CONDITIONAL FEASIBILITY STEERING IN DIFFUSION MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Terminal verifiers can identify successful diffusion outputs but provide no guidance on unfinished states. We introduce Conditional Feasibility Steering (CFS), a fixed-budget discovery method that uses one completed pilot per saved state to allocate further generation. CFS favors states with successful pilots, independently reselects a state for each fresh continuation, and retains all completed outputs. A closed-form allocation rule bounds the second moment of the optional importance correction. No verifier gradient or learned value model is required. Our analysis connects the information in binary feedback to variation in conditional feasibility and separates the yield contributions of computation reuse and informed allocation. Controlled experiments in molecular and image generation show higher verified yield than uniform allocation within the same branching structure. Independent refresh reduces finite-target estimation error relative to single-parent commitment, although rejection remains more accurate. CFS is designed for verified discovery; its unweighted outputs are not claimed to follow the base conditional distribution.

1 INTRODUCTION

Steering diffusion models toward task requirements is difficult when feedback is available only after generation is complete. A molecule must satisfy chemical and property constraints (Krenn et al., 2020; Bickerton et al., 2012); generated code must pass executable tests (Chen et al., 2021; Liu et al., 2023). Projection-based methods enforce constraints during denoising when a suitable projection operator can be constructed (Christopher et al., 2024; Sharma et al., 2024; Cardei et al., 2026). We consider a different access setting: a fixed, potentially non-differentiable verifier returns only pass/fail on complete samples, with no gradient or projection operator supplied. How can this limited feedback help generate more successful samples without retraining the model?

Rejection sampling offers a direct solution (Chopin et al., 2020), but pays for a full denoising trajectory on every attempt (Ho et al., 2020; Rombach et al., 2022). Rare success makes this costly (Cérou & Guyader, 2007; Wu et al., 2023). Saved intermediate states provide cheaper starting points for further generation: branching shares computation, while guidance based on sampled futures can inform which states to revisit (Jain et al., 2026; Li et al., 2024; Lew et al., 2023). Our starting observation is that even one binary success indicator can inform this choice. When states differ in their probability of eventual success, those that produced a successful first sample are more promising on average for subsequent generation. Accurately estimating each state's success probability is unnecessary.

We turn this observation into *Conditional Feasibility Steering (CFS)*. Generate several samples while saving an intermediate state from each trajectory. Then resume denoising more often from states whose first samples passed, choosing a state independently for every additional sample. Keep the initial samples as well as the new ones. Each completed trajectory thus provides both an output and evidence for choosing a reusable continuation parent. The method requires no learned value model or verifier gradient. A constraint on the importance weights controls how strongly successful states are favored, drawing on the role of weight moments in importance-sampling accuracy (Agapiou et al., 2017).

Our goal is to generate more distinct samples that pass the verifier under a fixed inference-time compute budget. Resuming denoising from an intermediate state already saves computation by reusing part of a trajectory. To measure the additional value of verifier feedback, we compare CFS with a baseline that samples intermediate states uniformly for further denoising. Both methods use the

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.

same intermediate states, the same initial rollouts to completion, and the same number of additional denoising steps. Estimating the base model's distribution conditioned on verifier success is evaluated separately.

Our contributions are threefold.

- **Verifier-guided completion reuse.** We introduce a fixed-stage sampler that uses binary pilot outcomes to redistribute fresh continuations across saved diffusion states, while retaining every completed sample. A closed-form, moment-constrained rule controls allocation strength without a learned value model.
- **Information and allocation analysis.** We characterize how one terminal observation reveals variation in conditional feasibility and separate the expected valid-yield gains from reuse and informed allocation. Under a fixed selection law, independent parent refresh weakly improves hit probability, expected distinct discovery and unnormalized estimator variance over single-parent commitment. Standard index weighting provides an optional target-preserving estimator.
- **Controlled discovery evaluation.** Molecular and image experiments isolate the incremental yield of binary feedback within a shared branching structure. Finite-target tests and additional-domain results delineate the limits: improved discovery does not imply better conditional-estimation accuracy, and allocation benefits are not universal.

2 BACKGROUND

Base generation law. For fixed context c, let X_t denote the complete decoder state, including memory needed to sample a base continuation. Time follows generation, from initial noise at $t = 0$ to the completed artifact at T. The operational base law is (Chopin et al., 2020; Moral, 2004)

$$P_c(dx_{0:T}) = \mu_c(dx_0) \prod_{t=1}^{T} Q_{t,c}(dx_t \mid x_{t-1}). \tag{1}$$

Here μ_c is the initial law, $Q_{t,c}$ is the implemented transition, and p_{base} is the terminal marginal. The decoder schedule, sampling temperature, scheduler state and all transition-relevant memory form part of this law.

Terminal conditioning. A fixed verifier $O(x, c) \in \{0, 1\}$ accepts a completed artifact, for example a valid molecule above a declared QED threshold (Bickerton et al., 2012). It need not be differentiable or meaningful on unfinished states. Conditioning gives (Moral, 2004; Wu et al., 2023)

$$Z_c = \mathbb{E}_{P_c}[O(X_T, c)], \qquad \pi_c(dx) = \frac{p_{\text{base}}(dx \mid c) O(x, c)}{Z_c}. \tag{2}$$

We assume $Z_c > 0$. This target excludes failures while preserving relative probabilities among accepted outputs.

Conditional feasibility. The success probability of a fresh base continuation is

$$h_t(x, c) = \mathbb{P}_{P_c}(O(X_T, c) = 1 \mid X_t = x)\,. \tag{3}$$

We do not assume access to exact h_t, its gradient or a learned value model. CFS obtains noisy evidence about this quantity from pilot completions generated within its sampling budget.

Related work. Splitting methods allocate simulation effort toward promising paths (Cérou & Guyader, 2007); TDS and FK steering use intermediate information within corrected particle sampling (Wu et al., 2023; Singhal et al., 2025). SVDD and nested SMC use candidate lookahead (Li et al., 2024; Chanchu et al., 2026), while DTS and language-model SMC reuse rollout information for generation (Jain et al., 2026; Lew et al., 2023). Recycling is also established in Monte Carlo methods (Dau & Chopin, 2022; Nguyen et al., 2014). CFS studies a restricted construction: one binary terminal observation per independent parent, a fixed moment-constrained allocation, and reuse of both pilot endpoints and saved states. It requires no learned value function and leaves the base continuation kernel unchanged. Its empirical test is the additional value of this allocation over equally resourced branching, not whether reuse itself is novel. Appendix F gives the extended comparison.

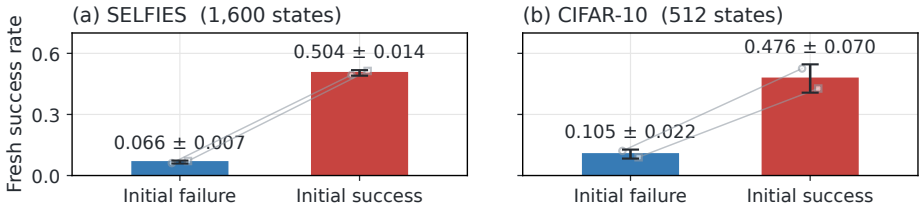

Figure 1: A successful first sample identifies saved states with higher fresh success rates. States are grouped by whether their first completed sample fails (blue) or passes (red) verification. Each state contributes eight fresh continuations, excluding the first sample, so both groups receive equal evaluation effort per state. Bars and error bars show means and sample SD across two seeds; connected markers show individual seed means.

3 Motivation: Reusing Intermediate Diffusion States

Resuming a saved diffusion state avoids recomputing its prefix. To allocate these cheaper continuations, we ask whether a successful pilot predicts success under fresh randomness. Let $Z \sim P_c(X_\tau \in \cdot)$ be a saved diffusion state and $H = h_\tau(Z, c)$ its probability of eventual success. Completing the first trajectory gives $O \mid H \sim \text{Bernoulli}(H)$. A new trajectory from the same state, using independent randomness, succeeds with the same probability H.

Proposition 1 (One-bit feasibility resolution)**.** *Let $\mu = \mathbb{E}[H] \in (0, 1)$ and $\sigma_h^2 = \text{Var}(H)$. Then*

$$\mathbb{E}[H \mid O = 1] = \mu + \frac{\sigma_h^2}{\mu}, \qquad \mathbb{E}[H \mid O = 0] = \mu - \frac{\sigma_h^2}{1 - \mu}, \tag{4}$$

and hence

$$\mathbb{E}[H \mid O = 1] - \mathbb{E}[H \mid O = 0] = \frac{\sigma_h^2}{\mu(1 - \mu)}. \tag{5}$$

The identity follows from $\mathbb{E}[HO] = \mathbb{E}[H^2]$ (Appendix A). A successful pilot identifies higher-feasibility states on average whenever $\text{Var}(H) > 0$, without identifying the best state in every run. If feasibility is constant, the pilot carries no such information.

This distinction is visible in real diffusion models. Figure 1 compares eight new samples from each saved state, grouped by whether its first completed sample failed or passed verification. The new-sample success rate is $0.066_{\pm 0.007}$ versus $0.504_{\pm 0.014}$ on SELFIES, and $0.105_{\pm 0.022}$ versus $0.476_{\pm 0.070}$ on CIFAR-10. Every state contributes the same number of new samples, so the difference does not arise from generating more samples from states with an initial success. This diagnostic reuses recorded generations; its protocol and scope are given in Appendix D.1.

Independent parent reselection spreads fresh continuations across the available states. Under a fixed selection law, it preserves expected valid count while weakly improving fresh-hit probability, expected distinct discovery and unnormalized estimator variance relative to selecting one parent for all continuations (Proposition 3). Retaining the pilots makes the information-gathering computation productive as well. Section 4 specifies this construction.

