolds in this continuous space, we can plug in continuous-domain inference-time search algorithms—like BFMT samplers—to categorical settings. The only modification required is a simple argmax projection at the very end of the generated path to yield the discrete output. Our empirical findings on a large-scale molecule generation task highlight BFMT's effectiveness for efficient search in the discrete domain.

3 Experiments and Results

Task Overview and Baselines We assess BFMT in two settings: *search* and *alignment*. In the *search* setting, an unconditional prior model must generate samples that satisfy fine-grained, class-specific objectives while operating under strict sampling and computational budgets. In the *alignment* setting, the goal is to guide a conditional prior model toward generating samples that satisfy complex, fine-grained constraints involving compositional and quantitative reasoning within fixed computational and sampling limits. We compare BFMT with a diverse collection of competitive, state-of-the-art methods, including the tree-based Diffusion Tree Search (DTS) (Jain et al., 2025), Feynman–Kac Steering (FKS) (Singhal et al., 2025), the SMC-based DAS method (Kim et al., 2025), and the recently introduced Meta-Flow-Map (MFM) approach (Potaptchik et al., 2026). To simulate feedback, we use two widely adopted reward models: the VQA score obtained with CLIP-FlanT5 (Lin et al., 2024) and the ImageReward score (Xu et al., 2023).

Search Experimental Setting: To evaluate the search capabilities of the BFMT, we utilize ImageNet-1K. We employ a distilled flow map model optimized jointly using the $\mathcal{L}_{\text{inst}}^{\text{distill}}$ and $\mathcal{L}_{\text{const}}^{\text{distill}}$ loss objectives as the prior. We evaluate BFMT and all competing methods on 200 randomly selected target classes, each paired with its corresponding fine-grained specifications. Examples of the target classes and prompts are provided in the Appendix K. We report results under two complementary metrics: mean reward and max reward. Mean reward captures the overall alignment efficacy of the sampler, reflecting how well it performs across all generated samples. Max reward, on the other hand, measures optimal alignment performance — a critical metric in downstream tasks where producing a single high-quality, target-aligned sample is more valuable than broad generation quality. For a fair evaluation, we include MFM (BoN) as a baseline only in the max reward comparison, and exclude it from the mean reward comparison. This choice reflects its design: MFM (BoN) optimizes for a single, highly aligned target sample rather than broad, population-wide alignment efficiency.

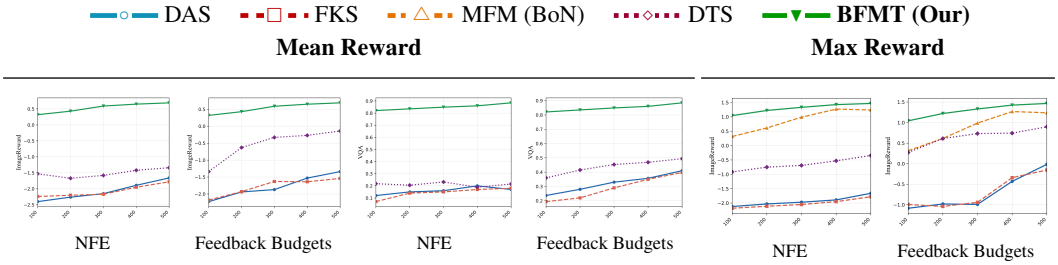


Figure 3: Search Performance Analysis on ImageNet Target Classes. (left) Mean reward; (right) Max reward.

Search Result: Fig. 3 demonstrates that BFMT consistently achieves superior reward scores using substantially fewer NFEs and feedback queries than all baselines. This extreme computational efficiency is driven by the bootstrap sufficient statistic mechanism, which synthesizes complete trajectories in a single NFE. Grounded in Proposition 4.3, this allows BFMT to cheaply simulate extensive diffusion horizons, directly exploiting quadratic convergence scaling to accelerate target alignment. Furthermore, BFMT maximizes feedback efficiency through the synergy of arbitrary-duration Flow-Map transitions and budget-aware node selection. This unlocks dynamic, hierarchical

search: early stages use small transitions and exploration-biased selection to broadly discover global macro-structures, while deeper stages leverage larger transitions and exploitation-focused selection to fine-tune granular details (Several detailed visualizations illustrating BFMT's hierarchical search strategies are presented in the Appendix F, G, H, I). Qualitative comparisons (Fig. 4 and additional visualizations are in the Appendix O) further confirm this precise target alignment.

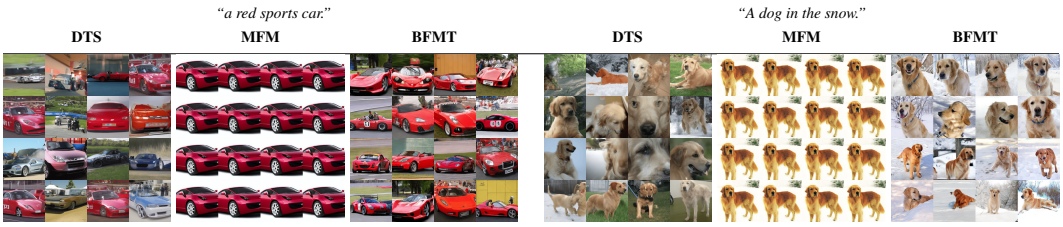
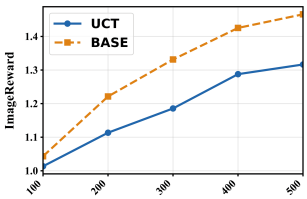


Figure 4: Search Visualizations.

Search Performance Evaluation with Best Reward In this section, we evaluate the search performance of BFMT against several baseline methods, including MFM Best-Of-N (BoN), which is specifically optimized for best-reward evaluation. Detailed comparative results are presented in the figure 3. Consistent with our earlier analysis, we use ImageReward for evaluation. Our experiments demonstrate that BFMT outperforms all baselines, including BoN. These results underscore a key strength of BFMT: beyond producing better target-aligned samples on average — as reflected in higher mean reward — it also excels at generating the single best target-aligned sample. This capability is particularly valuable in applications where producing an optimal output matters more than overall alignment capability of the sampler.

Impact of BASE To evaluate the effectiveness of the BASE, we compare its performance against a standard UCT-based mechanism across varying feedback budgets. As shown in Fig. 5, BASE achieves significantly higher reward scores under any given fixed budget. This demonstrates that incorporating budget awareness into the node selection process provides far more efficient control over the exploration-exploitation tradeoff than standard UCT. These results underscore the critical role of the BASE node selection strategy within the BFMT framework in enabling feedback-efficient search.

Figure 5: BASE vs. UCT.



The Impact of Bootstrap Sufficient Statistic (BSS): To evaluate the impact of the BSS mechanism within BFMT, we compare our approach against BFMT-ISI, a variant that replaces BSS with an alternative ODE-based, single-NFE trajectory generation scheme: mapping x_t to x_0 via a pretrained flow-map, followed by an Iterative Stochastic Interpolant (ISI) between x_0 and $\epsilon \sim \mathcal{N}(0, I)$ to construct intermediate nodes. As shown in Fig. 6, this alternative degrades sample quality. Furthermore, because it lacks the DDPM-like transitions guaranteed by BSS, it fails to construct valid trajectories, which hinders convergence to the target distribution and leads to lower reward scores (Fig. 6). We visualize the impact of BSS in Appendix E.

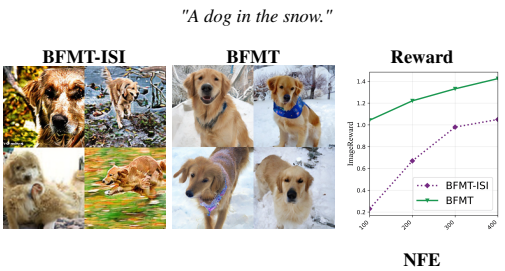


Figure 6: Impact of BSS.

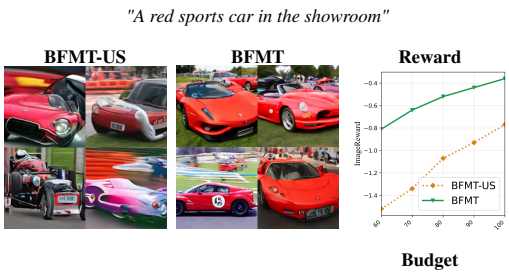


Figure 7: Impact of Dynamic Transition step.

Effect of Dynamic Transition Schedule on Search BFMT leverages Flow-Maps to enable dynamic transitions during trajectory rollout, providing precise control over the exploration-exploitation tradeoff, key to enable efficient search. By employing smaller transition steps during the early

denoising phase, BFMT achieves broader exploration. Conversely, larger steps in the final phases promote localized exploitation. Given a fixed NFE budget, this non-uniform transition scheduling enables BFMT to conduct a more efficient search compared to its uniform-step counterpart, BFMT-US. This enhanced exploratory capacity directly yields a more diverse set of generated samples aligning with the target prompt, while resulting in higher reward across varying NFE (see Fig. 7).

Results with Large-Scale Molecular Structure Generation Task We consider 10M molecules extracted from PubChem (Kim et al., 2023). The molecules are represented as SELFIES (Krenn et al., 2020), a robust string representation for molecular structures. These molecules are used to train Categorical Flow Maps (CFM), a generative flow-matching model developed by (Roos et al., 2026). We trained the CFM model for 16 epochs using the default hyperparameters and then used the trained model for inference-time search. We subsequently used BFMT to steer generation toward regions of chemical space with high (i) Quantitative Estimate of Drug-likeness (QED) and (ii) a combination of QED and synthetic accessibility (SA) scores. QED measures a molecule's drug-likeness, while SA estimates how easily it can be synthesized. Our goal is to generate molecules that satisfy these objectives and assess whether BFMT produces a higher proportion of desirable molecules than vanilla CFM sampling, while maintaining comparable validity and uniqueness. Fig. 8 illustrates the target property distributions alongside representative visualizations of the generated molecules. Empirically, BFMT consistently outperforms CFM by generating samples that more effectively satisfy the target criteria, maintaining this superior search performance across all evaluated metrics and feedback budgets. Table 1 reports the mean validity, uniqueness, QED, and the combination of QED and SA scores for samples generated under feedback budget 1000. The results indicate that BFMT effectively optimizes for target properties without compromising chemical validity. Ultimately, these results establish BFMT as a highly feedback-efficient sampler capable of navigating complex discrete spaces such as molecular design. Additional results across varying feedback budgets are presented in the Appendix M.

Table 1: Validity Check for the Generated Molecules.

Method	Validity	Unique.	Mean QED	Validity	Unique.	Mean QED&SA
CFM	0.9396	0.9968	0.4892	0.9396	0.9968	0.5173
BFMT	0.9558	0.9644	**0.5610**	0.9540	0.9746	**0.5746**

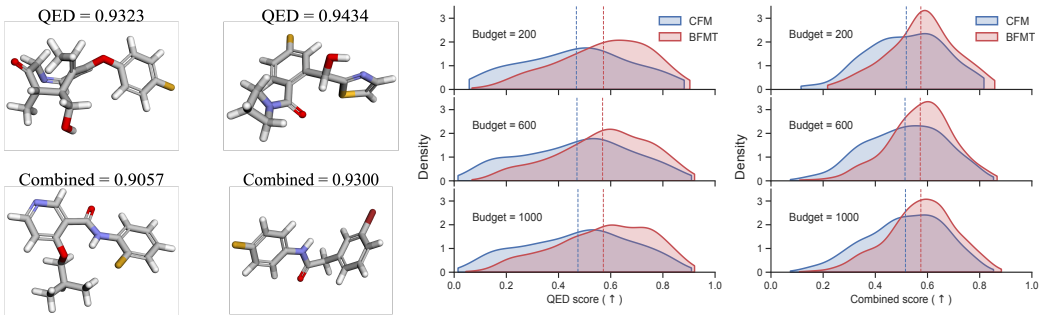


Figure 8: Effectiveness of BFMT on target-aligned feedback-constrained Molecule Generation Task.

