COUNTERFACTUAL PROBING FOR PARALLEL UNMASKING WITH HIDDEN FOREST STRUCTURE

Anonymous authors
Paper under double-blind review

ABSTRACT

Masked generative models offer parallel token prediction, but accurate parallel sampling must account for dependencies among tokens. When dependencies are unknown, finding safe batches also costs model evaluations. We study whether total evaluations, including discovery, can be sublinear in sequence length N; sublinear sequential depth then follows. We consider discrete distributions with hidden forest structure, accessed through a fixed approximate conditional oracle. Under explicit regularity conditions and uniform Hellinger error bounds, for any fixed target accuracy $\varepsilon \in (0, 1/8]$ and sufficiently large N, our sampler achieves seed-averaged total-variation error at most ε, with total masked-state submissions and sequential depth both bounded by $\widetilde{O}(N^C \varepsilon^{-a})$ for constants $0 < C < 1$ and $a > 0$. These guarantees use polynomial vocabulary size and an edge-response lower bound set by N and ε. The sampler shares evaluations of hypothetical reveals across dependence tests to identify safe parallel batches without requiring full recovery of the hidden forest. A tunable parameter trades probing cost against irreversible commit rounds. In the same class, any admissible irreversible product-commit sampler attaining the same seed-averaged accuracy requires $\Omega(N^c \varepsilon^b)$ counterfactual submissions or commit rounds in the worst case, for constants $c, b > 0$.

1 INTRODUCTION

Masked generative models predict missing tokens from a partially revealed sequence, allowing many positions to be processed in one model evaluation (Austin et al., 2021; Sahoo et al., 2024; Nie et al., 2025). Turning these predictions into a joint sample is less straightforward. Drawing several tokens independently can miss dependencies between them. With exact conditionals, revealing one token at a time samples exactly by the chain rule, in any fixed order, but uses N model evaluations in N sequential steps for a sequence of length N. The question is whether parallel unmasking can reduce this total cost.

Choosing which tokens to reveal together is itself part of the problem. Confidence, entropy, and adaptive ordering guide practical samplers (Chang et al., 2022; Ben-Hamu et al., 2025; Kim et al., 2025; Hayakawa et al., 2026). Yet the singleton predictions at one masked state do not, in general, determine the dependence among its masked positions. Evaluating hypothetical reveals can supply additional information: existing methods compare candidate continuations or test contextual independence across positions (Lee et al., 2025; Azangulov et al., 2026). These evaluations also contribute to the sampling cost. These methods do not establish a sublinear total-evaluation bound at fixed sampling accuracy. We therefore ask:

> *Can a sampler discover enough dependence structure and generate an accurate sample with $o(N)$ model evaluations in total, at fixed accuracy?*

To study this question, we consider distributions that factorize over hidden forests, classical models of discrete dependence (Chow & Liu, 1968; Tan et al., 2011). For a known forest, exact conditionals allow exact sampling in $O(\log N)$ rounds by revealing one centroid per residual component. Our goal is to exploit this conditional separation while accounting for the cost of discovering safe batches.

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.

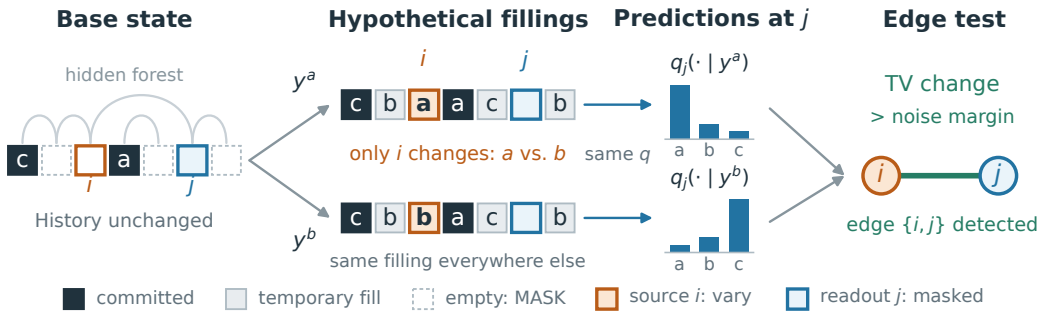


Figure 1: Counterfactual edge probing; faint arcs show the hidden forest. Keeping the committed state unchanged, vary only source i between two hypothetical fillings and compare the predicted rows at masked readout j. With the same fully filled background, a response above the oracle-noise margin detects $\{i, j\}$. Section 4.2 packs and aggregates such tests.

Key strategy: shared counterfactual probing. We compare singleton predictions from a fixed approximate conditional oracle under hypothetical token fillings, without committing the tested values (Figure 1). Sharing evaluations across many dependence tests reduces discovery cost. The resulting local structure guides generation, progressively simplifying the remaining forest. To generate tokens, we independently sample a batch from the oracle's singleton conditionals and commit it permanently. We count evaluations used for both probing and generation.

Main results, informally. *Under the target and oracle assumptions in Section 2, for any fixed sufficiently small accuracy $\varepsilon > 0$ and all sufficiently large N, there is a sampler with seed-averaged total-variation (TV) error at most ε using $\widetilde{O}(N^{2/3+1/(9s)})$ evaluations in total, which also gives sublinear sequential depth. Here $s > 1$ is the rank-frequency exponent. Conversely, every admissible irreversible product-commit sampler meeting the same accuracy target uniformly over the class requires worst-case budgets of $\Omega(N^c)$ probes or $\Omega(N^c)$ commit rounds for some $c > 0$ (Theorems 3.1–3.2).*

The model assumes detectable edge responses, controlled tail responses, a rank-frequency envelope, and polynomial vocabulary. Its uniform oracle accuracy also suffices for the N-step baseline to meet the output target (Appendix J.2).

Existing distribution-general samplers achieve smaller depth exponents with sufficient total budgets of $O(N)$ or $O(N \log N)$ (Anari et al., 2024; 2026). Dependence-adaptive guarantees can be sublinear on weakly dependent instances (Li & Cai, 2025; Zhao & Cai, 2026). Our guarantee also covers paths with constant dependence per edge and total correlation $\Theta(N)$ (Proposition I.1; Section 5).

Section 4 combines random-color probes, majority certification of low-degree neighborhoods, singleton peeling of the high-degree core, and parallel centroid commits, without first recovering the original forest. A public degree cutoff trades probing work against commit rounds.

2 Problem Formulation

Public and hidden information. The vocabulary and its order, class parameters, accuracy target, and algorithm parameters are public: known to the sampler and fixed before the hidden instance is chosen. The target, forest, exact marginals, ranks, and response witnesses are hidden. For the main results, fix $0 < \varepsilon \le 1/8$ before taking $N \to \infty$; we suppress dependence on N. The appendix retains the general finite setup.

For laws p, q on a finite alphabet, write $d_{\mathrm{TV}}(p, q) = \frac{1}{2}\sum_a |p(a) - q(a)|$, $h^2(p, q) = 1 - \sum_a \sqrt{p(a)q(a)}$, and $D_{\mathrm{KL}}(p\|q) = \sum_a p(a)\log(p(a)/q(a))$ (natural logarithms). Write Δ° for the strictly positive probability simplex.

2.1 HIDDEN FOREST AND REGULARITY

Definition 2.1 (Forest target and committed history)**.** Let $[N] = \{1, \dots, N\}$ and let $\mathcal{V}$ be a vocabulary of size V with public order $\prec$. The target $X \in \mathcal{V}^N$ has a strictly positive law $P \in \Delta^\circ(\mathcal{V}^N)$ and a hidden undirected forest $F = ([N], E)$ (Tan et al., 2011; Bhattacharyya et al., 2023). Write π_i for the marginal of X_i. A committed history $H_G = (G, x_G)$ leaves residual positions $U_G = [N] \setminus G$ and forest $F_G = F[U_G]$ (Definition A.1 and Equation (14)).

Assumption 2.2 (Forest factorization)**.** Choose one root per component, with root set $\mathrm{Roots}(F)$ and parent $\mathrm{pa}(j)$ for each nonroot. Assume the factorization (Assumption A.2):

$$P(x) = \prod_{u \in \mathrm{Roots}(F)} \pi_u(x_u) \prod_{j \notin \mathrm{Roots}(F)} P(X_j = x_j \mid X_{\mathrm{pa}(j)} = x_{\mathrm{pa}(j)}). \tag{1}$$

Definition 2.3 (Tail sets and directed response)**.** For distinct positions i, j, call i the *source* whose token value is varied and j the masked *readout* whose conditional distribution is inspected. For a complete boundary $z \in \mathcal{V}^{[N]\setminus\{i,j\}}$, write $\mathcal{T}_i(t) = \{a : \pi_i(a) \le t\}$ and $\mu_{i\to j}(\cdot \mid a, z) = \mathcal{L}_P(X_j \mid X_i = a,\, X_{[N]\setminus\{i,j\}} = z)$. The directed response is $\Omega_{i\to j}(z) = \max_{a,b\in\mathcal{V}} d_{\mathrm{TV}}(\mu_{i\to j}(\cdot \mid a, z), \mu_{i\to j}(\cdot \mid b, z))$ (Definition A.3).

Assumption 2.4 (Regularity and scaling for the main results)**.** Let $s > 1$ and $\omega \in (0, 1]$ be public.

- **(RT) Tail response.** For all distinct i, j, $t \in [0, 1]$, and complete boundaries z, $\max_{a,b\in\mathcal{T}_i(t)} d_{\mathrm{TV}}\big(\mu_{i\to j}(\cdot \mid a, z), \mu_{i\to j}(\cdot \mid b, z)\big) \le t$. For tails of size at most one, take the maximum as zero. Tail tokens are thus interchangeable within TV error t (Assumption A.4).
- **(UEN) Edge nondegeneracy.** Every edge $\{u, v\} \in E$ and complete boundary z satisfy $\min\{\Omega_{u\to v}(z), \Omega_{v\to u}(z)\} \ge \omega$. The witnessing token pair may depend on the direction and boundary (Assumption A.4).
- **(RF) Rank-frequency envelope.** Rank tokens $a_{i,k}$ by decreasing π_i (ties use $\prec$). For every position i and rank k, $\pi_i(a_{i,k}) \le k^{-s} + V^{-1}$ (Assumption A.5). No lower frequency bound or adjacent-rank separation is assumed. This bounds tokens above thresholds exceeding V^{-1} and, with RT, explicit token choices (Section 4.1).
- **Scaling.** For a public $\nu > 1/3$, take $V = N^{\nu+o(1)}$ and set $\omega = (\varepsilon^2/(32N))^{1/3}$ (Corollary G.3).

2.2 ORACLE ACCESS AND SAMPLING OBJECTIVE

One oracle evaluation is one *masked-state submission*; a *probe* is a noncommit counterfactual submission. Hypothetical-reveal tests also appear in PUNT (Azangulov et al., 2026); Definition 2.6 charges probes to the sampling budget.

Definition 2.5 (Frozen conditional oracle)**.** A masked state is $y \in (\mathcal{V} \cup \{\mathsf{MASK}\})^N$ with observed set $O(y) = \{i : y_i \ne \mathsf{MASK}\}$. For a masked readout $j \notin O(y)$, its exact row is $\mu_j(\cdot \mid y) = \mathcal{L}_P(X_j \mid X_{O(y)} = y_{O(y)})$. The oracle $q = \{q_j\}_{j\in[N]}$ is fixed and deterministic (Definition A.6). One submission returns probability vectors for a requested subset S of masked positions:

$$(y, S) \xrightarrow{\text{one masked-state submission}} \big\{q_j(\cdot \mid y)\big\}_{j\in S}, \qquad S \subseteq [N] \setminus O(y).$$

Each row is the full vector $q_j(\cdot \mid y) = (q_j(a \mid y))_{a\in\mathcal{V}} \in \Delta^\circ(\mathcal{V})$; the submission costs one evaluation, regardless of $|S|$.

Definition 2.6 (Sampling operations, resources, and risk)**.** An admissible algorithm uses q adaptively without remasking committed positions (Definition A.9). At $H_G = (G, x_G)$, both operations preserve $y_G = x_G$:

- *Probe.* Submit (y, S) with $S \subseteq [N] \setminus O(y)$ and inspect the probabilities $\{q_j(a \mid y) : j \in S,\ a \in \mathcal{V}\}$ returned together. The state y may temporarily reveal residual coordinates; only the transcript changes, while (G, x_G) stays fixed.

- *Commit.* Choose a nonempty batch $B \subseteq U_G$ from prior information, before the reply. Submit (y, B) with all residual positions masked, receive $\{q_j(\cdot \mid y)\}_{j\in B}$, and sample $z_B \sim \bigotimes_{j\in B} q_j(\cdot \mid y)$ to commit permanently.

Batches eventually partition $[N]$. Committing counterfactual values or using verify-and-accept steps is not allowed.

Resources (Definition A.11). Let $s = 1, 2, \ldots$ index the submissions actually performed, and let s_{pre} index the distinguished preprocessing all-mask readout (0 if absent). Define

$$Q_{\mathrm{cf}} := \big|\{s : s \neq s_{\mathrm{pre}},\ \text{submission } s \text{ is a probe}\}\big|,$$
$$R := \big|\{s : \text{submission } s \text{ performs a nonempty product commit}\}\big|.$$

The construction's total is $1 + Q_{\mathrm{cf}} + R$, including its one initial all-mask readout. Repeated states and parallel submissions count separately; positions and tests sharing one submission do not. The oracle depth D counts sequential oracle stages; returned probabilities and local computation are not charged separately.

Output law and risk. The algorithm's decision rule chooses submissions and batches from available information and a seed W independent of the hidden instance and primitive commit-draw randomness. Conditional on $W = w$, the output law $\widehat{P}^q_w$ integrates the commit draws (Definition A.10). Its seed-averaged risks are $\mathcal{R}_{\mathrm{TV}}(\mathcal{A}; P, q) = \mathbb{E}_W d_{\mathrm{TV}}(P, \widehat{P}^q_W)$ and $\mathcal{R}_{\mathrm{KL}}(\mathcal{A}; P, q) = \mathbb{E}_W D_{\mathrm{KL}}\Big(P \Big\| \widehat{P}^q_W\Big)$ (Definition F.1).

The risks average divergence over decision-rule seeds, before mixing their output laws. Convexity gives $d_{\mathrm{TV}}(P, \mathbb{E}_W \widehat{P}^q_W) \le \mathcal{R}_{\mathrm{TV}}(\mathcal{A}; P, q)$: the upper bound also controls the mixed output, while the lower bound concerns the stated seed-averaged criterion.

Assumption 2.7 (Uniform frozen-oracle accuracy: two cases)**.** For the fixed public accuracy ε, assume one of the following bounds uniformly over all valid state–readout pairs.

Case (i) (Hellinger, A2; Assumption A.7): $h^2(\mu_j(\cdot \mid y), q_j(\cdot \mid y)) \le \varepsilon^2/(8N)$.

Case (ii) (forward KL; Definition F.1): $D_{\mathrm{KL}}(\mu_j(\cdot \mid y) \| q_j(\cdot \mid y)) \le \varepsilon^2/(4N)$.

Both cases imply row-TV error at most $\varepsilon/(2\sqrt{N})$, uniformly over counterfactual and commit states. This uniform requirement is stronger than average prediction accuracy of a trained model.

Definition 2.8 (Target–oracle classes)**.** For each case, take all target–oracle tuples satisfying Assumptions 2.2 and 2.4 and the corresponding condition of Assumption 2.7. For case (i), denote this class by $\mathfrak{F}^{\mathrm{sh}}$ and write (P^I, q^I) for the target and oracle of instance $I \in \mathfrak{F}^{\mathrm{sh}}$ (Corollary G.3).

3 Main results

3.1 Sublinear sampling is possible

The positive result controls total submissions and sequential depth. The public algorithm parameter d trades discovery submissions against irreversible commit rounds; target degrees may be arbitrary.

Theorem 3.1 (Existence of a sublinear oracle sampler)**.** *For either target–oracle class in Definition 2.8 and every public integer $9 \le d \le N - 1$, there is an admissible sampler $\mathcal{A}_{\mathrm{pack}}$, depending only on the public class data and d, with the following guarantees for all sufficiently large N. Uniformly over the respective target–oracle class, on every execution path,*

$$Q_{\mathrm{cf}} = \widetilde{O}\Big(d^2 (N/\varepsilon^2)^{1/(3s)}\Big), \qquad R, D = \widetilde{O}(N/d). \tag{2}$$

Here $\widetilde{O}$ hides powers of $\log N$, with ε fixed. In case (i), $\sup_{I\in\mathfrak{F}^{\mathrm{sh}}} \mathcal{R}_{\mathrm{TV}}(\mathcal{A}_{\mathrm{pack}}; P^I, q^I) \le \varepsilon$. In case (ii), uniformly over its class, $\mathcal{R}_{\mathrm{KL}}(\mathcal{A}_{\mathrm{pack}}; P, q) \le \varepsilon^2$.

In particular, choosing $d = \lceil (N\omega^{1/s})^{1/3} \rceil$ gives the common balanced bound

$$1 + Q_{\mathrm{cf}} + R,\ D = \widetilde{O}\Big(N^{2/3+1/(9s)}/\varepsilon^{2/(9s)}\Big) = o(N). \tag{3}$$

Since $s > 1$, both total submissions and depth are sublinear at fixed admissible accuracy. Section 4 gives the construction and Corollary G.3 the finite calibration and logarithmic factors. Table 1 compares bounds under the respective target and oracle assumptions.

Table 1: Sufficient bounds at fixed positive accuracy and polynomial vocabulary. Mean: seed-averaged divergence; exp.: expected resources (ours are pathwise). TC/DTC: total/dual total correlation; $\widehat{T}$: a supplied bound on either. Path: Proposition I.1. Oracle conditions, accuracy dependence, and markers a, b: Appendices I–J.

Sampler / variant	Target / oracle	Submissions	Depth	Output
Sequential chain rule	Arbitrary / exact	N	N	Exact
One product batch	Arbitrary / exact	1	1	No error bound[a]
Anari et al. (2024)	Arbitrary / exact	$O(N)$, exp.	$\widetilde{O}(N^{2/3})$	Exact
Anari et al. (2026)	Arbitrary / noisy[b]	$O(N \log N)$, exp.	$\widetilde{O}(N^{1/2})$	TV
Li & Cai (2025)	TC/DTC / averaged prediction error	$O(1 + \mathrm{TC} + \mathrm{DTC})$ (path: $O(N)$)	same	Mean KL
Chen et al. (2026)	Supplied $\widehat{T}$ / exact	$\widetilde{O}(1 + \widehat{T})$ (path: $\widetilde{O}(N)$)	same	Mean KL
This paper, case (i)	Regular hidden forests / uniform Hellinger	$\widetilde{O}(N^{2/3+1/(9s)})$	same	Mean TV
This paper, case (ii)	Regular hidden forests / uniform forward KL	$\widetilde{O}(N^{2/3+1/(9s)})$	same	Mean KL

3.2 A QUERY–ROUND LOWER BOUND

The lower bound applies to every admissible sampler in the Hellinger/TV class $\mathfrak{F}^{\mathrm{sh}}$: each commit samples a product batch from the current singleton predictions and permanently fixes its tokens. Accuracy is measured by the seed-averaged TV risk in Definition 2.6.

Theorem 3.2 (Fixed-accuracy query–round lower bound)**.** *For all sufficiently large even N, every admissible algorithm with deterministic bounds $Q_{\mathrm{cf}} \le \overline{Q}$ and $R \le \overline{R}$ on every instance, seed, transcript, and execution path and $\sup_{I \in \mathfrak{F}^{\mathrm{sh}}} \mathcal{R}_{\mathrm{TV}}(\mathcal{A}; P^I, q^I) \le \varepsilon$ satisfies*

$$\overline{Q} = \Omega\Big((N/\varepsilon^2)^{1/(3s)}\Big) \quad \textit{or} \quad \overline{R} = \Omega\Big(N^{1/3}\varepsilon^{1/3}\Big). \tag{4}$$

The constants may depend on s, but not on the hidden instance or the algorithm. The sufficiently-large-N threshold may depend on the fixed public accuracy ε.

Lower-bound construction. The hard family is a hidden matching whose oracle reveals dependence only when a source is set to its private trigger token. One submitted state tests at most one candidate per source, even when many readout rows are returned. The proof converts collisions of unresolved matched endpoints within product-commit batches into a TV lower bound, showing that accuracy requires enough submissions or commit rounds. The hard family uses $\omega \asymp (\varepsilon^2/N)^{1/3}$: its nontrigger squared-Hellinger error is of order $\omega^3 \asymp \varepsilon^2/N$. Thus the signal floor vanishes as $N^{-1/3}$ at fixed accuracy; changing ε also changes the target class. Corollaries G.2 and G.3 give the finite bound and its specialization to Theorem 3.2.

4 ALGORITHMIC CONSTRUCTION

We now construct the sampler in Theorem 3.1. The key is to learn the residual forest as sampling removes vertices. Step 2–Step 3 share counterfactual probes to recover low-degree neighborhoods

with high probability. When all screens succeed, Step 4 uses these reports to shrink the high-degree core by singleton commits, and Step 5–Step 6 recover the remaining forest for parallel centroid commits (Algorithm 1).

Algorithm 1 Counterfactual-probing sampler: shared tests, claim-based peeling, and centroid commits. Full safeguards: Algorithm 2.

Input: N, ordered vocabulary $\mathcal{V}$, frozen oracle q, accuracy ε
Parameters: $d, J, \underline{t}, \delta_{\mathrm{tail}}, p, M_{\mathrm{rnd}}$ (Corollary G.3)
Step 1. Preprocess & initialize.
repeat
  Step 2. Draft & color. Fix a draft; draw fresh independent colorings
  Step 3. Packed screen.
  for each coloring $\tau = 1, \ldots, M_{\mathrm{rnd}}$ **do**
    Step 3.1. Split each readout color class into chunks
    for each nonempty chunk $\mathcal{C}$ and other source color **do**
      for each bank-or-tail column κ **do**
        Step 3.2. Submit the packed state; retain replies at $j \in \mathcal{C}$
  Step 3.3. Compute votes, strict majorities, and capped rows $A_j(H_G)$
  Step 4. Claim & peel. Form $\mathcal{P}(H_G)$; if empty or at the phase cap, **break**
  Freeze $\mathcal{P}(H_G)$ and commit its vertices singly in public order
Step 5. Terminal graph. Step 6. Centroid commits.