4 Conditional Feasibility Steering

CFS allocates fresh base continuations using terminal pilot outcomes. Algorithm 1 summarizes its fixed-stage implementation.

4.1 Generating and Reusing Intermediate States

For a fixed context, run K independent base trajectories to a chosen checkpoint τ and save their states $Z_1, \ldots, Z_K$. Complete each trajectory once, obtaining samples $Y_i^{(0)}$ and verification outcomes $O_i = O(Y_i^{(0)}, c)$. We refer to the saved states as parents and these first samples as pilots. Each

Algorithm 1 Conditional Feasibility Steering

Require: Base generator P_c, verifier $O(\cdot, c)$, identity map κ, checkpoint τ
Require: Fixed integers $K \geq 1$, $L \geq 0$, and finite moment budget $B \geq 1$
Ensure: Verified archive $\mathcal{A}$; weighted draws $\mathcal{W}$ for optional estimation

$\mathcal{A} \leftarrow \varnothing, \mathcal{W} \leftarrow [\,]$
for $i = 1, \ldots, K$ **do**
  Sample an independent base path; save $Z_i = X_\tau^{(i)}$ and pilot $Y_i^{(0)} = X_T^{(i)}$
  $O_i \leftarrow O(Y_i^{(0)}, c)$
  $\mathcal{A} \leftarrow \mathcal{A} \cup \{Y_i^{(0)} : O_i = 1\}$
  Append $(Y_i^{(0)}, O_i/(K+L))$ to $\mathcal{W}$
end for
Compute $s^\star$ using Equation 12 ▷ Uniform if all O_i agree
for $\ell = 1, \ldots, L$ **do**
  Draw $I_\ell \sim s^\star$ independently
  Restore Z_{I_ℓ}; sample a fresh base continuation $Y_\ell^{(1)}$
  $O_\ell^{(1)} \leftarrow O(Y_\ell^{(1)}, c)$
  $\mathcal{A} \leftarrow \mathcal{A} \cup \{Y_\ell^{(1)} : O_\ell^{(1)} = 1\}$
  Append $(Y_\ell^{(1)}, O_\ell^{(1)}/[(K+L)Ks_{I_\ell}^\star])$ to $\mathcal{W}$
end for
return $\mathrm{Deduplicate}_\kappa(\mathcal{A}), \mathcal{W}$ ▷ Keep repeated draws in $\mathcal{W}$

parent includes the full information needed to resume generation, so later samples can reuse its earlier denoising computation while drawing new randomness for the remaining steps.

The pilots serve two purposes: they are completed output candidates, and their success or failure provides evidence about further generation from the same state (Proposition 1). CFS keeps both the samples and all K saved states. The checkpoint, K and the number L of additional samples are fixed before seeing outcomes. Appendix A.1 specifies the base sampling laws and independence conditions; Appendix B describes state restoration and batching.

4.2 Steering subsequent denoising

A successful pilot makes its saved state a promising starting point for another sample. CFS therefore selects those states more often, while retaining a nonzero chance of selecting every state. Let s_i be the probability of choosing parent i. We choose

$$s^\star \in \arg\max_{s \in \Delta_K,\ s_i > 0} \sum_i s_i O_i \quad \text{subject to} \quad C(s) := \frac{1}{K^2} \sum_i \frac{1}{s_i} \leq B. \tag{6}$$

The objective increases the probability of returning to states with successful pilots. The constraint limits how uneven those probabilities can become: the correction $G_I = (Ks_I)^{-1}$ for a sampled index $I \sim s$ has second moment $C(s)$ (Appendix A.3). We fix $B = 2$ as a correction-moment budget, not an empirically optimal constant; $B = 1$ gives uniform selection. This directly controls the parent-selection correction (Agapiou et al., 2017), without fitting a success predictor. Binary outcomes admit a closed-form solution, uniform within the passing and failing groups; if all outcomes agree, CFS selects uniformly. The formula and derivation are in Appendix A.3, and the expected fresh-success gain is established in Proposition 2.

After fixing s, CFS independently draws a parent index for each of the L additional samples, restores that state, and completes denoising with fresh randomness. This keeps the same preference for promising states without committing the entire budget to one trajectory. Under a fixed selection law, independent refresh preserves expected valid count while improving or matching fresh-hit probability, expected distinct discovery and corrected estimator variance relative to single-parent commitment (Appendix A.4).

4.3 DISCOVERY AND OPTIONAL WEIGHTED ESTIMATION

All verified samples, including the initial pilots, contribute to discovery. For conditional estimation, however, fresh samples need a correction because their starting states were selected nonuniformly. Each pilot already has the base terminal marginal and needs no selection correction; a fresh sample $Y_\ell^{(1)}$ from parent I_ℓ receives the factor $(Ks_{I_\ell})^{-1}$. For a bounded test function f, the combined estimator is

$$\widehat{\gamma}(f) = \frac{1}{K+L}\left[\sum_{i=1}^{K} O(Y_i^{(0)}, c) f(Y_i^{(0)}) + \sum_{\ell=1}^{L} \frac{O(Y_\ell^{(1)}, c) f(Y_\ell^{(1)})}{Ks_{I_\ell}}\right]. \tag{7}$$

This factor recovers uniform selection over the recorded states in expectation, even though s depends on all pilot outcomes. This is a property of the weighted estimator; the unweighted archive is generally tilted and is not claimed to follow the base conditional distribution. With base continuations, complete state restoration and positive recorded selection probabilities,

$$\mathbb{E}[\widehat{\gamma}(f)] = \mathbb{E}_{P_c}[O(X_T, c) f(X_T)]. \tag{8}$$

Averaging independent histories and normalizing by their estimated accepted mass therefore consistently estimates $\pi_c(f)$. Appendix A.5 gives the formal statement, assumptions and proof. Finite normalized estimates can still be biased or undefined at zero accepted mass; the identity does not imply lower estimation error than rejection.

For search, valid samples are deduplicated by their canonical identities. For estimation, every draw retains its contribution, including repeated artifacts. One run completes $K + L$ samples while paying for each saved prefix once. Computational cost is given in Equation 18, with implementation details and rejection/uniform special cases in Appendix B. Discovery and conditional fidelity are evaluated separately in Section 5.

5 EXPERIMENTS

We evaluate three questions: **(1) Generation performance.** How does CFS compare with rejection, Late_branching and domain-specific baselines across generation tasks (Section 5.2)? **(2) Feasibility information.** At fixed parents, pilots and fresh budget, does verifier-informed allocation improve yield over Uniform_iid and alternative allocation rules (Section 5.3)? **(3) Refresh and fidelity.** How does independent parent refresh affect conditional-estimation error relative to single-parent commitment, uniform recycling and rejection (Section 5.4)? Section 6 examines when terminal evidence is informative and separates the matched allocation gain from differences in computational budget.

5.1 EXPERIMENTAL SETUP

Baselines. *Rejection* samples independent complete base trajectories and retains verifier-passing outputs (Chopin et al., 2020). *Late_branching* denotes shared-prefix sampling at a late checkpoint: each independent root produces multiple fresh base continuations without verifier-guided allocation. Early-checkpoint branching is a separate structural control. *Uniform_iid* is the matched-budget no-steering control: it uses the same parents, pilots and fresh budget as CFS but samples each fresh parent uniformly ($s_i = 1/K$). Domain-specific references are discrete FK correctors and nested SMC for molecules (Hasan et al., 2026; Chanchu et al., 2026); implementation details are in Appendix C.2. Main tables include sampling comparators, Uniform_iid and CFS; balanced counts and single-parent commitment are examined in the ablations.

Metrics. Table 1 reports distinct verified molecules (↑), Table 2 reports classifier-valid image draws (↑), and Table 3 reports distinct verified programs or valid routing completions (↑), all per group. Table 4 uses the corresponding molecular and image yields. Distinct molecules are deduplicated by canonical SMILES; image and routing counts retain multiplicity. CFS yield includes pilots and fresh completions. SD is across seed means and is given in the text when absent from the table. These are counts, not return probabilities; fidelity metrics are defined in Section 5.4.

Method	QM9-SELFIES distinct valid per K=4 pool QED > 0.55	QED > 0.60
Rejection (Chopin et al., 2020)	$0.450_{\pm 0.014}$	$0.075_{\pm 0.000}$
Late_branching	$2.745_{\pm 0.120}$	$0.487_{\pm 0.092}$
Discrete FK corrector (Hasan et al., 2026)	$1.977_{\pm 0.278}$	$0.337_{\pm 0.074}$
Nested SMC (Chanchu et al., 2026)	$0.467_{\pm 0.029}$	$0.087_{\pm 0.029}$
CFS	$\mathbf{3.195}_{\pm 0.127}$	$\mathbf{0.535}_{\pm 0.084}$

Table 1: QM9-SELFIES distinct verified yield/group (↑); rejection is pilot-only.

Method	CIFAR-10 Valid yield/group (V, ↑)
Rejection (Chopin et al., 2020)	$0.667_{\pm 0.055}$
Late_branching	$6.219_{\pm 0.980}$
CFS	$\mathbf{9.151}_{\pm 0.638}$

Table 2: CIFAR-10 valid yield/group (↑), mean ± seed SD; rejection is pilot-only

Common settings. CFS uses $K = 4$ and $B = 2$. Seed counts are study-specific: the QED > 0.55 allocation table uses two seeds, Table 2 and the analysis figures three. The separate CIFAR component-ablation cohort in Table 4 uses one seed. The posterior comparison uses one seed per domain. Tables and figures from different studies are not interchangeable estimates; their exact sources and schedules are listed in Appendices C.4 and D. SD, where shown, is across seed means, not individual samples.

5.2 Generation performance across domains

The domain panels retain the available baseline measurements; the controlled allocation comparisons appear in Table 4. The panels report verified yield, separating distinct artifacts from counts that retain repeated outputs.