Alignment Experimental Setting: We assess BFMT's alignment performance across two challenging tasks. First, for the compositional alignment task, we utilize 100 randomly selected prompts from GenAI-Bench (Jiang et al., 2024) that necessitate at least three advanced compositional skills. Second, for the quantity-aware alignment task, we evaluate using 100 prompts randomly sampled from the numeracy category of T2I-CompBench++ (Huang et al., 2025). Here, the held-out reward is defined as the negative Residual Sum of Squares (RSS) between the target object counts and the detected counts, where detections are obtained using DINO (Liu et al., 2024) and SAM (Kirillov et al., 2023). We employ an off-the-shelf consistency model (SANA) (Chen et al., 2025) as the prior for these experiments. We provide a detailed derivation of the reparametrization of the noise schedule used in SANA (see App. J) to facilitate its integration within the BFMT framework.

Alignment Result: Consistent with the search setting, Fig. 10 demonstrates that BFMT achieves superior reward scores compared to competing methods, using significantly fewer NFEs and excelling under a fixed feedback budget. These findings reinforce the efficacy of BFMT's highly effective, feedback-driven alignment capability—powered by its dynamic transition schedule and a budget-aware node selection mechanism. Qualitative comparisons are presented in Fig. 9 (additional viz. in App. P). The results on the quantity-aware alignment task are reported in the Appendix D.

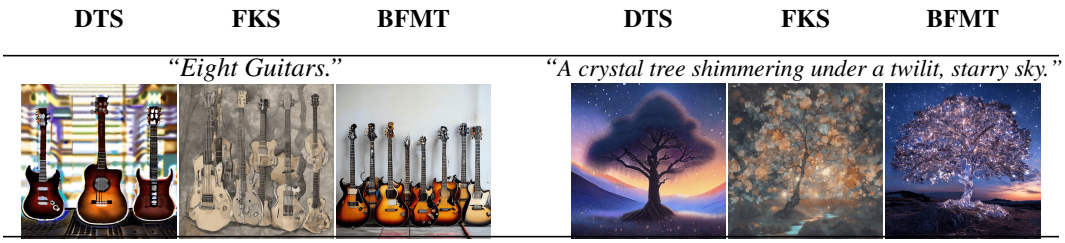


Figure 9: Alignment Visualization. (left) Quantity Prompt. (right) Compositional Prompt.

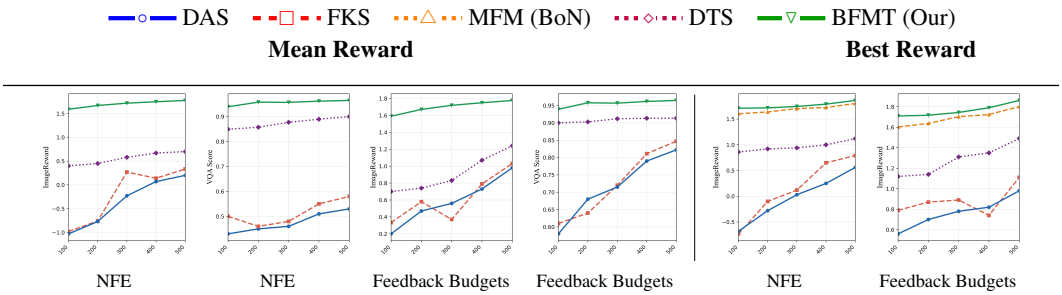


Figure 10: Analysis of Competitive Approaches on Compositional Alignment.

Related Work

Tree-Based Sampler For Search Tree-based samplers have been introduced for effective inference-time search and alignment. DTS (Jain et al., 2026) casts the denoising process as a finite-horizon tree, where, similar to MCTS, rollouts are used to continuously improve value estimates for intermediate noisy states. DTS thus enables a principled search in the space of denoising trajectories to identify the modes within high-volume regions of the target density. Unlike SMC-style samplers, tree-based frameworks backpropagate only terminal rewards, eliminating the need to evaluate rewards at every intermediate denoising step. While promising, current tree-based samplers (Jain et al., 2026; Guo et al., 2026) are severely bottlenecked by the massive number of function evaluations (NFEs) needed for node valuation, rendering them impractical for exploration-heavy tasks. Moreover, their dependence on rigid, uniform transitions restricts adaptive search, and heuristic workarounds like beam search (Li et al., 2025) fail to outperform standard UCT selection (Kocsis & Szepesvári, 2006).

SMC-based Sampler for Search Inference-time scaling has seen rapid adoption, with SMC serving as a primary framework. SMC methods approximate the target distribution by maintaining an evolving population of sample particles. SMC (Del Moral et al., 2006; Kim et al., 2025; Singhal et al., 2025) utilize potential functions—which typically approximate the soft value function—to iteratively weight and resample particles at each step. While numerous variants have recently been adapted for diffusion model alignment (Kim et al., 2025; Wu et al., 2023), classical SMC theoretically guarantees both exact target sampling and optimal value estimation only in the limit of infinite particles. In practice, repeated resampling inherently diminishes sample diversity due to high weight variance and imprecise value estimation. To combat particle weight degeneracy and debias particle-based guidance, several correction techniques (Skreta et al., 2025; Lee et al., 2025) have recently been proposed. However, these methods falter in high-dimensional domains, such as image synthesis, due to their reliance on dense reward gradients at every resampling step. Because exact gradients are rarely available in practice, relying on approximations inevitably injects systematic bias into the sampling trajectory. Although numerous gradient-free alignment methods circumvent the need for reward gradients via best-of-N sampling (Stiennon et al., 2020; Beirami et al., 2024), this generate-and-filter paradigm incurs severe computational overhead. Furthermore, because intervention occurs exclusively post-hoc at the final output state, these approaches fundamentally fail to actively steer the intermediate generative trajectory—a capability essential for feedback-efficient sampling.

Conclusion We introduce BFMT, a principled Flow-Map Tree sampler tailored for efficient feedback-efficient search and alignment. Crucially, BFMT enables compute-efficient dynamic search by constructing arbitrary-transition search trajectories with only a single NFE. Extensive experiments show BFMT's effectiveness and scalability across a diverse range of modalities.

AI USE STATEMENT

We used large language models (LLMs) solely as a writing assistant for language polishing and improving clarity of presentation. The LLMs were not involved in research ideation, methodological design, experimental execution, or result analysis. All scientific contributions and substantive writing were carried out by the authors.

REPRODUCIBILITY STATEMENT

To facilitate reproducibility, we have provided full implementation details, hyperparameter settings, and architectural specifications. Our complete source code, including training scripts and evaluation pipelines, is available via the zip file included in the supplementary material. Upon acceptance, we will make all code publicly available to facilitate easy reproduction.

REFERENCES

Ahmad Beirami, Alekh Agarwal, Jonathan Berant, Alexander D'Amour, Jacob Eisenstein, Chirag Nagpal, and Ananda Theertha Suresh. Theoretical guarantees on the best-of-n alignment policy. *arXiv preprint arXiv:2401.01879*, 2024.

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

Nicholas Boffi, Michael Albergo, and Eric Vanden-Eijnden. How to build a consistency model: Learning flow maps via self-distillation. *Advances in Neural Information Processing Systems*, 38: 33346–33382, 2026.

Nicholas M Boffi, Michael S Albergo, and Eric Vanden-Eijnden. How to build a consistency model: Learning flow maps via self-distillation. *arXiv preprint arXiv:2505.18825*, 2025.

Junsong Chen, Shuchen Xue, Yuyang Zhao, Jincheng Yu, Sayak Paul, Junyu Chen, Han Cai, Song Han, and Enze Xie. Sana-sprint: One-step diffusion with continuous-time consistency distillation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 16185–16195, 2025.

Adrien Couëtoux, Jean-Baptiste Hoock, Nataliya Sokolovska, Olivier Teytaud, and Nicolas Bonnard. Continuous upper confidence trees. In *International conference on learning and intelligent optimization*, pp. 433–445. Springer, 2011.

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

Yingqing Guo, Yukang Yang, Hui Yuan, and Mengdi Wang. Training-free guidance beyond differentiability: Scalable path steering with tree search in diffusion and flow models. *arXiv preprint arXiv:2502.11420*, 2025.

Yingqing Guo, Yukang Yang, Hui Yuan, and Mengdi Wang. Training-free guidance beyond differentiability: Scalable path steering with tree search in diffusion and flow models. *Advances in Neural Information Processing Systems*, 38:73343–73384, 2026.

Peter Holderrieth, Uriel Singer, Tommi Jaakkola, Ricky TQ Chen, Yaron Lipman, and Brian Karrer. Glass flows: Transition sampling for alignment of flow and diffusion models. *arXiv preprint arXiv:2509.25170*, 2025.

Peter Holderrieth, Douglas Chen, Luca Eyring, Ishin Shah, Giri Anantharaman, Yutong He, Zeynep Akata, Tommi Jaakkola, Nicholas Matthew Boffi, and Max Simchowitz. Diamond maps: Efficient reward alignment via stochastic flow maps. *arXiv preprint arXiv:2602.05993*, 2026.

Kaiyi Huang, Chengqi Duan, Kaiyue Sun, Enze Xie, Zhenguo Li, and Xihui Liu. T2i-compbench++: An enhanced and comprehensive benchmark for compositional text-to-image generation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 47(5):3563–3579, 2025.

Vineet Jain, Kusha Sareen, Mohammad Pedramfar, and Siamak Ravanbakhsh. Diffusion tree sampling: Scalable inference-time alignment of diffusion models. *arXiv preprint arXiv:2506.20701*, 2025.

Vineet Jain, Kusha Sareen, Mohammad Pedramfar, and Siamak Ravanbakhsh. Diffusion tree sampling: Scalable inference-time alignment of diffusion models. *Advances in Neural Information Processing Systems*, 38:145576–145615, 2026.

Dongfu Jiang, Max Ku, Tianle Li, Yuansheng Ni, Shizhuo Sun, Rongqi Fan, and Wenhu Chen. Genai arena: An open evaluation platform for generative models. *Advances in Neural Information Processing Systems*, 37:79889–79908, 2024.

Sunghwan Kim, Jie Chen, Tiejun Cheng, Asta Gindulyte, Jia He, Siqian He, Qingliang Li, Benjamin A Shoemaker, Paul A Thiessen, Bo Yu, et al. Pubchem 2023 update. *Nucleic acids research*, 51(D1): D1373–D1380, 2023.

Sunwoo Kim, Minkyu Kim, and Dongmin Park. Test-time alignment of diffusion models without reward over-optimization. *arXiv preprint arXiv:2501.05803*, 2025.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4015–4026, 2023.

Levente Kocsis and Csaba Szepesvári. Bandit based monte-carlo planning. In *European conference on machine learning*, pp. 282–293. Springer, 2006.

Mario Krenn, Florian Häse, AkshatKumar Nigam, Pascal Friederich, and Alan Aspuru-Guzik. Self-referencing embedded strings (selfies): A 100% robust molecular string representation. *Machine Learning: Science and Technology*, 1(4):045024, 2020.

Cheuk Kit Lee, Paul Jeha, Jes Frellsen, Pietro Lio, Michael Samuel Albergo, and Francisco Vargas. Debiasing guidance for discrete diffusion with sequential monte carlo. *arXiv preprint arXiv:2502.06079*, 2025.

Xiner Li, Masatoshi Uehara, Xingyu Su, Gabriele Scalia, Tommaso Biancalani, Aviv Regev, Sergey Levine, and Shuiwang Ji. Dynamic search for inference-time alignment in diffusion models. *arXiv preprint arXiv:2503.02039*, 2025.

Zhiqiu Lin, Deepak Pathak, Baiqi Li, Jiayao Li, Xide Xia, Graham Neubig, Pengchuan Zhang, and Deva Ramanan. Evaluating text-to-visual generation with image-to-text generation. In *European Conference on Computer Vision*, pp. 366–384. Springer, 2024.

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

Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *European conference on computer vision*, pp. 38–55. Springer, 2024.

Peter Potaptchik, Adhi Saravanan, Abbas Mammadov, Alvaro Prat, Michael S Albergo, and Yee Whye Teh. Meta flow maps enable scalable reward alignment. *arXiv preprint arXiv:2601.14430*, 2026.