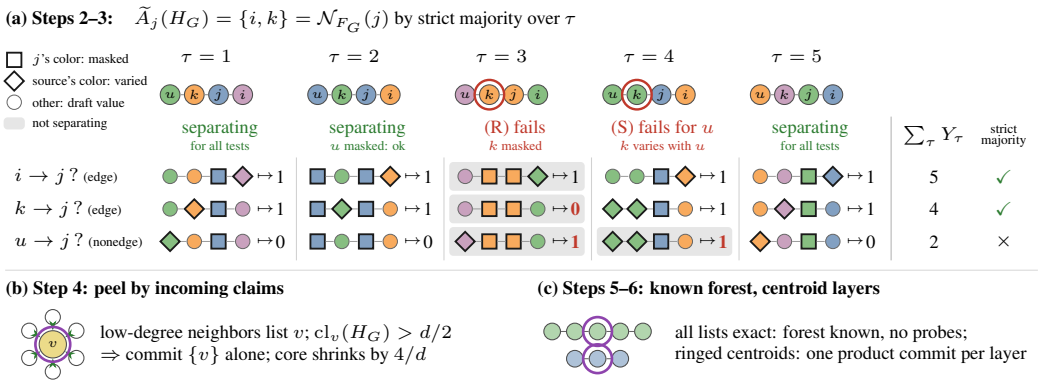


Figure 2: Shared tests and majority recovery. (a) Each cell represents all bank/tail columns; each readout color has one chunk. Equal-color pairs (no diamond) get vote 0 without probing. Grey: nonseparation; red: illustrative errors. (b) Singleton peels contract the core per phase. (c) One centroid per component forms a terminal batch.

4.1 Building a single-source edge test

We prepare the edge test that Section 4.2 will share across pairs. Fix a history $H_G = (G, x_G)$, distinct residual positions i, j, and a query-free draft f. The single-source state y^a reveals x_G, sets $y^a_i = a$, masks j, and fixes all other residual coordinates to their draft values (Figure 1). Its exact reply $\mu_{i\to j}(\cdot \mid a, z)$ has boundary $z = (x_G, f_{U_G\setminus\{i,j\}})$. For a nonedge this row is constant in a by forest separation (Lemma E.2); for an edge its TV diameter is at least ω by Assumption 2.4 (UEN). Testing all ordered pairs and tokens separately costs $|U_G|(|U_G| - 1)V$ submissions.

Banks and tail representatives (Step 1). To reduce token choices, we test a local vocabulary bank and one representative of the omitted tokens. Write $\varepsilon_0 := \varepsilon/(2\sqrt{N})$ for the row-TV error bound in either case (Equations (103) and (195)). On the threshold grid $\mathcal{G}$ with floor $\underline{t}$, choose $t = \max\{u \in \mathcal{G} : u \le \delta_{\mathrm{tail}}\}$, the largest threshold allowed by the tail budget. One all-mask submission gives the local vocabulary banks $\widehat{\mathcal{B}}_i := \{a \in \mathcal{V} : q_i(a \mid \mathsf{MASK}^N) \ge t - \varepsilon_0\}$ and tail representatives $b_i = \min_{\prec}(\mathcal{V} \setminus \widehat{\mathcal{B}}_i)$ (Definition D.1). The standard draft reuses these rows without another submission (Equation (114)).

Assumption 2.4 (RT) allows each omitted token to be replaced by b_i within TV error $\delta_{\mathrm{tail}} = \omega/2$. Assumption 2.4 (RF) bounds $\ell_{\max} = \max_i |\widehat{\mathcal{B}}_i| = O(\omega^{-1/s})$.

A resolvable edge signal. The test queries every bank token and the tail representative, then compares $\max_{a,a' \in \widehat{\mathcal{B}}_i \cup \{b_i\}} d_{\mathrm{TV}}(q_j(\cdot \mid y^a), q_j(\cdot \mid y^{a'}))$ with $2\varepsilon_0$. This observed diameter is at most $2\varepsilon_0$ for a nonedge; for an edge it is at least $\omega - \delta_{\mathrm{tail}} - 2\varepsilon_0 > 2\varepsilon_0$ when $\delta := 4\varepsilon_0 + \delta_{\mathrm{tail}} < \omega$. Under the scaling in Assumption 2.4, our parameter choices make preprocessing feasible and ensure $\delta < \omega$ for all sufficiently large N (Corollary G.3). Lemma E.6 proves the diameter bound.

4.2 Packing tests with random colorings

We share the tests of Section 4.1 across source and readout groups. For each coloring, Step 3.1 splits color classes into chunks of at most $\lceil N/J \rceil$ positions. Step 3.2 masks one chunk and varies another color class through bank-or-tail columns κ: each source takes its own token value, while committed values and the remaining draft stay fixed. There are $\Lambda \le \ell_{\max} + 1$ submitted columns; Equations (115) and (116) specify padding and the empty-bank case. The same replies test every source–readout pair between the two groups. Step 2 draws M_{rnd} independent colorings with independent uniform colors $h_\tau(i) \in [p]$, $p = 8(d+1)$. All screen probes are fixed before replies: one oracle stage (Algorithm 3). Coloring also shares evaluations in sparse Jacobian estimation (Coleman & Moré, 1983).

For $i \to j$, the single-source test is preserved if every neighbor of j is revealed and all except i stay fixed. Call a coloring *separating* when:

- *Readout separation (R).* No neighbor of j has j's color, so none is masked.
- *Source separation (S).* No neighbor of j other than i has i's color, so none varies with i.

For separating colorings with distinct source/readout colors, exact replies equal their single-source counterparts (Lemma E.5). Figure 2(a) shows possible collision errors and majority voting.

Voting and capped candidate lists (Step 3.3). Each coloring casts one vote on whether i and j are adjacent, using the shared replies from Step 3.2. For differently colored i, j, compare the conditional distributions returned at j across the columns probing i's color class. Let $D_\tau(i, j \mid H_G)$ be their maximum pairwise TV distance (Equation (125)). Applying the threshold from Section 4.1 gives the vote $Y_\tau(i, j \mid H_G) = \mathbf{1}\{D_\tau(i, j \mid H_G) > 2\varepsilon_0\}$. Equal-color pairs have no corresponding probe and receive vote zero. We retain candidates supported by a strict majority of all M_{rnd} colorings:

$$\widetilde{A}_j(H_G) := \left\{ i \in U_G \setminus \{j\} : \sum_{\tau=1}^{M_{\mathrm{rnd}}} Y_\tau(i, j \mid H_G) > M_{\mathrm{rnd}}/2 \right\}.$$

The returned list $A_j(H_G)$ equals $\widetilde{A}_j(H_G)$ when it contains at most d candidates; otherwise, it retains the d candidates with the highest vote totals, breaking ties by public position order. Random hashing followed by majority aggregation also appears in junta learning (Bshouty & Costa, 2018).

Low-degree recovery (Step 2–Step 3). For a readout j with $d_{F_G}(j) \le d$, conditions (R) and (S) forbid at most $2d$ color equalities. Conditional on the past and the fixed draft, each equality has probability $1/p$. Thus, each fresh coloring separates any fixed pair (i, j) with probability at least $1 - 2d/p > 3/4$.

By the single-source margin $\delta < \omega$ from Section 4.1, every separating coloring gives the correct edge/nonedge vote. This includes same-color separating pairs: condition (R) excludes an edge, so their zero vote is correct. A strict separating majority therefore classifies the pair correctly, regardless of the remaining votes. Let $\mathsf{Iso}(H_G)$ denote the analysis-only event that such a majority holds for every $i \ne j$ with $d_{F_G}(j) \le d$. On this event, each low-degree candidate list equals the true neighborhood. Its size is already at most d, so truncation leaves it unchanged (Lemma E.6):

$$A_j(H_G) = \mathcal{N}_{F_G}(j) \quad \text{for every } j \in U_G \text{ with } d_{F_G}(j) \le d. \tag{5}$$

By independence across colorings, Hoeffding's inequality bounds the probability that a fixed pair lacks a strict separating majority by $e^{-M_{\mathrm{rnd}}/8}$. A union bound over pairs gives

$$\Pr\big(\mathsf{Iso}(H_G)^{\mathsf{c}} \mid \text{past including the current draft}\big) \le N^2 e^{-M_{\mathrm{rnd}}/8}. \tag{6}$$

Fresh colors make this bound valid at every adaptively reached screen. A further union bound over the capped number of screen calls controls failure along the full execution (Lemma E.4).

4.3 Peeling the high-degree core, then centroid layers

Step 4 commits vertices with many incoming neighbor reports singly. When all screens succeed, repeated peeling removes the high-degree core, enabling forest recovery and parallel centroid commits (Step 5–Step 6).

Define the incoming claim count and the peel set (also Equation (106)) by

$$\mathrm{cl}_v(H_G) := |\{u \in U_G : v \in A_u(H_G)\}|, \qquad \mathcal{P}(H_G) := \{v \in U_G : \mathrm{cl}_v(H_G) > d/2\}. \tag{7}$$

Why peel by incoming claims (Step 4)? A high-degree vertex's own report may be unreliable, but each low-degree neighbor reports it correctly. Peeling may also select low-degree vertices; singleton commits introduce no within-batch dependence error. The size cap controls their cost: on successful screens, $|A_u(H_G)| \le d_{F_G}(u)$, so

$$(d/2)|\mathcal{P}(H_G)| \le \sum_v \mathrm{cl}_v(H_G) = \sum_u |A_u(H_G)| \le \sum_u d_{F_G}(u) < 2|U_G|.$$

Thus a phase uses fewer than $4|U_G|/d$ singleton rounds.

The high-degree core shrinks. Let $\mathcal{V}^{\mathrm{hi}}(H_G) = \{v : d_{F_G}(v) > d\}$. On a successful screen, an unpeeled high-degree vertex has at most $d/2$ low-degree neighbors, since each contributes a claim; it therefore has more than $d/2$ neighbors in $\mathcal{V}^{\mathrm{hi}}(H_G)$. The forest induced on $\mathcal{V}^{\mathrm{hi}}(H_G)$ has degree sum below $2|\mathcal{V}^{\mathrm{hi}}(H_G)|$, so fewer than $(4/d)|\mathcal{V}^{\mathrm{hi}}(H_G)|$ high-degree vertices remain unpeeled when $\mathcal{V}^{\mathrm{hi}}(H_G)$ is nonempty. Deleting vertices cannot raise degrees; the next history H'_G satisfies

$$|\mathcal{V}^{\mathrm{hi}}(H'_G)| \le (4/d)|\mathcal{V}^{\mathrm{hi}}(H_G)|. \tag{8}$$

Finish without further probes (Step 5–Step 6). On successful-screen paths, the $O(\log N)$ phase cap or an empty peel set ensures residual maximum degree at most d (Lemma E.7). Since screening precedes either test, Equation (5) makes every terminal row exact. Joining reported neighbors therefore recovers the true residual forest. The exact joint conditional of one centroid per component factorizes. Centroid deletion halves component sizes, giving logarithmically many layers (Lemmas E.2 and H.4). These layers form a vertex ranking (Iyer et al., 1988), with higher ranks assigned to earlier commits. Only commit submissions remain: later steps delete vertices of the known forest rather than rediscovering it.

4.4 Resources and accuracy

Each screen uses at most $\Lambda M_{\mathrm{rnd}} p(p + J)$ submissions: columns, colorings, source colors, and readout chunks. With $J = d$, the bank bound, phase count, and peel-size bound yield

$$Q_{\mathrm{cf}} = \widetilde{O}(\omega^{-1/s} d^2), \qquad R, D = \widetilde{O}(N/d).$$

Larger d increases screening cost but reduces commit rounds; balancing gives Equation (3). Public caps enforce the bounds even on failed-screen paths (Lemma E.8); Algorithm 2 specifies singleton cycle repair and the round-cap all-residual product-commit fallback.

On paths where all screens succeed, the target conditional factorizes over every commit batch. The appendix bounds oracle error and the contribution of failed screens using separate TV and forward-KL arguments (Lemmas E.9 to E.11 and Theorem F.3). The calibration in Corollary G.3 gives the guarantees of Theorem 3.1 for the sampler in Algorithm 1.

5 Related work

Masked generation and dependence-aware decoding. Under standard absorbing masks, the denoising target can be expressed using time-independent clean-data conditionals (Ou et al., 2025; Zheng et al., 2025), motivating our frozen-oracle model. Most closely related,

PUNT (Azangulov et al., 2026) uses hypothetical reveals and shares evaluations across contextual-independence tests, but does not establish a sublinear total-evaluation guarantee or an end-to-end sampling-error bound. Our random-color screen instead certifies local neighborhoods and yields sublinear total submissions under the hidden-forest target–oracle assumptions of Theorem 3.1. Fu et al. (2025) lower-bound rounds under a confidence-threshold restriction; our lower bound trades probes against commit rounds over the full admissible interface.

Parallel sampling guarantees. Distribution-general conditional-oracle samplers provide exact- and noisy-oracle guarantees with smaller depth exponents (Anari et al., 2024; 2026); Table 1 compares their row-query bounds in our submission units. Dependence-based schedule analyses control error through TC/DTC, effective TC, or adaptive estimates (Li & Cai, 2025; Chen et al., 2026; Dmitriev et al., 2026a; Zhao & Cai, 2026). Our class contains a path on which TC, DTC, and effective TC are all $\Theta(N)$, while the construction exploits local conditional separation. Appendices I–J give the source conditions and path calculations; these substitutions compare sufficient upper bounds, not lower bounds on those samplers.

Learning hidden structure. Structure learning uses subset queries (Angluin & Chen, 2008; Abasi & Bshouty, 2019), samples (Chow & Liu, 1968; Dasarathy et al., 2016), or Gaussian covariance entries (Lugosi et al., 2021). Tree Ising laws can be learned in global TV without exact tree recovery (Daskalakis & Pan, 2021). Our conditional-row probes yield local certificates and charge discovery to sampling; Appendix J.3 gives further historical context.

6 Finite-size experiments

We evaluate four hidden forest families with an exact frozen conditional oracle at five sizes from $N = 8192$ to 16384. Figure 3 compares total oracle submissions under the fixed empirical batch-KL criterion $K + 2\,\mathrm{SE}_{\mathrm{MC}} \le 10^{-10} L_0$, where K sums product-to-joint batch KL along the sampled history, not the target-to-output risk. The family-specific reference L_0 is the one-batch KL cost at $N = 2048$ (Appendix K). We use empirical tuning; the targets and parameters need not satisfy the theorem's sufficient conditions. Our sampler's submission counts are consistent with sublinear scaling over the tested finite-size range. These exact-oracle experiments isolate submission cost; learned-model accuracy and runtime are outside their scope.

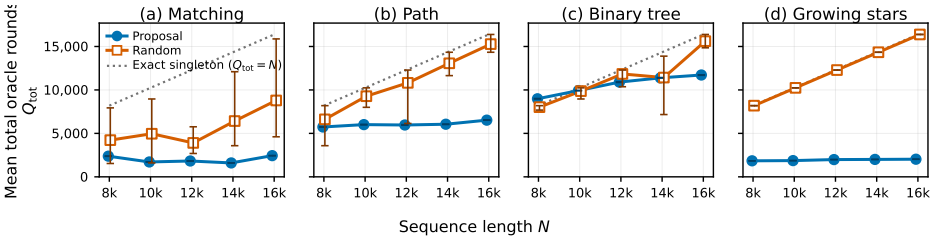


Figure 3: Total oracle submissions $Q_{\mathrm{tot}} = Q_{\mathrm{pre}} + Q_{\mathrm{cf}} + R$, including preprocessing, probes, and nonempty commit rounds. Points: six-run means; whiskers: min–max ranges (not standard errors or confidence intervals). Proposal settings are fixed per forest and N; random budgets are selected per run on coupled curves. Proposal ranges (≤ 78 rounds) may lie inside markers. Dotted line: exact singleton $Q_{\mathrm{tot}} = N$, not a cap. Horizontal ticks: $1\mathrm{k} = 1024$.

7 Conclusion

We establish sublinear parallel unmasking under hidden-forest assumptions, including dependence-discovery costs. The main practical challenge is to turn shared probing into computational savings with learned denoisers. Open directions include closing the probe–commit tradeoff gap for irreversible sampling, understanding whether remasking improves this tradeoff when resampling costs are counted, and extending the guarantees beyond forests.

AI USE STATEMENT

Generative AI tools were used extensively during this work, not only for language editing. GPT-5.6 Sol and GPT-6 Astra were used throughout the research and writing process. Claude Fable 5.1 and Claude Opus 5.5 were additionally used for writing assistance and the generation of Figures 1 and 2. Iterative brainstorming and technical discussions with the GPT models contributed to the development of central proof ideas. The GPT models also assisted in developing and refining mathematical arguments, making proof steps explicit and rigorous, and drafting and revising the proofs. AI tools also assisted with experimental design. The GPT models were also used to write code for the numerical experiments. Further assistance included literature search, translation, and the preparation of scientific figures, as well as manuscript organization, expository revision, and LaTeX preparation. The authors verified the proofs and cross-checked the cited sources. The authors take responsibility for the final content of this work, including its mathematical claims, proofs, citations, and AI-assisted text.

REFERENCES

Hasan Abasi and Nader H. Bshouty. On learning graphs with edge-detecting queries. In *Proceedings of the 30th International Conference on Algorithmic Learning Theory*, volume 98 of *Proceedings of Machine Learning Research*, pp. 3–30. PMLR, 2019. URL https://proceedings.mlr.press/v98/abasi19a.html.

Nima Anari, Ruiquan Gao, and Aviad Rubinstein. Parallel sampling via counting. In *Proceedings of the 56th Annual ACM Symposium on Theory of Computing*, pp. 537–548. Association for Computing Machinery, 2024. doi: 10.1145/3618260.3649744. URL https://arxiv.org/abs/2408.09442.

Nima Anari, Carlo Baronio, CJ Chen, Alireza Haqi, Frederic Koehler, Anqi Li, and Thuy-Duong Vuong. Parallel sampling via autospeculation. In *Proceedings of the 58th Annual ACM Symposium on Theory of Computing*, pp. 1168–1179. Association for Computing Machinery, 2026. doi: 10.1145/3798129.3800828. URL https://arxiv.org/abs/2511.07869.

Dana Angluin and Jiang Chen. Learning a hidden graph using $O(\log n)$ queries per edge. *Journal of Computer and System Sciences*, 74(4):546–556, 2008. doi: 10.1016/j.jcss.2007.06.006. URL https://doi.org/10.1016/j.jcss.2007.06.006.

Jacob Austin, Daniel D. Johnson, Jonathan Ho, Daniel Tarlow, and Rianne van den Berg. Structured denoising diffusion models in discrete state-spaces. In *Advances in Neural Information Processing Systems*, volume 34, pp. 17981–17993, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/958c530554f78bcd8e97125b70e6973d-Abstract.html.

Iskander Azangulov, Teodora Pandeva, Niranjani Prasad, Javier Zazo, and Sushrut Karmalkar. Parallel sampling from masked diffusion models via conditional independence testing. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=XjcHRIu0iF.

Vansh Bansal, Cholyeon Cho, Syamantak Kumar, Sujay Sanghavi, and Purnamrita Sarkar. Understanding parallel samplers in masked diffusion via random walks on graphs. *arXiv preprint arXiv:2606.22976*, 2026. URL https://arxiv.org/abs/2606.22976.

Florent Becker, Martín Matamala, Nicolas Nisse, Ivan Rapaport, Karol Suchan, and Ioan Todinca. Adding a referee to an interconnection network: What can(not) be computed in one round. *arXiv preprint arXiv:1009.4447*, 2010. URL https://arxiv.org/abs/1009.4447v2.

Heli Ben-Hamu, Itai Gat, Daniel Severo, Niklas S. Nolte, and Brian Karrer. Accelerated sampling from masked diffusion models via entropy bounded unmasking. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/510e22f4a2da5212a64f1591736e2eaf-Abstract-Conference.html.

Arnab Bhattacharyya, Sutanu Gayen, Eric Price, Vincent Y. F. Tan, and N. V. Vinodchandran. Near-optimal learning of tree-structured distributions by Chow and Liu. *SIAM Journal on Computing*, 52(3):761–793, 2023. doi: 10.1137/22M1489678. URL https://epubs.siam.org/doi/10.1137/22M1489678.

Antonio Blanca, Zongchen Chen, Daniel Štefankovič, and Eric Vigoda. Complexity of high-dimensional identity testing with coordinate conditional sampling. In *Proceedings of the Thirty-Sixth Conference on Learning Theory*, volume 195 of *Proceedings of Machine Learning Research*, pp. 1774–1790. PMLR, 2023. URL https://proceedings.mlr.press/v195/blanca23a.html.

Guy Bresler and Mina Karzand. Learning a tree-structured Ising model in order to make predictions. *The Annals of Statistics*, 48(2):713–737, 2020. doi: 10.1214/19-AOS1808. URL https://arxiv.org/abs/1604.06749.

Nader H. Bshouty and Areej Costa. Exact learning of juntas from membership queries. *Theoretical Computer Science*, 742:82–97, 2018. doi: 10.1016/j.tcs.2017.12.032. URL https://arxiv.org/abs/1706.06934v1.

Changxiao Cai and Gen Li. Confidence-based decoding is provably efficient for diffusion language models. *arXiv preprint arXiv:2603.22248*, 2026. URL https://arxiv.org/abs/2603.22248.

Andrew Campbell, Joe Benton, Valentin De Bortoli, Thomas Rainforth, George Deligiannidis, and Arnaud Doucet. A continuous time framework for discrete denoising models. In *Advances in Neural Information Processing Systems*, volume 35, pp. 28266–28279, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/b5b528767aa35f5b1a60fe0aaeca0563-Abstract-Conference.html.

Clément L. Canonne, Dana Ron, and Rocco A. Servedio. Testing probability distributions using conditional samples. *SIAM Journal on Computing*, 44(3):540–616, 2015. doi: 10.1137/130945508. URL https://doi.org/10.1137/130945508.

Clément L. Canonne, Xi Chen, Gautam Kamath, Amit Levi, and Erik Waingarten. Random restrictions of high-dimensional distributions and uniformity testing with subcube conditioning. In *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms*, pp. 321–336. SIAM, 2021. doi: 10.1137/1.9781611976465.21. URL https://arxiv.org/abs/1911.07357.

Huiwen Chang, Han Zhang, Lu Jiang, Ce Liu, and William T. Freeman. MaskGIT: Masked generative image transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11315–11325, 2022. URL https://openaccess.thecvf.com/content/CVPR2022/html/Chang_MaskGIT_Masked_Generative_Image_Transformer_CVPR_2022_paper.html.

Haoxuan Chen, Yinuo Ren, Lexing Ying, and Grant M. Rotskoff. Accelerating diffusion models with parallel sampling: Inference at sub-linear time complexity. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/f162fa05675e3db4a733aafc081653cf-Abstract-Conference.html.

Sitan Chen, Kevin Cong, and Jerry Li. Optimal inference schedules for masked diffusion models. In *Proceedings of the Thirty-Ninth Conference on Learning Theory*, volume 336 of *Proceedings of Machine Learning Research*, pp. 1279–1311. PMLR, 2026. URL https://proceedings.mlr.press/v336/chen26e.html.