Molecular generation. QM9-SELFIES (Ramakrishnan et al., 2014; Krenn et al., 2020) provides the primary molecular setting. Success requires validity and the declared QED threshold. Table 1 reports distinct verified yield at QED thresholds 0.55 and 0.60. Cohort sizes, comparison budgets and allocation protocols are specified in Appendix C.4.

The rejection row in Table 1 contains only the four pilots (204 model sample-steps at $T = 51$), whereas the QED > 0.55 allocation experiment uses 816 steps. Its Late_branching row uses the Uniform_iid records, providing the no-steering comparison rather than a separate additional control. Only this allocation contrast matches the CFS budget. Nested SMC estimates intermediate weights through extra terminal rollouts rather than a supplied intermediate score; its computation therefore includes internal lookahead that does not directly increase the number of final output particles (Appendix C.2).

Image generation. CIFAR and MNIST evaluate image generation using frozen target-class classifiers, with allocation comparisons in Section 5.3. MNIST targets digit 7: valid yield is $2.391_{\pm 0.798}$ under balanced uniform, $2.401_{\pm 0.618}$ under Uniform_iid, and $5.453_{\pm 1.484}$ under CFS. Table 2 reports the three-seed CIFAR-10 verified-yield measurements also shown in Figure 3: pilot-only rejection, uniform late branching and CFS. The pilot-only reference uses less work; late branching and CFS share the fresh-generation budget. Protocols are given in Appendices C.4 and D. Classifier-valid yield does not establish semantic diversity or conditional fidelity.

Executable generation and routing. HumanEval/MBPP test executable correctness; PCB routing requires complete connectivity and zero DRC violations. Steering and held-out tests are distinguished whenever they differ. Table 3 reports distinct verified program yield and valid routing yield. PCB is an exploratory transfer study; its task coverage and model configuration are specified in Appendix C.4. The MBPP comparison uses one seed.

Task / verified yield	Rejection†	Late_branching	CFS
Code generation			
HumanEval (Chen et al., 2021)	2.92	4.242	**4.283**
MBPP (Austin et al., 2021)§	3.50	**5.775**	5.675
Hardware design			
PCB / LLaDA⋆ (Nie et al., 2026)	$0.019_{\pm 0.032}$	$0.363_{\pm 0.081}$	$\mathbf{0.644}_{\pm 0.076}$

Table 3: Code and hardware transfer. HumanEval/MBPP report mean distinct valid programs per group at $K=4$; PCB reports valid completions per group. ⋆PCB metric per Appendix C.4. §MBPP uses one seed. †Code rejection entries are budget-normalized estimates from separate rejection runs, not direct matched-budget measurements (Appendix C.4).

Additional constraint domains. Sudoku and SAT are reported together in Appendix E, including no detected gain on Sudoku and zero hits on SAT. The SAT comparison also has a budget mismatch; neither result supports a broad cross-domain advantage.

5.3 Ablations: feasibility steering and parent refresh

The controlled CFS–uniform comparisons share parents, pilots, fresh budget and model work. Uniform_iid removes steering, balanced uniform fixes per-parent counts, and binary single commits all fresh draws to one selected parent. Table 4 also includes separately collected single-parent measurements; those rows are not a fully paired comparison with every other row. All variants retain pilots (Appendix C.4).

Variant	steering	refresh	SELFIES distinct	CIFAR valid	MNIST valid
Uniform bal.	×	✓	2.808	7.141	2.391
Uniform_iid	×	✓	2.745	6.703	2.401
Binary single	✓	×	2.630	9.979	5.901
CFS	✓	✓	**3.195**	**9.781**	**5.453**

Table 4: Component ablations: verified yield/group (↑).

Component effects. CFS improves over Uniform_iid by $0.450_{\pm 0.007}$ distinct molecules/group on SELFIES and $3.052_{\pm 0.875}$ valid images/group on MNIST. The single-seed CIFAR difference is 3.078. Binary single has higher observed image yield than CFS, so these results do not support a yield advantage from refresh. Under the same selection law, theory predicts equal expected valid counts; the refresh benefit concerns dependence and distinct discovery.

Allocation rule	SELFIES distinct (↑)	CIFAR valid (↑)
Uniform balanced	3.010	5.906
Posterior greedy	3.210	6.953
CFS	**3.485**	**8.406**

Table 5: Allocation rules: verified yield/group (↑).

Allocation rule comparison. Table 5 reports yield including pilots and compares balanced uniform, Beta–Bernoulli posterior greedy and CFS on a separate cohort, with the same parents, pilots and model work within each domain. CFS yields more verified outputs than the tested posterior rule, whose objective is fresh-hit probability rather than yield. Appendix C.4 gives the protocol and scope; Section 6.2 discusses the interpretation.

5.4 Parent refresh and conditional fidelity

Two enumerable targets compare rejection, uniform recycling, binary single and CFS at matched work. Refresh reduces unnormalized error relative to single-parent commitment, but rejection has lower unnormalized error, normalized error, TV and extinction on both targets. These are correctness and dependence diagnostics, not evidence that CFS is preferable for estimation. Appendix D.4 reports the measurements.

6 Analysis: When and Why Does CFS Help?

We connect terminal information to allocation gain, compare model work with distinct verified yield, and distinguish valid yield from distinct discovery.

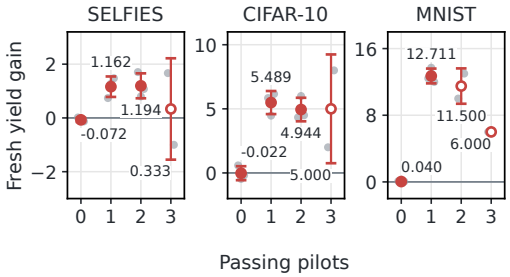

Figure 2: Fresh allocation gain by pilot outcomes (mean ± seed SD); open markers denote fewer than five groups.

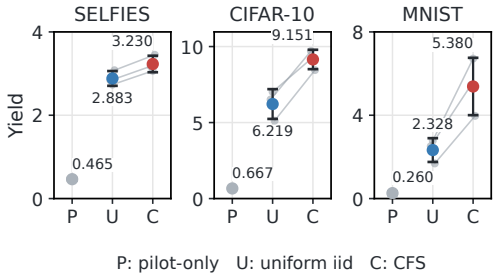

Figure 3: Yield under pilot-only, Uniform_iid and CFS (mean ± seed SD). Only the latter two use matched work.

6.1 When does terminal evidence help?

A successful pilot predicts higher fresh success when feasibility varies across saved states (Proposition 1; Figure 1). Figure 2 tests the allocation benefit by grouping runs by their number of passing pilots and measuring CFS–Uniform_iid gain on fresh continuations. Pilot counts are observable selection evidence, not estimates of latent feasibility heterogeneity. Sampling and aggregation details are in Appendix D.

With no passing pilot, CFS selects uniformly and observed gains are near zero. With one passing pilot, fresh gains are $1.162_{\pm 0.381}$, $5.489_{\pm 0.900}$ and $12.711_{\pm 0.887}$ for SELFIES, CIFAR and MNIST. Mixed-label groups show positive mean gains, but sparse upper bins do not establish a trend with pilot composition or latent feasibility variation. The all-pass case is unobserved.

6.2 Where does the yield gain come from?

Figure 3 compares CFS and Uniform_iid using the same saved states, pilots, fresh-sample budget and model work. Yield includes pilots and fresh outputs. The paired gains are $0.347_{\pm 0.021}$ for SELFIES, $2.932_{\pm 0.568}$ for CIFAR and $3.052_{\pm 0.834}$ for MNIST, isolating the benefit of verifier-informed allocation. These three-seed analysis runs differ from the allocation-table runs, explaining the different absolute means.

CFS also outperforms the tested Beta–Bernoulli posterior rule (Table 5). This supports the simple binary rule in the tested setting, but does not explain the difference: posterior greedy optimizes fresh-hit probability rather than yield, and the policies also differ in count dependence (Appendix C.4).

6.3 Compute–yield comparison

Figure 4 compares available SELFIES operating points at QED > 0.55, counting all model forwards, including model-based probes. The allocation variants use 816 forwards per group; NSMC uses 812, whereas DFKC uses 2,416 (816 main-trajectory and 1,600 probe forwards). The initial-sample reference uses only four complete trajectories, or 204 forwards; it is not full-budget rejection. Late_branching and Uniform_iid share records and are shown once.

The plot uses three seeds per method, with 200 groups per seed for the allocation variants and initial-sample reference, and 100 for DFKC/NSMC. Yield is the number of distinct verified molecules; error bars are sample SD across seed means. These are individual operating points on separately sampled cohorts, not a common-cohort compute frontier. The CFS–Uniform_iid contrast remains paired at matched work; multi-budget and full-budget rejection comparisons are still pending.

Interpreting checkpoint effects. A later checkpoint reduces the cost of each fresh suffix, but repeated continuations can produce the same artifact. The checkpoint comparison therefore asks whether cheaper generation translates into more distinct verified outputs. Appendix A.6 separates the expected valid-yield gains from reuse and binary information; predicting distinct yield additionally requires output-collision probabilities.

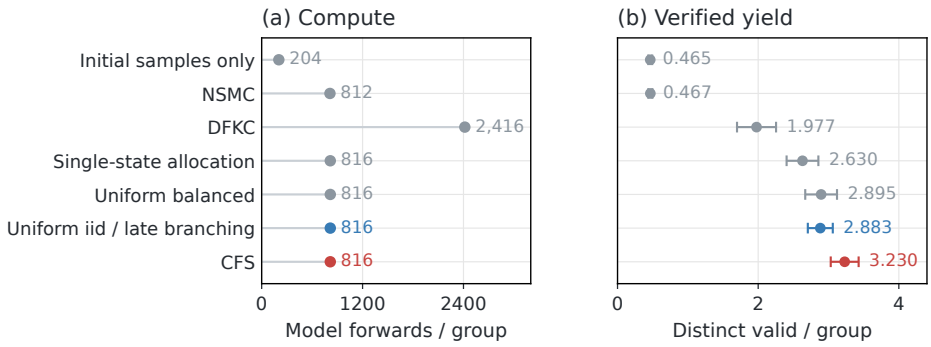

Figure 4: Compute and distinct verified yield at available SELFIES operating points. Model work includes probe generation; yield is mean ± seed SD.