Daan Roos, Oscar Davis, Floor Eijkelboom, Michael Bronstein, Max Welling, Ismail Ilkan Ceylan, Luca Ambrogioni, and Jan-Willem van de Meent. Categorical flow maps. *arXiv preprint arXiv:2602.12233*, 2026.

Kenneth H Rosen and Kamala Krithivasan. *Discrete mathematics and its applications*, volume 6. McGraw-hill New York, 1999.

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. *arXiv preprint arXiv:2501.06848*, 2025.

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

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. Learning to summarize with human feedback. *Advances in neural information processing systems*, 33:3008–3021, 2020.

Masatoshi Uehara, Yulai Zhao, Kevin Black, Ehsan Hajiramezanali, Gabriele Scalia, Nathaniel Lee Diamant, Alex M Tseng, Sergey Levine, and Tommaso Biancalani. Feedback efficient online fine-tuning of diffusion models. *arXiv preprint arXiv:2402.16359*, 2024.

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

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

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. *Advances in Neural Information Processing Systems*, 36:15903–15935, 2023.

BFMT: ENHANCING SEARCH CAPABILITIES OF TREE SAMPLER VIA BOOTSTRAP FLOW-MAP TREE (APPENDIX)

APPENDIX TABLE OF CONTENTS

Appendix: Theoretical Derivations **Page**

1 Derivation of BFMT Convergence Rate 14
2 Bootstrap Sufficient Statistic Realizes DDPM-Like Stochastic Path 18
3 Derivation of Equation 11 21

Appendix: Quantitative Results on the Quantity-Aware Alignment Task

1 Detailed Results on the Quantity Aware Alignment Task 22
2 Additional Quantitative Comparisons with Baselines on the Quantity-aware Alignment Task 22

Appendix: Additional Ablation Studies and Qualitative Analysis of BFMT

1 Visualizing the Impact of Bootstrap Sufficient Statistic 22
2 A Visual Illustration of BFMT's Exploratory Search Strategy 23
3 Additional Visual Illustration of BFMT's Exploratory Search Strategy 24
4 Hierarchical Exploration Strategy of BFMT 25
5 Additional Visualizations of Hierarchical Exploration Strategy of BFMT 26

Appendix: Additional Implementation Details, Results, and Analysis

1 Derivation of Noise Schedule to Adapt IMPFM on Alignment Tasks 26
2 Details of Class and Prompt Detail in the Search Experiment 28
3 Details about the Evaluation Dataset for Alignment Tasks 29
4 Additional Results With Molecule Generation Task 29
5 Implementation Details 29

Appendix: Additional Qualitative Visualizations on Search and Alignment Task

1 Additional Comparative Search Visualizations 29
2 Additional Competitive Alignment Visualization 30

A PROOF OF PROPOSITION 4.3

Proposition 4.3 (BFMT Convergence Rate)

Let $r(x_0)$ be bounded and L-Lipschitz continuous, M be the feedback budget, and T be the maximum tree depth. Under standard regularity assumptions, the Total Variation distance between the BFMT sampling policy $\hat{q}_M(x_0)$ and the target optimal policy $\pi^*(x_0)$ is bounded:

$$D_{TV}(\hat{q}_M \parallel \pi^*) \leq \mathcal{O}\left(\beta T^2 M^{-1/4}\right)$$

Proof. Let M be the total number of trajectory rollouts starting at the root, and T be the maximum depth of the tree. We assume that the reward function is bounded, $r(x) \in [0, R_{max}]$. Consequently, the true soft value function is also bounded: $V_t(x) \in [0, R_{max}]$ for all t. We also assume that the transition kernel $p_\theta(x_{t-1} \mid x_t)$ and the reward function $r(x)$ are L-Lipschitz continuous with respect to the state space.

Step 1: Bound The Single-Node Value Estimation Error: At any intermediate node x_t visited N times: BFMT utilizes progressive widening to limit the continuous branching factor to $B = \lceil CN^\alpha \rceil$ for some constant C and $\alpha \in (0, 1)$. The empirical soft value estimate is defined as:

$$\hat{V}_t(x_t) = \frac{1}{\beta} \log \left(\frac{1}{B} \sum_{i=1}^{B} \exp\left(\beta\, \hat{V}_{t-1}(x_{t-1}^{(i)})\right) \right)$$

The true soft value is defined as:

$$V_t(x_t) = \frac{1}{\beta} \log \mathbb{E}_{x_{t-1} \sim \mathrm{BSS}^1\left(x_0 \sim p_{0|t}(\cdot|x_t)\right)} \left[\exp\left(\beta\, V_{t-1}(x_{t-1})\right)\right]$$

The error $\epsilon(N) = |\hat{V}_t(x_t) - V_t(x_t)|$ at this node stems from two distinct sources:

- *Discretization Error (ϵ_{disc}): Approximating the continuous integral over* $\mathrm{BSS}^1(x_0 \sim p_{0|t}(\cdot|x_t))$ *with a finite sum of B branches. Under the Lipschitz assumption, standard Monte Carlo integration error scales as* $\mathcal{O}(B^{-1/2})$. Next, we prove this statement.
  Firstly, the Monte-Carlo estimator is unbiased. To show this, let's assume, $f(x_{t-1}) = \exp\left(\beta\, V_{t-1}(x_{t-1})\right)$, $\hat{p}_{\hat{v}}(x_{t-1}|x_t) := \mathrm{BSS}^1(x_0 \sim p_{0|t}(\cdot|x_t))$. Specifically, we are looking at the integral:

  $$I = \mathbb{E}_{x_{t-1} \sim \hat{p}_{\hat{v}}(\cdot)}[f(x_{t-1})] = \int f(x_{t-1}) \hat{p}_{\hat{v}}(x_{t-1}) dx_{t-1}$$

  Now, our Monte Carlo estimator using B samples $x_{t-1}^{(1)}, x_{t-1}^{(2)}, \ldots, x_{t-1}^{(B)} \sim \hat{p}_{\hat{v}}(\cdot)$ is:

  $$\hat{I}_B = \frac{1}{B} \sum_{i=1}^{B} f(x_{t-1}^{(i)})$$

  $$\mathbb{E}[\hat{I}_B] = \mathbb{E}\left[\frac{1}{B} \sum_{i=1}^{B} f(x_{t-1}^{(i)})\right]$$

  By linearity of expectation:

  $$\mathbb{E}[\hat{I}_B] = \frac{1}{B} \sum_{i=1}^{B} \mathbb{E}[f(x_{t-1}^{(i)})] = \frac{1}{B} \sum_{i=1}^{B} I = I$$

  which shows that the Monte-Carlo estimator is unbiased.
  Secondly, because *the estimator is unbiased, the mean squared error (MSE) equals the variance of the estimator*. We must first ensure that the variance of a single sample,

Var$(f(x_{t-1}))$, is finite. As we assume bounded rewards: $V_{t-1}(x_{t-1}) \in [0, R_{max}]$, therefore, $f(x_{t-1}) = \exp(\beta V_{t-1}(x_{t-1}))$ is also bounded:

$$1 \leq f(x_{t-1}) \leq \exp(\beta\, R_{max})$$

Both the transition kernel and the reward function are Lipschitz continuous. Since V_{t-1} is derived from Lipschitz rewards and transitions, it is Lipschitz continuous. The composition of a Lipschitz function with an exponential function (over a bounded domain) remains Lipschitz. Because $f(x_{t-1})$ is bounded on a compact domain, its true variance σ^2 must be finite:

$$\sigma^2 = \text{Var}(f(x_{t-1})) = \mathbb{E}[f(x_{t-1})^2] - (\mathbb{E}[f(x_{t-1})])^2 < \infty$$

Specifically, since $f(x_{t-1}) \in [1, \exp(\beta\, R_{max})]$, we can crudely bound the variance using Popoviciu's inequality:

$$\sigma^2 \leq \frac{(\exp(\beta\, R_{max}) - 1)^2}{4}$$

We now calculate the variance of the estimator using the average of our B samples.

$$\text{Var}(\hat{I}_B) = \text{Var}\left(\frac{1}{B}\sum_{i=1}^{B} f(x_{t-1}^{(i)})\right)$$

Because the B samples are drawn independently, the variance of the sum is the sum of the variances:

$$\text{Var}(\hat{I}_B) = \frac{1}{B^2}\sum_{i=1}^{B} \text{Var}(f(x_{t-1}^{(i)}))$$

Since all samples are drawn from the same distribution, they all have the same variance σ^2:

$$\text{Var}(\hat{I}_B) = \frac{1}{B^2}(B \cdot \sigma^2) = \frac{\sigma^2}{B}$$

The expected estimation error is typically quantified by the Root Mean Square Error (RMSE), which is the standard deviation of the estimator.

$$\text{RMSE}(\hat{I}_B) = \sqrt{\text{Var}(\hat{I}_B)} = \sqrt{\frac{\sigma^2}{B}} = \frac{\sigma}{\sqrt{B}}$$

Because σ is a finite constant independent of the number of samples B, we can express the asymptotic convergence rate in Big-O notation:

$$\epsilon_{disc} = \text{Error} = \mathcal{O}\left(\frac{1}{\sqrt{B}}\right) = \mathcal{O}(B^{-1/2})$$

- *Statistical Estimation Error (ϵ_{stat}): The downstream values $\hat{V}_{t-1}$ are estimated using finite rollouts. Leveraging Pigeonhole Principle (Rosen & Krithivasan, 1999), with N total visits distributed across B branches, each child receives a minimum polynomial budget fraction $N(x_t) \geq \Omega(B^\gamma)$, where $\gamma = (1-\alpha)^t$, which implies $\gamma > 0$. However, under standard regularity assumptions, we relax the polynomial node visitation count relation and let each child receive $n \approx N/B$ visits. By Hoeffding's inequality on the bounded log-sum-exp operator, this error scales as $\mathcal{O}(n^{-1/2})$.* Next, we prove this statement.
  For a specific child node, let $X_1, X_2, \ldots, X_n$ represent the outcomes (returns) of the n independent rollouts starting from that node. The empirical value estimate $\hat{V}_{t-1}$ is simply the sample average of these returns:

$$\hat{V}_{t-1} = \frac{1}{n}\sum_{i=1}^{n} X_i$$

We assume that the reward function—and consequently the true value function V_{t-1}—is strictly bounded within the interval $[0, R_{max}]$. Therefore, every individual rollout return X_i is a bounded random variable:

$$0 \leq X_i \leq R_{max} \quad \text{for all } i$$

Now, Hoeffding's inequality provides a strict upper bound on the probability that the sample mean of independent, bounded random variables deviates from its true expected value by more than a certain margin, ϵ. For random variables $X_i \in [a, b]$, Hoeffding's inequality states:

$$P(|\hat{V}_{t-1} - V_{t-1}| \geq \epsilon) \leq 2\exp\left(-\frac{2n\epsilon^2}{(b-a)^2}\right)$$

Substituting our specific bounds $a = 0$ and $b = R_{max}$:

$$P(|\hat{V}_{t-1} - V_{t-1}| \geq \epsilon) \leq 2\exp\left(-\frac{2n\epsilon^2}{R_{max}^2}\right)$$

We aim to find the error bound ϵ that holds with a high probability $1 - \delta$ (where δ is a very small failure probability, e.g., 0.05).To do this, we set the right side of the inequality to δ:

$$2\exp\left(-\frac{2n\epsilon^2}{R_{max}^2}\right) = \delta$$

Now, we solve for ϵ algebraically, which yields:

$$\epsilon = R_{max}\sqrt{\frac{\ln(2/\delta)}{2n}}$$

For any chosen confidence level δ (which is fixed), the terms R_{max} and $\sqrt{\ln(2/\delta)/2}$ are just constants. Let's denote this combined constant K. We can rewrite the equation as:

$$\epsilon = K \cdot \frac{1}{\sqrt{n}}$$

Therefore, with high probability $1 - \delta$, the estimation error for any single child node shrinks proportionally to the inverse square root of the number of visits n, i.e.,

$$\epsilon_{stat} = \mathcal{O}(n^{-1/2})$$

In the BFMT framework, the values of the B children are aggregated using a soft-maximum (the log-sum-exp function). The standard log-sum-exp function, defined as $f(z) = \frac{1}{\beta}\log\sum\exp(\beta z_i)$, is a well-known non-expansive operator.Specifically, it is 1-Lipschitz continuous with respect to the maximum norm (L-infinity norm). This implies that if the inputs (the children's $\hat{V}_{t-1}$ estimates) have a maximum error of $\mathcal{O}(n^{-1/2})$, passing them through the log-sum-exp operator will not amplify the error. The final aggregated statistical error at the parent node remains bounded by $\mathcal{O}(n^{-1/2})$.