Xi Chen, Rajesh Jayaram, Amit Levi, and Erik Waingarten. Learning and testing junta distributions with subcube conditioning. In *Proceedings of the 34th Conference on Learning Theory*, volume 134 of *Proceedings of Machine Learning Research*, pp. 1060–1113. PMLR, 2021. URL https://proceedings.mlr.press/v134/chen21b.html.

C. K. Chow and C. N. Liu. Approximating discrete probability distributions with dependence trees. *IEEE Transactions on Information Theory*, 14(3):462–467, 1968. doi: 10.1109/TIT.1968.1054142. URL https://doi.org/10.1109/TIT.1968.1054142.

Thomas F. Coleman and Jorge J. Moré. Estimation of sparse Jacobian matrices and graph coloring problems. *SIAM Journal on Numerical Analysis*, 20(1):187–209, 1983. doi: 10.1137/0720013. URL https://epubs.siam.org/doi/10.1137/0720013.

Gautam Dasarathy, Aarti Singh, Maria-Florina Balcan, and Jong H. Park. Active learning algorithms for graphical model selection. In *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, volume 51 of *Proceedings of Machine Learning Research*, pp. 1356–1364. PMLR, 2016. URL https://proceedings.mlr.press/v51/dasarathy16.html.

Constantinos Daskalakis and Qinxuan Pan. Sample-optimal and efficient learning of tree Ising models. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, 2021. doi: 10.1145/3406325.3451006. URL https://arxiv.org/abs/2010.14864v2.

Daniil Dmitriev, Zhihan Huang, and Yuting Wei. Efficient sampling with discrete diffusion models: Sharp and adaptive guarantees. In *Proceedings of the Thirty-Ninth Conference on Learning Theory*, volume 336 of *Proceedings of Machine Learning Research*, pp. 2038–2104. PMLR, 2026a. URL https://proceedings.mlr.press/v336/dmitriev26a.html.

Daniil Dmitriev, Zhihan Huang, and Yuting Wei. Provably adaptive sampling with uniform and remasking discrete diffusion models. *arXiv preprint arXiv:2608.23554*, 2026b. URL https://arxiv.org/abs/2608.23554.

Guhao Feng, Yihan Geng, Jian Guan, Wei Wu, Liwei Wang, and Di He. Theoretical benefit and limitation of diffusion language model. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/2318d75a06437eaa257737a5cf3ab83c-Abstract-Conference.html.

Weiming Feng, Yuxin Sun, and Yitong Yin. What can be sampled locally? *Distributed Computing*, 33:227–253, 2020. doi: 10.1007/s00446-018-0332-8. URL https://doi.org/10.1007/s00446-018-0332-8.

Hengyu Fu, Baihe Huang, Virginia Adams, Charles Wang, Venkat Srinivasan, and Jiantao Jiao. From bits to rounds: Parallel decoding with exploration for diffusion language models. *arXiv preprint arXiv:2511.21103*, 2025. URL https://arxiv.org/abs/2511.21103.

Noah Golowich, Ankur Moitra, and Dhruv Rohatgi. The power of test-time training for approximate sampling. *arXiv preprint arXiv:2606.11437*, 2026. URL https://arxiv.org/abs/2606.11437.

Satoshi Hayakawa, Yuhta Takida, Masaaki Imaizumi, Hiromi Wakaki, and Yuki Mitsufuji. Distillation of discrete diffusion through dimensional correlations. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 22259–22297. PMLR, 2025. URL https://proceedings.mlr.press/v267/hayakawa25a.html.

Satoshi Hayakawa, Yuhta Takida, Masaaki Imaizumi, Hiromi Wakaki, and Yuki Mitsufuji. Demystifying MaskGIT sampler and beyond: Adaptive order selection in masked diffusion. *Transactions on Machine Learning Research*, 2026. URL https://openreview.net/forum?id=mKlW68i2Ig.

Zhengfu He, Tianxiang Sun, Qiong Tang, Kuanning Wang, Xuanjing Huang, and Xipeng Qiu. DiffusionBERT: Improving generative masked language models with diffusion models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 4521–4534. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.acl-long.248. URL https://aclanthology.org/2023.acl-long.248/.

David Heckerman, David Maxwell Chickering, Christopher Meek, Robert Rounthwaite, and Carl Kadie. Dependency networks for inference, collaborative filtering, and data visualization. *Journal of Machine Learning Research*, 1:49–75, 2000. URL https://jmlr.org/papers/v1/heckerman00a.html.

Jonathan Ho, Ajay N. Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 33, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/4c5bcfec8584af0d967f1ab10179ca4b-Abstract.html.

Ananth V. Iyer, H. Donald Ratliff, and G. Vijayan. Optimal node ranking of trees. *Information Processing Letters*, 28(5):225–229, 1988. doi: 10.1016/0020-0190(88)90194-9. URL https://research.ibm.com/publications/optimal-node-ranking-of-trees.

Haozhe Jiang, Nika Haghtalab, and Lijie Chen. Diffusion language models are provably optimal parallel samplers. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/2f992baabfee60ab6add5e4f63b414b1-Abstract-Conference.html.

Jaeyeon Kim, Kulin Shah, Vasilis Kontonis, Sham M. Kakade, and Sitan Chen. Train for the worst, plan for the best: Understanding token ordering in masked diffusions. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 30749–30768. PMLR, 2025. URL https://proceedings.mlr.press/v267/kim25ah.html.

Hugo Lavenant and Giacomo Zanella. Error bounds and optimal schedules for masked diffusions with factorized approximations. *arXiv preprint arXiv:2510.25544*, 2025. URL https://arxiv.org/abs/2510.25544.

Sanghyun Lee, Seungryong Kim, Jongho Park, and Dongmin Park. Lookahead unmasking elicits accurate decoding in diffusion language models. *arXiv preprint arXiv:2511.05563*, 2025. URL https://arxiv.org/abs/2511.05563v1.

Jose Lezama, Tim Salimans, Lu Jiang, Huiwen Chang, Jonathan Ho, and Irfan Essa. Discrete predictor-corrector diffusion models for image synthesis. In *Proceedings of the 11th International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=VM8batVBWvg.

Gen Li and Changxiao Cai. Breaking AR's sampling bottleneck: Provable acceleration via diffusion language models. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/111298628bfb153ee1f84b10fec3a8b9-Abstract-Conference.html.

Xiang Li, John Thickstun, Ishaan Gulrajani, Percy Liang, and Tatsunori B. Hashimoto. Diffusion-LM improves controllable text generation. In *Advances in Neural Information Processing Systems*, volume 35, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/1be5bc25d50895ee656b8c2d9eb89d6a-Abstract.html.

Yuchen Li, Alexandre Kirchmeyer, Aashay Mehta, Yilong Qin, Boris Dadachev, Kishore Papineni, Sanjiv Kumar, and Andrej Risteski. Promises and pitfalls of generative masked language modeling: Theoretical framework and practical guidelines. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 27969–28017. PMLR, 2024. URL https://proceedings.mlr.press/v235/li24af.html.

Yuchen Liang, Renxiang Huang, Lifeng Lai, Ness Shroff, and Yingbin Liang. Absorb and converge: Provable convergence guarantee for absorbing discrete diffusion models. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-0685. URL https://papers.neurips.cc/paper_files/paper/2025/hash/1d571ea833394630c1dec71664f16cd9-Abstract-Conference.html.

Yuchen Liang, Zhiheng Tan, Ness Shroff, and Yingbin Liang. Sharp convergence rates for masked diffusion models. *arXiv preprint arXiv:2602.22505*, 2026. URL https://arxiv.org/abs/2602.22505.

Anji Liu, Oliver Broadrick, Mathias Niepert, and Guy Van den Broeck. Discrete copula diffusion. In *Proceedings of the 13th International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/dd1fef536655685898a6602bfbf16857-Abstract-Conference.html.

Han Liu, Min Xu, Haijie Gu, Anupam Gupta, John Lafferty, and Larry Wasserman. Forest density estimation. *Journal of Machine Learning Research*, 12(25):907–951, 2011. URL https://jmlr.org/papers/v12/liu11a.html.

Gábor Lugosi, Jakub Truszkowski, Vasiliki Velona, and Piotr Zwiernik. Learning partial correlation graphs and graphical models by covariance queries. *Journal of Machine Learning Research*, 22 (203):1–41, 2021. URL https://jmlr.org/papers/v22/20-1137.html.

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. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/48b383b24230e0e6e649d9c98dae4d8c-Abstract-Conference.html.

Jingyang Ou, Shen Nie, Kaiwen Xue, Fengqi Zhu, Jiacheng Sun, Zhenguo Li, and Chongxuan Li. Your absorbing discrete diffusion secretly models the conditional distributions of clean data. In *Proceedings of the 13th International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/a365e37c18fb91af547a2f0012a89e98-Abstract-Conference.html.

Yong-Hyun Park, Chieh-Hsin Lai, Satoshi Hayakawa, Yuhta Takida, and Yuki Mitsufuji. Jump your steps: Optimizing sampling schedule of discrete diffusion models. In *Proceedings of the 13th International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/f00aa3e7878d109618f5edeb36e2c6d3-Abstract-Conference.html.

Yury Polyanskiy and Yihong Wu. *Information Theory: From Coding to Learning*. Cambridge University Press, 2025. doi: 10.1017/9781108966351. URL https://people.lids.mit.edu/yp/homepage/data/itbook-export.pdf.

Yinuo Ren, Haoxuan Chen, Grant M. Rotskoff, and Lexing Ying. How discrete and continuous diffusion meet: Comprehensive analysis of discrete diffusion models via a stochastic integral framework. In *Proceedings of the 13th International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/6a9305d8e1dc254308a2c2e918108007-Abstract-Conference.html.

Subham Sekhar Sahoo, Marianne Arriola, Yair Schiff, Aaron Gokaslan, Edgar Marroquin, Justin T. Chiu, Alexander Rush, and Volodymyr Kuleshov. Simple and effective masked diffusion language models. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/eb0b13cc515724ab8015bc978fdde0ad-Abstract-Conference.html.

Andy Shih, Suneel Belkhale, Stefano Ermon, Dorsa Sadigh, and Nima Anari. Parallel sampling of diffusion models. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/0d1986a61e30e5fa408c81216a616e20-Abstract-Conference.html.

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 2256–2265. PMLR, 2015. URL https://proceedings.mlr.press/v37/sohl-dickstein15.html.

Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021. URL https://arxiv.org/abs/2011.13456.

Vincent Y. F. Tan, Animashree Anandkumar, and Alan S. Willsky. Learning high-dimensional Markov forest distributions: Analysis of error rates. *Journal of Machine Learning Research*, 12(45):1617–1653, 2011. URL https://jmlr.org/papers/v12/tan11a.html.

Tim van Erven and Peter Harremoës. Rényi Divergence and Kullback–Leibler Divergence. *IEEE Transactions on Information Theory*, 60(7):3797–3820, 2014. doi: 10.1109/TIT.2014.2320500. URL https://arxiv.org/abs/1206.2459.

Martin J. Wainwright. The data geometry of masking diffusion: Certified-optimal schedules via unmasking growth complexity. *arXiv preprint arXiv:2608.13520*, 2026. URL https://arxiv.org/abs/2608.13520.

Shintaro Wakasugi and Taiji Suzuki. State size independent statistical error bound for discrete diffusion models. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-4637. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/caabce23491e5418f4c9ea348359fcd2-Abstract-Conference.html.

Yu Yao, Huanjian Zhou, Andi Han, Wei Huang, and Masashi Sugiyama. Accelerating discrete diffusion models with parallel-in-time sampling. *arXiv preprint arXiv:2607.00773*, 2026. URL https://arxiv.org/abs/2607.00773.

Shaorong Zhang, Longxuan Yu, Rob Brekelmans, Luhan Tang, Salman Asif, and Greg Ver Steeg. Generation order and parallel decoding in masked diffusion models: An information-theoretic perspective. *arXiv preprint arXiv:2602.00286*, 2026. URL https://arxiv.org/abs/2602.00286.

Zikun Zhang, Zixiang Chen, and Quanquan Gu. Convergence of score-based discrete diffusion models: A discrete-time analysis. In *Proceedings of the 13th International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/560b4a3ad23137d641258d8807de13e9-Abstract-Conference.html.

Yunxiao Zhao and Changxiao Cai. Adaptation to intrinsic dependence in diffusion language models. *arXiv preprint arXiv:2602.20126*, 2026. URL https://arxiv.org/abs/2602.20126.

Kaiwen Zheng, Yongxin Chen, Hanzi Mao, Ming-Yu Liu, Jun Zhu, and Qinsheng Zhang. Masked diffusion models are secretly time-agnostic masked models and exploit inaccurate categorical sampling. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/9e3b203e72c4e058de26d02a92a81844-Abstract-Conference.html.

A Full problem setup

We use the size convention of Section 2: N is fixed within each finite statement, and size dependence is suppressed, including in P, V, ω and the accuracy parameters. In rate statements, parameters declared constant are independent of N; every public sequence and its deterministic remainder is fixed uniformly over the class before any hidden target or oracle is chosen.

Reading route to the main claims. The formal setup below fixes the objects used by all proofs. The following routes give the dependencies of the main claims:

- *Lower bound in case (i), Theorem 3.2.* Appendix B defines the matching family and states its finite lower bound. Appendix C proves the bound and verifies class inclusion in Proposition C.2. Then the adjacent Corollaries G.2 and G.3 in Appendix G give the finite calibration and the fixed-accuracy deduction of the main theorem.
- *Upper bounds in both cases, Theorem 3.1.* Appendix D specifies the executable sampler. Appendix E proves the shared screen, peeling, and resource bounds, and the TV guarantee in case (i) (Theorem E.1). Appendix F proves the forward-KL guarantee in case (ii) (Theorem F.3), using Lemma F.2. Lemma G.1 and Corollary G.3 verify the finite calibration and derive Theorem 3.1 with accuracy fixed before $N \to \infty$.
- *Comparison in Table 1.* Appendix I proves the row-to-state accounting and the path's TC/DTC scales; Appendix J records the assumptions of the source results being substituted.

- *Finite-size experiments, Section 6.* Appendix K gives the targets, accuracy test, tuning, resource accounting, and evaluation protocol.

The elementary auxiliary facts in Appendix H are referenced at their point of use. That appendix also collects the supplementary selector-invariance result.

Short intuition paragraphs precede the main proof steps. There, $\simeq$ compares scales up to fixed positive multiplicative constants in the stated regime. An upper bound alone is written with $\lesssim$ or an explicit inequality, and subpolynomial factors are displayed when relevant. All formal statements retain their finite constants and domains.

Remark (Supplementary context). Unnumbered remarks give supplementary context beyond the proof route above. Assumptions, resource definitions, conditioning arguments, and restrictions needed to apply a result are stated in the main exposition.

For each $N \geq 3$, let $[N] := \{1, \dots, N\}$ be the set of positions and let $\mathcal{V}$ be a finite vocabulary of size $V \geq 2$, equipped with a public total order $\prec$. For a finite set $\mathcal{A}$, write $\Delta(\mathcal{A})$ for its probability simplex and $\Delta^\circ(\mathcal{A})$ for the strictly positive part. For $p, q \in \Delta(\mathcal{A})$, we use the normalizations

$$d_{\mathrm{TV}}(p,q) := \frac{1}{2} \sum_{a \in \mathcal{A}} |p(a) - q(a)|, \qquad h^2(p,q) := 1 - \sum_{a \in \mathcal{A}} \sqrt{p(a)q(a)}. \tag{9}$$

For $q \in \Delta^\circ(\mathcal{A})$, also write

$$\chi^2(p\|q) := \sum_{a \in \mathcal{A}} \frac{(p(a) - q(a))^2}{q(a)}. \tag{10}$$

Also define $D_{\mathrm{KL}}(p\|q) := \sum_a p(a) \log(p(a)/q(a))$, with natural logarithms and the convention $0 \log(0/q(a)) = 0$. For $x \in \mathbb{R}$, let $[x]_+ := \max\{x, 0\}$.

All quantities explicitly declared public below, together with the algorithm and the law of its internal random seed, are fixed before the target instance is chosen. The hidden target law and, subsequently, the frozen oracle may be chosen adversarially. Local vocabulary banks $\widehat{\mathcal{B}}_i$, tail representatives, drafts, and graph certificates are constructed from oracle replies by the algorithm.

A.1 Target law and hidden forest

We expand Definition 2.1 and Assumption 2.2.

Definition A.1 (Forest-structured target law)**.** Let $X = (X_1, \dots, X_N) \in \mathcal{V}^N$ have a hidden strictly positive law $P \in \Delta^\circ(\mathcal{V}^N)$, and let $F = ([N], E)$ be a hidden undirected forest. The exact position marginal is

$$\pi_i := \mathcal{L}_P(X_i) \in \Delta^\circ(\mathcal{V}), \qquad i \in [N]. \tag{11}$$

For each orientation $i \to j$ of an edge $\{i, j\} \in E$, let $K_{i\to j}(\cdot \mid a) \in \Delta^\circ(\mathcal{V})$ be an endpoint kernel for every $a \in \mathcal{V}$.

Assumption A.2 (Structural forest law (S0–C0))**. Forest factorization (S0).** Fix one hidden reference root in each connected component. If $\mathrm{Roots}(F)$ is the resulting root set and $\mathrm{pa}(j)$ is the induced parent of a nonroot vertex j, assume

$$P(x) = \prod_{u \in \mathrm{Roots}(F)} \pi_u(x_u) \prod_{j \notin \mathrm{Roots}(F)} K_{\mathrm{pa}(j)\to j}(x_j \mid x_{\mathrm{pa}(j)}). \tag{12}$$

Endpoint compatibility (C0). For every $\{i, j\} \in E$ and $a, b \in \mathcal{V}$, assume

$$\pi_i(a) K_{i\to j}(b \mid a) = \pi_j(b) K_{j\to i}(a \mid b). \tag{13}$$

Moving a root across one edge replaces the left-hand side of 13 by its right-hand side and leaves every other factor unchanged. Iterating this replacement along the unique path between two roots shows that (S0)–(C0) yield the same factorization for every componentwise root choice.

To obtain these kernels from Assumption 2.2, set $K_{i\to j}(b \mid a) := P(X_j = b \mid X_i = a)$ for each oriented edge. Then (S0) is the main-text factorization, and (C0) follows from

$$\pi_i(a) K_{i\to j}(b \mid a) = P(X_i = a, X_j = b) = \pi_j(b) K_{j\to i}(a \mid b).$$

For a committed set $G \subseteq [N]$ and realized values $x_G \in \mathcal{V}^G$, define the history, uncommitted set, and residual forest by

$$H_G := (G, x_G), \qquad U_G := [N] \setminus G, \qquad F_G := F[U_G]. \tag{14}$$

Strict positivity makes every such history admissible. For any graph $H = (W, E_H)$, we write $\mathcal{N}_H(v) := \{u \in W : \{u, v\} \in E_H\}$ for its neighborhood and $d_H(v) := |\mathcal{N}_H(v)|$ for its degree. The induced graph $H[S]$, for $S \subseteq W$, has vertex set S and edge set $\{\{u, v\} \in E_H : u, v \in S\}$; $V(T)$ denotes the vertex set of a connected component T.

A.2 Response regularity

The response quantities below correspond to Definition 2.3; the general regularity assumptions correspond to the RT, UEN, and RF items of Assumption 2.4. For simplicity, the main text and Corollary G.3 use $\alpha = L = C = 1$.

Definition A.3 (Marginal tails and intrinsic responses)**.** For a source position $i \in [N]$ and threshold $t \in [0, 1]$, define the exact marginal tail

$$\mathcal{T}_i(t) := \{a \in \mathcal{V} : \pi_i(a) \le t\}. \tag{15}$$

For distinct positions i, j and a complete boundary $z \in \mathcal{V}^{[N]\setminus\{i,j\}}$, define the boundary-conditioned row

$$\mu_{i\to j}(\cdot \mid a, z) := \mathcal{L}_P\big(X_j \mid X_i = a, X_{[N]\setminus\{i,j\}} = z\big)\,. \tag{16}$$

For an edge $e = \{u, v\} \in E$ and such a complete boundary, define the directed intrinsic response

$$\Omega_{v\to u}(z) := \max_{a,b\in\mathcal{V}} d_{\mathrm{TV}}\big(\mu_{v\to u}(\cdot \mid a, z), \mu_{v\to u}(\cdot \mid b, z)\big). \tag{17}$$

Assumption A.4 (Response regularity (RT–UEN))**.** Let $\alpha > 0$, $L \in [0, \infty)$, and $\omega \in (0, 1]$ be public.

Power tail-response continuity (RT). For all distinct $i, u \in [N]$, every $t \in [0, 1]$, and every complete assignment $z \in \mathcal{V}^{[N]\setminus\{i,u\}}$, assume

$$\max_{a,b\in\mathcal{T}_i(t)} d_{\mathrm{TV}}\big(\mu_{i\to u}(\cdot \mid a, z), \mu_{i\to u}(\cdot \mid b, z)\big) \le Lt^{\alpha}. \tag{18}$$

The maximum is defined to be zero when the tail is empty or a singleton.

Two-sided edge nondegeneracy (UEN). For every edge $e = \{u, v\} \in E$ and every complete boundary z, assume

$$\min\big\{\Omega_{v\to u}(z), \Omega_{u\to v}(z)\big\} \ge \omega. \tag{19}$$

Only the scale ω is public; the forest, edge direction, boundary, and maximizing token pair remain hidden.

Assumption A.5 (Asymptotic vocabulary and frequency envelope (RF))**.** For asymptotic rate statements, let $\nu > 0$, $s > 1$, and $C \ge 1$ be public constants, and assume $V = N^{\nu+o(1)}$. Order the vocabulary at each position by exact marginal mass,

$$\pi_i(a_{i,1}) \ge \cdots \ge \pi_i(a_{i,V}), \tag{20}$$

with ties broken by $\prec$, and assume uniformly in $i \in [N]$ and $k \in [V]$ that

$$\pi_i(a_{i,k}) \le C(k^{-s} + V^{-1}). \tag{21}$$

No lower Zipf bound or adjacent-rank separation is assumed. This assumption is used only to turn finite bounds into uniform asymptotic rates; it is not needed for finite correctness once threshold feasibility and the realized bank sizes are fixed.

A.3 FROZEN CONDITIONAL ORACLE

The two cases of Assumption 2.7 share the frozen oracle interface below, with different accuracy conditions and output objectives:

- *Case (i) (Hellinger/TV).* The uniform squared-Hellinger condition is (A2) below; the output objective is the seed-averaged TV risk in Definition A.10. The normalized Hellinger radius ε^{H} remains independent of the output-TV tolerance ε.
- *Case (ii) (forward KL).* Definition F.1 defines the uniform forward row-KL condition and the seed-averaged forward-KL output objective.

The main text uses $\varepsilon^{\mathrm{H}} = \varepsilon/2$ in case (i) and $\varepsilon^{\mathrm{KL}} = \varepsilon^2/2$ in case (ii), with the latter calibrated by Corollary F.4. Corollary G.3 gives both main-text specializations.