7 Conclusion and Limitations

CFS uses binary terminal feedback to allocate further generation from saved diffusion states. In the controlled SELFIES, CIFAR and MNIST allocation experiments, CFS increases verified yield over uniform allocation within the same branching structure. The method combines a closed-form moment-constrained selector with independent parent refresh and retention of every completed sample, without a learned value model or verifier gradients.

This finding does not extend to every task or metric: Sudoku shows no detected allocation gain. The benefit depends on variation in conditional feasibility and can be offset by duplicate outputs. Independent refresh changes output dependence, not expected valid count under a fixed selection law. Optional importance weighting preserves unnormalized expectations but does not make the discovery archive conditionally distributed or improve estimation over rejection in our finite tests. The controlled evidence concerns allocation at the tested budgets; it does not establish dominance across budgets or semantic diversity, and the MBPP comparison uses one seed. We study one fixed checkpoint and a fixed allocation, without adaptive or multi-round claims. Finally, verifier success certifies only the tested property, not general semantic quality or held-out correctness.

AI USE STATEMENT

AI assistance was used for editorial revisions, reference checking, and figure-generation code.

ETHICS STATEMENT

Passing a terminal verifier certifies only its declared tests. Generated code and physical designs require independent validation before deployment; experimental test suites do not establish general safety or correctness.

REPRODUCIBILITY STATEMENT

Appendices specify the generator and verifier configurations, sampling schedules, seed counts, aggregation rules and proof assumptions. Completed measurements and pending comparisons are identified separately. All pilot, lookahead and fresh-generation costs must be included when reproducing the compute–yield evaluation.

REFERENCES

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

Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, et al. Program synthesis with large language models. *arXiv preprint arXiv:2108.07732*, 2021.

G Richard Bickerton, Gaia V Paolini, Jérémy Besnard, Sorel Muresan, and Andrew L Hopkins. Quantifying the chemical beauty of drugs. *Nature chemistry*, 4(2):90–98, 2012.

Michael Cardei, Jacob K Christopher, Bhavya Kailkhura, Tom Hartvigsen, and Ferdinando Fioretto. Constrained discrete diffusion. *Advances in Neural Information Processing Systems*, 38:12383–12416, 2026.

Frédéric Cérou and Arnaud Guyader. Adaptive multilevel splitting for rare event analysis. *Stochastic Analysis and Applications*, 25(2):417–443, 2007.

Lohithsai Yadala Chanchu, Hany Abdulsamad, and Christian A Naesseth. Discrete diffusion inference-time control with nested sequential monte carlo. *arXiv preprint arXiv:2608.20123*, 2026.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde De Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021.

Nicolas Chopin, Omiros Papaspiliopoulos, et al. *An introduction to sequential Monte Carlo*, volume 4. Springer, 2020.

Jacob K Christopher, Stephen Baek, and Ferdinando Fioretto. Constrained synthesis with projected diffusion models. *Advances in Neural Information Processing Systems*, 37:89307–89333, 2024.

Hai-Dang Dau and Nicolas Chopin. Waste-free sequential monte carlo. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 84(1):114–148, 2022.

Mohsin Hasan, Viktor Ohanesian, Artem Gazizov, Yoshua Bengio, Alán Aspuru-Guzik, Roberto Bondesan, Marta Skreta, and Kirill Neklyudov. Discrete feynman-kac correctors. *arXiv preprint arXiv:2601.10403*, 2026.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

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.

GR Kanagachidambaresan. Introduction to kicad design for breakout and circuit designs. In *Role of Single Board Computers (SBCs) in rapid IoT Prototyping*, pp. 165–175. Springer, 2021.

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.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

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

Alexander K Lew, Tan Zhi-Xuan, Gabriel Grand, and Vikash K Mansinghka. Sequential monte carlo steering of large language models using probabilistic programs. *arXiv preprint arXiv:2306.03081*, 2023.

Xiner Li, Yulai Zhao, Chenyu Wang, Gabriele Scalia, Gokcen Eraslan, Surag Nair, Tommaso Biancalani, Shuiwang Ji, Aviv Regev, Sergey Levine, et al. Derivative-free guidance in continuous and discrete diffusion models with soft value-based decoding. *arXiv preprint arXiv:2408.08252*, 2024.

Fredrik Lindsten, Michael I Jordan, and Thomas B Schön. Particle gibbs with ancestor sampling. *The Journal of Machine Learning Research*, 15(1):2145–2184, 2014.

Jiawei Liu, Chunqiu Steven Xia, Yuyao Wang, and Lingming Zhang. Is your code generated by chatgpt really correct? rigorous evaluation of large language models for code generation. *Advances in neural information processing systems*, 36:21558–21572, 2023.

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

Thi Le Thu Nguyen, Francois Septier, Gareth W Peters, and Yves Delignon. Improving smc sampler estimate by recycling all past simulated particles. In *2014 IEEE Workshop on Statistical Signal Processing (SSP)*, pp. 117–120. IEEE, 2014.

Shen Nie, Fengqi Zhu, Zebin You, Xiaolu Zhang, Jingyang Ou, Jun Hu, Jun Zhou, Yankai Lin, Ji-Rong Wen, and Chongxuan Li. Large language diffusion models. *Advances in Neural Information Processing Systems*, 38:50608–50646, 2026.

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

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.

Kartik Sharma, Srijan Kumar, and Rakshit Trivedi. Diffuse, sample, project: plug-and-play controllable graph generation. In *Forty-first International Conference on Machine Learning*, 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. *arXiv preprint arXiv:2501.06848*, 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.

The appendix follows the main paper's claims: Appendix A gives the theoretical results, Appendix B describes their implementation, and Appendix C specifies the experiments. Appendices D and E document the diagnostic analyses and additional domain results. Appendix F discusses related work.

A Theoretical results and proofs

This section establishes what binary feedback reveals about future success, derives the selection rule, and proves the effects of independent state reselection and importance correction. All results condition on a fixed task context c. Counts and checkpoint times are fixed before observing outcomes, and each saved state contains the information needed to resume the base process.

A.1 Sampling law and assumptions

Fix $K \geq 1$, $L \geq 0$ and $0 \leq \tau < T$ before observing any generation outcomes. At a fixed context c, the saved states and first completed samples follow

$$Z_i \overset{\text{iid}}{\sim} P_c(X_\tau \in \cdot), \qquad Y_i^{(0)} \sim P_c(X_T \in \cdot \mid Z_i), \qquad O_i = O(Y_i^{(0)}, c). \tag{9}$$

The initial trajectories are independent; pilot streams are independent conditional on the saved states. Let $\mathcal{U}$ contain all saved states and pilots, and let $s = s(\mathcal{U})$ be the positive selection law fixed from these observations. Conditional on $\mathcal{U}$, subsequent samples follow

$$I_\ell \overset{\text{iid}}{\sim} s, \qquad Y_\ell^{(1)} \sim P_c(X_T \in \cdot \mid Z_{I_\ell}), \qquad \ell = 1, \ldots, L. \tag{10}$$

Conditional on the complete state Z_{I_ℓ}, each fresh continuation uses the original base kernel and new independent randomness, including when an index is repeated. Write $h_i = h_\tau(Z_i, c)$. The indices are iid conditional on $\mathcal{U}$, not generally independent after averaging over the random pilot pool. A single-parent control instead draws $I \sim s$ once and uses Z_I for all L fresh samples. Its marginal selection law is the same, but its outputs share the selected index. Complete state restoration includes all scheduler or decoder memory; neither a cached endpoint nor replay of a completed suffix counts as a fresh draw.

A.2 Information in one binary observation

The first result concerns a population of saved states, before any particular state is selected. Write $H = h_\tau(Z, c)$, $\mu = \mathbb{E}[H]$ and $\sigma_h^2 = \mathrm{Var}(H)$. A pilot outcome satisfies $O \mid H \sim \mathrm{Bernoulli}(H)$.

Proposition 1. Since $\mathbb{E}[O \mid H] = H$, $\mathbb{P}(O = 1) = \mu$ and $\mathbb{E}[HO] = \mathbb{E}[H^2] = \mu^2 + \sigma_h^2$. Dividing by μ gives the first conditional mean. Similarly, $\mathbb{E}[H(1 - O)] = \mu - \mu^2 - \sigma_h^2$ gives the second. Subtraction yields the stated difference. The conditional means are only defined for $0 < \mu < 1$; at the endpoints all pilots have the same label almost surely. □

A.3 Selection under a second-moment constraint

We next derive how the observed labels determine the selection probabilities. The constraint controls the importance ratio for selecting a state; it does not constrain all sources of estimation error.

Correction moment. For $I \sim s$, define $G_I = (K s_I)^{-1}$ and $u_i = 1/K$. Then

$$\mathbb{E}_s[G_I] = 1, \qquad \mathbb{E}_s[G_I^2] = C(s), \qquad \mathrm{Var}_s(G_I) = C(s) - 1 = D_{\chi^2}(u \| s). \tag{11}$$

Thus $B - 1$ bounds the variance of the parent-index correction, not the variance of a normalized rare-event estimate. Positive probabilities preserve support for every saved state.