Combining these via the triangle inequality:

$$\epsilon(N) \leq \mathcal{O}\left(\frac{1}{\sqrt{B}}\right) + \mathcal{O}\left(\frac{1}{\sqrt{N/B}}\right); n \approx \frac{N}{B}$$

Substituting $B = CN^{\alpha}$:

$$\epsilon(N) \leq \mathcal{O}\left(N^{-\alpha/2}\right) + \mathcal{O}\left(N^{-(1-\alpha)/2}\right)$$

To find the tightest bound, we balance the two error terms by setting $\alpha = 0.5$. This yields a single-node convergence rate of:

$$\epsilon(N) \leq \mathcal{O}\left(N^{-1/4}\right). \tag{13}$$

Step 2: Backward Induction for Error Propagation: We must now propagate this error from the leaves ($t = 0$) up to the root ($t = T$). Let ϵ_t denote the maximum absolute error in the value estimate at depth t. By the definition of the Bellman backup, the error at depth t is bounded by the intrinsic estimation error at that node plus the maximum inherited error from its children at depth $t - 1$:

$$\epsilon_t \leq \epsilon_{t-1} + \mathcal{O}\left(N_t^{-1/4}\right)$$

Because BFMT focuses its search budget M along the highest-value trajectories (according to the BASE selection mechanism), the visitation count N_t along the sampled trajectory remains proportional to M, i.e., $N_t \approx cM$ for some constant $c \in (0, 1]$. Unrolling the recursion from $t = T$ to $t = 0$, we sum the errors across the horizon:

$$\sup_{x_t}|\hat{V}_t(x_t) - V_t(x_t)| \leq \sum_{t=1}^{T}\mathcal{O}\left((cM)^{-1/4}\right) = \mathcal{O}\left(TM^{-1/4}\right)$$

Step 3: Bound the Step-Wise Policy Divergence: The BFMT sampling policy at timestep t is an empirically perturbed Boltzmann distribution:

$$\hat{q}(x_{t-1} \mid x_t) = \frac{\hat{p}_{\hat{v}}(x_{t-1} \mid x_t) \exp(\beta\, \hat{V}_{t-1}(x_{t-1}))}{\int \hat{p}_{\hat{v}}(x'_{t-1} \mid x_t) \exp(\beta\, \hat{V}_{t-1}(x'_{t-1}))dx'}$$

We bound the Total Variation distance between this empirical policy and the optimal target policy $\pi^*(x_{t-1} \mid x_t)$. For any two Boltzmann distributions over the same base measure differing only by their potential functions, the TV distance is bounded by the maximum absolute difference between those potentials multiplied by β:

$$D_{TV}(\hat{q}(x_{t-1}|x_t) \parallel \pi^*_t(x_{t-1}|x_t)) \leq 2\beta \sup_{x_{t-1}} |\hat{V}_{t-1}(x_{t-1}) - V_{t-1}(x_{t-1})|$$

. Next, we present the detailed derivation of the above statement.

Let $\hat{p}_{\hat{v}}(x_{t-1}|x_t)$ be a shared base measure. We define two Boltzmann distributions, the empirical policy $\hat{q}(x_{t-1}|x_t)$ and the target policy $\pi^*(x_{t-1}|x_t)$, parameterized by a temperature scalar β and potential functions $\hat{V}_{t-1}(x_{t-1})$ and $V_{t-1}(x_{t-1})$, respectively:

$$\hat{q}(x_{t-1}|x_t) = \frac{\hat{p}_{\hat{v}}(x_{t-1}|x_t) \exp(\beta \hat{V}_{t-1}(x_{t-1}))}{\hat{Z}} \quad \text{and} \quad \pi^*(x_{t-1}|x_t) = \frac{\hat{p}_{\hat{v}}(x_{t-1}|x_t) \exp(\beta V_{t-1}(x_{t-1}))}{Z^*}$$

where $\hat{Z}$ and Z^* are the respective normalizing constants (partition functions). Let $\epsilon = \sup_{x_{t-1}} |\hat{V}_{t-1}(x_{t-1}) - V_{t-1}(x_{t-1})|$ be the maximum absolute difference between the potential functions. We aim to prove that:

$$D_{TV}(\hat{q}(x_{t-1}|x_t) \parallel \pi^*_t(x_{t-1}|x_t)) \leq 2\beta\epsilon$$

To this end, Let $\Delta(x_{t-1}) = \hat{V}(x_{t-1}) - V(x_{t-1})$ be the difference between the potential functions at any point x_{t-1}. By definition, $-\epsilon \leq \Delta(x_{t-1}) \leq \epsilon$. Multiplying by the temperature parameter β and exponentiating gives us strict bounds on the ratio of the unnormalized exponential terms:

$$e^{-\beta\epsilon} \leq \exp(\beta\Delta(x_{t-1})) \leq e^{\beta\epsilon}$$

$$e^{-\beta\epsilon} \leq \frac{\exp(\beta\hat{V}_{t-1}(x_{t-1}))}{\exp(\beta V_{t-1}(x_{t-1}))} \leq e^{\beta\epsilon} \tag{14}$$

The partition function Z^* for the true distribution is defined as $Z^* = \int \hat{p}_{\hat{v}}(x_{t-1}|x_t) \exp(\beta V_{t-1}(x_{t-1}))dx_{t-1}$. The partition function $\hat{Z}$ for the empirical distribution is:

$$\hat{Z} = \int \hat{p}_{\hat{v}}(x_{t-1}|x_t) \exp(\beta\hat{V}_{t-1}(x_{t-1}))dx_{t-1} = \int \hat{p}_{\hat{v}}(x_{t-1}|x_t) \exp(\beta V_{t-1}(x_{t-1})) \exp(\beta\Delta(x_{t-1}))dx_{t-1}$$

Using our bounds from Eqn. 14, we can factor the maximum deviations out of the integral:

$$\hat{Z} \leq \int \hat{p}_{\hat{v}}(x_{t-1}|x_t) \exp(\beta V_{t-1}(x_{t-1}))e^{\beta\epsilon}dx_{t-1} = e^{\beta\epsilon}Z^*$$

$$\hat{Z} \geq \int \hat{p}_{\hat{v}}(x_{t-1}|x_t) \exp(\beta V_{t-1}(x_{t-1}))e^{-\beta\epsilon}dx_{t-1} = e^{-\beta\epsilon}Z^*$$

Rearranging these inequalities gives us bounds on the ratio of the two normalizing constants:

$$e^{-\beta\epsilon} \leq \frac{Z^*}{\hat{Z}} \leq e^{\beta\epsilon}$$

We can now evaluate the ratio of the two full probability distributions $\frac{\hat{q}(x_{t-1}|x_t)}{\pi^*(x_{t-1}|x_t)}$:

$$\frac{\hat{q}(x_{t-1}|x_t)}{\pi^*(x_{t-1}|x_t)} = \frac{\exp(\beta\hat{V}_{t-1}(x_{t-1}))}{\exp(\beta V_{t-1}(x_{t-1}))} \cdot \frac{Z^*}{\hat{Z}}$$

Since we have lower bounds for both fractions on the right side, we can multiply them to find the minimum possible ratio between the two distributions:

$$\frac{\hat{q}(x_{t-1}|x_t)}{\pi^*(x_{t-1}|x_t)} \geq e^{-\beta\epsilon} \cdot e^{-\beta\epsilon} = e^{-2\beta\epsilon}$$

This implies that for all x_{t-1}:

$$\hat{q}(x_{t-1}|x_t) \geq e^{-2\beta\epsilon}\pi^*(x_{t-1}|x_t) \tag{15}$$

An alternative but mathematically equivalent definition of Total Variation distance between two probability measures is:

$$D_{TV}(\hat{q}(x_{t-1}|x_t) \parallel \pi^*(x_{t-1}|x_t)) = 1 - \int \min(\hat{q}(x_{t-1}|x_t), \pi^*(x_{t-1}|x_t))dx_{t-1}$$

Using the inequality established in 15, we know that $\min(\hat{q}(x_{t-1}|x_t), \pi^*(x_{t-1}|x_t))$ must be at least $e^{-2\beta\epsilon}\pi^*(x_{t-1}|x_t)$. Therefore:

$$\int \min(\hat{q}(x_{t-1}|x_t), \pi^*(x_{t-1}|x_t))dx_{t-1} \geq \int e^{-2\beta\epsilon}\pi^*(x_{t-1}|x_t)dx_{t-1} = e^{-2\beta\epsilon}\int \pi^*(x_{t-1}|x_t)dx_{t-1}$$

Because $\pi^*(x_{t-1}|x_t)$ is a valid probability distribution, its integral over the entire domain is exactly 1.

$$\int \min(\hat{q}(x_{t-1}|x_t), \pi^*(x_{t-1}|x_t))dx \geq e^{-2\beta\epsilon}$$

Substitute this back into the TV distance definition:

$$D_{TV}(\hat{q}(x_{t-1}|x_t) \parallel \pi^*(x_{t-1}|x_t)) \leq 1 - e^{-2\beta\epsilon}$$

For any $y \geq 0$, the exponential inequality $1 - e^{-y} \leq y$ holds true. Setting $y = 2\beta\epsilon$:

$$1 - e^{-2\beta\epsilon} \leq 2\beta\epsilon$$

Thus, we conclude that:

$$D_{TV}(\hat{q}(x_{t-1}|x_t) \parallel \pi^*(x_{t-1}|x_t)) \leq 2\beta\epsilon = 2\beta \sup_{x_{t-1}} |\hat{V}_{t-1}(x_{t-1}) - V_{t-1}(x_{t-1})|$$

Substituting our propagated value error from **Step 2**:

$$D_{TV}(\hat{q}(x_{t-1}|x_t) \parallel \pi^*(x_{t-1}|x_t)) \leq \mathcal{O}\left(\beta T M^{-1/4}\right)$$

Step 4: Full Trajectory Divergence: Finally, we bound the divergence of the entire generated sample x_0. Following (Benton et al., 2024), and by using the chain rule for Total Variation distance over Markov chains, the total error is at most the sum of the step-wise errors over the T steps of the diffusion horizon:

$$D_{TV}(\hat{q}_M(x_0) \parallel \pi^*(x_0)) \leq \sum_{t=1}^{T} D_{TV}(\hat{q}(x_{t-1}|x_t)) \parallel \pi^*(x_{t-1}|x_t))$$

$$D_{TV}(\hat{q}_M \parallel \pi^*) \leq T \times \mathcal{O}\left(\beta T M^{-1/4}\right) = \mathcal{O}\left(\beta T^2 M^{-1/4}\right)$$

This completes the proof. □

B Proof of Proposition 4.1

Proposition 4.1 (Bootstrap Sufficient Statistic Realizes DDPM-Like Stochastic Path)

Let $0 < t'' < t' < t$ be any two consecutive transition time steps in a trajectory, and $p^{\text{DDPM}}_{t''|t'}$ be the DDPM transition kernel. Then for $x_{t''}$ obtained via 8, starting from x_t:

$$x_{t''} \sim p^{\text{DDPM}}_{t''|t'}(\cdot \mid x_t) \tag{16}$$

Proof. We present the proof in two steps.