Definition A.6 (Frozen conditional oracle)**.** As in Definition 2.5, let $\mathsf{MASK} \notin \mathcal{V}$. For a masked state $y \in (\mathcal{V} \cup \{\mathsf{MASK}\})^N$, define

$$M(y) := \{i : y_i = \mathsf{MASK}\}, \qquad O(y) := [N] \setminus M(y). \tag{22}$$

For $j \in M(y)$, the hidden exact singleton conditional is

$$\mu_j(a \mid y) := P\big(X_j = a \mid X_{O(y)} = y_{O(y)}\big), \qquad a \in \mathcal{V}. \tag{23}$$

The all-mask state is $y^\perp := (\mathsf{MASK}, \ldots, \mathsf{MASK})$, so $\mu_i(\cdot \mid y^\perp) = \pi_i$.

The oracle is one deterministic frozen map

$$q : \{(y, j) : y \in (\mathcal{V} \cup \{\mathsf{MASK}\})^N,\ j \in M(y)\} \longrightarrow \Delta^\circ(\mathcal{V}). \tag{24}$$

Equivalently, write $q = \{q_j\}_{j \in [N]}$, with $q(y, j) = q_j(\cdot \mid y)$ and $q_j(a \mid y)$ its probability mass at token a.

One submission, a family of probability vectors. One oracle evaluation submits a single masked state y and returns full vocabulary probability vectors at the requested masked positions:

$$(y, S) \xrightarrow{\text{one masked-state submission}} \big\{q_j(\cdot \mid y)\big\}_{j \in S}, \qquad S \subseteq M(y).$$

Here S selects the readouts; each returned row is the whole vector $q_j(\cdot \mid y) = (q_j(a \mid y))_{a \in \mathcal{V}}$. The charge is one submission, regardless of $|S|$ or how many tests use the returned probabilities. Every occurrence of the same (y, j) returns the same row, across readout sets and operations; repeated submissions still count as separate evaluations. Probes and commits in Definition A.9 use this same interface and the same frozen rows.

Assumption A.7 (Case (i): frozen-oracle accuracy (A2))**.** For a public $\varepsilon^{\mathrm{H}} \in [0, 1]$, assume uniformly over all valid (y, j) that

$$h^2\big(\mu_j(\cdot \mid y), q_j(\cdot \mid y)\big) \le \frac{(\varepsilon^{\mathrm{H}})^2}{2N}. \tag{25}$$

Oracle errors may be adversarially correlated across states, positions, and calls; no independence, unbiasedness, concentration, or fresh-noise condition is imposed. Let $\mathcal{Q}^{\mathrm{Hel}}(P; \varepsilon^{\mathrm{H}})$ denote the class of frozen maps satisfying (A2).

Remark A.8 (Oracle metric calibration). The parameter ε^{H} is a normalized oracle budget: the per-row Hellinger distance in (A2) is at most $\varepsilon^{\mathrm{H}}/\sqrt{2N}$, not ε^{H}. Lemma H.1 therefore gives the row-TV bound $\varepsilon^{\mathrm{H}}/\sqrt{N}$ used by the screen; this is a consequence of (A2), not an additional assumption. Conversely, that row-TV bound alone does not imply (A2). For example, for sufficiently small $e > 0$, the positive binary laws $(1 - 2e, 2e)$ and $(1 - e, e)$ have TV distance e but squared Hellinger distance at least $e(\sqrt{2} - 1)^2/2 > e^2/2$.

By Lemma H.2, either of the following uniform bounds is sufficient for (A2):

$$\sup_{(y,j)} D_{\mathrm{KL}}(\mu_j(\cdot \mid y) \| q_j(\cdot \mid y)) \le \frac{(\varepsilon^{\mathrm{H}})^2}{N} \quad \text{or} \quad \sup_{(y,j)} D_{\mathrm{KL}}(q_j(\cdot \mid y) \| \mu_j(\cdot \mid y)) \le \frac{(\varepsilon^{\mathrm{H}})^2}{N}. \tag{26}$$

These are *sufficient conditions*, not necessary bounds on the KL error of an oracle satisfying (A2): Remark H.3 gives a strictly positive counterexample to the converse. Imposing either KL condition leaves the upper theorem valid with its stated output-TV bound; it does not assert an output-KL bound for the randomized sampler.

A.4 ALGORITHMS, RISK, AND RESOURCES

Both operations below use the row family of Definition A.6. A *probe* inspects its probabilities without changing the committed history; a *commit* samples a chosen batch from these rows and permanently fixes the sampled values. The algorithm requirements and resource counts below apply to both cases. They give the full interface of Definition 2.6.

Definition A.9 (Admissible algorithms)**. Algorithm requirements.** An admissible algorithm accesses $q = \{q_j\}_{j\in[N]}$ adaptively and satisfies the following conditions.

- *Observable decisions and independent randomness.* Let W be decision-rule randomness independent of the hidden instance and of the primitive random sources used for product-commit draws. The *decision rule* (also called the controller) uses public data, previous oracle replies, committed values, and its internal state and seed to choose the next operation (probe or commit), its submitted state, and any commit batch. Once the seed is fixed, these choices are deterministic functions of the observable history. The sampled token values may depend on W through the decision rule's choices, but the primitive commit-draw randomness is separate from that seed. The forest, exact marginals, endpoint kernels, hidden ranks, edge directions, and response witnesses remain hidden.

- *No remasking and history-compatible submissions.* At a realized history $H_G = (G, x_G)$, every submitted state must agree with the committed values: $y_i = x_i$ for all $i \in G$. Committed positions stay revealed in all subsequent submissions, with the same values. Uncommitted positions in U_G may be masked or temporarily assigned vocabulary tokens for a probe.

- *Probes (counterfactual submissions).* Choose a history-compatible state y and a subset of masked positions $S \subseteq M(y)$. One submission (y, S) returns $\{q_j(\cdot \mid y)\}_{j\in S}$ together, allowing the algorithm to inspect all probabilities $\{q_j(a \mid y) : j \in S,\ a \in \mathcal{V}\}$. This operation updates the observable transcript but leaves $H_G = (G, x_G)$ unchanged; temporary assignments are not committed. Probes may be adaptive and interleaved with commits. They may also use $O(y) = G$, with every residual position masked: additional hypothetical reveals are optional.

- *Irreversible product commits.* Let t index commits only, and let G_{t-1} be the positions committed before round t, with $G_0 = \emptyset$. Choose a nonempty batch $B_t \subseteq U_{G_{t-1}}$ from the observable past and W, *before* receiving this commit's oracle reply. Submit the state

$$(y_t)_i = \begin{cases} x_i, & i \in G_{t-1}, \\ \mathsf{MASK}, & i \in U_{G_{t-1}}. \end{cases} \tag{27}$$

  One submission (y_t, B_t) returns the probability vectors $\{q_j(\cdot \mid y_t)\}_{j\in B_t}$ from the same interface. Draw $z_{B_t} \in \mathcal{V}^{B_t}$ according to

$$\bigotimes_{j\in B_t} q_j(\cdot \mid y_t), \tag{28}$$

  and permanently update $G_t := G_{t-1} \cup B_t$ and $x_{B_t} := z_{B_t}$, retaining the earlier values $x_{G_{t-1}}$. The product in equation 28 draws the batch coordinates independently conditional on the current history and selected batch. Temporary probe assignments cannot replace this draw, and the draw is not filtered by an accept–reject or verify-and-accept step.

- *Completion.* The batches are disjoint and satisfy $\bigcup_{t=1}^{R} B_t = [N]$.

Operation-indexed actions and visible history. Index the submissions actually performed, probes and commits together, by consecutive positive integers $s = 1, 2, \ldots$; repeated or parallel submissions have separate indices. This operation index is distinct from the rank-frequency exponent in equation 21. Let s_t be the operation index of commit t, and write $t_s = t$ when $s = s_t$. Fix the decision-rule seed w. The complete visible history immediately before operation s is h^{op}_{s-}, initialized by the public data and seed-fixed initial state, $h^{\mathrm{op}}_{1-} := h_{\mathrm{init}}(w)$. The selected action has the form

$$A_s = \mathsf{Next}_w(h^{\mathrm{op}}_{s-}) = \begin{cases} (\mathsf{query}, y^{\mathrm{op}}_s, J_s), & \text{probe}, \\ (\mathsf{commit}, y^{\mathrm{op}}_s, B_{t_s}), & \text{commit}. \end{cases} \tag{29}$$

Here query denotes a probe, J_s is its requested readout set (S in the probe rule), with $J_s \subseteq M(y_s^{\mathrm{op}})$. At the current committed history (G, x_G), a probe state satisfies

$$(y_s^{\mathrm{op}})_i = x_i \quad (i \in G), \qquad (y_s^{\mathrm{op}})_i \in \mathcal{V} \cup \{\mathsf{MASK}\} \quad (i \notin G).$$

The decision rule in Equation (29) selects these uncommitted entries and J_s from the visible history and fixed seed; any token assignments there are temporary and may all be omitted. At a commit, the state instead has all uncommitted positions masked:

$$y_s^{\mathrm{op}} = y_{t_s}, \qquad (y_{t_s})_i = \begin{cases} x_i, & i \in G, \\ \mathsf{MASK}, & i \notin G, \end{cases}$$

as in Equation (27), with $G = G_{t_s - 1}$.

At the current committed history (G, x_G), the vertex roles are:

Role	Probe	Commit t
Temporarily assigned positions	$O(y_s^{\mathrm{op}}) \setminus G$	$\varnothing$
Requested readouts	J_s	B_t
Newly committed positions	$\varnothing$	B_t

If z_s is the vector drawn at a commit, the ordinary observation and the visible-history update are

$$o_s^{\mathrm{vis}} := \begin{cases} (q_j(\cdot \mid y_s^{\mathrm{op}}))_{j \in J_s}, & \text{query}, \\ ((q_j(\cdot \mid y_s^{\mathrm{op}}))_{j \in B_{t_s}}, z_s), & \text{commit}, \end{cases} \tag{30}$$

$$h_{(s+1)-}^{\mathrm{op}} := (h_{s-}^{\mathrm{op}}, A_s, o_s^{\mathrm{vis}}). \tag{31}$$

The decision-rule internal state is a deterministic function of this history and w. This full-history representation does not require the implementation to store the entire transcript.

Definition A.10 (Output law and case (i) risk)**.** For an admissible algorithm from Definition A.9, let $\widehat{X}$ be its output. Conditional on $W = w$, let $\widehat{P}_w^q$ be its law after integrating all product-commit draws. Define the seed-averaged total-variation risk by

$$\mathcal{R}_{\mathrm{TV}}(\mathcal{A}; P, q) := \mathbb{E}_W d_{\mathrm{TV}}\big(P, \widehat{P}_W^q\big), \tag{32}$$

and the oracle-robust risk by

$$\overline{\mathcal{R}}_{\mathrm{TV}}(\mathcal{A}; P) := \sup_{q \in \mathcal{Q}^{\mathrm{Hel}}(P; \varepsilon^{\mathrm{H}})} \mathcal{R}_{\mathrm{TV}}(\mathcal{A}; P, q). \tag{33}$$

The expectation in equation 32 is the expectation of the conditional-output TV, rather than the TV after first mixing over W. In case (ii), the same conditional output law is evaluated by the seed-averaged forward-KL risk of Definition F.1.

Definition A.11 (Resources and all-path budgets)**.** These definitions apply to both accuracy cases. In the counts below, s ranges over the positive-integer indices of submissions actually performed, as in Equation (29); t indexes commits only. Let s_{pre} be the index of the one distinguished preprocessing all-mask readout, with $s_{\mathrm{pre}} = 0$ if it is not used. Define

$$Q_{\mathrm{cf}} := \big|\{s : s \neq s_{\mathrm{pre}}, \text{ submission } s \text{ is a probe}\}\big|,$$
$$R := \big|\{s : \text{submission } s \text{ performs a nonempty product commit}\}\big|.$$

Each probe other than that preprocessing readout contributes one to Q_{cf}; each commit contributes one to R and zero to Q_{cf}, regardless of its batch size. A probe at the commit state equation 27 still counts toward Q_{cf}: the operation performed, rather than the masked state alone, determines its counter. Every other all-mask probe also counts toward Q_{cf}. Reusing returned rows in several tests adds no submission, whereas submitting the same state again does. The actual total number of submissions is therefore $Q_{\mathrm{cf}} + R + \mathbf{1}\{s_{\mathrm{pre}} > 0\}$.

The one distinguished all-mask readout is kept separate through the accounting flag $Q_{\mathrm{marg}} \in \{0, 1\}$: set it to one when that readout is charged, and to zero when it is omitted from the reported count (or is not used). The reported *noncommit* count is

$$Q_{\mathrm{nc}} := Q_{\mathrm{cf}} + Q_{\mathrm{marg}}. \tag{34}$$

Thus Q_{nc} excludes commits. When the preprocessing readout is used and charged, the full count is $Q_{\mathrm{nc}} + R = 1 + Q_{\mathrm{cf}} + R$, as in the main construction. Setting $Q_{\mathrm{marg}} = 0$ omits only that preprocessing charge; it does not make other probes free.

Sequential depth and bandwidth. D counts sequential oracle stages. A stage may contain multiple submissions whose inputs are fixed before receiving that stage's replies; later inputs that depend on those replies require another stage. Parallel submissions therefore share depth but each retains its own submission charge. Returned-row counts, scalar bandwidth, and local computation are not included in these interaction counts; Lemma E.8 separately accounts for the construction's rows.

Deterministic all-path budgets. The budgets $\overline{Q}, \overline{R}$ satisfy $Q_{\mathrm{cf}} \le \overline{Q}$ and $R \le \overline{R}$ on every instance, seed, transcript, and realized path. A lower bound on these budgets does not assert the same lower bound on every realized counter.

Example (Submission counting)**.** One preprocessing all-mask readout, two probes of the same state, and one commit of all N positions use $Q_{\mathrm{cf}} = 2$ and $R = 1$. Charging preprocessing gives $Q_{\mathrm{marg}} = 1$, $Q_{\mathrm{nc}} = 3$, and $Q_{\mathrm{nc}} + R = 4$ evaluations. Returning several rows per probe or using each row in several tests does not change these counts.

Quantifier order. The public class data and the algorithm are fixed first. A hidden target law satisfying Assumptions A.2 and A.4, together with Assumption A.5 where asymptotic rates are claimed, is then chosen. In case (i), a single frozen oracle satisfying Assumption A.7 is chosen after the target law; the algorithm is then executed and the robust risk in equation 33 is evaluated. Case (ii) uses the same order with the oracle class and output risk of Definition F.1.

B FINITE MATCHING LOWER-BOUND STATEMENTS

This section defines the hard matching family and states its finite TV query–round lower bound and target-error tradeoff. Appendix C proves these results and verifies class membership in Proposition C.2. Appendix G then places the finite main-text calibration (Corollary G.2) immediately before the fixed-accuracy derivation (Corollary G.3) of the case (i) lower bound in Theorem 3.2. The finite upper bounds for the two cases are Theorems E.1 and F.3.

B.1 FINITE MATCHING LOWER BOUND

Definition B.1 (Finite hard matching family)**.** Let N be even, let $1 \le m < V$ be integers, and suppose

$$m\rho < 1, \qquad 0 < \eta \le \rho \le \frac{1}{2}. \tag{35}$$

Use the public vocabulary and common marginal

$$\mathcal{V} = \{a_1, \dots, a_m\} \sqcup \mathcal{B}^{\mathrm{bg}}, \qquad |\mathcal{B}^{\mathrm{bg}}| = V - m, \qquad \pi(a_c) = \rho, \qquad \pi(b) = \frac{1 - m\rho}{V - m}. \tag{36}$$

The candidate bank $\{a_1, \dots, a_m\}$ and the background set are public. An instance $I = (\mathcal{M}, \Theta)$ consists of a hidden perfect matching $\mathcal{M}$ on $[N]$ and hidden trigger indices $\Theta = (\Theta_i)_{i=1}^N \in [m]^N$. Put

$$\phi_i(a) := \mathbf{1}\{a = a_{\Theta_i}\} - \rho, \qquad \sigma_\rho^2 := \rho(1 - \rho). \tag{37}$$

For $e = \{i, j\} \in \mathcal{M}$, define

$$P_e^0(a, b) := \pi(a)\pi(b), \tag{38}$$

$$P_e^I(a, b) := \pi(a)\pi(b)\left[1 + \eta\frac{\phi_i(a)\phi_j(b)}{\sigma_\rho^2}\right], \tag{39}$$

$$P^I := \bigotimes_{e \in \mathcal{M}} P_e^I, \qquad K_{i \to j}^I(b \mid a) := \pi(b)\left[1 + \eta\frac{\phi_i(a)\phi_j(b)}{\sigma_\rho^2}\right]. \tag{40}$$

The bounds in equation 35 make these laws strictly positive: the three possible likelihood-ratio values are displayed in equation 50 and checked in Step 1 of the proof of Lemma C.1. If $\mathrm{mate}_{\mathcal{M}}(j)$

denotes the mate of j, the selected frozen oracle is, for every masked readout j and $i := \mathrm{mate}_{\mathcal{M}}(j)$,

$$q_j^I(\cdot \mid y) := \begin{cases} \pi, & i \in M(y), \\ K^I_{i\to j}(\cdot \mid y_i), & i \in O(y) \text{ and } y_i = a_{\Theta_i}, \\ \pi, & i \in O(y) \text{ and } y_i \neq a_{\Theta_i}. \end{cases} \tag{41}$$

Let $\mathfrak{I}^{\mathrm{hard}}[V, m, \rho, \eta]$ be the family obtained by ranging over all $\mathcal{M}$ and Θ. The public candidate labels are granted to the algorithm; the matching and trigger indices remain hidden.

For integers $\overline{Q}, \overline{R}$, let $\mathfrak{A}(\overline{Q}, \overline{R})$ be the admissible algorithms whose counterfactual-submission and nonempty-commit counters are at most $\overline{Q}$ and $\overline{R}$, respectively, on every instance, seed, transcript, and realized path. Define

$$\mathfrak{R}^{\mathrm{hard}}_{\mathrm{TV}}(\overline{Q}, \overline{R}; V, m, \rho, \eta) := \inf_{\mathcal{A}\in\mathfrak{A}(\overline{Q},\overline{R})} \sup_{I\in\mathfrak{I}^{\mathrm{hard}}[V,m,\rho,\eta]} \mathcal{R}_{\mathrm{TV}}(\mathcal{A}; P^I, q^I). \tag{42}$$

The theorem below uses only the finite hard-family conditions. Corollary G.2 later chooses these parameters to satisfy the case (i) oracle condition for the main lower bound.

Theorem B.2 (Nonlinear finite adaptive query–round lower bound)**.** *Assume only the basic hard-family conditions equation 35; in particular, no restriction on $N\eta^2$ is imposed. Then*

$$\begin{aligned} 0 \le \overline{Q} \le m/8, \qquad & 1 \le \overline{R} \le N/16384 \\ \Longrightarrow \quad & \mathfrak{R}^{\mathrm{hard}}_{\mathrm{TV}}(\overline{Q}, \overline{R}; V, m, \rho, \eta) \\ & \ge \frac{1}{4}\left[1 - \exp\left\{-\frac{N\eta^2}{98304\overline{R}}\right\}\right]. \end{aligned} \tag{43}$$

Corollary B.3 (Nonlinear finite target-error tradeoff)**.** *Assume equation 35, fix $\varepsilon \in (0, 1/8]$, and suppose an admissible algorithm has worst-case pathwise budgets $(\overline{Q}, \overline{R})$ and*

$$\sup_{I\in\mathfrak{I}^{\mathrm{hard}}[V,m,\rho,\eta]} \mathcal{R}_{\mathrm{TV}}(\mathcal{A}; P^I, q^I) \le \varepsilon. \tag{44}$$

Then at least one of

$$\overline{Q} > m/8, \qquad \overline{R} > N/16384, \qquad \overline{R} \ge \frac{N\eta^2}{786432\varepsilon} \tag{45}$$

holds.

Remark B.4 (Where the oracle slack enters). The selected oracle equation 41 is exact except in its third case, where a revealed nontrigger source value a at the mate i of j produces the row π instead of the exact row $K^I_{i\to j}(\cdot \mid a)$. The difference of these two rows is $\pi(b)\eta\phi_i(a)\phi_j(b)/\sigma_\rho^2$ with $\phi_i(a) = -\rho$, so by equation 47 they are at total-variation distance exactly $\eta\rho$, and the squared Hellinger bound equation 48 is the only place where Assumption A.7 is consumed. The resulting constraint $\eta^2\rho \le (\varepsilon^{\mathrm{H}})^2/(2N)$ in equation 54 yields the cubic signal calibration when $\eta = \rho = \omega$.

C Proof of the lower bound

This section proves the finite matching lower bound in Theorem B.2 and states and proves its class inclusion in Proposition C.2. The matching proof allows randomized, adaptive, value-dependent algorithms, interleaved counterfactual submissions, and unrestricted returned bandwidth. Only deterministic envelopes for the two pathwise counters in Definition A.11 are bounded. The next step toward Theorem 3.2 is in Appendix G: Corollary G.2 gives the finite calibration, immediately followed by the fixed-accuracy derivation in Corollary G.3.

C.1 Proof overview

The goal is to show that few counterfactual submissions and few commit rounds cannot both yield small sampling error. We use a hidden perfect matching: each position has one hidden trigger among m candidates, and the oracle conceals an edge's dependence until a trigger is hit. The argument has three steps.

1. At most one candidate per source per submission. A state can return many rows but tests at most one candidate at each source. Conditional uniformity bounds the probability of hitting a given source before retirement within $\overline{Q}$ submissions by $\overline{Q}/m$. Summing over positions gives $\mathbb{E}D_{\mathrm{gen}} \le N\overline{Q}/m$, where D_{gen} counts query-retired edges.

2. Few commit rounds force collisions. Every edge not retired by a query must be retired at its first endpoint commit. Few rounds therefore require large active batches. Conditional uniformity makes such batches likely to contain both endpoints of an unresolved edge: a *collision*. Edge counting and a conditional Laplace bound give $Z \gtrsim N/\overline{R}$ collisions with constant probability under the budget conditions of Theorem B.2.

3. Collisions turn hidden dependence into sampling error. At a collision, sampling independently from the correct marginals π misses the edge's dependence. The one-edge squared Hellinger loss $d_{\mathrm{edge}} \asymp \eta^2$ gives a local affinity factor $1 - d_{\mathrm{edge}}$. The midpoint identity combines these factors along adaptive paths to bound output error. Here query-disclosure and collision probabilities use the instance–midpoint law equation 64, with the decision-rule seed fixed; the final conclusion concerns the sampler's seed-averaged TV risk. The table locates the results that implement these three steps.