Closed-form allocation. Let $S = \sum_i O_i$ and $a = S/K$. For $0 < S < K$, the closed-form solution is

$$\begin{aligned} r^\star(a, B) &= \frac{2a + B - 1 + \sqrt{(B-1)^2 + 4(B-1)a(1-a)}}{2B}, \\ s_i^\star &= \begin{cases} r^\star / S, & O_i = 1, \\ (1 - r^\star)/(K - S), & O_i = 0. \end{cases} \end{aligned} \tag{12}$$

For a fixed passing-group mass r, convexity of $1/s_i$ implies that equal probabilities within each group minimize $C(s)$. Thus an optimum can be sought among group-uniform laws, where

$$C(s) = \frac{a^2}{r} + \frac{(1-a)^2}{1-r} = 1 + \frac{(r-a)^2}{r(1-r)}.$$

The objective is r. For $0 < a < 1$ and finite $B > 1$, its maximum is the larger root of $(r-a)^2 = (B-1)r(1-r)$, giving Equation 12. It lies in $(a, 1)$ and retains positive support. Jensen's inequality makes uniform the only feasible law at $B = 1$. Constant labels use uniform as a tie-break; $B < 1$ is infeasible.

Proposition 2 (Expected fresh-feasibility gain)**.** *Under the independent-parent construction in Equation 9, condition on a mixed pilot-label vector o. Set $a = K^{-1}\sum_i o_i$ and $r = r^\star(a, B)$. Averaging over parents conditional on these labels,*

$$\mathbb{E}\left[\sum_i (s_i^\star - 1/K)h_i \,\middle|\, O_{1:K} = o\right] = (r-a)\frac{\sigma_h^2}{\mu(1-\mu)} \geq 0. \tag{13}$$

For $B > 1$ and nonzero feasibility variance, the gain is positive on mixed-label pools. This is an expectation over latent parents conditional on the observed labels, not a guarantee for every fixed realized pool. It concerns valid yield, not distinct discovery or conditional-estimation error.

Proposition 2. The pairs (H_i, O_i) are iid. Conditional on the full label vector, each H_i therefore has the conditional mean determined by its own label. Write these means as μ_1 and μ_0. The expected fresh success under the guided law is $r\mu_1 + (1-r)\mu_0$, versus $a\mu_1 + (1-a)\mu_0$ under uniform allocation. Subtract and apply Proposition 1. □

Exchangeability alone is insufficient: other labels can change the posterior mean of H_i when parents are dependent. For parents iid conditional on a shared prefix $X = x$, the same proof uses $\mu_x = \mathbb{E}[H \mid x]$ and $\sigma_{h,x}^2 = \mathrm{Var}(H \mid x)$ and conditions on x throughout. Between-prefix variation is not evidence for within-pool allocation. These are population identities; a particular passing-pilot parent can still have lower h_i than a failing-pilot parent.

A.4 Independent state reselection

Conditional on $\mathcal{U}$, the fresh no-hit probabilities are $\sum_i s_i(1-h_i)^L$ for single-parent sampling and $(\sum_i s_i(1-h_i))^L$ for independent refresh. Distinct discoveries use a fixed canonical artifact identity. For bounded f, the following proposition compares corrected fresh means under the same pool and law s.

Proposition 3 (Effect of independent refresh)**.** *For a fixed pool, law s and $L \geq 1$, independent parent refresh has no smaller fresh-hit probability or expected distinct valid discovery than the single-parent control. Their expected valid counts are equal. For the corrected fresh mean*

$$F(f) = \frac{1}{L}\sum_{\ell=1}^{L} \frac{O(Y_\ell^{(1)}, c) f(Y_\ell^{(1)})}{K s_{I_\ell}}, \qquad m_i(f) = \mathbb{E}[O(Y,c)f(Y) \mid Z_i],$$

the conditional variance difference is

$$\mathrm{Var}(F_{\mathrm{single}} \mid \mathcal{U}) - \mathrm{Var}(F_{\mathrm{multi}} \mid \mathcal{U}) = \left(1 - \frac{1}{L}\right)\mathrm{Var}_{I\sim s}\left[\frac{m_I(f)}{K s_I}\right] \geq 0. \tag{14}$$

Proposition 3. Fix $\mathcal{U}$ and s. For $L \geq 1$, convexity of x^L gives

$$\left(\sum_i s_i(1-h_i)\right)^L \leq \sum_i s_i(1-h_i)^L.$$

Both schemes have expected valid count $L\sum_i s_i h_i$. Let κ map raw outputs to a fixed finite or countable canonical identity space $\mathcal{Y}$. Define $p_i(y) = \mathbb{P}(O(Y,c) = 1, \kappa(Y) = y \mid Z_i)$, the

probability of a verified output with identity y from parent i. Its occurrence probabilities are $1-(1-\sum_i s_i p_i(y))^L$ under independent refresh and $\sum_i s_i[1-(1-p_i(y))^L]$ under a single parent. The same inequality orders these probabilities. Sum over $y \in \mathcal{Y}$ using nonnegativity to obtain distinct-discovery dominance. For total pilot-plus-fresh distinct discovery, restrict the sum to artifacts absent from the fixed pilot archive; the pilot contribution is common.

For the variance identity, define $W = O(Y,c)f(Y)/(Ks_I)$, $q_i = \mathbb{E}[W \mid I=i,\mathcal{U}] = m_i(f)/(Ks_i)$ and $v_i = \mathrm{Var}(W \mid I=i,\mathcal{U})$. Under one parent, the variance of the fresh mean is $\sum_i s_i v_i/L + \mathrm{Var}_s(q_I)$. Under independent refresh it is $[\sum_i s_i v_i + \mathrm{Var}_s(q_I)]/L$. Their difference is Equation 14. □

The statements are equalities at $L=1$. They compare the same allocation law and do not order binary and uniform allocation. For the recycled estimator conditional on the pilots, the variance difference is multiplied by $(L/(K+L))^2$. Conditional means agree, so the corresponding unconditional unnormalized variance ordering also follows. Normalizing by the random accepted mass does not preserve this ordering in general.

A.5 Target preservation and normalized estimation

The estimator in Equation 7 combines all initial and additional samples. The correction recovers the unnormalized target in expectation; normalization requires aggregation over independent runs.

Proposition 4 (All-completion target identity)**.** *Assume fixed K, L, τ, base-distributed parents, conditionally independent base pilots and fresh continuations, complete state restoration, a fixed deterministic verifier, and recorded positive selection probabilities. Then Equation 8 holds for every bounded f. Ratios $\sum_{r=1}^n \widehat{\gamma}_r(f)/\sum_{r=1}^n \widehat{\gamma}_r(1)$ from iid histories at the same context and fixed sampling configuration consistently estimate $\pi_c(f)$ when $Z_c > 0$.*

Proposition 4. Let $m_i(f) = \mathbb{E}[O(Y,c)f(Y) \mid Z_i]$ and condition on all parents and pilots. Fresh independence and the recorded selection law give

$$\mathbb{E}\left[\frac{O(Y_\ell^{(1)},c)f(Y_\ell^{(1)})}{Ks_{I_\ell}}\,\middle|\,\mathcal{U}\right] = \sum_i s_i \frac{m_i(f)}{Ks_i} = \frac{1}{K}\sum_i m_i(f).$$

Each parent has the base checkpoint marginal, hence $\mathbb{E}[m_i(f)] = \mathbb{E}_{P_c}[Of]$. Every pilot also has the base terminal marginal. Taking expectations in Equation 7 proves the identity, despite the dependence of s on the pilots. Only parent marginals are needed for this identity; their iid property is used for the stronger binary-information result above.

For bounded f, absolute integrability follows by the same identity applied to $|f|$. The iid laws of large numbers make numerator and denominator converge to $\mathbb{E}_{P_c}[Of]$ and $Z_c > 0$, proving ratio consistency. Within-history endpoints need not be independent. Deduplicating weighted draws, outcome-dependent stopping, or choosing K, L after seeing pilots is outside this fixed-count argument. □

The local moment bound ensures $s_i \geq 1/(K^2 B)$ and controls each fresh index ratio. Rare acceptance, parent-pool variability and normalization can still dominate estimation error.

Using the estimator. For n independent runs, the self-normalized estimator is

$$\widehat{\pi}_n(f) = \frac{\sum_{r=1}^n \widehat{\gamma}_r(f)}{\sum_{r=1}^n \widehat{\gamma}_r(1)}. \tag{15}$$

A zero denominator means that no positive verification weight was obtained (extinction). Resampling repeatedly from one weighted sample collection does not create independent runs. Search metrics use all verified pilot and additional samples; conditional estimates retain every draw with its recorded weight.

Archive return versus yield. Yield can distinguish allocation rules even when return cannot. If any pilot passes, archive return is already one; if all pilots fail, CFS selects uniformly. Consequently, CFS and Uniform_iid have the same archive-return probability under the same pilot law and fixed budget, although their expected valid and distinct counts can differ.

A.6 BUDGET DECOMPOSITION AND THE MISSING COLLISION TERM

Assume iid base parents, fixed K, L, τ, one conditionally independent pilot per parent, constant cost per model sample-step, $0 < \mu < 1$, and a fixed binary moment budget B. Define

$$A_K(\mu, B) = \mathbb{E}_{S\sim \mathrm{Bin}(K,\mu)}[r^\star(S/K, B) - S/K], \qquad g_\tau = A_K(\mu, B)\frac{\sigma_\tau^2}{\mu(1-\mu)},$$

with $\sigma_\tau^2 = \mathrm{Var}(h_\tau(X_\tau))$, $r^\star(0, B) = 0$ and $r^\star(1, B) = 1$. Then the valid-yield and model-work decomposition is

$$\mathbb{E}[V_{\mathrm{CFS}}] = (K + L)\mu + Lg_\tau, \qquad C = KT + L(T - \tau). \tag{16}$$

Proposition 5 (Valid-yield decomposition at fixed work)**.** *Equation 16 holds. At work C, independent rejection produces $n = \lfloor C/T \rfloor$ complete samples and has expected valid yield $n\mu$, so*

$$\mathbb{E}[V_{\mathrm{CFS}}] - \mathbb{E}[V_{\mathrm{rej}}] = (K + L - n)\mu + Lg_\tau.$$

Proof. The pilots contribute $K\mu$. Before observing pilots, the uniform fresh success probability is μ. Averaging Proposition 2 over $S \sim \mathrm{Bin}(K, \mu)$ adds g_τ per fresh draw; constant-label pools add zero. Summing gives the yield expression. Each of K pilot trajectories costs T and each fresh suffix costs $T - \tau$. Rejection uses n full trajectories and leaves $C - nT$ unused. □

This decomposition includes pilot cost and assumes equal model-step cost. Variable-length decoding, costly verification and wall-clock batching require their own cost model.