Step 1: We first show that starting at x_t, the children of x_t (i.e., $x_{t^{child}}$) (i.e., equivalent to a Single transition) obtained via 8, is a valid DDPM transition. Given x_t as the starting node of the trajectory, and $x_{t^{child}}$ is the next node in the trajectory. We first compute $r^* = g(x_t, x_{t^{child}})$, then for $\epsilon \sim \mathcal{N}(0, I_d)$, we define the random variable:

$$\tilde{X}_{t^{child}} = \alpha_{t^{child}} S_{r^*,t}(x_{r^*}, x_t) \tag{17}$$

We aim to prove that:

$$\tilde{p}(\tilde{X}_{t^{child}} = x_{t^{child}} | X_t = x_t) = p^{\text{DDPM}}_{t^{child}|t}(X_{t^{child}} = x_{t^{child}} | X_t = x_t) \tag{18}$$

Where the forward process of DDPM is defined via a transition kernel $q_{t|t'}$ in reverse-time given by:

$$q_{t|t^{child}}(x_t | x_{t^{child}}) = \mathcal{N}\left(\frac{\alpha_t}{\alpha_{t^{child}}} x_{t^{child}}; \left(\sigma_t^2 - \frac{\alpha_t^2}{\alpha_{t^{child}}^2}\sigma_{t^{child}}^2\right) I\right), \quad (t < t^{child}) \tag{19}$$

The time-reversal SDE or DDPM transitions are then given by:

$$p^{\text{DDPM}}_{t^{child}|t}(x_{t^{child}} | x_t) = q_{t|t^{child}}(x_t | x_{t^{child}}) \frac{p_{t^{child}}(x_{t^{child}})}{p_t(x_t)} \tag{20}$$

In other words, we must demonstrate that the conditional distribution of $\tilde{X}_{t^{child}}$ given $X_t = x_t$ exactly matches the DDPM transition kernel $p^{\text{DDPM}}_{t^{child}|t}$. To establish this, it suffices to show that their corresponding joint distributions are identical:

$$\tilde{p}(\tilde{X}_{t^{child}} = x_{t^{child}}, X_t = x_t) = p_{t^{child},t}(X_{t^{child}} = x_{t^{child}}, X_t = x_t) \tag{21}$$

for $x_t \sim p_t$. As the DDPM transition of $q_{t|t^{child}}$, we know that:

$$p_{t,t^{child}}(x_t, x_{t^{child}}) = \int q_{t|t^{child}}(x_t | x_{t^{child}}) p_{t^{child}}(x_{t^{child}} | x_1) p_{\text{data}}(x_1) dx_1 \tag{22}$$

where $q_{t|t^{child}}(x_t | x_{t^{child}}) = \mathcal{N}\left(\frac{\alpha_t}{\alpha_{t^{child}}} x_{t^{child}}; \left(\sigma_t^2 - \frac{\alpha_t^2}{\alpha_{t^{child}}^2}\sigma_{t^{child}}^2\right) I\right)$ and $p_{t^{child}}(x_{t^{child}} | x_1) = \mathcal{N}(\alpha_{t^{child}} x_1; \sigma_{t^{child}}^2 I)$.

Now, we get a sample $(X_t, X_{t^{child}}) \sim p_{t,t^{child}}(x_t, x_{t^{child}})$ via:

$$\begin{aligned} X_1 &\sim p_{\text{data}}, \\ X_{t^{child}} &= \alpha_{t^{child}} X_1 + \sigma_{t^{child}} \epsilon_1, \\ X_t &= \frac{\alpha_t}{\alpha_{t^{child}}} X_{t^{child}} + \sqrt{\sigma_t^2 - \frac{\alpha_t^2}{\alpha_{t^{child}}^2}\sigma_{t^{child}}^2}\, \epsilon_2, \quad \epsilon_1, \epsilon_2 \sim \mathcal{N}(0, I_d) \end{aligned}$$

Next, to sample from $\tilde{p}(\tilde{x}_{t^{child}}, x_t)$, we can perform the following steps : Sample $X_1 \sim p_{\text{data}}$ first and then renoise to obtain X_t and X_{r^*} independently and then apply stochastic interpolant to obtain $X_{t^{child}}$. Specifically, we perform the following steps:

$$X_1 \sim p_{\text{data}}, \tag{23}$$

$$X_{r^*} = \bar{\alpha}_{r^*} X_1 + \bar{\sigma}_{r^*} \tilde{\epsilon}_1 \tag{24}$$

$$X_t = \alpha_t X_1 + \sigma_t \tilde{\epsilon}_2 \tag{25}$$

$$X_{t^{child}} = \alpha_{t^{child}} \frac{\bar{\alpha}_{r^*}\sigma_t^2 X_{r^*} + \alpha_t \bar{\sigma}_{r^*}^2 X_t}{\bar{\alpha}_{r^*}^2 \sigma_t^2 + \alpha_t^2 \bar{\sigma}_{r^*}^2} \tag{26}$$

We obtain 26 by following the definition of the GLASS velocity field (Holderrieth et al., 2025) and GLASS probability path (Holderrieth et al., 2025). Here, $\bar{\gamma} = \rho \sigma_{t^{child}} \sigma_t^{-1}$, $\bar{\alpha} = \alpha_{t^{child}} - \bar{\gamma}\alpha_t$, $\bar{\sigma}^2 = \sigma_{t^{child}}^2 (1 - \rho^2)$. ρ is the correlation parameter.

Next, we aim to show that $(X_t, X_{t^{child}})$ sampled via this procedure results in the same distribution as sampling them via the SDE/DDPM. But it is sufficient if we can show that their distribution coincides conditioned on X_1. Therefore, let's fix $X_1 = x_1$. Then, for the DDPM, it holds that:

$$(X_t, X_{t^{child}}) | X_1 = x_1 \sim \mathcal{N}\left(\begin{pmatrix} \alpha_t x_1 \\ \alpha_{t^{child}} x_1 \end{pmatrix}, \begin{pmatrix} \sigma_t^2 & \frac{\alpha_t}{\alpha_{t^{child}}}\sigma_{t^{child}}^2 \\ \frac{\alpha_t}{\alpha_{t^{child}}}\sigma_{t^{child}}^2 & \sigma_{t^{child}}^2 \end{pmatrix}\right) \tag{27}$$

Similarly, covariance can be derived conditioned on X_1 using equation 26 and the fact that X_t, X_{r^*} are independent given X_1:

$$\begin{aligned}\text{Cov}(X_t, X_{t^{child}}|X_1) &= \alpha_{t^{child}} \frac{\bar{\alpha}_{r^*}\sigma_t^2 \text{Cov}(X_{r^*}, X_t|X_1) + \alpha_t \bar{\sigma}_{r^*}^2 \text{Cov}(X_t, X_t|X_1)}{\bar{\alpha}_{r^*}^2\sigma_t^2 + \alpha_t^2\bar{\sigma}_{r^*}^2} \\ &= \alpha_{t^{child}} \frac{\alpha_t \bar{\sigma}_{r^*}^2 \text{Cov}(X_t, X_t|X_1)}{\bar{\alpha}_{r^*}^2\sigma_t^2 + \alpha_t^2\bar{\sigma}_{r^*}^2} \\ &= \frac{\alpha_{t^{child}}\alpha_t \bar{\sigma}_{r^*}^2 \text{Var}(X_t|X_1)}{\bar{\alpha}_{r^*}^2\sigma_t^2 + \alpha_t^2\bar{\sigma}_{r^*}^2} \\ &= \frac{\alpha_{t^{child}}\alpha_t \bar{\sigma}_{r^*}^2 \sigma_t^2}{\bar{\alpha}_{r^*}^2\sigma_t^2 + \alpha_t^2\bar{\sigma}_{r^*}^2}\end{aligned}$$

We know by definition:

$$\begin{aligned}\sigma_{t^{child}}^2 = \alpha_{t^{child}}^2 g(t^{child}) = \alpha_{t^{child}}^2 g(t^*(r^*, t)) &= \alpha_{t^{child}}^2 \frac{\sigma_t^2\bar{\sigma}_{r^*}^2}{\bar{\alpha}_{r^*}^2\sigma_t^2 + \alpha_t^2\bar{\sigma}_{r^*}^2} \\ &= \frac{\alpha_{t^{child}}}{\alpha_t}\left(\frac{\alpha_{t^{child}}\alpha_t\bar{\sigma}_{r^*}^2\sigma_t^2}{\bar{\alpha}_{r^*}^2\sigma_t^2 + \alpha_t^2\bar{\sigma}_{r^*}^2}\right) \\ &= \frac{\alpha_{t^{child}}}{\alpha_t}\text{Cov}(X_t, X_{t^{child}}|X_1)\end{aligned}$$

Similarly, we can show that:

$$\begin{aligned}\text{Var}(X_{t^{child}}|X_1) &= \alpha_{t^{child}}^2 \frac{\bar{\alpha}_{r^*}^2\sigma_t^4\bar{\sigma}_{r^*}^2 + \alpha_t^2\bar{\sigma}_{r^*}^4\sigma_t^2}{(\bar{\alpha}_{r^*}^2\sigma_t^2 + \alpha_t^2\bar{\sigma}_{r^*}^2)^2} \\ &= \alpha_{t^{child}}^2 \left(\frac{\bar{\alpha}_{r^*}^2\sigma_t^2 + \alpha_t^2\bar{\sigma}_{r^*}^2}{\bar{\alpha}_{r^*}^2\sigma_t^2 + \alpha_t^2\bar{\sigma}_{r^*}^2}\right)\frac{\sigma_t^2\bar{\sigma}_{r^*}^2}{\bar{\alpha}_{r^*}^2\sigma_t^2 + \alpha_t^2\bar{\sigma}_{r^*}^2} \\ &= \alpha_{t^{child}}^2 \frac{\sigma_t^2\bar{\sigma}_{r^*}^2}{\bar{\alpha}_{r^*}^2\sigma_t^2 + \alpha_t^2\bar{\sigma}_{r^*}^2} \\ &= \alpha_{t^{child}}^2 g(t^{child}) = \sigma_{t^{child}}^2\end{aligned}$$

Hence, we can write that:

$$(X_t, X_{t^{child}})|X_1 = x_1 \sim \mathcal{N}\left(\begin{pmatrix}\alpha_t x_1 \\ \alpha_{t^{child}} x_1\end{pmatrix}, \begin{pmatrix}\sigma_t^2 & \frac{\alpha_t}{\alpha_{t^{child}}}\sigma_{t^{child}}^2 \\ \frac{\alpha_t}{\alpha_{t^{child}}}\sigma_{t^{child}}^2 & \sigma_{t^{child}}^2\end{pmatrix}\right) \tag{28}$$

which precisely matches the distribution of the memoryless SDE conditioned on X_1. By marginalizing over $X_1 \sim p_{\text{data}}$, we establish that the joint distributions are identical, fulfilling equation 21. This joint equality directly implies the equivalence of the target distributions conditioned on $X_t = x_t$.

Step 2: Proof by Induction Having established the relationship between x_t and $x_{t^{child}}$, we now aim to show that the child of $x_{t^{child}}$ obtained via 8 also follows the DDPM transition kernel. To this end, we simply consider

$x_t = x_{t^{child}}$, and the next transition time step in the trajectory is:

t^{child} = time step of the **child of** $x_{t^{child}}$.

Assuming this, we can compute the updated $r^* = g(x_t, x_{t^{child}})$, and as before, we can compute $\tilde{X}_{t^{child}} = \alpha_{t^{child}} S_{r^*,t}(x_{r^*}, x_t)$. Then, following the results in Step 1, we can conclude that the child of $x_{t^{child}}$ is also a valid DDPM transition from $x_{t^{child}}$.

By iteratively repeating this procedure as defined in 8, we can conclude that any intermediate node in the trajectory (say $x_{t'}$) has a corresponding child $x_{t''}$ that also follows a valid DDPM transition. In other words, $x_{t''}$ obtained via 8, starting from x_t follows:

$$x_{t''} \sim p_{t''|t'}^{\text{DDPM}}(\cdot \mid x_t) \tag{29}$$

This concludes the proof.

□

C Derivation of Equation 11

Let's assume $\hat{p}_{\hat{v}}(x_{t-1}|x_t) = \mathrm{BSS}^1(x_0 \sim p_{0|t}(\cdot|x_t))$.