Proof step	Key idea and conclusion	References
Construct a hard matching family	Nontrigger oracle replies hide dependence while each edge retains loss $d_{\mathrm{edge}} \asymp \eta^2$. The inclusion calculation checks the target assumptions and oracle error under the stated calibration.	Section C.2; Lemma C.1 and Proposition C.2
Relate adaptive commits to output error	Later batches depend on sampled values. The midpoint identity expresses output affinity as an expectation of products of local affinities along these adaptive paths.	Section C.3; Lemma C.3
Limit what probes can reveal	The unresolved matching and triggers remain conditionally uniform. One candidate per source per submission gives $\mathbb{E}D_{\mathrm{gen}} \le N\overline{Q}/m$.	Section C.4; Lemmas C.4 and C.5
Force collisions with few rounds	Edge counting forces large active batches; a conditional Laplace bound then gives $Z \gtrsim N/\overline{R}$ with constant probability under the budgets in equation 46.	Sections C.5 to C.7; Lemmas C.6 and C.7 and Proposition C.8
Convert collisions into sampling error	Each collision contributes a factor $1 - d_{\mathrm{edge}}$. The midpoint identity gives a prior-averaged loss; averaging over seeds and selecting a worst-case instance yields the TV lower bound.	Sections C.8 and C.9; Lemma C.9 and Theorem B.2
Recover the main-text lower bound	Calibrate the matching witness at the public parameters, then take the fixed-accuracy limit to obtain Theorem 3.2.	Corollaries G.2 and G.3

Table 2: Proof steps for the lower bound.

Quantitative summary. For every fixed decision-rule seed w, the finite hypotheses and constants give the following chain:

$$\begin{aligned} &\overline{Q} \le m/8, \quad 1 \le \overline{R} \le N/16384 \\ \Longrightarrow \quad &\mathbb{E}D_{\mathrm{gen}} \le N/8 \\ \Longrightarrow \quad &\mathbb{P}\left(Z \ge \frac{N}{8192\overline{R}}\right) \ge \frac{1}{4} \\ \Longrightarrow \quad &\mathbb{E}_{I\sim\Pi} h^2(P^I, \widehat{P}^I_w) \ge \frac{1}{4}\left[1 - \exp\left\{-\frac{N\eta^2}{98304\overline{R}}\right\}\right]. \end{aligned} \tag{46}$$

The collision step combines the unresolved-matching invariant, the edge-counting identity, and the conditional Laplace bound. The last step uses the adaptive midpoint identity and $d_{\mathrm{edge}} \ge \eta^2/12$.

C.2 HARD-FAMILY LOCAL FACTS AND CLASS MEMBERSHIP

This subsection proves the one-edge calculations in Lemma C.1: positivity and common marginals, the edge response, the selected-oracle error, and the one-edge Hellinger loss. The same calculations then verify Proposition C.2, placing the finite hard family inside the shared target/oracle class used by both bounds.

Intuition. There are two different error scales. Removing one dependence edge costs $d_{\mathrm{edge}} \simeq \eta^2$, whereas replacing a nontrigger conditional by the neutral marginal costs at most order $\eta^2\rho$. The extra factor ρ lets the frozen oracle hide failed trigger tests while preserving a larger loss when two unresolved endpoints are committed together.

For the moment calculations, if $A \sim \pi$, then the trigger indicator at position i is Bernoulli with mean ρ. Hence

$$\mathbb{E}\phi_i(A) = 0, \qquad \mathbb{E}\phi_i(A)^2 = \sigma_\rho^2, \qquad \mathbb{E}|\phi_i(A)| = 2\sigma_\rho^2. \tag{47}$$

Indeed, the feature is $1-\rho$ with probability ρ and $-\rho$ otherwise, so

$$\begin{aligned}\mathbb{E}\phi_i &= \rho(1-\rho) - (1-\rho)\rho = 0,\\ \mathbb{E}\phi_i^2 &= \rho(1-\rho)^2 + (1-\rho)\rho^2 = \rho(1-\rho),\\ \mathbb{E}|\phi_i| &= \rho(1-\rho) + (1-\rho)\rho = 2\rho(1-\rho).\end{aligned}$$

Lemma C.1 (Local facts for the hard matching family)**.** *Under equation 35, every member of $\mathfrak{I}^{\mathrm{hard}}[V, m, \rho, \eta]$ has the following properties.*

1. **Positivity, marginals, and factorization.** *Each edge law in equation 39 is strictly positive, has both marginals equal to π, and the product law satisfies the structural forest assumptions.*
2. **Edge response.** *For either orientation $i \to j$ of a matching edge, the largest TV change in the endpoint kernel equals η.*
3. **Oracle error.** *Let j be any readout, let $i := \mathrm{mate}_{\mathcal{M}}(j)$, and let $a \in \mathcal{V} \setminus \{a_{\Theta_i}\}$ be a revealed nontrigger value at i. The selected oracle is exact outside this case, while in this case*
$$h^2\big(K^I_{i\to j}(\cdot \mid a), \pi\big) \le \frac{\eta^2\rho}{2(1-\rho)} \le \eta^2\rho. \tag{48}$$
4. **One-edge Hellinger loss.** *For every matching edge e, uniformly over its two private trigger indices, the value below depends only on (ρ, η), not on e or I:*
$$\frac{\eta^2}{12} \le d_{\mathrm{edge}} := h^2\big(P^I_e, P^0_e\big) \le \frac{\eta^2}{2}. \tag{49}$$

Proof. 1. Positivity, common marginals, and forest factorization. Fix $e = \{i, j\}$. Summing equation 39 over its second coordinate and using equation 47 gives

$$\sum_b P^I_e(a,b) = \pi(a)\left[1 + \eta\frac{\phi_i(a)}{\sigma_\rho^2}\mathbb{E}\phi_j(B)\right] = \pi(a),$$

where $B \sim \pi$. The other marginal follows symmetrically. The likelihood ratio with respect to P^0_e takes the three values

$$1 + \eta\frac{\phi_i(a)\phi_j(b)}{\sigma_\rho^2} = \begin{cases} 1 + \eta(1-\rho)/\rho, & a, b \text{ are their respective triggers},\\ 1 - \eta, & \text{exactly one of } a, b \text{ is its trigger},\\ 1 + \eta\rho/(1-\rho), & a, b \text{ are both nontriggers}. \end{cases} \tag{50}$$

The conditions $0 < \eta \le \rho \le 1/2$ place every value in $[1/2, 2]$, proving strict positivity. The marginal identity also gives the endpoint compatibility

$$\pi(a)K^I_{i\to j}(b \mid a) = P^I_e(a,b) = \pi(b)K^I_{j\to i}(a \mid b).$$

Every component is one edge, so multiplying these identities over the hidden matching proves the required forest factorization for either root choice.

2. Directed endpoint response. For source values a, a',

$$d_{\mathrm{TV}}\big(K^I_{i\to j}(\cdot \mid a), K^I_{i\to j}(\cdot \mid a')\big) = \frac{\eta|\phi_i(a)-\phi_i(a')|}{2\sigma_\rho^2}\sum_b \pi(b)|\phi_j(b)| = \eta|\phi_i(a)-\phi_i(a')|. \tag{51}$$

The feature equals $1-\rho$ at the unique trigger and $-\rho$ elsewhere. Its largest difference is therefore one, so the maximum in equation 51 is exactly η.

3. Oracle error at a revealed nontrigger. Because the target is a product over matching edges, the exact row at a readout j is π when its mate is masked and is $K^I_{i\to j}(\cdot \mid y_i)$ when the mate i is revealed. Thus equation 41 is exact in its first two cases. In the third case, $\phi_i(y_i) = -\rho$, so the likelihood ratio is

$$\frac{K^I_{i\to j}(b \mid y_i)}{\pi(b)} = 1 - \frac{\eta}{1-\rho}\phi_j(b).$$

Substitution into the definition of chi-square divergence gives

$$\chi^2\big(K^I_{i\to j}(\cdot \mid y_i)\big\|\pi\big) = \sum_b \pi(b)\left(\frac{\eta\phi_j(b)}{1-\rho}\right)^2 = \frac{\eta^2}{(1-\rho)^2}\rho(1-\rho) = \eta^2\frac{\rho}{1-\rho}. \tag{52}$$

For a likelihood ratio r, $(\sqrt{r}-1)^2 \le (r-1)^2$. The normalization in equation 9 therefore implies $h^2(p,q) \le \chi^2(p\|q)\,/2$. Combining this with equation 52 and $\rho \le 1/2$ proves equation 48.

4. One-edge squared-Hellinger loss. Finally, put

$$Z_e(a,b) := \frac{\phi_i(a)\phi_j(b)}{\sigma_\rho^2}.$$

Under P_e^0, each endpoint independently equals its trigger token with probability ρ. Thus

$$Z_e = \begin{cases} (1-\rho)/\rho, & \text{with probability } \rho^2 \quad \text{(both triggers)},\\ -1, & \text{with probability } 2\rho(1-\rho) \quad \text{(one trigger)},\\ \rho/(1-\rho), & \text{with probability } (1-\rho)^2 \quad \text{(neither trigger)}. \end{cases}$$

This law depends only on ρ, not on the edge or its trigger indices. Equation 47 and independence under P_e^0 give

$$\mathbb{E}_{P_e^0} Z_e = \frac{\mathbb{E}\phi_i\,\mathbb{E}\phi_j}{\sigma_\rho^2} = 0, \qquad \mathbb{E}_{P_e^0} Z_e^2 = \frac{\mathbb{E}\phi_i^2\,\mathbb{E}\phi_j^2}{(\sigma_\rho^2)^2} = 1.$$

Since $\mathrm{d}P_e^I/\mathrm{d}P_e^0 = 1+\eta Z_e$, the definition of squared Hellinger distance gives the first line below. For the second, multiply $\sqrt{1+\eta Z_e}-1$ by $(\sqrt{1+\eta Z_e}+1)/(\sqrt{1+\eta Z_e}+1)$:

$$d_{\mathrm{edge}} = \frac{1}{2}\mathbb{E}_{P_e^0}\big(\sqrt{1+\eta Z_e}-1\big)^2 = \frac{\eta^2}{2}\mathbb{E}_{P_e^0}\frac{Z_e^2}{(\sqrt{1+\eta Z_e}+1)^2}. \tag{53}$$

Consequently d_{edge} depends only on (ρ,η), so it is the same for every matching component and every choice of its trigger indices. By equation 50, the denominator lies between one and $(\sqrt{2}+1)^2 < 6$. Equation 49 follows. □

Intuition for class inclusion. The four checks have separate jobs: $V\rho > 1$ puts the candidate bank above the background, $\rho \le m^{-s}$ gives the frequency envelope, $\omega \le \eta$ gives the edge signal, and $\eta^2\rho \le (\varepsilon^{\mathrm{H}})^2/(2N)$ hides nontrigger replies within the oracle budget. In the main calibration, $\eta = \rho = \omega$, so the last check becomes the cubic constraint $\omega^3 \le (\varepsilon^{\mathrm{H}})^2/(2N)$.

Proposition C.2 (Finite hard-family inclusion)**.** *In addition to equation 35, let $s > 1$, use the same public vocabulary of size V, set $\alpha = L = C = 1$, and suppose*

$$V\rho > 1, \qquad \rho \le m^{-s}, \qquad \omega \le \eta, \qquad \eta^2\rho \le \frac{(\varepsilon^{\mathrm{H}})^2}{2N}. \tag{54}$$

Then every target in $\mathfrak{I}^{\mathrm{hard}}[V, m, \rho, \eta]$ satisfies the structural and response assumptions in Appendix A, as well as the finite rank–frequency inequality in (RF; Equation (21)*), and its selected oracle equation 41 belongs to $\mathcal{Q}^{\mathrm{Hel}}(P^I; \varepsilon^{\mathrm{H}})$. If the public vocabulary sequence also obeys the vocabulary-growth condition in (RF; Assumption A.5), then the full (RF; Assumption A.5) assumption holds. Under that additional asymptotic condition, this matching family is a contained subfamily of the same public problem class used by the upper theorem.*

Proof of Proposition C.2. 1. Forest structure and the edge-signal condition (UEN; Equation (19)*).* Apply Lemma C.1. Its first conclusion proves strict positivity and Assumption A.2. Complete boundary assignments outside a matching edge do not alter that edge's law, so equation 51 makes both directed intrinsic responses exactly η. The condition $\omega \le \eta$ therefore proves (UEN; Equation (19)).

2. Response-tail regularity (RT; Equation (18)*).* For (RT; Equation (18)), first consider distinct positions i, j that are not matched. Conditioning on a complete boundary fixes the mates of i and j, and the product over matching components makes $\mu_{i\to j}(\cdot \mid a, z)$ independent of a. The response diameter is therefore zero.

It remains to consider a matching edge. Fix a threshold t.

- If $t < \rho$, the marginal tail contains no candidate token and hence no trigger. Every source value in the tail has the same centered feature $-\rho$, so its response diameter is zero.
- If $t \ge \rho$, the response diameter is at most $\eta \le \rho \le t$.

Thus (RT; Equation (18)) holds with $\alpha = L = 1$.

3. Rank–frequency regularity (RF; Equation (21)*).* The inequality $V\rho > 1$ is equivalent to the background mass in equation 36 being strictly less than V^{-1}, since

$$\frac{1 - m\rho}{V - m} < \frac{1}{V} \quad \Longleftrightarrow \quad V - Vm\rho < V - m \quad \Longleftrightarrow \quad V\rho > 1.$$

Hence the m candidates occupy the first m marginal ranks. For $k \le m$,

$$\pi(a_k) = \rho \le m^{-s} \le k^{-s};$$

for $k > m$, the vocabulary-floor term V^{-1} suffices. This proves the rank–frequency inequality in (RF; Equation (21)) with $C = 1$. The separate vocabulary-growth clause in (RF; Assumption A.5) holds whenever the public sequence V has the declared asymptotic scaling.

4. Uniform oracle accuracy (A2). The only nonzero oracle error is bounded by equation 48. The final condition in equation 54 therefore places it below $(\varepsilon^{\mathrm{H}})^2/(2N)$, proving (A2). All required finite conditions hold for every hidden matching and trigger vector. Together with the stated vocabulary-growth condition, this proves the asserted inclusion in the shared asymptotic class. □

C.3 ADAPTIVE MIDPOINT COMPARISON AND INSTANCE PRIOR

We relate the target law to the sampler's output law through the discrepancies at individual commits. Three points guide the comparison:

- *Goal.* Express the output affinity, or equivalently one minus the squared-Hellinger error, in terms of local commit affinities.
- *Obstacle.* A different committed value can change subsequent probes, batches, and the stopping round. Thus local affinities depend on the realized history; the fixed-product affinity formula does not directly combine them.
- *Bridge.* Normalize the geometric mean of the two commit kernels at each history. These normalized kernels define a common path law under which the product of local affinities equals the output affinity in expectation (Lemma C.3).

Fixed objects. Fix an admissible algorithm with finite pathwise budgets $(\overline{Q}, \overline{R})$, an instance $I = (\mathcal{M}, \Theta)$ of Definition B.1, and one value w of the decision-rule seed. Put

$$\widehat{P}^I_w := \widehat{P}^{q^I}_w. \tag{55}$$

The following three executions differ in their commit kernels, defined below. The exact comparison process has terminal law P^I; Step 2 of the proof verifies this even for adaptive batches.

Execution	Commit kernel at history h	Law and role
Exact comparison	$P^I_t(\cdot \mid h)$: true joint conditional	P^I: target output law.
Product-commit sampler	$\widehat{P}^I_t(\cdot \mid h)$: product of oracle rows	$\widehat{P}^I_w$: output law being evaluated.
Adaptive midpoint	$m^I_t(\cdot \mid h)$: normalized geometric mean	$\mathbb{M}^I_w$: analytic path law for averaging local affinities.

- *Shared operations.* All three use the same seed-fixed decision rule and the same frozen oracle q^I for probes. At a common history they select the same batch $B_t(h)$.
- *Changed operation.* Only the law of the committed values changes. The resulting histories, and hence later batches, can differ between executions.

Definition (Commit kernels and adaptive midpoint law)**.** *History and batch.* Immediately before commit t, after the intervening probes, let h be the complete observable history: committed positions and values, the earlier oracle transcript, and the decision-rule state. Thus $h = h^{\mathrm{op}}_{s_t-}$ in the operation notation of Equations (29) and (31). Write $G(h)$ for the committed set, $x_{G(h)}$ for its values, and $y_t(h)$ for the commit state in equation 27. For fixed (I, w), the probes and batch selection are deterministic functions of the observed past; write $B_t(h)$ for the selected batch. The seed w is suppressed from local kernels because h includes the seed-fixed decision-rule state, but retained in full-law and terminal-round notation.

Commit kernels. For $z \in \mathcal{V}^{B_t(h)}$, define

$$P^I_t(z \mid h) := P^I\big(X_{B_t(h)} = z \mid X_{G(h)} = x_{G(h)}\big), \tag{56}$$

$$\widehat{P}^I_t(z \mid h) := \prod_{j \in B_t(h)} q^I_j(z_j \mid y_t(h)). \tag{57}$$

Local affinity and midpoint law. Define

$$a_t(I, h) := \sum_{z \in \mathcal{V}^{B_t(h)}} \sqrt{P^I_t(z \mid h)\widehat{P}^I_t(z \mid h)}, \tag{58}$$

$$m^I_t(z \mid h) := \frac{\sqrt{P^I_t(z \mid h)\widehat{P}^I_t(z \mid h)}}{a_t(I, h)}. \tag{59}$$

Let $\mathbb{M}^I_w$ be the analytic adaptive path law that runs the same decision rule and frozen-oracle probes, drawing every commit batch from $m^I_t(\cdot \mid h)$.

Why the construction is well defined.

- *Conditioning on the history.* For fixed (I, w), every reachable h can be reconstructed from $x_{G(h)}$: replay the seed-fixed decision rule from the initial state and, at each earlier commit, supply the coordinates of $x_{G(h)}$ selected by that commit. The frozen oracle reproduces the intervening replies, batches, and decision-rule states. Conversely, h records $x_{G(h)}$. Thus conditioning on this h adds no random event beyond $X_{G(h)} = x_{G(h)}$, justifying equation 56.
- *Normalization.* Strict positivity makes the denominator nonzero. Cauchy–Schwarz gives $0 < a_t(I, h) \le 1$. In particular,

$$\sum_{z \in \mathcal{V}^{B_t(h)}} m_t^I(z \mid h) = \frac{\sum_{z \in \mathcal{V}^{B_t(h)}} \sqrt{P_t^I(z \mid h)\widehat{P}_t^I(z \mid h)}}{a_t(I, h)} = \frac{a_t(I, h)}{a_t(I, h)} = 1.$$

- *Pathwise budgets.* Both commit kernels are strictly positive, so the midpoint kernel has the same support at every reachable history. It therefore traverses the same decision tree as the implemented sampler. Every midpoint path has at most $\overline{Q}$ counterfactual submissions and at most $\overline{R}$ nonempty commits.

From local affinity to output error. The defining factorization is

$$\sqrt{P_t^I(z \mid h)\widehat{P}_t^I(z \mid h)} = a_t(I, h) m_t^I(z \mid h).$$

Along a complete path, the normalized m_t^I factors supply its midpoint probability, while the $a_t(I, h)$ factors retain the local affinity losses. The next identity averages their product over the adaptive paths.

Lemma C.3 (Adaptive midpoint identity)**.** *For every fixed instance I and decision-rule seed w,*

$$1 - h^2(P^I, \widehat{P}_w^I) = \mathbb{E}_{H \sim \mathbb{M}_w^I} \prod_{t=1}^{T_w^I(H)} a_t(I, H_{t-1}), \tag{60}$$

where $T_w^I(H) \le \overline{R}$ is the terminal number of nonempty commit batches on the midpoint path. Here H_{t-1} denotes the complete observable history immediately before commit t, after all intervening counterfactual operations. The subscript counts earlier commits, not individual query operations.

The remaining lower-bound argument uses this identity at two levels:

- *Local penalty.* Lemma C.9 bounds the factors $a_t(I, h)$ using unresolved edges whose endpoints are committed together.
- *Global error.* The survivor and collision arguments count these events under the instance–midpoint law equation 64. The identity converts that count into an output-error bound.

Proof. 1. Reconstruct the unique adaptive path. Fix a complete output $x \in \mathcal{V}^N$. Starting with the empty history, replay the deterministic seed-fixed decision rule. Whenever it chooses $B_t(h_{t-1}(x))$, feed it $z_t(x) := x_{B_t(h_{t-1}(x))}$. This recursively determines the histories, batches, and terminal round $T_w^I(x)$. No remasking makes the realized batches disjoint and exhaustive. Put $G_t(x) := G_{t-1}(x) \cup B_t(h_{t-1}(x))$, with $G_0(x) = \varnothing$.

2. Factor the exact and product output masses. The exact conditional factors telescope:

$$\begin{aligned} &\prod_{t=1}^{T_w^I(x)} P_t^I(z_t(x) \mid h_{t-1}(x)) \\ &\quad = \prod_{t=1}^{T_w^I(x)} \frac{P^I(X_{G_t(x)} = x_{G_t(x)})}{P^I(X_{G_{t-1}(x)} = x_{G_{t-1}(x)})} = \frac{P^I(X_{[N]} = x)}{1} = P^I(x). \end{aligned}$$

Here the numerator at round t is the denominator at round $t+1$; the initial event has probability one and the terminal set is $[N]$. This identity and the product-commit rule therefore give

$$P^I(x) = \prod_{t=1}^{T_w^I(x)} P_t^I(z_t(x) \mid h_{t-1}(x)), \tag{61}$$

$$\widehat{P}_w^I(x) = \prod_{t=1}^{T_w^I(x)} \widehat{P}_t^I(z_t(x) \mid h_{t-1}(x)). \tag{62}$$

The batch-selection rule has no likelihood factor because it is deterministic given the displayed history.

3. Normalize the midpoint path mass. For the same terminal output, the midpoint path mass is

$$\mathbb{M}_w^I(x) := \prod_{t=1}^{T_w^I(x)} m_t^I(z_t(x) \mid h_{t-1}(x)). \tag{63}$$

To verify that this path mass sums to one, first account for paths that finish at different rounds. Nonempty batches are disjoint, so every path has at most N rounds. For this calculation only, pad each finished path to N rounds by setting $B_t(h) = \varnothing$, $\mathcal{V}^{\varnothing} = \{\varnothing\}$, and $m_t^I(\varnothing \mid h) = 1$ after termination. These extra factors leave equation 63 unchanged. Each complete output corresponds to exactly one padded path, and conversely, because the seed-fixed decision rule is deterministic and the genuine batches are disjoint and exhaustive.