For a complete Markov state and fixed terminal event, the filtration $\mathcal{F}_t$ makes $h_t(X_t)$ a bounded martingale. For $u \geq t$,

$$\mathrm{Var}(h_u(X_u)) = \mathrm{Var}(h_t(X_t)) + \mathbb{E}[(h_u(X_u) - h_t(X_t))^2].$$

This need not hold for an incomplete state summary or a changing verifier.

Fix a canonical identity map κ into a finite or countable space $\mathcal{Y}$. For any realized multiset of outputs, let N_y count verified draws with identity y. Then $V - D = \sum_y (N_y - 1)_+ \leq \sum_y \binom{N_y}{2} = C_{\mathrm{pair}}$. Taking expectations gives

$$\mathbb{E}[V] - \mathbb{E}[C_{\mathrm{pair}}] \leq \mathbb{E}[D] \leq \mathbb{E}[V]. \tag{17}$$

If $p_y = \mathbb{P}_{P_c}(O(X_T, c) = 1, \kappa(X_T) = y)$, rejection's exact expected distinct count is

$$D_{\mathrm{rej}}(n) = \sum_{y\in\mathcal{Y}} [1 - (1 - p_y)^n].$$

A sufficient, not necessary, condition for CFS distinct superiority is

$$(K + L)\mu + Lg_\tau - \mathbb{E}[C_{\mathrm{pair,CFS}}] > D_{\mathrm{rej}}(n).$$

The collision term includes pilot–pilot, pilot–fresh and fresh–fresh pairs with their actual selection dependence. Two models can share the same feasibility process while assigning successful outcomes to one artifact or many artifacts; their valid yield is identical but their distinct yield differs. An optimal distinct-discovery checkpoint therefore cannot be inferred from feasibility moments and suffix cost alone. The pair bound can be loose or negative and must not be reported as an attainable yield.

B SAMPLING IMPLEMENTATION AND COMPUTATIONAL COST

The algorithm below implements the sampling law in Appendix A.1. We describe the selection probabilities actually used, the computation required by one run, and the checks needed to preserve that law. Domain-specific settings appear in Appendix C.4.

Sampling procedure. One independent run generates K base trajectories to a fixed time τ, saves their states, and completes one initial sample from each state (the pilot). It then generates L additional samples from these saved states. The base sampler, verifier, τ, K and L are fixed before observing the pilots. Saved states remain unchanged when reused, and each continuation draws new independent randomness. Exact restoration includes the latent, conditioning and scheduler memory for images, or the token/mask configuration and any other transition-relevant memory for discrete models.

Step	One independent CFS run
1	Simulate K independent base trajectories to τ and save their complete states $Z_{1:K}$.
2	Generate one completed sample from each saved state using independent randomness. Evaluate the verifier and retain all K samples.
3	Compute the selection probabilities s with $B = 2$. Check that they sum to one, remain positive, and satisfy $C(s) \leq B$ within numerical tolerance.
4	Draw L state indices independently from s, or sample their counts from the equivalent multinomial distribution. Generate a new base continuation for each selection. Keep s fixed throughout this step.
5	Verify the additional samples. Assign weights $O_i/(K + L)$ to the pilots and $O(Y_\ell^{(1)}, c)/[(K + L)Ks_{I_\ell}]$ to the additional samples. Count every draw in the estimator, including duplicates; remove duplicates only when measuring distinct discovery.

Table 6: Sampling and weighting in one CFS run.

Computing selection probabilities. The implementation solves the moment-constrained optimization numerically for binary outcomes; Equation 12 gives its closed-form solution. Identical pilot outcomes and $B = 1$ give uniform selection. Importance weights use the probabilities actually used to sample indices, including any recorded numerical fallback to uniform selection. Selection probabilities and sample counts are determined before observing the additional outcomes.

At $L = 0$, CFS reduces to K independent base samples followed by rejection of failures. At $B = 1$, each additional sample selects a saved state uniformly and independently. Balanced uniform selection instead fixes nearly equal sample counts per state, changing the dependence between selections; Uniform_iid is therefore the direct control for feasibility steering.

Computational cost. We count one base transition for one sample as one sample-step. Reaching a saved state costs $c_{\text{pre}} = \tau$, and completing a sample from that state costs $c_{\text{suffix}} = T - \tau$. One run requires

$$C_{\text{CFS}} = Kc_{\text{pre}} + (K + L)c_{\text{suffix}}, \qquad N_{\text{verifier}} = K + L. \tag{18}$$

Here N_{verifier} counts one verification per completed sample before any identical-sample caching. Additional quality evaluations are counted separately. Choose L within the computational budget before observing pilots; report unused budget, batching and elapsed time.

Implementation checks. For each run, record verification outcomes, selected state indices, selection probabilities, sample identities and weights, together with configuration identifiers and measured costs. Report sample-steps, actual verifier calls, independent-evaluator cost and elapsed time separately. Check that restoring a state leaves it unchanged, continuations use independent randomness, and recorded verification outcomes agree with re-evaluation. These checks support correct implementation but do not establish real-model conditional-estimation accuracy. The normalized estimator and its limitations are stated in Appendix A.5.

C Experimental setup and evaluation criteria

This section defines what is generated, what counts as success, which methods are compared, and how each reported experiment is configured. Diagnostic measurements are detailed separately in Appendix D. A group denotes one sampling run at a fixed task or context; its completed samples are dependent observations, not independent replicates. Different tables and diagnostic figures use different experimental groups; their sizes and replication counts are specified below.

C.1 Tasks and verification

Each task provides a base generator and a binary test on completed samples. CFS saves the complete generation state at time τ and resumes the same base process with independent randomness. The verifier and, where needed, the rule for identifying distinct outputs are fixed before evaluation.

Molecules. QM9 (Ramakrishnan et al., 2014) molecules are represented as SELFIES (Krenn et al., 2020) and generated by the uniform-noise decoder. Success requires molecular validity and

QED (Bickerton et al., 2012) above 0.55 or 0.60, according to the experiment. A saved state contains the token/mask configuration and transition-relevant decoder memory. Distinct verified molecules are identified by canonical SMILES.

Images. CIFAR-10 (Krizhevsky et al., 2009) and MNIST (LeCun et al., 1998) use frozen DDPMs (Ho et al., 2020) and target-class classifiers; the MNIST allocation experiment targets digit 7. Restoration preserves the latent, scheduler memory and conditioning. New diffusion noise supplies conditional variation; deterministic replay would not. Image hashes identify exact duplicates, not semantic diversity. An independent classifier, when available, supplies a separate evaluation rather than changing the steering predicate.

Programs and hardware. HumanEval (Chen et al., 2021), MBPP (Austin et al., 2021) and Verilog use LLaDA decoding (Nie et al., 2026). Programs must pass the declared test suite; Verilog modules must compile and pass functional simulation. PCB generation uses LLaDA to produce a track/via description, with success requiring complete connectivity and zero KiCad DRC violations (Kanagachidambaresan, 2021). Saved states include all masked-token or routing-decoder memory. Fresh outputs are checked with the task's tests or DRC checker; cached verification is reused only for identical outputs under the same context and verifier version. Steering and held-out tests are distinguished when both are used. PCB reports valid count because no canonical distinct-board metric is used. Solver-based repair is a different generation policy.

Sudoku and SAT use the same terminal-verification principle. Their task definitions, initialization differences and results are collected in Appendix E.

C.2 Sampling comparisons

We distinguish changes to the sampling schedule from changes to the selection rule. Rejection and branching compare complete sampling schemes; the allocation controls isolate components of CFS.

Reference sampling schemes. Rejection generates independent full base trajectories and accepts passing outputs (Chopin et al., 2020). Branching generates independent trajectories to a chosen time and multiple base continuations from each saved state. Early/late branching changes this time; more-root branching changes the number of independent initial trajectories. These are internally constructed schedules, not implementations of Diffusion Tree Sampling. Their actual work, number of completed samples and unused budget determine whether a comparison is resource matched.

Component controls. Uniform_iid uses the same saved states, pilots and additional-sample budget as CFS but chooses each state uniformly and independently. Uniform balanced assigns nearly equal counts to each state, changing count dependence. Binary single uses CFS's guided selection probabilities but selects one state for all additional samples. All three retain pilots and use the declared sample-weighting rule. Posterior greedy is a separate allocation alternative: a Beta$(1, 1)$ prior is updated by one pilot per state, and additional counts are chosen greedily for the working model's fresh-hit probability. Its objective differs from verified yield, and no unbiased all-output correction is claimed for it.

External comparators and cost matching. Molecular comparisons include adaptations of discrete FK correctors and nested SMC (Hasan et al., 2026; Chanchu et al., 2026); historical image references include TDS and FK steering (Wu et al., 2023; Singhal et al., 2025), which are not part of the current main image table. Intermediate classifier or gradient access is disclosed separately from terminal binary verification. A matched comparison fixes the base model, target predicate and eligible output population and accounts for each method's actual computation. Model sample-steps, verifier calls, additional scoring and elapsed time are distinct costs.