The joint target density over the full chain $(x_0, \ldots, x_{t-1}, x_t)$ is given by:

$$\pi^*(x_0, \ldots, x_{t-1}, x_t) = \frac{1}{Z}\hat{p}_{\hat{v}}(x_0, \ldots, x_{t-1}, x_t)\exp(\beta r(x_0)), \tag{30}$$

where Z represents the normalization constant of this joint density.

The marginal joint density of (x_t, x_{t-1}) under π^* is:

$$\begin{aligned}
\pi^*(x_t, x_{t-1}) &= \frac{1}{Z}\int \hat{p}_{\hat{v}}(x_0, \ldots, x_{t-1}, x_t)\exp(\beta r(x_0))\, dx_0 \ldots dx_{t-2} \\
&= \frac{1}{Z}\hat{p}_{\hat{v}}(x_t)\hat{p}_{\hat{v}}(x_{t-1} \mid x_t)\left(\int \hat{p}_{\hat{v}}(x_0, \ldots, x_{t-2} \mid x_{t-1})\exp(\beta r(x_0))\, dx_0 \ldots dx_{t-2}\right) \\
&= \frac{1}{Z}\hat{p}_{\hat{v}}(x_t)\mathrm{BSS}^1(x_0 \sim p_{0|t}(\cdot \mid x_t))\exp(\beta V_{t-1}(x_{t-1})).
\end{aligned} \tag{31}$$

Similarly, the marginal density of x_t under π^* is:

$$\pi^*(x_t) = \frac{1}{Z}\hat{p}_{\hat{v}}(x_t)\exp(\beta V_t(x_t)). \tag{32}$$

By dividing these two marginals, we get the transitions under the optimal policy:

$$\begin{aligned}
\pi^*(x_{t-1} \mid x_t) &= \frac{\pi^*(x_t, x_{t-1})}{\pi^*(x_t)} \\
&= \frac{\hat{p}_{\hat{v}}(x_{t-1} \mid x_t)\exp(\beta V_{t-1}(x_{t-1}))}{\exp(\beta V_t(x_t))} \\
&= \frac{\hat{p}_{\hat{v}}(x_{t-1} \mid x_t)\exp(\beta V_{t-1}(x_{t-1}))}{\int \hat{p}_{\hat{v}}(x_{t-1} \mid x_t)\exp(\beta V_{t-1}(x_{t-1}))\, dx_{t-1}} \\
&= \frac{\mathrm{BSS}^1(x_0 \sim p_{0|t}(\cdot|x_t))\exp(\beta V_{t-1}(x_{t-1}))}{\int \mathrm{BSS}^1(x_0 \sim p_{0|t}(\cdot|x_t))\exp(\beta V_{t-1}(x_{t-1}))\, dx_{t-1}}.
\end{aligned} \tag{33}$$

The above relation gives the optimal policy from Equation 11.

D Detailed Results on the Quantity Aware Alignment Task

Here, we present results for the quantity-aware alignment task. Quantitative comparisons are detailed in Fig. 11. Mirroring the patterns observed in the compositional alignment task, BFMT achieves high reward scores (corresponding to lower RSS scores) using significantly fewer feedback budgets. These findings further validate the efficacy of BFMT for complex, online feedback-driven alignment tasks.

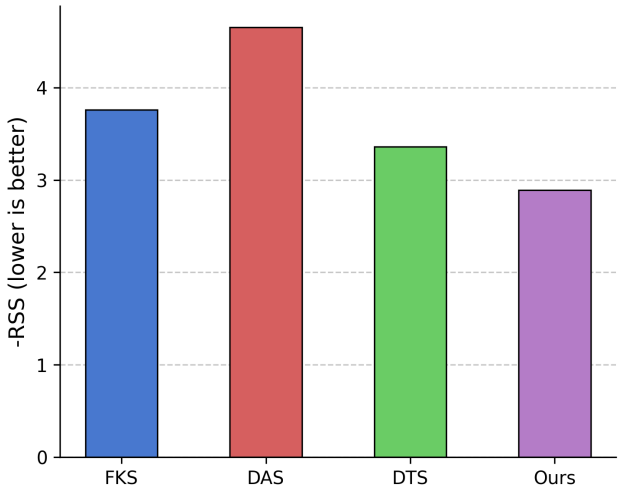


Figure 11: Performance Comparisons on Quantity-Aware Alignment Task.

Additional Quantitative Comparisons with Baselines on the Quantity-aware Alignment Task We also evaluate the alignment performance of BFMT against several baselines in Figure 12, using ImageReward as the evaluation metric. BFMT consistently outperforms all baselines across both NFE and feedback budgets, achieving higher mean reward and further underscoring the effectiveness of the BFMT framework in the online feedback-driven quantity-aware alignment settings. These findings further strengthen the impact of BFMT.

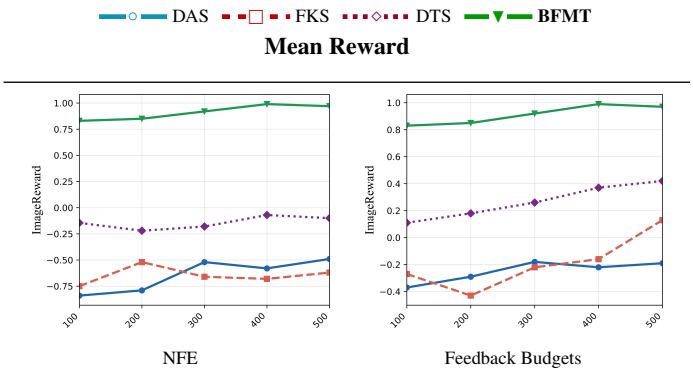


Figure 12: Alignment Performance Analysis on Imagenet Classes.

E Visualizing the Impact of Bootstrap Sufficient Statistic

In the main paper (Section 3), we analyze the impact of the Bootstrap Sufficient Statistic (BSS) both qualitatively and quantitatively. Here, we present additional visualizations highlighting that the BSS mechanism is crucial for generating high-quality, target-aligned samples, particularly under strict NFE constraints. As depicted in Fig. 13, an alternative method relying on an iterative stochastic interpolant fails to produce high-quality, aligned images. In contrast, the BSS-based approach succeeds because it guarantees a DDPM-like trajectory rollout, thereby ensuring accurate sampling from the true target distribution.

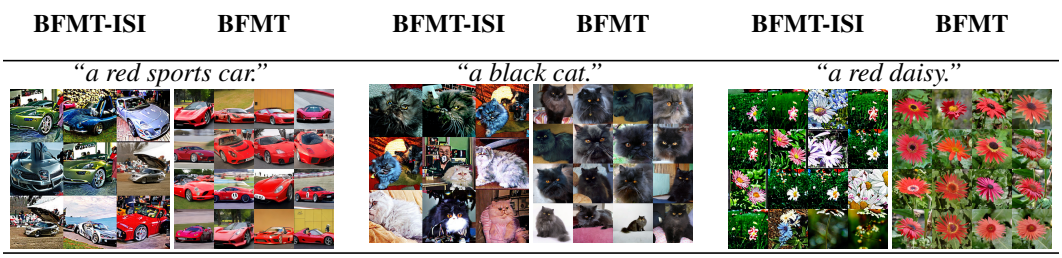


Figure 13: Additional Visualizations on the impact of BSS.

F A Visual Illustration of BFMT's Exploratory Search Strategy

Figure 14 visualizes the evolution of BFMT's search strategy across different feedback stages. In the early stages, BFMT explores broadly, producing semantically diverse samples within the target category, albeit without precise alignment to the fine-grained target prompt. As feedback accumulates and the search matures, BFMT progressively converges toward samples that closely match the target prompt — reflecting an efficient search process. This exploration-then-exploitation dynamic underscores BFMT's strong exploratory capability, which is fundamental to its success in online feedback-driven settings.



Figure 14: Exploration Strategy of BFMT in Online Feedback-Driven Search Settings.

G Additional Visual Illustration of BFMT's Exploratory Search Strategy

Figures 15 and 16 illustrate how BFMT's search strategy evolves across feedback stages. Initially, BFMT explores broadly, generating semantically diverse samples within the target category, though not yet precisely aligned with the fine-grained target prompt. As feedback accumulates, it progressively converges toward well-aligned samples, showcasing an effective exploration-to-exploitation dynamic that is central to its success in online feedback-driven settings.

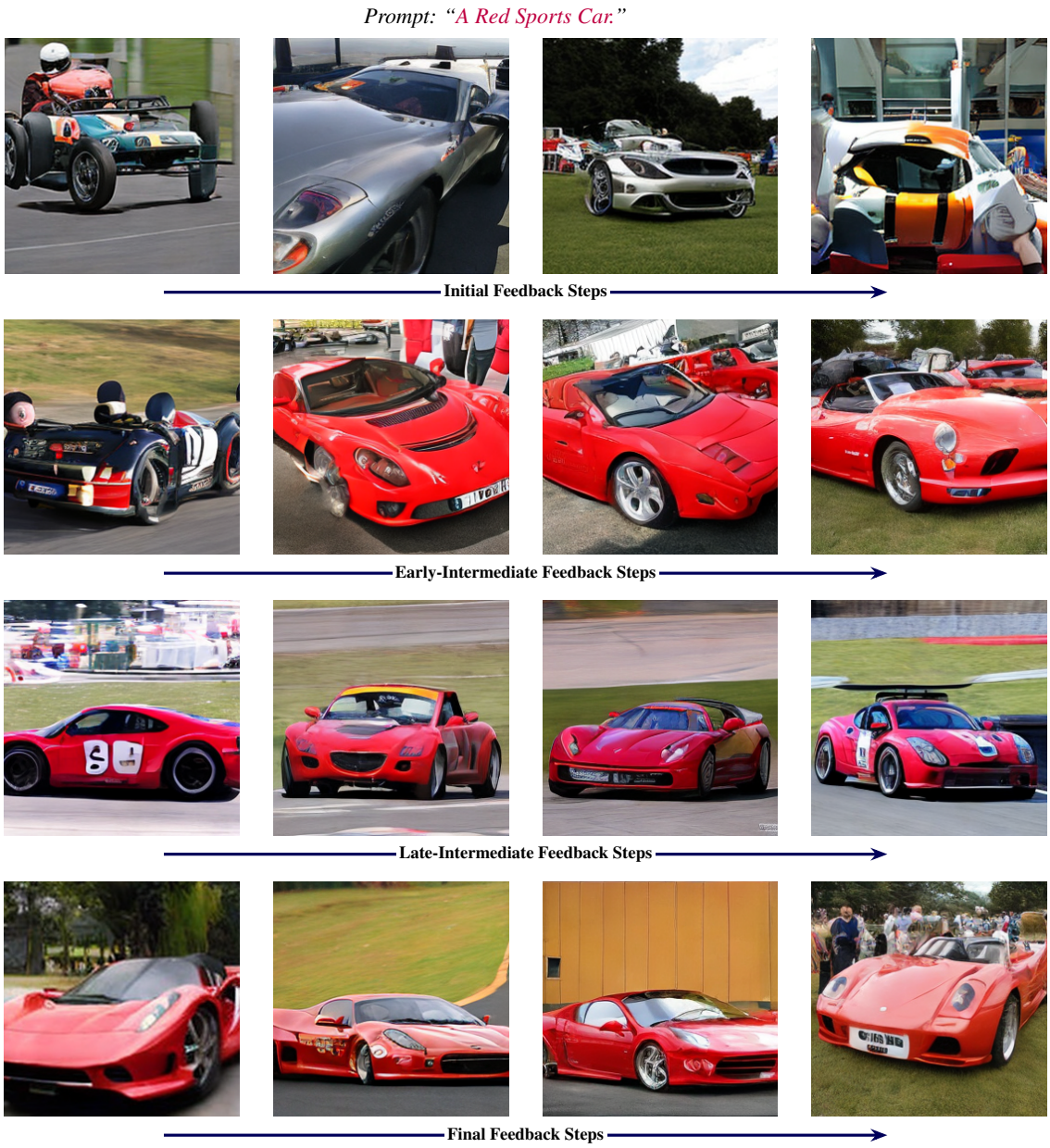


Figure 15: Exploration Strategy of BFMT on Online Feedback-Driven Search Settings.