Write h_{t-1} for the history determined by the preceding assignments $z_1, \dots, z_{t-1}$. For any $1 \le r \le N$, hold $z_1, \dots, z_{r-1}$ fixed. The earlier factors do not depend on z_r, so the local normalization above gives

$$\begin{aligned}
&\sum_{z_r \in \mathcal{V}^{B_r(h_{r-1})}} \prod_{t=1}^{r} m_t^I(z_t \mid h_{t-1}) \\
&\quad = \left[\prod_{t=1}^{r-1} m_t^I(z_t \mid h_{t-1})\right] \underbrace{\sum_{z_r \in \mathcal{V}^{B_r(h_{r-1})}} m_r^I(z_r \mid h_{r-1})}_{=1} \\
&\quad = \prod_{t=1}^{r-1} m_t^I(z_t \mid h_{t-1}).
\end{aligned}$$

Applying this equality for $r = N, N-1, \dots, 1$ removes the nested sums from the innermost one outwards and ends with the empty product 1:

$$\sum_{x \in \mathcal{V}^N} \mathbb{M}_w^I(x) = \sum_{z_1 \in \mathcal{V}^{B_1(h_0)}} \cdots \sum_{z_N \in \mathcal{V}^{B_N(h_{N-1})}} \prod_{t=1}^{N} m_t^I(z_t \mid h_{t-1}) = 1.$$

4. Convert the affinity sum into a midpoint expectation. Substituting $\sqrt{P_t^I \widehat{P}_t^I} = a_t(I, h) m_t^I$ into the terminal affinity sum yields

$$\begin{aligned}
&\mathrm{Aff}(P^I, \widehat{P}_w^I) \\
&\quad = \sum_{x \in \mathcal{V}^N} \sqrt{P^I(x) \widehat{P}_w^I(x)} \\
&\quad = \sum_{x \in \mathcal{V}^N} \mathbb{M}_w^I(x) \prod_{t=1}^{T_w^I(x)} a_t(I, h_{t-1}(x)) \\
&\quad = \mathbb{E}_{H \sim \mathbb{M}_w^I} \prod_{t=1}^{T_w^I(H)} a_t(I, H_{t-1}).
\end{aligned}$$

Since $\mathrm{Aff}(p, q) = 1 - h^2(p, q)$, this is equation 60. □

Instance prior for the remaining proof. Keep the seed w fixed. Draw an instance $I = (\mathcal{M}, \Theta)$ from the prior Π as follows:

- *Matching.* $\mathcal{M}$ is uniform over all perfect matchings of $[N]$.
- *Triggers.* $\Theta_1, \dots, \Theta_N$ are independent uniform elements of $[m]$, independent of $\mathcal{M}$.

All random variables used in the remaining proof are evaluated under the joint analytic law

$$I \sim \Pi, \qquad H \mid (I, w) \sim \mathbb{M}_w^I. \tag{64}$$

C.4 Unresolved edges, survivor invariant, and query retirement

We need two facts under the instance–midpoint law equation 64:

- *The distribution left unresolved.* Lemma C.4 shows that the remaining matching is uniform and its triggers are independent uniform remaining candidates. This permits the conditional collision calculation below.
- *The number of query retirements.* Lemma C.5 bounds $\mathbb{E}D_{\mathrm{gen}} \le N\overline{Q}/m$. A submission tests at most one candidate at each source, so probes can retire only a limited number of edges before commits touch them.

We first define retirement and the information used to condition these two statements. The decision-rule seed w remains fixed. Use the operation clock, actions A_s, and visible histories h_{s-}^{op} from Equations (29) and (31). For a query, write $y = y_s^{\mathrm{op}}$ and $J = J_s$. At commit t, $s = s_t$ and the batch is $B_t = B_{t_s}$.

Definition (Unresolved edges and the analytic record)**.** *The test at one source.* For any submitted state y and position $i \in [N]$, put

$$\mathsf{Test}_i(y) := \begin{cases} c, & y_i = a_c \text{ for some } c \in [m], \\ \bot, & y_i = \mathsf{MASK} \text{ or } y_i \in \mathcal{B}^{\mathrm{bg}}, \end{cases} \qquad \mathsf{Hit}_i(y) := \mathbf{1}\{\mathsf{Test}_i(y) = \Theta_i\}. \tag{65}$$

Thus $y_i = a_c$ tests the equality $\Theta_i = c$. A mask or background value tests no candidate.

Unresolved edges and vertices. Let $\mathcal{M}_s^{\mathrm{unres}}$ be the edges unresolved immediately before operation s. Initialize $\mathcal{M}_1^{\mathrm{unres}} := \mathcal{M}$ and update by

$$\mathcal{M}_{s+1}^{\mathrm{unres}} := \begin{cases} \{e \in \mathcal{M}_s^{\mathrm{unres}} : \sum_{i \in e} \mathsf{Hit}_i(y) = 0\}, & \text{query of } y, \\ \{e \in \mathcal{M}_s^{\mathrm{unres}} : e \cap B_t = \varnothing\}, & \text{commit } t. \end{cases} \tag{66}$$

Define the unresolved vertex set by

$$\mathcal{U}_s^{\mathrm{op}} := \bigcup_{e \in \mathcal{M}_s^{\mathrm{unres}}} e. \tag{67}$$

Thus a query retires an edge when at least one endpoint hits; a commit retires it when $e \cap B_t \neq \varnothing$. In either case both endpoints leave $\mathcal{U}_s^{\mathrm{op}}$. A query commits neither endpoint, and a commit fixes only those in $e \cap B_t$. Because each update removes whole edges, $\mathcal{M}_s^{\mathrm{unres}}$ is a perfect matching on $\mathcal{U}_s^{\mathrm{op}}$. The target law remains $P^I(x) = \prod_{e \in \mathcal{M}} P_e^I(x_e)$, and the frozen oracle q^I remains unchanged. In particular, an uncommitted vertex may lie on a retired edge.

Retirement record. For each newly retired edge, record its identity and both trigger indices:

$$\mathsf{Ret}_s := \left\{ \left(e, (\Theta_i)_{i \in e}\right) : e \in \mathcal{M}_s^{\mathrm{unres}} \setminus \mathcal{M}_{s+1}^{\mathrm{unres}} \right\}. \tag{68}$$

Endpoint tuples use the public position order. Let $D_{\mathrm{gen}} := \sum_{s:\, A_s \text{ is a query}} |\mathcal{M}_s^{\mathrm{unres}} \setminus \mathcal{M}_{s+1}^{\mathrm{unres}}|$. Simultaneous hits on both endpoints count as one retired edge.

Remaining candidates. Initially $\mathcal{C}_{i,1}^{\mathrm{op}} := [m]$. For each surviving $i \in \mathcal{U}_{s+1}^{\mathrm{op}}$, put

$$\mathcal{C}_{i,s+1}^{\mathrm{op}} := \begin{cases} \mathcal{C}_{i,s}^{\mathrm{op}} \setminus \{\mathsf{Test}_i(y)\}, & \text{query of } y, \\ \mathcal{C}_{i,s}^{\mathrm{op}}, & \text{commit.} \end{cases} \tag{69}$$

A surviving endpoint has hit bit zero, so any candidate tested there has failed. Subtracting $\perp$ or a previously excluded candidate changes nothing. Candidate sets are needed only while the vertex survives.

Analytic observations and history. Use the ordinary observation o_s^{vis} from Equation (30), with $q = q^I$. At a commit, $z_s = X_{B_t}$ is drawn from the midpoint kernel $m_t^I(\cdot \mid h_{s-}^{\mathrm{op}})$ of Equation (59). The additional information is:

- At a query, the entire hit-bit vector $\big(\mathsf{Hit}_i(y)\big)_{i\in\mathcal{U}_s^{\mathrm{op}}}$ from Equation (65), including bits whose mate rows were not requested. Evaluate these bits on the same pre-operation set $\mathcal{U}_s^{\mathrm{op}}$, then apply Equation (66).
- At each retirement, the tuples in Ret_s from Equation (68). Commit values X_{B_t} are already part of the ordinary observation, not additional hidden information.

Formally, initialize $\mathcal{H}_{1-}^{\mathrm{op}} := h_{1-}^{\mathrm{op}}$ and define

$$\mathsf{Obs}_s := \begin{cases} \big(o_s^{\mathrm{vis}}, (\mathsf{Hit}_i(y))_{i\in\mathcal{U}_s^{\mathrm{op}}}\big), & \text{query of } y, \\ o_s^{\mathrm{vis}}, & \text{commit}, \end{cases} \tag{70}$$

$$\mathcal{H}_{(s+1)-}^{\mathrm{op}} := (\mathcal{H}_{s-}^{\mathrm{op}}, A_s, \mathsf{Obs}_s, \mathsf{Ret}_s). \tag{71}$$

The decision rule uses only the ordinary observations in Equation (31); it does not receive hit bits or retirement disclosures. Its internal state is recoverable from that visible history. Define the analytic sigma-field and hidden survivor state by

$$\begin{aligned} \mathcal{F}_{s-}^{\mathrm{op}} &:= \sigma(\mathcal{H}_{s-}^{\mathrm{op}}, \mathcal{U}_s^{\mathrm{op}}), \\ \mathsf{S}_s^{\mathrm{op}} &:= \big(\mathcal{M}_s^{\mathrm{unres}}, (\Theta_i)_{i\in\mathcal{U}_s^{\mathrm{op}}}\big). \end{aligned} \tag{72}$$

The visible history is a projection of the analytic record, so $\sigma(h_{s-}^{\mathrm{op}}) \subseteq \mathcal{F}_{s-}^{\mathrm{op}}$. The unresolved vertices are recoverable by removing all recorded retired endpoints from $[N]$; their remaining matching and triggers are not explicitly recorded. The survivor state $\mathsf{S}_s^{\mathrm{op}}$ is not itself an entry of the record. Here *unresolved* means survival under Equation (66), not a claim about the algorithm's knowledge.

One state supplies at most one test per source, even if many returned rows are inspected. Its submission contributes one to $Q_{\mathrm{cf}} \le Q$, as in Definition A.11. A state inducing no tests still counts, except for the separately accounted distinguished preprocessing readout. Retired edges can still affect ordinary oracle replies; they are excluded only from subsequent unresolved-edge collision counts.

State immediately before a commit. At operation s_t, use the shorter pre-commit notation for the objects in Equations (67), (69), (71) and (72):

$$\begin{aligned} \mathcal{U}_t &:= \mathcal{U}_{s_t}^{\mathrm{op}}, & \mathcal{C}_{i,t} &:= \mathcal{C}_{i,s_t}^{\mathrm{op}}, \\ \mathcal{H}_{t-} &:= \mathcal{H}_{s_t-}^{\mathrm{op}}, & \mathcal{F}_{t-} &:= \mathcal{F}_{s_t-}^{\mathrm{op}}, \\ \mathsf{S}_t &:= \mathsf{S}_{s_t}^{\mathrm{op}}, & H_{t-1} &= h_{s_t-}^{\mathrm{op}}. \end{aligned} \tag{73}$$

Here H_{t-1} is the complete visible history used in Lemma C.3, after any queries preceding commit t. There may be queries before commit 1, so $\mathcal{U}_1$ need not equal $[N]$. The next operation after commit t is $s_t + 1$; intervening queries are processed before the next commit boundary s_{t+1}.

Why uniformity can survive an observation. A failed test removes one candidate. A hit or a commit instead retires the affected edge and records both its triggers. We show that, once this retired record is fixed, the probability of the complete new observation is the same for every compatible remaining matching and trigger assignment. Equal prior weights therefore give equal posterior weights.

Lemma C.4 (Unresolved-matching invariant)**.** *Under the joint law equation 64,*

$$\mathsf{S}_t \mid \mathcal{F}_{t-} \sim \mathrm{Unif}\big(\mathrm{PM}(\mathcal{U}_t)\big) \otimes \bigotimes_{i \in \mathcal{U}_t} \mathrm{Unif}(\mathcal{C}_{i,t}), \tag{74}$$

where $\mathrm{PM}(U)$ *denotes the perfect matchings of* U*. In particular, the unresolved matching is conditionally uniform, and every surviving trigger is conditionally uniform on its untested candidates. The same product law holds at every analytic operation boundary, with the current unresolved set, remaining-candidate sets, and analytic sigma-field in place of their pre-commit notation above.*

Simplified proof via conditional independence. We prove the operation-boundary statement and then specialize it to the pre-commit boundaries. We use the operation-time notation already defined in Equations (67), (69), (71) and (72), without introducing shorter names for these objects. The desired induction step has two parts:

1. after conditioning on the newly retired edges and their triggers, the new survivor state still has the required product-uniform law; and
2. the new observation is conditionally independent of that survivor state once the retirement record is fixed.

Indeed, these two statements give

$$\begin{aligned}&\mathcal{L}\big(\mathsf{S}^{\mathrm{op}}_{s+1} \,\big|\, \mathcal{F}^{\mathrm{op}}_{s-}, \mathsf{Ret}_s, \mathsf{Obs}_s\big)\\ &\qquad = \mathcal{L}\big(\mathsf{S}^{\mathrm{op}}_{s+1} \,\big|\, \mathcal{F}^{\mathrm{op}}_{s-}, \mathsf{Ret}_s\big)\,,\end{aligned}$$

and the conditioning field on the left is precisely $\mathcal{F}^{\mathrm{op}}_{(s+1)-}$, by Equations (71) and (72).

1. Initialize the induction. Before operation 1, the unresolved matching is uniform on $\mathrm{PM}([N])$, and the trigger indices are mutually independent and uniform on $[m]$, independently of the matching. Thus the required product law holds at the initial boundary.

Now assume that, conditionally on $\mathcal{F}^{\mathrm{op}}_{s-}$,

$$\mathsf{S}^{\mathrm{op}}_s \sim \mathrm{Unif}\big(\mathrm{PM}(\mathcal{U}^{\mathrm{op}}_s)\big) \otimes \bigotimes_{i \in \mathcal{U}^{\mathrm{op}}_s} \mathrm{Unif}(\mathcal{C}^{\mathrm{op}}_{i,s}).$$

The action A_s is already determined by $\mathcal{F}^{\mathrm{op}}_{s-}$, because the seed-fixed decision rule uses the visible history contained in this field.

2. Condition first on the retirement record. Fix a retirement record of positive conditional probability,

$$\mathfrak{r} = \{(e, (\theta^{\mathfrak{r}}_i)_{i \in e}) : e \in E_0\}, \qquad D_0 := \bigcup_{e \in E_0} e,$$

where E_0 is the set of edges retired in operation s and D_0 is its endpoint set. On $\{\mathsf{Ret}_s = \mathfrak{r}\}$,

$$\mathcal{U}^{\mathrm{op}}_{s+1} = \mathcal{U}^{\mathrm{op}}_s \setminus D_0.$$

For every $i \in \mathcal{U}^{\mathrm{op}}_{s+1}$, the update already defined in equation 69 reads

$$\mathcal{C}^{\mathrm{op}}_{i,s+1} = \begin{cases} \mathcal{C}^{\mathrm{op}}_{i,s} \setminus \{\mathsf{Test}_i(y)\}, & A_s \text{ is a query of } y,\\ \mathcal{C}^{\mathrm{op}}_{i,s}, & A_s \text{ is a commit.}\end{cases}$$

For an edge $e = \{i, j\} \in \mathcal{M}^{\mathrm{unres}}_s$, the update equation 66 is equivalently

$$e \in \mathcal{M}^{\mathrm{unres}}_{s+1} \quad \Longleftrightarrow \quad \begin{cases} \mathsf{Hit}_i(y) = \mathsf{Hit}_j(y) = 0, & A_s \text{ is a query of } y,\\ e \cap B_t = \varnothing, & A_s \text{ is commit } t.\end{cases}$$

In the query case, a candidate θ_i at a surviving endpoint obeys

$$\theta_i \in \mathcal{C}^{\mathrm{op}}_{i,s+1} \quad \Longleftrightarrow \quad \theta_i \in \mathcal{C}^{\mathrm{op}}_{i,s} \text{ and } \theta_i \neq \mathsf{Test}_i(y).$$

This also covers $\mathsf{Test}_i(y) = \perp$, because trigger indices lie in $[m]$. Since the fixed record has positive conditional probability, each $e \in E_0$ satisfies the complementary retirement condition: at least one endpoint hits in the query case, and $e \cap B_t \neq \varnothing$ in the commit case. Hence the complete support of the new survivor state, after fixing $\mathfrak{r}$, is

$$\mathrm{PM}(\mathcal{U}_{s+1}^{\mathrm{op}}) \times \prod_{i \in \mathcal{U}_{s+1}^{\mathrm{op}}} \mathcal{C}_{i,s+1}^{\mathrm{op}}.$$

We now verify explicitly that every element of this Cartesian support has the same conditional weight. Fix $\mathfrak{z} = \left(M', (\theta_i)_{i \in \mathcal{U}_{s+1}^{\mathrm{op}}}\right) \in \mathrm{PM}(\mathcal{U}_{s+1}^{\mathrm{op}}) \times \prod_{i \in \mathcal{U}_{s+1}^{\mathrm{op}}} \mathcal{C}_{i,s+1}^{\mathrm{op}}$. Together, $\mathfrak{z}$ and $\mathfrak{r}$ reconstruct exactly one pre-operation state: its unresolved matching is $M' \dot\cup E_0$, its trigger is θ_i on $\mathcal{U}_{s+1}^{\mathrm{op}}$, and its trigger is $\theta_i^{\mathfrak{r}}$ on D_0. Conversely, membership in the displayed support ensures that this reconstructed state produces the fixed retirement record. Since

$$\mathcal{U}_s^{\mathrm{op}} = \mathcal{U}_{s+1}^{\mathrm{op}} \dot\cup D_0,$$

these two constructions are inverse to each other. The induction hypothesis therefore gives the same joint mass to every compatible $\mathfrak{z}$:

$$\begin{aligned}
&\mathbb{P}\big(\mathsf{S}_{s+1}^{\mathrm{op}} = \mathfrak{z},\, \mathsf{Ret}_s = \mathfrak{r} \,\big|\, \mathcal{F}_{s-}^{\mathrm{op}}\big) \\
&\quad = \mathbb{P}\left(\begin{array}{l} \mathcal{M}_s^{\mathrm{unres}} = M' \dot\cup E_0, \\ \Theta_i = \theta_i \quad (i \in \mathcal{U}_{s+1}^{\mathrm{op}}), \\ \Theta_i = \theta_i^{\mathfrak{r}} \quad (i \in D_0) \end{array} \middle| \, \mathcal{F}_{s-}^{\mathrm{op}} \right) \\
&\quad = \underbrace{\frac{1}{|\,\mathrm{PM}(\mathcal{U}_s^{\mathrm{op}})|} \prod_{i \in \mathcal{U}_s^{\mathrm{op}}} \frac{1}{|\mathcal{C}_{i,s}^{\mathrm{op}}|}}_{=:c_s}.
\end{aligned}$$

The common weight c_s is independent of $\mathfrak{z}$.

Normalize over the full support. The number of compatible survivor configurations is

$$K_{\mathfrak{r}} := |\,\mathrm{PM}(\mathcal{U}_{s+1}^{\mathrm{op}})| \prod_{i \in \mathcal{U}_{s+1}^{\mathrm{op}}} |\mathcal{C}_{i,s+1}^{\mathrm{op}}|.$$

These configurations partition $\{\mathsf{Ret}_s = \mathfrak{r}\}$, and each has joint weight c_s. Therefore

$$\mathbb{P}\big(\mathsf{Ret}_s = \mathfrak{r} \,\big|\, \mathcal{F}_{s-}^{\mathrm{op}}\big) = K_{\mathfrak{r}} c_s.$$

For every compatible $\mathfrak{z}$, it follows that

$$\begin{aligned}
&\mathbb{P}\big(\mathsf{S}_{s+1}^{\mathrm{op}} = \mathfrak{z} \,\big|\, \mathcal{F}_{s-}^{\mathrm{op}}, \mathsf{Ret}_s = \mathfrak{r}\big) \\
&\quad = \frac{\mathbb{P}(\mathsf{S}_{s+1}^{\mathrm{op}} = \mathfrak{z},\, \mathsf{Ret}_s = \mathfrak{r} \mid \mathcal{F}_{s-}^{\mathrm{op}})}{\mathbb{P}(\mathsf{Ret}_s = \mathfrak{r} \mid \mathcal{F}_{s-}^{\mathrm{op}})} \\
&\quad = \frac{c_s}{K_{\mathfrak{r}} c_s} = \frac{1}{K_{\mathfrak{r}}} \\
&\quad = \frac{1}{|\,\mathrm{PM}(\mathcal{U}_{s+1}^{\mathrm{op}})|} \prod_{i \in \mathcal{U}_{s+1}^{\mathrm{op}}} \frac{1}{|\mathcal{C}_{i,s+1}^{\mathrm{op}}|}.
\end{aligned}$$

The last line is exactly the mass function of

$$\mathrm{Unif}\big(\mathrm{PM}(\mathcal{U}_{s+1}^{\mathrm{op}})\big) \otimes \bigotimes_{i \in \mathcal{U}_{s+1}^{\mathrm{op}}} \mathrm{Unif}(\mathcal{C}_{i,s+1}^{\mathrm{op}}).$$

Thus conditioning on retirement has not coupled the surviving matching and triggers.

3. A query observation contains no further survivor information. Suppose $A_s = (\mathsf{query}, y, J)$. We check the complete observation edge by edge.

- On an edge retired before operation s, its identity and triggers are already in $\mathcal{F}^{\mathrm{op}}_{s-}$, so every requested frozen-oracle row on that edge is fixed.
- On an edge retired by this query, its identity and both triggers are in $\mathfrak{r}$. Together with the submitted state y, these values fix its hit bits and every requested row.
- Let $e = \{i, j\}$ survive, and suppose the row at $j \in J$ is requested. Then neither endpoint hits: $\mathsf{Hit}_i(y) = \mathsf{Hit}_j(y) = 0$. Moreover, the edge has not met an earlier commit, so both endpoints are uncommitted. If i is masked, the selected frozen-oracle rule returns π. If i is temporarily revealed, survival implies $y_i \neq a_{\Theta_i}$, and the same rule again returns π. In formulas,
$$q^I_j(\cdot \mid y) = \begin{cases} \pi, & i \in M(y), \\ \pi, & i \in O(y) \setminus G(h^{\mathrm{op}}_{s-}),\ y_i \neq a_{\Theta_i}. \end{cases}$$
Thus the requested row is independent of the identity of i and of the surviving trigger values.