Paired comparisons. For paired comparisons, methods may share an independently generated sequence of conditional samples from each saved state. Each method uses the number of samples prescribed by its own selection rule, starting from the beginning of that sequence. This couples the methods for comparison while preserving each method's sampling law. We distinguish the computation each method would require if run alone from the actual computation used to generate the shared samples once.

Domain	T	τ	L	Samples/group	Sample-steps/group
SELFIES	51	34	36	40	816
CIFAR / MNIST	200	133	35	39	3,145
PCB / LLaDA	128	96	16	20	1,024
Sudoku	51	25	20	24	724

Table 7: CFS allocation schedules and sample-step costs.

C.3 Metrics and uncertainty

Verified yield and return. For group r, let $\mathcal{A}_r$ contain all completed draws eligible for discovery, including repeated outputs, and let κ identify a canonical artifact. Verified yield and distinct verified yield are

$$V_r = \sum_{y \in \mathcal{A}_r} O(y, c), \qquad D_r = \big|\{\kappa(y) : y \in \mathcal{A}_r,\ O(y, c) = 1\}\big|.$$

CFS includes pilots and additional samples in $\mathcal{A}_r$. Archive return is $R_{\mathrm{arch},r} = \mathbf{1}\{V_r > 0\}$. Earlier reference panels instead use a declared final-output population $\mathcal{F}_r$, with $R_{\mathrm{out},r} = \mathbf{1}\{\exists y \in \mathcal{F}_r : O(y, c) = 1\}$; excluded preliminary samples are not retroactively added. Tables report the appropriate group means (↑). The passing fraction $V_r/|\mathcal{A}_r|$ is a different quantity. Independent joint-pass yield is $\sum_{y \in \mathcal{A}_r} O(y, c) O_{\mathrm{blind}}(y, c)$.

Conditional-estimation error. On finite targets, let $g(x) = p_{\mathrm{base}}(x \mid c) O(x, c)$ and let $\widehat{g}_r$ be the weighted estimate from group r. Unnormalized atom MSE is

$$\frac{1}{n} \sum_{r=1}^{n} \|\widehat{g}_r - g\|_2^2.$$

This sums error over atoms without dividing by their number. Write $z_r = \sum_x \widehat{g}_r(x)$, $n_+ = \sum_r \mathbf{1}\{z_r > 0\}$ and $\widehat{\pi}_r = \widehat{g}_r / z_r$ when $z_r > 0$. Normalized atom MSE and TV are

$$\frac{1}{n_+} \sum_{r : z_r > 0} \|\widehat{\pi}_r - \pi_c\|_2^2, \qquad \frac{1}{n_+} \sum_{r : z_r > 0} \frac{1}{2} \sum_x |\widehat{\pi}_r(x) - \pi_c(x)|.$$

Extinction is $1 - n_+/n$; normalized metrics are undefined if $n_+ = 0$. Lower values are better for these error metrics. Duplicate draws retain their weights in the estimator. An atom-MSE measurement is not an observable-MSE measurement.

Replication and uncertainty. We average within each seed and report the mean and sample SD of seed means, with denominator $n_{\mathrm{seed}} - 1$. A single seed does not support a between-seed SD. Paired difference SD is computed from per-seed differences, not from marginal SDs. Groups, tasks or prompts are the replication units; multiple outputs from one group do not increase this count. Group-bootstrap intervals, where retained below, use 20,000 paired resamples and are pointwise. They describe a different uncertainty quantity from seed SD and are not converted into it. Threshold variants sharing seeds are not independent replications.

C.4 Experimental configurations

The common CFS settings are $K = 4$ and $B = 2$. Generation length, saved-state time, additional-sample count and seed count depend on the experiment. Table 7 collects the current allocation schedules; the paragraphs below identify which measurements use them. Counts refer to sampling groups, not the number of unique tasks. Sample-step costs in Table 7 follow Equation 18; other sampling schemes require separate accounting.

SELFIES. The QED > 0.55 uniform/CFS comparison in Tables 1 and 4 uses 200 groups per seed over two seeds (400 total). The > 0.60 comparison uses 200 groups per seed over three seeds (600 total). Each group costs 816 sample-steps. Distinct counts are checked against canonical SMILES and terminal verification records. The rejection and branching entries drawn from these records use the

same 200 groups per seed and threshold-specific seed counts. The molecular literature comparisons use 100 groups per seed and three seeds; they are separate sampling runs. The pilot-only rejection summary uses less work, and the Late_branching row repeats Uniform_iid measurements. Neither is an additional full-budget structural experiment.

The discrete FK and nested SMC comparison budgets are reported as 16 full-trajectory equivalents, corresponding to 816 model sample-steps for $T = 51$. Nested SMC uses $N_{\text{outer}} = 10$ terminal particles; its output count is distinct from the forward-computation budget. The matched-forward comparison does not imply equal terminal-sample counts. In the terminal-only NSMC adapter, inner terminal rollouts estimate intermediate weights and incur additional model work; the final reported outputs are outer particles. This is not uniform, information-free resampling, and low final yield alone does not identify its cause.

An additional QED > 0.70 condition is omitted from Table 1 because the reported evaluations produced no passing samples. The accompanying unconditional check generated 14,500 samples per seed, with maximum observed QED approximately 0.66. This is an empirical zero-hit observation, not proof that higher-QED molecules are outside the generator's support.

The CFS–Uniform_iid group-bootstrap interval at threshold 0.55 is $[0.305, 0.600]$. At threshold 0.60, balanced uniform, Uniform_iid and CFS yield 0.480, 0.487 and 0.535 distinct molecules per group, respectively, with paired gain 0.048 and pointwise 95% interval $[0.012, 0.088]$. The threshold changes both pilot selection evidence and final verification.

CIFAR and MNIST. The uniform/CFS CIFAR ablation cohort has 64 groups from one seed. MNIST uses 64 groups per seed over three seeds (192 total). Both schedules use 3,145 sample-steps under a 3,200-step budget. The CIFAR base transition law includes the `fixed_small` diffusion variance setting. The paired CFS–Uniform_iid group-bootstrap intervals are $[1.8902, 4.3906]$ for this CIFAR cohort and $[2.234, 3.896]$ for MNIST. Separate three-seed analysis figures are described in Appendix D; their estimates are not pooled with these table cohorts.

Table 2 instead uses the three-seed analysis cohort in Figure 3: 64 groups per seed, 192 in total, with means and sample SD across seeds. Its rejection, Late_branching and CFS rows correspond to pilot-only, Uniform_iid and CFS, respectively, and count classifier-valid outputs including pilots. The four pilot-only trajectories use 800 sample-steps; Uniform_iid and CFS each use 3,145. This replaces the earlier class/population return panel and is distinct from the one-seed CIFAR ablation cohort.

Historical text-to-image reference (outside the main comparison). The reference panel uses stochastic DDIM with $\eta_{\text{DDIM}} = 1$, 25 steps, 40 prompts and a development-pinned ImageReward threshold of 1.5583. Rejection generates 24 independent trajectories per prompt and succeeds on 28/40 prompts, using 24,000 sample-steps, 960 verifier calls and 877 seconds. This paragraph describes the original rejection reference. This historical measurement is not part of the current image-domain comparison.

HumanEval and MBPP. The task manifests contain 164 HumanEval tasks and 378 MBPP tasks. Results use the declared executable test suite and report distinct passing programs per group; the MBPP table reports one seed. The code rejection entries are inherited budget-normalized estimates from separate $K = 1$ rejection runs, whereas the branching and CFS entries use $K = 4$ pools. They are not direct measurements under a shared budget or a paired design. The available provenance does not specify a reproducible conversion for these estimates, so they should not be used to infer a measured rejection advantage or disadvantage.

PCB. An earlier PCB evaluation used six boards from the canonical beginning of the `eval_first100` subset of `pcb_eval_set_v1.json`, with four sampling groups per board and seed: 24 groups per seed and 72 over three seeds. This six-board study is distinct from the expanded PCB panel retained in Table 3; its sample counts and return intervals must not be attributed to that panel. The expanded study targets 20 boards and four groups per board per seed.

The sampler is LLaDA-8B with rank-128 LoRA fine-tuning on the mixed real/synthetic v2 corpus, using checkpoint `pretrain_pcb_llada_lora_v1/v7_pcb_mixed_v2_r128_20260920_151335/step_5000`. The CFS schedule in Table 7 generates 20 samples per group. Uniform_iid and CFS share saved states and pilot observations; rejection and Late_branching are

separate sampling controls with their own sample counts. The reported V counts outputs passing connectivity and DRC verification, without deduplication. In the earlier six-board study, archive-return estimates lie between 0.042 and 0.056 and paired intervals include zero. Yield retains multiplicity discarded by return.

For reproducibility, the six board identifiers are `0002_rufs_simple_kicad_schema_and_pcb_v1`, `0003_kitspace_grove_adaptor`, `0005_AVR-Playground_hello_world`, `0008_CapPCB_CapPcb`, `0013_crossover-schiit-stack_xover4schiit`, and `0017_Blink-Eras_AVR_ISP_Pogo`.

Posterior-allocation comparison. Table 5 uses a separate single-seed experiment in each domain: 200 SELFIES groups and 64 CIFAR groups under the corresponding schedules in Table 7. Methods share saved states, pilots and conditional samples; counts are fixed before the additional outcomes are observed. There is one seed per domain, so between-seed SD is unavailable. These results are not pooled with the uniform/CFS confirmation cohorts.

Finite-target comparison. The mode-imbalanced and sparse enumerable targets each use 4,000 groups per seed over three seeds. We fix $K = 4$, $B = 2$, $\tau = \lfloor 2T/3 \rfloor$ and a budget of $16T$ sample-steps, giving $L = \lfloor (16T - K\tau)/(T - \tau) \rfloor - K$. Uniform balanced, Uniform_iid, binary single and CFS retain pilots and additional samples; rejection generates 16 independent base outputs. The runner samples from exactly enumerated conditional continuation laws. It tests the weighting construction against a known target, not real-model conditional fidelity. Checks cover state restoration, independent randomness, selection support and moments, sample counts, costs and verification outcomes.