Figure 16: Exploration Strategy of BFMT on Online Feedback-Driven Search Settings.

H Hierarchical Exploration Strategy of BFMT

Here, we illustrate BFMT's hierarchical search strategy and its enhanced exploration capability, enabled by the dynamic transition schedule. As shown in the figure 17, smaller transition steps in

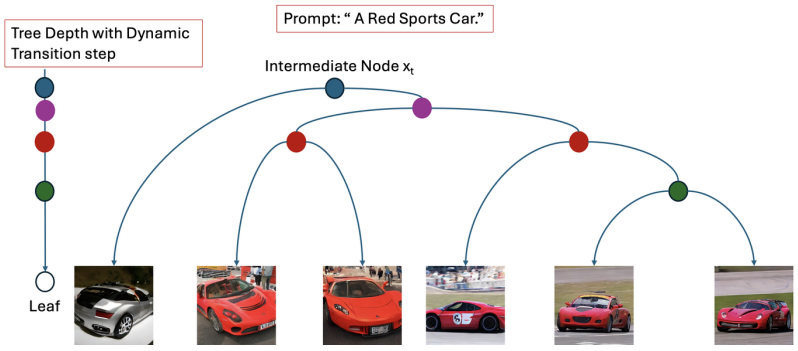


Figure 17: BFMT's Hierarchical Search Strategy.

the early stages allow BFMT to explore the space more broadly (e.g., BFMT samples a car with a different color and a different background), while larger transition steps in the later stages enable it to focus on and sample from high-utility modes (e.g., BFMT samples a red sports car).

I Additional Visualizations of Hierarchical Exploration Strategy of BFMT

Figures 18 and 19 present additional visualizations of BFMT's hierarchical exploration strategy, further reinforcing its effectiveness in enabling efficient online feedback-driven search.

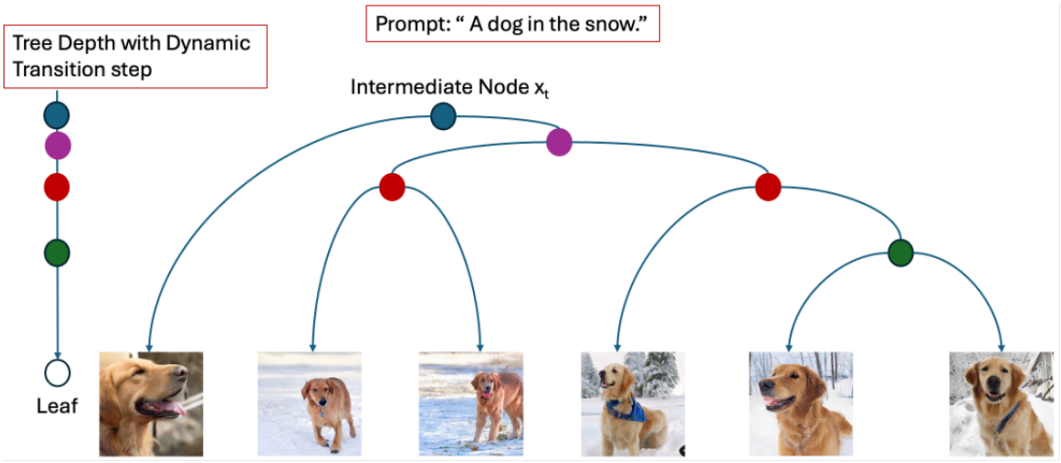


Figure 18: BFMT's Hierarchical Search Strategy.

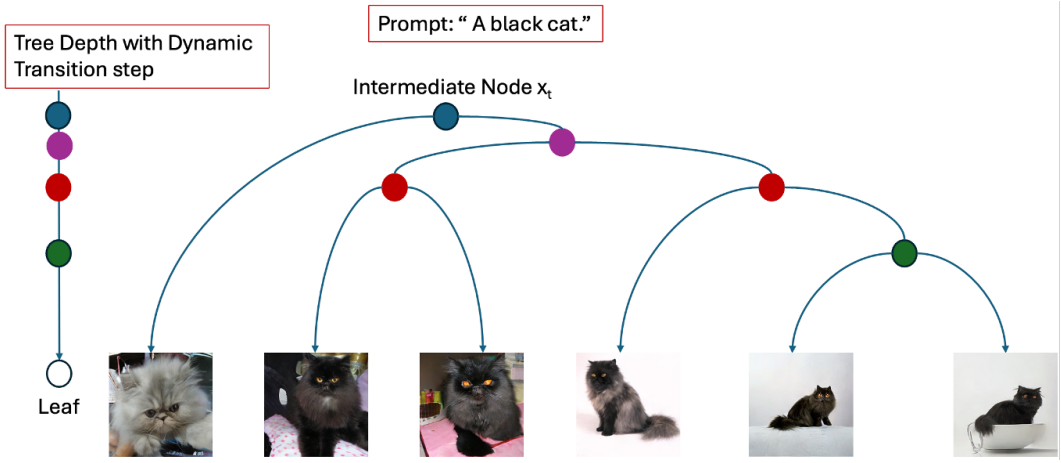


Figure 19: BFMT's Hierarchical Search Strategy.

J Adapting the TrigFlow Noise Schedule to the BFMT Framework

In this section, we describe how we adapt the trigonometric noise schedule from TrigFlow Chen et al. (2025) and the consistency model (SANA-Sprint) Chen et al. (2025) to our BFMT framework, which operates under a different noise schedule. We first review the TrigFlow parameterization, then describe the consistency model reparameterization, and finally derive the adaptation of the noise schedule to our setting.

J.1 Background: TrigFlow Transformation

TrigFlow introduces a trigonometric reparameterization of the Flow Matching timestep, establishing a bijective mapping between the standard FM timestep $t_{FM} \in [0, 1]$ and a cosine-based schedule

parameterized by t_{cm}. Specifically, the mapping is defined as:

$$t_{FM} = \frac{\sin(t_{cm})}{\sin(t_{cm}) + \cos(t_{cm})}, \tag{34}$$

where t_{cm} controls the balance between noise and signal through trigonometric interpolation. This mapping ensures that $t_{FM} = 0$ corresponds to a clean sample and $t_{FM} = 1$ corresponds to pure noise, while providing a smoother, geometry-aware interpolation path compared to a linear schedule.

Given the mapping in Eq. equation 34, the noisy latent $X_{t_{FM}}$ is expressed in terms of the original latent $X_{t_{cm}}$ as:

$$X_{t_{FM}} = \frac{X_{t_{cm}}}{\sigma_d} \sqrt{t_{FM}^2 + (1 - t_{FM})^2}, \tag{35}$$

where σ_d is a scaling factor that normalizes the latent representation. The factor $\sqrt{t_{FM}^2 + (1 - t_{FM})^2}$ accounts for the norm of the trigonometric interpolation coefficients, ensuring consistent scaling across timesteps.

Under this reparameterization, the score network $\hat{F}_\theta$ is reformulated to operate on the rescaled latent $X_{t_{cm}}/\sigma_d$ at timestep t_{cm}. The relationship between the TrigFlow score network and the velocity field V_θ of the standard Flow Matching model is:

$$\begin{aligned} \hat{F}_\theta\left(\frac{X_{t_{cm}}}{\sigma_d},\, t_{cm},\, y\right) = \frac{1}{\sqrt{t_{FM}^2 + (1 - t_{FM})^2}} \\ \times \Big[(1 - 2t_{FM})\, X_{t,FM} + \left(1 - 2t_{FM} + 2t_{FM}^2\right) V_\theta(X_{t_{FM}},\, t_{FM},\, y)\Big], \end{aligned} \tag{36}$$

where y is the conditioning input (e.g., a text prompt). This reformulation shows that the TrigFlow score network can be expressed as a linear combination of the noisy latent and the predicted velocity field, weighted by time-dependent coefficients that are derived analytically from the trigonometric schedule.

J.2 SANA-Sprint Reparameterization

SANA-Sprint builds upon TrigFlow by introducing a consistency-style denoising function f_θ that directly maps a noisy latent X_t to a clean estimate $\hat{X}_0$ in a single step. Under the TrigFlow time parameterization, this function takes the form:

$$f_\theta(X_t, t) = \cos(t_{cm}) \cdot X_t - \sin(t_{cm}) \cdot \sigma_d \cdot F_\theta\left(\frac{X_t}{\sigma_d},\, t_{cm}\right), \tag{37}$$

where $X_{cm} = X_t/\text{Scale}$ is the rescaled latent. Intuitively, Eq. equation 37 can be understood as a trigonometric decomposition: the $\cos(t_{cm})$ term retains the signal component of X_t, while the $\sin(t_{cm})$ term removes the noise component as predicted by F_θ. This yields the denoised estimate:

$$\hat{X}_0 = \cos(t_{cm})\, X_{t_{cm}} - \sin(t_{cm})\, \hat{F}_\theta, \tag{38}$$

which provides a clean, geometry-consistent reconstruction of the original sample from any noisy intermediate state.

J.3 Adapting to Our BFMT Framework with the Sufficient Statistic (S) Schedule

Our BFMT framework uses a different noise schedule compared to TrigFlow. Specifically, we adopt the S schedule, which defines a signal-to-noise ratio (SNR) function that differs from the trigonometric parameterization used in TrigFlow. To adapt the SANA-Sprint framework to our setting, we must establish a consistent mapping between the two schedules.

Setting up the Schedule Correspondence. Under the S schedule, the signal and noise coefficients at timestep $t = t_{cm}$ are defined as:

$$\alpha_t = \cos t = \sqrt{s}, \qquad \sigma_t = \sin t = \sqrt{1 - s}, \tag{39}$$

where $s \in [0, 1]$ is the schedule variable. This gives the forward noising process:

$$X_t = \left(\alpha_t \hat{X}_0 + \sigma_t \frac{\varepsilon}{\sigma_d}\right) \cdot \sigma_d = \cos(t)\, \hat{X}_0 \cdot \sigma_d + \sin(t)\, \varepsilon, \tag{40}$$

which is a standard affine interpolation between the clean sample $\hat{X}_0$ and Gaussian noise $\varepsilon \sim \mathcal{N}(0, I)$, scaled by the latent normalization factor σ_d.

Defining the SNR Function. Under the S schedule, the SNR function $g(s)$ and its transformed counterpart $\tilde{g}(\alpha_t)$ are defined as:

$$g(s) = \frac{1}{s} - 1, \qquad \tilde{g}(\alpha_t) = \frac{1}{\alpha_t} - 1 = \frac{1}{\cos^2(t_{cm})} - 1, \tag{41}$$

where $\tilde{g}(\alpha_t)$ is the SNR expressed as a function of the signal coefficient $\alpha_t = \cos(t_{cm})$. This function is monotonically decreasing in t_{cm}, consistent with the intuition that noise increases as the timestep progresses.

Deriving the Optimal Transition Timestep $r^* = r^*_{cm}$ for computing DDPM transition via Sufficient Statistic. A key step in adapting the noise schedule is determining the optimal intermediate timestep r^* at which to perform the consistency transition between two noise levels t_{cm} and $t_{cm'}$ (with $t_{cm'} < t_{cm}$, i.e., $t_{cm'}$ is closer to the clean data). The optimal r^* is derived by matching the SNR levels under the S schedule, yielding:

$$\tilde{g}(r^*) = \frac{\tilde{g}(\cos t_{cm'}) \cdot \tilde{g}(\cos t_{cm})}{\tilde{g}(\cos t_{cm}) - \tilde{g}(\cos t_{cm'})}, \tag{42}$$

which can be solved by applying the inverse of g to both sides:

$$\hat{r}^* = g^{-1}\left(\frac{\tilde{g}(\cos t_{cm'}) \cdot \tilde{g}(\cos t_{cm})}{\tilde{g}(\cos t_{cm}) - \tilde{g}(\cos t_{cm'})}\right). \tag{43}$$

Eq. equation 43 gives the optimal SNR level r^* in the s-domain. To recover the corresponding trigonometric timestep r^*_{cm}, we use the relation $\alpha_t = \cos(t_{cm}) = \sqrt{s}$, which gives:

$$\cos(r^*_{cm}) = d\sqrt{r^*} \quad \Longrightarrow \quad r^*_{cm} = \arccos\left(\sqrt{r^*}\right). \tag{44}$$

This recovers the optimal intermediate timestep r^*_{cm} in the trigonometric domain, which can be directly used within the TrigFlow/SANA-Sprint framework.