The hit bits on surviving endpoints are all zero, while every other hit bit and row is fixed by $(\mathcal{F}^{\mathrm{op}}_{s-}, \mathsf{Ret}_s = \mathfrak{r})$. Hence the whole query observation Obs_s is determined by $(\mathcal{F}^{\mathrm{op}}_{s-}, \mathsf{Ret}_s)$ on the compatible support. Equivalently, for every compatible $\mathfrak{z}$ and observation value o,
$$\begin{aligned} &\mathbb{P}\big(\mathsf{Obs}_s = o \,\big|\, \mathcal{F}^{\mathrm{op}}_{s-}, \mathsf{Ret}_s = \mathfrak{r}, \mathsf{S}^{\mathrm{op}}_{s+1} = \mathfrak{z}\big) \\ &\quad = \mathbb{P}\big(\mathsf{Obs}_s = o \,\big|\, \mathcal{F}^{\mathrm{op}}_{s-}, \mathsf{Ret}_s = \mathfrak{r}\big) \in \{0, 1\}. \end{aligned}$$
In particular,
$$\mathsf{Obs}_s \perp\!\!\!\perp \mathsf{S}^{\mathrm{op}}_{s+1} \mid (\mathcal{F}^{\mathrm{op}}_{s-}, \mathsf{Ret}_s) \qquad \text{for a query operation.}$$

4. A commit observation depends only on retired edge components. Suppose $A_s = (\mathsf{commit}, y_t, B_t)$, so $s = s_t$. By Equation (66), every unresolved edge that meets B_t is included in Ret_s. Consequently every surviving edge satisfies
$$e \in \mathcal{M}^{\mathrm{unres}}_{s+1} \quad \Longrightarrow \quad e \cap B_t = \varnothing.$$
Equivalently,
$$\mathcal{U}^{\mathrm{op}}_{s+1} \cap B_t = \varnothing.$$
We now verify that the normalized midpoint law preserves this separation.

Fix a compatible instance I. For every edge $e \in \mathcal{M}$, define probability masses on $\mathcal{V}^{e \cap B_t}$ by
$$\begin{aligned} p_e(z_{e\cap B_t}) &:= P^I_e\Big(X_{e\cap B_t} = z_{e\cap B_t} \Big| X_{e\cap G(h^{\mathrm{op}}_{s-})} = x_{e\cap G(h^{\mathrm{op}}_{s-})}\Big), \\ \widehat{p}_e(z_{e\cap B_t}) &:= \prod_{j \in e\cap B_t} q^I_j(z_j \mid y_t(h^{\mathrm{op}}_{s-})). \end{aligned}$$
Both are normalized masses. For p_e, this is the defining normalization of a conditional law; for $\widehat{p}_e$, it follows coordinate by coordinate:
$$\sum_{z_{e\cap B_t}} p_e(z_{e\cap B_t}) = 1, \qquad \sum_{z_{e\cap B_t}} \widehat{p}_e(z_{e\cap B_t}) = \prod_{j\in e\cap B_t} \left[\sum_{z_j \in \mathcal{V}} q^I_j(z_j \mid y_t(h^{\mathrm{op}}_{s-}))\right] = 1.$$
When $e \cap B_t = \varnothing$, there is one empty assignment and both masses equal 1 on it.

Because the target factorizes over matching edges and the commit rule uses the product of its requested oracle rows,
$$\begin{aligned} P^I_t(z \mid h^{\mathrm{op}}_{s-}) &= \frac{\displaystyle\prod_{e\in\mathcal{M}} P^I_e\Big(X_{e\cap B_t} = z_{e\cap B_t},\, X_{e\cap G(h^{\mathrm{op}}_{s-})} = x_{e\cap G(h^{\mathrm{op}}_{s-})}\Big)}{\displaystyle\prod_{e\in\mathcal{M}} P^I_e\Big(X_{e\cap G(h^{\mathrm{op}}_{s-})} = x_{e\cap G(h^{\mathrm{op}}_{s-})}\Big)} \\ &= \prod_{e\in\mathcal{M}} p_e(z_{e\cap B_t}), \\ \widehat{P}^I_t(z \mid h^{\mathrm{op}}_{s-}) &= \prod_{e\in\mathcal{M}} \widehat{p}_e(z_{e\cap B_t}). \end{aligned}$$

The first equality conditions the edge-product target on the already committed values; the denominators are positive by strict positivity of the hard family. The blocks $e \cap B_t$, including empty ones, are disjoint and have union B_t. Thus restriction to these blocks and the distributive law give, for arbitrary functions f_e, the finite-product identity

$$\sum_{z \in \mathcal{V}^{B_t}} \prod_{e \in \mathcal{M}} f_e(z_{e \cap B_t}) = \prod_{e \in \mathcal{M}} \left[\sum_{u \in \mathcal{V}^{e \cap B_t}} f_e(u) \right].$$

Applying it with $f_e(u) = \sqrt{p_e(u)\widehat{p}_e(u)}$ gives the midpoint normalizer. Strict positivity of the hard family and frozen oracle gives $\sum_u \sqrt{p_e(u)\widehat{p}_e(u)} > 0$ for every edge, so all following componentwise divisions are valid.

Substituting this normalizer into Equation (59) now gives

$$\begin{aligned} m_t^I(z \mid h_{s-}^{\mathrm{op}}) &= \frac{\displaystyle\prod_{e \in \mathcal{M}} \sqrt{p_e(z_{e \cap B_t})\widehat{p}_e(z_{e \cap B_t})}}{\displaystyle\prod_{e \in \mathcal{M}} \left[\sum_{u \in \mathcal{V}^{e \cap B_t}} \sqrt{p_e(u)\widehat{p}_e(u)} \right]} \\ &= \prod_{e \in \mathcal{M}} \frac{\sqrt{p_e(z_{e \cap B_t})\widehat{p}_e(z_{e \cap B_t})}}{\displaystyle\sum_{u \in \mathcal{V}^{e \cap B_t}} \sqrt{p_e(u)\widehat{p}_e(u)}}. \end{aligned}$$

For a surviving edge, $e \cap B_t = \varnothing$, so its normalized factor is 1. Every nontrivial factor belongs either to an edge retired earlier, whose identity and triggers are in $\mathcal{F}_{s-}^{\mathrm{op}}$, or to an edge retired now, whose identity and triggers are in $\mathfrak{r}$. Thus the midpoint probability of the observed commit value does not depend on $\mathsf{S}_{s+1}^{\mathrm{op}}$. The accompanying oracle rows have the same property: rows on previously retired edges are fixed by $\mathcal{F}_{s-}^{\mathrm{op}}$, and rows on newly retired edges are fixed after conditioning on $\mathfrak{r}$.

More explicitly, fix a compatible $\mathfrak{z}$. Together with $(\mathcal{F}_{s-}^{\mathrm{op}}, \mathsf{Ret}_s = \mathfrak{r})$, it specifies the compatible instance I. For a commit observation $(\mathbf{q}, z)$,

$$\begin{aligned} &\mathbb{P}\big(\mathsf{Obs}_s = (\mathbf{q}, z) \,\big|\, \mathcal{F}_{s-}^{\mathrm{op}}, \mathsf{Ret}_s = \mathfrak{r}, \mathsf{S}_{s+1}^{\mathrm{op}} = \mathfrak{z}\big) \\ &\quad = \mathbf{1}\Big\{\mathbf{q} = \big(q_j^I(\cdot \mid y_t(h_{s-}^{\mathrm{op}}))\big)_{j \in B_t}\Big\}\, m_t^I(z \mid h_{s-}^{\mathrm{op}}). \end{aligned}$$

Every row and every nonunit edge factor on the right is determined by $(\mathcal{F}_{s-}^{\mathrm{op}}, \mathsf{Ret}_s)$; changing $\mathfrak{z}$ changes only survivor-edge factors, which equal 1. Thus the displayed likelihood is the same for all compatible survivor configurations. Hence

$$\mathsf{Obs}_s \perp\!\!\!\perp \mathsf{S}_{s+1}^{\mathrm{op}} \mid (\mathcal{F}_{s-}^{\mathrm{op}}, \mathsf{Ret}_s) \qquad \text{for a commit operation.}$$

5. Close the induction. Step 3 proves for a query, and Step 4 proves for a commit, that the conditional law of the new observation does not change when the new survivor state is added to the conditioning information:

$$\begin{aligned} &\mathcal{L}\big(\mathsf{Obs}_s \,\big|\, \mathcal{F}_{s-}^{\mathrm{op}}, \mathsf{Ret}_s, \mathsf{S}_{s+1}^{\mathrm{op}}\big) \\ &\quad = \mathcal{L}\big(\mathsf{Obs}_s \,\big|\, \mathcal{F}_{s-}^{\mathrm{op}}, \mathsf{Ret}_s\big). \end{aligned}$$

Thus, for every pair $(\mathfrak{r}, o)$ of positive conditional probability and every compatible survivor configuration $\mathfrak{z}$,

$$\begin{aligned} &\mathbb{P}\big(\mathsf{S}_{s+1}^{\mathrm{op}} = \mathfrak{z} \,\big|\, \mathcal{F}_{s-}^{\mathrm{op}}, \mathsf{Ret}_s = \mathfrak{r}, \mathsf{Obs}_s = o\big) \\ &\quad = \mathbb{P}\big(\mathsf{S}_{s+1}^{\mathrm{op}} = \mathfrak{z} \,\big|\, \mathcal{F}_{s-}^{\mathrm{op}}, \mathsf{Ret}_s = \mathfrak{r}\big) \\ &\quad = \frac{1}{|\operatorname{PM}(\mathcal{U}_{s+1}^{\mathrm{op}})|} \prod_{i \in \mathcal{U}_{s+1}^{\mathrm{op}}} \frac{1}{|\mathcal{C}_{i,s+1}^{\mathrm{op}}|}. \end{aligned}$$

The last equality is the normalization computed in Step 2.

It remains only to identify the conditioning field. The selected action A_s is $\mathcal{F}^{\mathrm{op}}_{s-}$-measurable, and the next unresolved set is recovered from the current set and retirement record by

$$\mathcal{U}^{\mathrm{op}}_{s+1} = \mathcal{U}^{\mathrm{op}}_{s} \setminus \bigcup_{(e,\cdot)\in \mathsf{Ret}_s} e.$$

The current unresolved set is itself recoverable from $\mathcal{H}^{\mathrm{op}}_{s-}$ by the previously recorded retirements. Expanding Equations (71) and (72) and using these two measurability facts gives

$$\begin{aligned}\mathcal{F}^{\mathrm{op}}_{(s+1)-} &= \sigma\left(\mathcal{H}^{\mathrm{op}}_{(s+1)-}, \mathcal{U}^{\mathrm{op}}_{s+1}\right)\\ &= \sigma\left(\mathcal{H}^{\mathrm{op}}_{s-}, A_s, \mathsf{Obs}_s, \mathsf{Ret}_s, \mathcal{U}^{\mathrm{op}}_{s+1}\right)\\ &= \sigma\left(\mathcal{F}^{\mathrm{op}}_{s-}, \mathsf{Ret}_s, \mathsf{Obs}_s\right).\end{aligned}$$

Consequently, the preceding conditional mass is also the conditional mass given $\mathcal{F}^{\mathrm{op}}_{(s+1)-}$. Therefore

$$\mathsf{S}^{\mathrm{op}}_{s+1} \mid \mathcal{F}^{\mathrm{op}}_{(s+1)-} \sim \mathrm{Unif}\left(\mathrm{PM}(\mathcal{U}^{\mathrm{op}}_{s+1})\right) \otimes \bigotimes_{i\in\mathcal{U}^{\mathrm{op}}_{s+1}} \mathrm{Unif}(\mathcal{C}^{\mathrm{op}}_{i,s+1}).$$

This proves the product law at every operation boundary.

Finally, the event that operation s is commit t is known at the starting boundary:

$$\{s_t = s\} = \left\{\sum_{r<s} \mathbf{1}\{A_r \text{ is a commit}\} = t-1,\ A_s \text{ is a commit}\right\} \in \mathcal{F}^{\mathrm{op}}_{s-}.$$

Applying the operation-boundary law on each event $\{s_t = s\}$, and using $\mathsf{S}^{\mathrm{op}}_{s_t} = \mathsf{S}_t$ and $\mathcal{F}^{\mathrm{op}}_{s_t-} = \mathcal{F}_{t-}$, proves equation 74. □

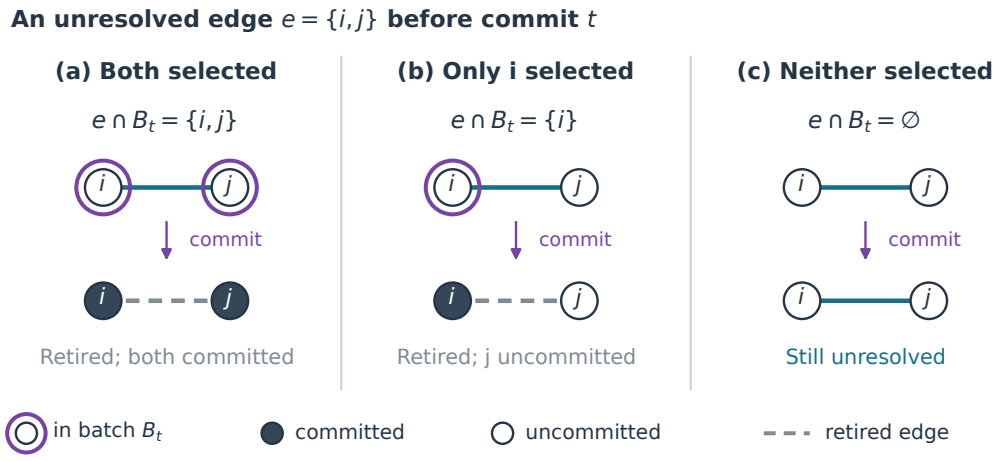


Figure 4: Three possible batch selections on an initially unresolved edge. Each panel shows the edge before (top) and after (bottom) one commit. Purple rings mark membership in B_t; filled nodes are committed. Selecting both endpoints gives a collision; selecting one gives a crossing edge. Both events retire the whole edge, but in (b) j remains uncommitted. Selecting neither endpoint leaves the edge unresolved. The proof records the edge identity and both triggers in (a)–(b); the target matching itself is unchanged.

One source, at most one candidate per submission. Recall from Equations (66) and (68) that the number of query-retired edges is

$$D_{\mathrm{gen}} = \sum_{s:\ A_s \text{ is a query}} |\mathcal{M}^{\mathrm{unres}}_{s} \setminus \mathcal{M}^{\mathrm{unres}}_{s+1}|.$$

Commit retirements do not contribute, and simultaneous hits at both endpoints count as one edge. A state y tests at most one candidate $\mathsf{Test}_i(y)$ at each source i, regardless of how many returned rows are inspected. Together with the conditional uniformity in Lemma C.4, this gives the following bound directly.

Lemma C.5 (Equality-test disclosure)**.** *For any algorithm using at most $\overline{Q}$ counterfactual submissions,*

$$\mathbb{E}D_{\mathrm{gen}} \le \frac{N\overline{Q}}{m}. \tag{75}$$

Proof. All probabilities and expectations use the joint instance–midpoint law equation 64, with the decision-rule seed w fixed. The execution below is the midpoint execution. It suffices to consider integer $\overline{Q}$: for a noninteger budget, apply the integer result to $\lfloor \overline{Q} \rfloor$, whose bound is no larger. If $\overline{Q} = 0$, there are no query hits and $D_{\mathrm{gen}} = 0$. If $\overline{Q} \ge m$, the bound follows from $D_{\mathrm{gen}} \le N/2 \le N\overline{Q}/m$. Hence assume $1 \le \overline{Q} < m$.

1. Bound the hit probability at one query. Fix a vertex i. Number counted counterfactual submissions by $k = 1, 2, \ldots$; commits may occur between them. The distinguished all-mask preprocessing readout tests no candidate and is not included in this count. At the k-th such query, let s be its operation index and write $y = y_s^{\mathrm{op}}$, as in Equation (29). Recall that $\mathcal{U}_s^{\mathrm{op}} = \bigcup_{e \in \mathcal{M}_s^{\mathrm{unres}}} e$ is the unresolved vertex set just before this operation (Equation (67)). The past and selected action determine whether operation s is the k-th counted query. Thus the operation-boundary conclusion of Lemma C.4 applies here even when the query time and state are chosen adaptively.

If $i \in \mathcal{U}_s^{\mathrm{op}}$, each of the preceding $k-1$ counted submissions has removed at most one candidate at i. Commits leave the candidate set of a surviving vertex unchanged, by Equation (69). Consequently,

$$|\mathcal{C}_{i,s}^{\mathrm{op}}| \ge m - (k-1) = m - k + 1.$$

Conditional on $\mathcal{F}_{s-}^{\mathrm{op}}$, the submitted state is fixed and Θ_i is uniform on $\mathcal{C}_{i,s}^{\mathrm{op}}$. Therefore

$$\begin{aligned} \mathbb{P}(\mathsf{Hit}_i(y) = 1 \mid \mathcal{F}_{s-}^{\mathrm{op}}) &= \mathbb{P}(\Theta_i = \mathsf{Test}_i(y) \mid \mathcal{F}_{s-}^{\mathrm{op}}) \\ &= \frac{\mathbf{1}\{\mathsf{Test}_i(y) \in \mathcal{C}_{i,s}^{\mathrm{op}}\}}{|\mathcal{C}_{i,s}^{\mathrm{op}}|} \le \frac{1}{m-k+1}. \end{aligned} \tag{76}$$

A mask, background token, or previously excluded candidate gives numerator zero. If the edge has already retired, there can be no further hit *while unresolved*, regardless of later oracle replies.

2. Accumulate the probabilities without assuming independence. For $0 \le k \le \overline{Q}$, define

$$p_{i,k} := \mathbb{P}\begin{pmatrix} i \text{ has a query hit while its edge is unresolved} \\ \text{among the first } k \text{ counted submissions} \end{pmatrix}, \qquad p_{i,0} = 0.$$

If the execution contains fewer than k counted submissions, this event uses all those that occurred; no extra queries are executed. A hit retires the edge, so i can have at most one hit while unresolved. The increment $p_{i,k} - p_{i,k-1}$ is therefore the probability of such a hit at query k.

To bound this increment, condition on each possible analytic history just before query k, apply equation 76 when i is unresolved, and average over these histories. This gives

$$\begin{aligned} p_{i,k} - p_{i,k-1} &\le \frac{\mathbb{P}(\text{query } k \text{ occurs and } i \text{ is unresolved just before it})}{m-k+1} \\ &\le \frac{1 - p_{i,k-1}}{m-k+1}. \end{aligned} \tag{77}$$

For the last inequality, an unresolved vertex has not hit earlier. Earlier retirement by a mate hit or a commit, as well as early termination, can only remove paths from the event in the numerator. Simultaneous hits at i and its mate still count as a hit at i: the edge was unresolved immediately before that query.

Rearranging equation 77 step by step yields

$$\begin{aligned}1 - p_{i,k} &= 1 - p_{i,k-1} - (p_{i,k} - p_{i,k-1}) \\ &\geq (1 - p_{i,k-1}) \left(1 - \frac{1}{m-k+1}\right) \\ &= (1 - p_{i,k-1}) \frac{m-k}{m-k+1}.\end{aligned}$$

Starting from $1 - p_{i,0} = 1$ and applying this inequality successively,

$$\begin{aligned}1 - p_{i,\overline{Q}} &\geq \prod_{k=1}^{\overline{Q}} \frac{m-k}{m-k+1} \\ &= \frac{m-1}{m} \frac{m-2}{m-1} \cdots \frac{m-\overline{Q}}{m-\overline{Q}+1} \\ &= \frac{m-\overline{Q}}{m}.\end{aligned}$$

For $\overline{Q} = 1$ the product has just its first factor. The cancellation therefore gives

$$p_{i,\overline{Q}} \leq 1 - \frac{m-\overline{Q}}{m} = \frac{\overline{Q}}{m}. \tag{78}$$

This multiplication iterates conditional probability bounds; it does not assume independent queries.

3. Count retired edges by their successful endpoints. At a query, an unresolved edge retires exactly when at least one of its endpoints hits. For binary bits b_u, b_v, $\mathbf{1}\{b_u + b_v \geq 1\} \leq b_u + b_v$; two simultaneous hits still retire only one edge. Hence, pathwise,

$$\begin{aligned}D_{\text{gen}} &= \sum_{s:\, A_s \text{ is a query}} \sum_{\{u,v\} \in \mathcal{M}_s^{\text{unres}}} \mathbf{1}\{\mathsf{Hit}_u(y_s^{\text{op}}) + \mathsf{Hit}_v(y_s^{\text{op}}) \geq 1\} \\ &\leq \sum_{i=1}^{N} \sum_{\substack{s:\, A_s \text{ is a query} \\ i \in \mathcal{U}_s^{\text{op}}}} \mathsf{Hit}_i(y_s^{\text{op}}).\end{aligned}$$

The change from edges to vertices uses that every unresolved vertex lies on exactly one matching edge. For each i, the inner sum is either zero or one, because its first hit retires its edge. It is thus the indicator of the event defining $p_{i,\overline{Q}}$. Taking expectations and using $\mathbb{E}\mathbf{1}_E = \mathbb{P}(E)$ gives

$$\begin{aligned}\mathbb{E}D_{\text{gen}} &\leq \sum_{i=1}^{N} \mathbb{E}\left[\sum_{\substack{s:\, A_s \text{ is a query} \\ i \in \mathcal{U}_s^{\text{op}}}} \mathsf{Hit}_i(y_s^{\text{op}})\right] \\ &= \sum_{i=1}^{N} p_{i,\overline{Q}} \leq \sum_{i=1}^{N} \frac{\overline{Q}}{m} = \frac{N\overline{Q}}{m}.\end{aligned}$$

No independence between vertices is used. □

C.5 Common collision bookkeeping

We count how matching edges leave the unresolved-edge set; this pathwise identity supplies the batch-size constraint in the collision lower-tail proof. Recall that $\mathcal{U}_t$ contains the vertices on edges still unresolved immediately before commit t, after any preceding queries. At this boundary, operation s_t, the unresolved edge set is $\mathcal{M}_{s_t}^{\text{unres}}$, as in Equations (66) and (73). Put $n_t := |\mathcal{U}_t|$. Removing whole edges makes this number even.

For the selected batch B_t, define its active part and collision count:

$$B_t^\circ := B_t \cap \mathcal{U}_t, \qquad n_t^\circ := |B_t^\circ|, \qquad Z_t := \left|\{e \in \mathcal{M}_{s_t}^{\text{unres}} : e \subseteq B_t^\circ\}\right|, \qquad Z := \sum_t Z_t. \tag{79}$$

An unresolved edge has no committed endpoint, so $\mathcal{U}_t \subseteq U_{G_{t-1}}$. The inclusion can be strict: query-retired endpoints and the uncommitted mate of a commit-retired endpoint remain uncommitted but are no longer active. Figure 4(b) shows the latter case. A later commit of that mate creates no new unresolved-edge collision.

Let $R^\circ := |\{t : n_t^\circ \geq 1\}| \leq \overline{R}$ be the number of rounds with a nonempty active batch.