D Diagnostic analyses and interpretation

This section documents the measurements behind the motivation and analysis figures. First, equal numbers of conditional samples test whether initial success predicts subsequent success. Second, paired allocation comparisons measure the yield gained from using that information. These analyses reuse recorded generations; they are not additional independent experiments.

D.1 Does initial success predict subsequent success?

Figure 1 uses the completed SELFIES and CIFAR batches from two seeds: 200 SELFIES groups or 64 CIFAR groups per seed, totaling 1,600 and 512 saved states. Pilot and additional samples use separate random streams. Balanced allocation guarantees at least nine additional samples per SELFIES state and eight per CIFAR state. We use the first eight from every state, regardless of its initial outcome or guided selection count, excluding the pilot itself.

For seed r and pilot outcome o, the estimate is

$$\widehat{p}_{r,o} = \frac{1}{n_{r,o}} \sum_{i:O_i=o} \frac{1}{8} \sum_{j=1}^{8} O(Y_{ij}^{\mathrm{fresh}}, c).$$

States receive equal weight within each outcome group, and the two seed means are averaged equally. Error bars are seed SD. SELFIES has 1,420 initial failures and 180 successes; CIFAR has 430 and 82. Duplicate artifacts remain draws and are not removed. The pass-minus-fail gaps are $0.438_{\pm 0.006}$ and $0.371_{\pm 0.047}$, respectively; restricting to mixed-label groups preserves the direction in both seeds. This establishes an observed association, not a noise-corrected estimate of latent feasibility variance. The analysis preserves state identities and separates pilot and fresh outcomes.

D.2 How is the allocation gain measured?

Figures 2 and 3 use three seeds, with 200 SELFIES or 64 CIFAR/MNIST groups per seed. The first bins groups by the number of passing pilots and measures CFS–Uniform_iid gain using only additional samples. Bins are defined before these outcomes; they do not estimate latent feasibility heterogeneity. Seed-bin means are averaged equally, and SD is omitted when only one seed contributes. The second figure includes pilots in yield and compares pilot-only generation with Uniform_iid and CFS.

Allocation	Valid draws / 24	Archive return
Uniform balanced	18.280	1.000
Uniform_iid	18.293	1.000
CFS	18.153	1.000

Table 8: Sudoku valid yield and archive return (↑), including pilots.

Pilot-only generation uses less work; the reported Late_branching entry uses the Uniform_iid records. Only the uniform-iid/CFS contrast fixes both sample budget and computation.

D.3 Interpretation of the diagnostics

Pilot-count bins do not estimate latent feasibility variance. Estimating the relation between that variance and allocation gain requires separate conditional draws or cross-fitting. Shared initialization must be conditioned on, and locating repeated outputs requires canonical identities and ancestry records. The uniform–CFS contrast isolates allocation; a refresh comparison must instead hold the selection law fixed between single- and multi-parent sampling.

D.4 Finite-target estimation results

On two enumerable targets, we compare rejection, uniform recycling, binary single-parent allocation and CFS at matched work (4,000 groups each; Appendix C.4). We report unnormalized atom MSE (↓), normalized atom MSE and TV among non-extinct groups (↓), and extinction (↓).

Binary multi-parent sampling reduces unnormalized squared error relative to binary single-parent sampling from $0.058_{\pm 0.005}$ to $0.021_{\pm 0.000}$ on the mode-imbalanced fixture and from $0.093_{\pm 0.004}$ to $0.030_{\pm 0.001}$ on the sparse fixture. Its extinction rates fall from $0.372_{\pm 0.003}$ to $0.205_{\pm 0.004}$ and from $0.221_{\pm 0.005}$ to $0.126_{\pm 0.004}$, respectively. Rejection has lower unnormalized error on both fixtures ($0.009_{\pm 0.000}$ and $0.017_{\pm 0.000}$), and binary multi-parent error is close to Uniform_iid ($0.020_{\pm 0.000}$ and $0.030_{\pm 0.001}$). Rejection also has lower normalized error, TV and extinction in both fixtures. This evaluation samples exactly enumerated conditional suffix laws; it is not a real-model fidelity measurement.

E Additional domain experiments

Sudoku and SAT examine settings where the available binary feedback does not produce a detected allocation advantage. We report each task's sampling configuration, observed result and interpretation limits. These experiments do not share the molecular or image evaluation groups.

E.1 Sudoku

Task and sampling configuration. The UDLM generator produces complete grids, accepted only when every row, column and subgrid is valid. This evaluates grid generation rather than solving a collection of clue-conditioned puzzles. We use $(T, \tau, K, L, B) = (51, 25, 4, 20, 2)$ and 50 groups per seed over three seeds (150 total). Within each group, four trajectories share an initial noise state and use independent subsequent randomness. The saved states are therefore treated as conditionally independent given the initial state. All allocation methods receive the same states and pilot outcomes.

Uniform balanced generates five additional samples per state; Uniform_iid and CFS independently select a state for each additional sample. Each method produces 24 samples at a cost of $4 \times 25 + 24 \times 26 = 724$ sample-steps, leaving 92 under the 816-step budget. The outcome is valid count. Terminal identities were not retained, so the aggregate field named distinct count does not support a distinct-grid measurement.

Results. CFS minus Uniform_iid is -0.140 valid samples/group, with pointwise 95% paired bootstrap interval $[-0.353, 0.080]$ from 20,000 resamples. A two-sided paired t-test gives $p = 0.208$, without adjustment across domains. The mean selection-correction second moment is 1.673; every

recorded group satisfies $B = 2$ within numerical tolerance. We detect no allocation benefit, but do not establish equivalence or a negative population effect.

The initial four samples produce 3.027 valid outputs on average (passing fraction 0.757). This summary costs 204 sample-steps and is not a full-budget rejection comparison. The Late_branching summary repeats the Uniform_iid measurements. Every group contains an initial success, making archive return one for all methods.

Interpretation. A high passing fraction and saturated return do not establish low conditional-feasibility variation. Proposition 2 depends on both the change in selection mass and feasibility variance, here conditional on the shared initial state. Since $0 \leq \mathrm{Var}(H) \leq \mu(1-\mu)$, a small denominator alone does not predict small gain. Independent conditional samples would be needed to identify a weak-information mechanism. This experiment establishes neither that mechanism nor conditional-distribution accuracy.

E.2 SAT

Task and sampling configuration. The implementation uses a `UniformBinaryDenoiser` on satisfiable random 3-SAT instances with 20 variables and 91 clauses; unsatisfiable instances are excluded when constructing the evaluated set. The run uses $(T, \tau, K, L, B) = (20, 8, 4, 36, 2)$ and 200 groups per seed over three seeds (600 total). Generation starts from the all-mask state, and acceptance requires satisfying every clause.

Results. Uniform balanced, Uniform_iid and CFS produce no valid pilot or additional sample in all 600 groups. Every pilot vector is all-fail, so CFS uses uniform selection. Pilot-only rejection uses less work, and the Late_branching summary repeats Uniform_iid measurements. Recorded counts do not support a distinct-assignment claim.

Interpretation and budget limitation. The recorded work is $4 \times 8 + 40 \times 12 = 512$ sample-steps per method and group, exceeding the declared $16T = 320$ budget by 192. The methods share this workload, but the run is excluded from cap-matched claims. With no successful pilots, it tests only the uniform-allocation fallback, not informative feasibility steering. Zero observed hits do not establish zero target mass.

F Extended related work

CFS connects terminal verification to selective reuse of intermediate states. We position it relative to methods that enforce constraints directly, use sampled futures for guidance, reuse generated paths, or correct a modified sampling law.

Projection-based constraint enforcement. Projected Diffusion Models formulate constrained synthesis through projections during sampling (Christopher et al., 2024). PRODIGY constructs projection operators for hard graph constraints, including non-differentiable constraints (Sharma et al., 2024); Constrained Discrete Diffusion integrates differentiable constraint optimization into discrete generation (Cardei et al., 2026). These approaches exploit an operational representation of the constraints. CFS instead assumes only a terminal binary verifier and a resumable base generator: it reallocates continuations without constructing a projection or modifying the base denoising kernel.

Lookahead information and candidate selection. SVDD and nested SMC use future-value information to guide candidate selection (Li et al., 2024; Chanchu et al., 2026). CFS shares this lookahead perspective but uses one binary completion outcome per saved state, without a learned feasibility predictor. Its selection law favors successful states subject to a bound on the second moment of the importance correction. The comparison with uniform selection isolates the value of this terminal information under the same generation budget.

Reusing generated paths. Diffusion Tree Sampling (DTS), distinct from TDS, reuses explored paths and rollout backups (Jain et al., 2026). Particle Gibbs with ancestor sampling retains a reference trajectory within an MCMC kernel (Lindsten et al., 2014); waste-free SMC and particle recycling

reuse intermediate or past samples (Dau & Chopin, 2022; Nguyen et al., 2014). CFS retains all initial completions and generates additional samples by independently reselecting a saved state for each continuation. Standard index importance ratios combine initial and additional samples in an estimator that preserves unnormalized target expectations.

Conditional generation and correction. Rejection supplies direct conditional samples; branching shares prefix computation; importance sampling corrects a changed proposal; and SMC combines weighting with an interacting particle population (Chopin et al., 2020). Twisted Diffusion Sampling (TDS), FK steering, discrete correctors and nested SMC adapt related ideas to diffusion generation (Wu et al., 2023; Singhal et al., 2025; Hasan et al., 2026; Chanchu et al., 2026). Comparisons with hard terminal verification must match the target, information access and output contract.