Computing the DDPM Transition via Sufficient Statistic Given the optimal transition timestep r^*_{cm}, the DDPM transition sample $X_{t_{cm'}}$ at the new timestep $t_{cm'}$ is obtained by applying the Sufficient Statistic scheme as defined in Equation 9 in the main paper.

K Details of Class and Prompt Detail in the Search Experiment

In the following table 2, we detail the randomly selected ImageNet target classes and their fine-grained specifications used to evaluate the competing methods on the online feedback-driven search task.

Target Class	**Fine-Grained Prompt**
Car	A Red Sports car.
Cat	A Black Cat.
Dog	A Dog in the Snow.
Pepper	A Green Bell Pepper.
Daisy	A Red Daisy.

Table 2: Target classes and their corresponding fine-grained prompts for the Online Feedback-driven Search task.

L Details about the Evaluation Dataset for Alignment Tasks

We provide the evaluation prompts for both the compositional and quantity-aware alignment tasks in separate Excel files, available via the zip file.

M Additional Results with Molecule Generation Task

Tables 3 and 4 report the mean validity, uniqueness, QED, and composite QED and SA scores for samples generated under feedback budgets 200 and 600, respectively. These results further highlight BFMT's ability to efficiently generate valid, target-aligned molecules even under tightly constrained feedback budgets.

Table 3: Validity Check for the Generated Molecules.

Method	Uniqueness	Validity	Mean QED	Uniqueness	Validity	Mean QED+SA
CFM	0.9993	0.9410	0.4876	0.9993	0.9410	0.5148
BFMT	0.9823	0.9620	**0.5489**	0.9778	0.9500	**0.5627**

Table 4: Validity Check for the Generated Molecules.

Method	Uniqueness	Validity	Mean QED	Uniqueness	Validity	Mean QED+SA
CFM	0.9989	0.9410	0.4925	0.9989	0.9410	0.5202
BFMT	0.9774	0.9579	**0.5575**	0.9766	0.9588	**0.5702**

N Implementation Details

Implementation details are available in the provided GitHub link. Finally, all experiments are implemented in PyTorch and conducted on 3 NVIDIA A100 GPUs. Our training and inference code will be made public.

O Additional Comparative Search Visualizations

We provide additional qualitative results for the online feedback-driven search task in Figure 20, comparing BFMT against the most competitive baselines. These visualizations further corroborate the ability of BFMT to produce diverse, target-aligned samples under online feedback.



Figure 20: Additional Search Visualizations.

P Additional Competitive Alignment Visualization

Figures 24, 21, 22, 23 present additional qualitative comparisons in the online feedback-driven alignment setting, further validating the effectiveness of BFMT over competitive baselines.

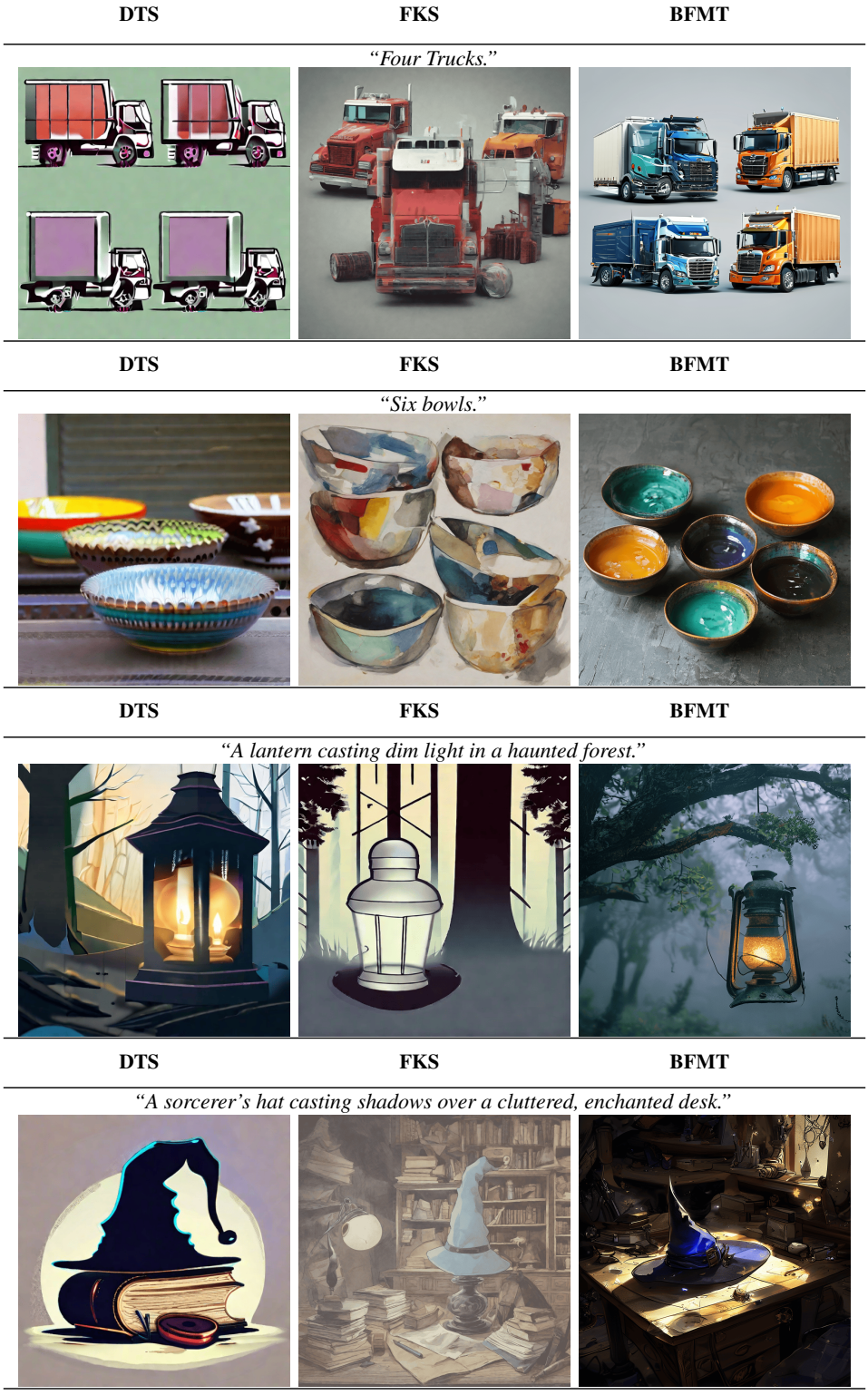


Figure 21: Additional Alignment Visualization.

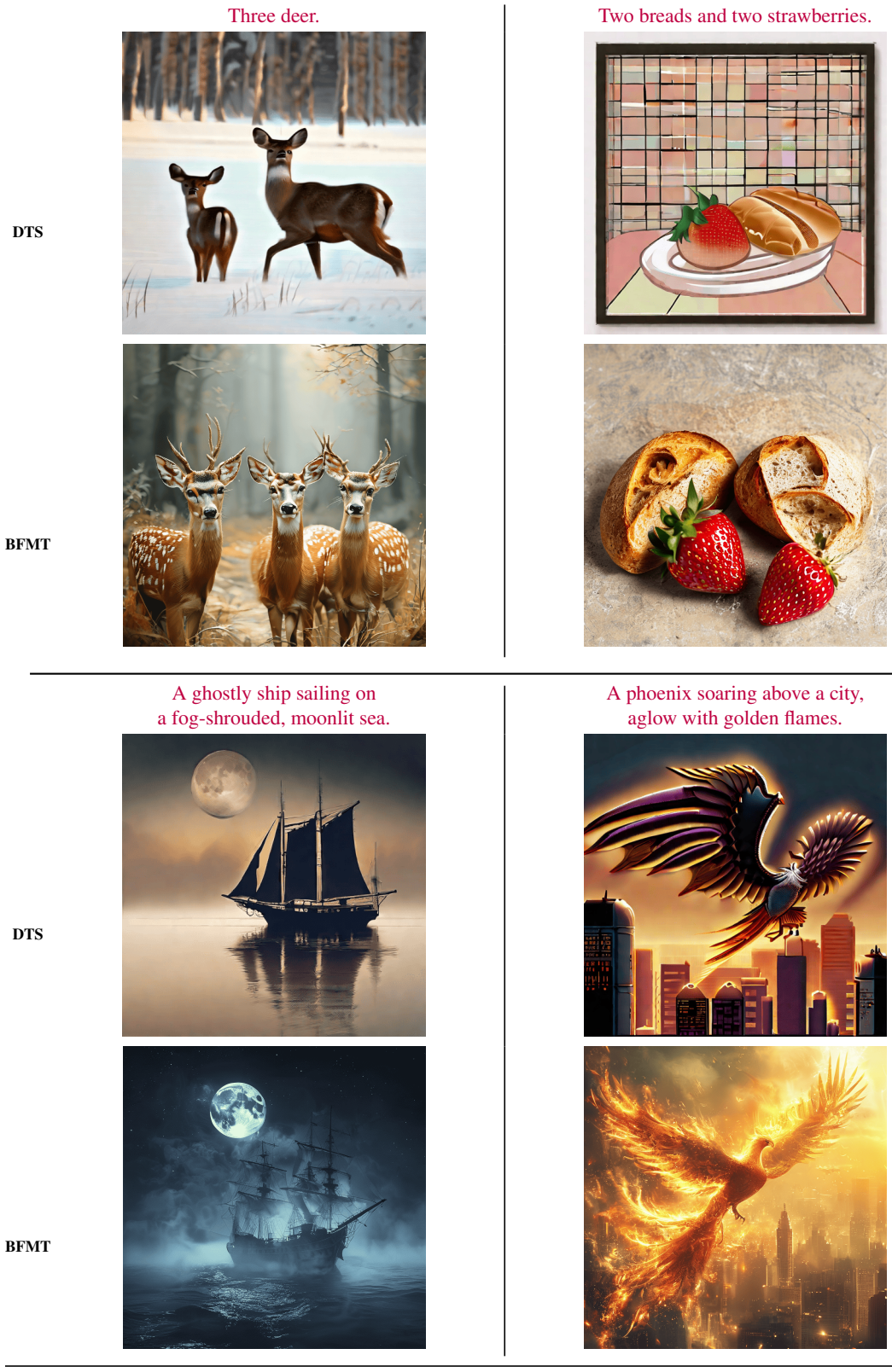


Figure 22: Additional Alignment Visualization.



Figure 23: Additional Alignment Visualization.

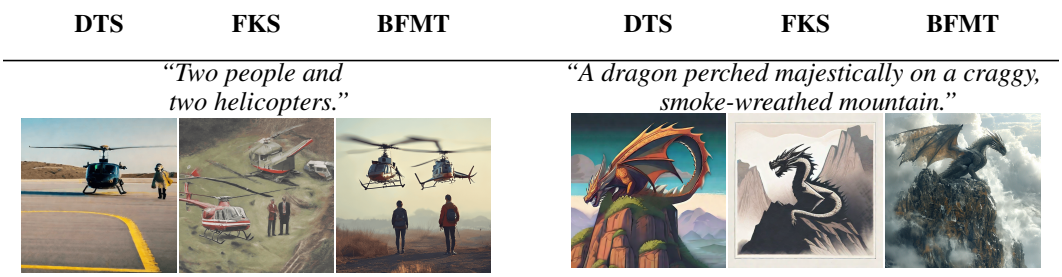


Figure 24: Alignment Visualization. (left) Quantity Prompt. (right) Compositional Prompt.