Lemma C.6 (Pathwise edge-counting identity)**.** *Under the retirement rules in Equation* (66)*, every complete execution satisfies the pathwise identity*

$$\frac{N}{2} = D_{\mathrm{gen}} + \sum_t (n_t^\circ - Z_t). \tag{80}$$

Consequently,

$$\sum_{t:n_t^\circ \geq 1} (n_t^\circ - 1) = \frac{N}{2} - D_{\mathrm{gen}} + Z - R^\circ \geq \frac{N}{2} - D_{\mathrm{gen}} - \overline{R}. \tag{81}$$

Proof. An active batch touches exactly $n_t^\circ - Z_t$ previously unresolved edges. Indeed, its Z_t internal edges use $2Z_t$ vertices, while every remaining active vertex lies on a distinct crossing edge. Thus the number touched is

$$\underbrace{Z_t}_{\text{internal edges}} + \underbrace{(n_t^\circ - 2Z_t)}_{\text{crossing edges}} = n_t^\circ - Z_t.$$

Since every position is eventually committed, every original matching edge is retired exactly once: either at a successful counterfactual trigger test (query retirement), or at its first active commit batch (commit retirement). Summing over the $N/2$ original edges proves equation 80.

Finally, $\sum_{t:n_t^\circ \geq 1}(n_t^\circ - 1) = \sum_t n_t^\circ - R^\circ$. Substituting $\sum_t n_t^\circ = N/2 - D_{\mathrm{gen}} + Z$ from equation 80, then using $Z \geq 0$ and $R^\circ \leq \overline{R}$, gives equation 81. □

C.6 A Laplace bound for uniform-matching collisions

The next lemma is a finite combinatorial statement independent of the hard oracle and the adaptive decision rule. It gives a conditional Laplace bound used below to derive an adaptive lower-tail estimate.

Intuition. For a fixed set of k vertices in a uniform matching on n vertices, each unordered pair is an edge with probability $1/(n-1)$. Hence the expected number of internal edges is

$$\binom{k}{2}\frac{1}{n-1} = \frac{k(k-1)}{2(n-1)} \simeq \frac{k^2}{n} \qquad (k \geq 2).$$

To obtain a lower-tail estimate rather than only a mean, the proof exposes order k edges, each with conditional success probability at least order k/n. Their Laplace factors multiply by successive conditioning; the exposures need not be independent.

Lemma C.7 (Internal-edge Laplace transform)**.** *Let M be a uniformly random perfect matching on a labeled set U of even cardinality $n \geq 2$. Fix $S \subseteq U$, put $k := |S|$, and let*

$$Z_S := |\{e \in M : e \subseteq S\}|.$$

Then, for every $t > 0$,

$$\mathbb{E}e^{-tZ_S} \leq \exp\left\{-\frac{1-e^{-t}}{32}\frac{[k-1]_+^2}{n}\right\}. \tag{82}$$

Proof. 1. Handle sets of size at most three. For $k \in \{0,1\}$, both sides of equation 82 equal one. For $k \in \{2,3\}$, two internal matching edges would require four distinct vertices of S. Thus $Z_S \in \{0,1\}$, and

$$\mathbb{E}Z_S = 0 \cdot \mathbb{P}(Z_S = 0) + 1 \cdot \mathbb{P}(Z_S = 1) = \mathbb{P}(Z_S = 1).$$

Also, writing the edge count as a sum of indicators and taking expectations,

$$\mathbb{E}Z_S = \sum_{\{u,v\}\subseteq S} \mathbb{P}(\{u,v\} \in M) = \frac{\binom{k}{2}}{n-1} = \frac{k(k-1)}{2(n-1)}. \tag{83}$$

Writing $a_t := 1 - e^{-t} \in (0,1)$, the same two possible values give

$$\begin{aligned}\mathbb{E}e^{-tZ_S} &= \mathbb{P}(Z_S = 0) + e^{-t}\mathbb{P}(Z_S = 1)\\ &= 1 - (1-e^{-t})\mathbb{P}(Z_S = 1) = 1 - a_t\mathbb{E}Z_S.\end{aligned}$$

Using $1 - v \le e^{-v}$ for $v \ge 0$, we obtain

$$\mathbb{E}e^{-tZ_S} = 1 - a_t\mathbb{E}Z_S \le e^{-a_t\mathbb{E}Z_S} \le \exp\left\{-\frac{a_t}{32}\frac{(k-1)^2}{n}\right\},$$

where the last inequality follows directly from equation 83 for $k = 2, 3$.

2. Expose edges and bound each conditional success probability. It remains to consider $k \ge 4$. Partition $S = A \sqcup B$ with $|A| = \lfloor k/2 \rfloor$ and $|B| = \lceil k/2 \rceil$, and put $m := \lfloor k/4 \rfloor$. We reveal edges of the original matching M on the full set U, not of a matching restricted to S. Only the starting endpoint is required to lie in A; its mate can lie in A, B, or $U \setminus S$.

Fix an ordering of U. Let $U_0 := U$, and for $j = 1, \dots, m$ define

$$\begin{aligned} a_j &:= \min(A \cap U_{j-1}), & b_j &:= \mathrm{mate}_M(a_j),\\ e_j &:= \{a_j, b_j\} \in M, & U_j &:= U_{j-1} \setminus \{a_j, b_j\}.\end{aligned}$$

Thus U_{j-1} consists of the vertices whose matching edges have not yet been revealed; these vertices are already matched in M. The minimum uses the fixed ordering, so the choice of a_j depends only on the earlier revealed edges. It is well-defined because each previous edge removes at most two vertices from A, and

$$|A \cap U_{j-1}| \ge \lfloor k/2 \rfloor - 2(j-1) \ge 2m - 2(j-1) \ge 2 > 0 \qquad (1 \le j \le m).$$

Define the success indicator by

$$X_j := \mathbf{1}_{\{b_j \in B \cap U_{j-1}\}}, \qquad \{X_j = 1\} = \{\mathrm{mate}_M(a_j) \in B \cap U_{j-1}\}.$$

Since every starting endpoint a_ℓ lies in A, each previous edge e_ℓ removes at most one vertex from B. Consequently,

$$|B \cap U_{j-1}| \ge \lceil k/2 \rceil - (j-1) \ge k/2 - (m-1) \ge k/4, \qquad |U_{j-1}| = n - 2(j-1).$$

Conditional on $e_1, \dots, e_{j-1}$, the remaining matching is uniform on U_{j-1}: every perfect matching there has exactly one extension by the already revealed edges. Each possible mate of a_j has the same number of matching completions, so

$$\mathbb{P}(b_j = v \mid e_1, \dots, e_{j-1}) = \frac{1}{|U_{j-1}| - 1} \quad \text{for } v \in U_{j-1} \setminus \{a_j\}.$$

Summing over the allowable mates in B, which never include $a_j \in A$, gives

$$\begin{aligned}\mathbb{P}(X_j = 1 \mid e_1, \dots, e_{j-1}) &= \sum_{v \in B \cap U_{j-1}} \mathbb{P}(b_j = v \mid e_1, \dots, e_{j-1})\\ &= \frac{|B \cap U_{j-1}|}{|U_{j-1}| - 1} \ge \frac{k/4}{n} = \frac{k}{4n}.\end{aligned}$$

The earlier indicators $X_1, \dots, X_{j-1}$ are determined by these revealed edges. Adding them to the conditioning therefore gives the same probability:

$$\mathbb{P}(X_j = 1 \mid X_1, \dots, X_{j-1}, \text{all preceding exposed edges}) \ge u := \frac{k}{4n}. \tag{84}$$

Removing both endpoints after each exposure makes $e_1, \dots, e_m$ distinct. If $X_j = 1$, then $a_j \in A$ and $b_j \in B$, so $e_j \subseteq A \cup B = S$. Therefore

$$\sum_{j=1}^{m} X_j = |\{e_j : 1 \le j \le m,\ X_j = 1\}| \le |\{e \in M : e \subseteq S\}| = Z_S.$$

3. Iterate conditional Laplace factors without independence. For one exposure, the Bernoulli identity and the preceding probability bound give

$$\begin{aligned}\mathbb{E}[e^{-tX_j} \mid \text{preceding exposed edges}] &= 1-(1-e^{-t})\Pr(X_j=1 \mid \text{preceding exposed edges}) \\ &\le 1-u(1-e^{-t}).\end{aligned}$$

For $1 \le r \le m$, condition on the first $r-1$ exposed edges. The earlier factors are measurable with respect to this record, so

$$\begin{aligned}\mathbb{E}e^{-t\sum_{j=1}^{r}X_j} &= \mathbb{E}\Big[e^{-t\sum_{j=1}^{r-1}X_j}\mathbb{E}(e^{-tX_r} \mid \text{first } r-1 \text{ exposed edges})\Big] \\ &\le [1-u(1-e^{-t})]\,\mathbb{E}e^{-t\sum_{j=1}^{r-1}X_j}.\end{aligned}$$

Starting with the empty sum at $r=0$, whose exponential is one, and iterating to $r=m$ gives the power below without an independence assumption:

$$\begin{aligned}\mathbb{E}e^{-tZ_S} &\le \mathbb{E}\exp\left\{-t\sum_{j=1}^{m}X_j\right\} \\ &\le \{1-u(1-e^{-t})\}^m \le \exp\{-mu(1-e^{-t})\}. \end{aligned} \tag{85}$$

Since $m \ge k/8$ for $k \ge 4$, $mu \ge k^2/(32n) \ge (k-1)^2/(32n)$. Substitution in equation 85 proves equation 82. □

C.7 An adaptive lower tail for batch collisions

The conditional Laplace bound and the edge-counting identity now force many collisions with constant probability under the joint instance–midpoint law. This is the input needed to accumulate edge-level discrepancy nonlinearly; no independence between commit rounds is assumed. Recall that $Z=\sum_t Z_t$ counts the unresolved edges whose two endpoints are committed in the same active batch, as defined in equation 79.

Intuition. If order N unresolved vertices were spread evenly across $\overline{R}$ batches while the unresolved pool had size order N, the collision scale would be

$$\overline{R}\frac{(N/\overline{R})^2}{N} = \frac{N}{\overline{R}}.$$

The proof does not assume even batches or a fixed pool: the edge-counting identity forces enough total active batch mass, Cauchy–Schwarz gives the same $N/\overline{R}$ scale, and the conditional Laplace bound turns that pathwise mass into a constant-probability collision event.

Proposition C.8 (Adaptive collision lower tail)**.** *Fix a decision-rule seed under the joint law equation 64. If $\overline{Q} \le m/8$ and $1 \le \overline{R} \le N/8$, then*

$$\mathbb{P}\left(Z \ge \frac{N}{8192\overline{R}}\right) \ge \frac{1}{2} - \exp\left\{-\frac{N}{8192\overline{R}}\right\}. \tag{86}$$

In particular, if $\overline{R} \le N/16384$, then

$$\mathbb{P}\left(Z \ge \frac{N}{8192\overline{R}}\right) \ge \frac{1}{4}. \tag{87}$$

Proof. 1. Fix the pre-exposure information and conditional Laplace bound. For an actual commit round t, let B_t be the batch selected by the seed-fixed decision rule. Recall the existing notation:

$$B_t^\circ = B_t \cap \mathcal{U}_t, \qquad n_t = |\mathcal{U}_t|, \qquad n_t^\circ = |B_t^\circ|,$$
$$Z_t = |\{e \in \mathcal{M}_{s_t}^{\mathrm{unres}} : e \subseteq B_t^\circ\}|, \qquad Z = \sum_t Z_t.$$

Thus B_t is the full commit batch, whereas B_t° contains only its currently unresolved vertices.

After termination, append empty rounds until the total number of indexed rounds is $\overline{R}$. In every appended round set $B_t = B_t^\circ = \mathcal{U}_t = \varnothing$, so $n_t = n_t^\circ = Z_t = 0$. These rounds add no collision, and hence $Z = \sum_{t=1}^{\overline{R}} Z_t$. Introduce only the following additional abbreviation:

$$A_t := \begin{cases} [n_t^\circ - 1]_+^2 / n_t, & n_t > 0, \\ 0, & n_t = 0. \end{cases}$$

The pre-commit analytic field $\mathcal{F}_{t-}$ determines both the selected batch B_t and the unresolved set $\mathcal{U}_t$, hence also $B_t^\circ = B_t \cap \mathcal{U}_t$. Selecting B_t adds no observation: the matching edges incident to B_t° and the midpoint values X_{B_t} have not yet been exposed. Thus n_t, n_t°, A_t, and all quantities from earlier rounds are $\mathcal{F}_{t-}$-measurable. For appended empty rounds, use the terminal field.

By Lemma C.4, conditional on $\mathcal{F}_{t-}$, the matching on $\mathcal{U}_t$ is uniform. For $n_t \geq 2$, apply Lemma C.7 with $U = \mathcal{U}_t$, $S = B_t^\circ$, $n = n_t$, $k = n_t^\circ$, and Laplace parameter one. Since $1 - e^{-1} \geq 1/2$, this gives

$$\mathbb{E}\big[e^{-Z_t} \mid \mathcal{F}_{t-}\big] \leq \exp\left\{-\frac{1 - e^{-1}}{32} A_t\right\} \leq e^{-A_t/64}. \tag{88}$$

The unresolved set has even cardinality, so $n_t = 1$ cannot occur. If $n_t = 0$, then $n_t^\circ = Z_t = A_t = 0$, and the same display holds with equality.

2. Iterate the exponential expectation bound. Let $r = 0, \ldots, \overline{R}$ count completed commit rounds, not individual operations. Define

$$\mathsf{M}_r := \exp\left\{-\sum_{t=1}^{r} Z_t + \frac{1}{64}\sum_{t=1}^{r} A_t\right\}, \qquad \mathsf{M}_0 := 1. \tag{89}$$

Both M_{r-1} and A_r are measurable with respect to $\mathcal{F}_{r-}$, the field immediately before commit r after any intervening counterfactual operations. Hence equation 88 and the tower property give

$$\mathbb{E}\mathsf{M}_r = \mathbb{E}\Big[\mathsf{M}_{r-1} e^{A_r/64} \mathbb{E}(e^{-Z_r} \mid \mathcal{F}_{r-})\Big] \leq \mathbb{E}\mathsf{M}_{r-1}.$$

Iteration yields

$$\mathbb{E}\mathsf{M}_{\overline{R}} \leq 1. \tag{90}$$

This is the only concentration step; adaptivity is absorbed into the successive conditional expectations.

3. Turn edge counting into a pathwise Laplace budget. By Lemma C.5 and $\overline{Q} \leq m/8$,

$$\mathbb{E}D_{\mathrm{gen}} \leq \frac{N\overline{Q}}{m} \leq \frac{N(m/8)}{m} = \frac{N}{8}.$$

Because $D_{\mathrm{gen}} \geq 0$, Markov's inequality gives

$$\mathbb{P}(D_{\mathrm{gen}} > N/4) \leq \frac{\mathbb{E}D_{\mathrm{gen}}}{N/4} \leq \frac{N/8}{N/4} = \frac{1}{2}.$$

Taking the complement, $\mathbb{P}(D_{\mathrm{gen}} \leq N/4) = 1 - \mathbb{P}(D_{\mathrm{gen}} > N/4)$, therefore yields

$$\mathbb{P}(D_{\mathrm{gen}} \leq N/4) \geq \frac{1}{2}. \tag{91}$$

On this event, the pathwise counting identity equation 81 in Lemma C.6, the bounds $R^\circ \leq \overline{R} \leq N/8$, and $Z \geq 0$ imply

$$\sum_{t=1}^{\overline{R}} [n_t^\circ - 1]_+ \geq \frac{N}{2} - D_{\mathrm{gen}} - \overline{R} \geq \frac{N}{8}. \tag{92}$$

Since $n_t \leq N$, the definition of A_t implies $A_t \geq [n_t^\circ - 1]_+^2 / N$. This also holds when $n_t = 0$, because then both sides are zero. Cauchy–Schwarz over the $\overline{R}$ indexed rounds (including appended

empty rounds) now gives

$$\sum_{t=1}^{\overline{R}} A_t \geq \frac{1}{N}\sum_{t=1}^{\overline{R}} [n_t^\circ - 1]_+^2 \geq \frac{1}{N\overline{R}}\left(\sum_{t=1}^{\overline{R}} [n_t^\circ - 1]_+\right)^2 \geq \frac{N}{64\overline{R}}. \tag{93}$$

4. Subtract the small-collision event. Set $x := N/(8192\overline{R})$. On the intersection $\{D_{\text{gen}} \leq N/4\} \cap \{Z < x\}$, equation 93 yields

$$\log \mathsf{M}_{\overline{R}} = -Z + \frac{1}{64}\sum_{t=1}^{\overline{R}} A_t > -x + \frac{N}{4096\overline{R}} = x.$$

Using equation 90 and Markov's inequality,

$$\mathbb{P}(D_{\text{gen}} \leq N/4,\ Z < x) \leq \mathbb{P}(\mathsf{M}_{\overline{R}} > e^x) \leq \frac{\mathbb{E}\mathsf{M}_{\overline{R}}}{e^x} \leq \frac{1}{e^x} = e^{-x}.$$

Combining with equation 91,

$$\begin{aligned}\mathbb{P}(Z \geq x) &\geq \mathbb{P}(D_{\text{gen}} \leq N/4,\ Z \geq x)\\ &= \mathbb{P}(D_{\text{gen}} \leq N/4) - \mathbb{P}(D_{\text{gen}} \leq N/4,\ Z < x)\\ &\geq \frac{1}{2} - e^{-x},\end{aligned}$$

which proves equation 86. If $\overline{R} \leq N/16384$, then $x \geq 2$ and $1/2 - e^{-2} > 1/4$, proving equation 87. □

C.8 Collision-affinity bound

We combine the midpoint identity with the one-edge Hellinger loss to convert unresolved collisions into a multiplicative affinity penalty. Recall our normalization: for probability masses p, q on the same finite set $\mathcal{X}$, $\mathrm{Aff}(p,q) := \sum_{x\in\mathcal{X}} \sqrt{p(x)q(x)}$. The squared Hellinger distance is $h^2(p,q) := \frac{1}{2}\sum_{x\in\mathcal{X}}(\sqrt{p(x)} - \sqrt{q(x)})^2 = 1 - \mathrm{Aff}(p,q)$.

Intuition. Each unresolved edge whose endpoints are committed together contributes the factor $1 - d_{\text{edge}}$. Since $d_{\text{edge}} \simeq \eta^2$,

$$1 - (1 - d_{\text{edge}})^Z \simeq \min\{1, \eta^2 Z\}.$$

For justification, $1 - (1-d)^z \leq \min\{1, zd\}$ for integer $z \geq 0$, while $1 - (1-d)^z \geq 1 - e^{-zd} \geq (1 - e^{-1})\min\{1, zd\}$. This is an edge-penalty scale; the next lemma places its expectation below the actual squared-Hellinger risk.

Lemma C.9 (Affinity loss from unresolved collisions)**.** *For every fixed decision-rule seed w,*

$$\mathbb{E}_{I\sim\Pi} h^2(P^I, \widehat{P}_w^I) \geq \mathbb{E}_{I,H}\left[1 - (1 - d_{\text{edge}})^Z\right], \tag{94}$$

where the expectation on the right is under equation 64.

Proof. 1. Specify the edge components at a fixed history. Fix I, w and a reachable pre-commit history h, and write $B_t = B_t(h)$. As in Step 4 of Lemma C.4, the two component laws on $\mathcal{V}^{e\cap B_t}$, for $e \in \mathcal{M}$, are

$$p_e(z_{e\cap B_t}) := P_e^I(z_{e\cap B_t} \mid x_{e\cap G(h)}),$$
$$\widehat{p}_e(z_{e\cap B_t}) := \prod_{j\in e\cap B_t} q_j^I(z_j \mid y_t(h)).$$

The first line is the conditional probability of the current edge coordinates given the already committed coordinates of that edge. Both masses sum to one; on an empty coordinate set both equal one on the empty assignment. The conditional product calculation in that step gives

$$P_t^I(z \mid h) = \prod_{e \in \mathcal{M}} p_e(z_{e \cap B_t}), \qquad \widehat{P}_t^I(z \mid h) = \prod_{e \in \mathcal{M}} \widehat{p}_e(z_{e \cap B_t}).$$

These products are at fixed I, h; each component is a block of batch coordinates from one edge, not necessarily a single vertex.

2. Compute the factor of an unresolved collision. Partition the unresolved edges according to the number of selected endpoints:

$$\mathcal{E}_t^{(r)} := \{e \in \mathcal{M}_{s_t}^{\mathrm{unres}} : |e \cap B_t^\circ| = r\}, \qquad r \in \{0, 1, 2\}. \tag{95}$$

Thus the collision-edge set and its size are

$$\mathcal{E}_t^{(2)} = \{e \in \mathcal{M}_{s_t}^{\mathrm{unres}} : e \subseteq B_t^\circ\}, \qquad |\mathcal{E}_t^{(2)}| = Z_t, \qquad B_t^\circ = B_t \cap \mathcal{U}_t.$$

For $e = \{i, j\} \in \mathcal{E}_t^{(2)}$, both endpoints belong to B_t and neither belongs to $G(h)$. Thus the exact component has no conditioning within e, while both oracle rows see their mate masked:

$$p_e(z_i, z_j) = P_e^I(z_i, z_j), \qquad \widehat{p}_e(z_i, z_j) = \pi(z_i)\pi(z_j) = P_e^0(z_i, z_j).$$

Here the oracle equality uses equation 41, and the absence of conditioning from other edges uses the target's edgewise product law. Consequently, $\mathrm{Aff}(p_e, \widehat{p}_e) = 1 - h^2(P_e^I, P_e^0) = 1 - d_{\mathrm{edge}}$, where d_{edge} is defined in equation 49. In terms of the actual edge masses, this is

$$\sum_{z_i, z_j} \sqrt{P_e^I(z_i, z_j) P_e^0(z_i, z_j)} = 1 - d_{\mathrm{edge}}. \tag{96}$$

3. Bound every remaining component and apply product affinity. Since p_e and $\widehat{p}_e$ are probability masses, $0 \le \mathrm{Aff}(p_e, \widehat{p}_e) \le 1$ by Cauchy–Schwarz. This covers all noncollision components, including previously retired edges where the oracle need not be exact. More specifically, an unresolved crossing edge has $p_e = \widehat{p}_e = \pi$, and an edge with $e \cap B_t = \varnothing$ has both masses equal to one on the empty assignment; these components have affinity exactly one.

Apply Lemma H.1 with index set $J = \mathcal{M}$, spaces $\mathcal{X}_e = \mathcal{V}^{e \cap B_t}$, and component laws $p_e, \widehat{p}_e$. The two product laws to which it applies are exactly $P_t^I(\cdot \mid h)$ and $\widehat{P}_t^I(\cdot \mid h)$ from Step 1. By the definition equation 58 and that lemma,

$$\begin{aligned} a_t(I, h) &= \mathrm{Aff}\Big(P_t^I(\cdot \mid h), \widehat{P}_t^I(\cdot \mid h)\Big) \\ &= \prod_{e \in \mathcal{M}} \mathrm{Aff}(p_e, \widehat{p}_e) \\ &= \left(\prod_{e \in \mathcal{E}_t^{(2)}} (1 - d_{\mathrm{edge}})\right) \left(\prod_{e \in \mathcal{M} \setminus \mathcal{E}_t^{(2)}} \mathrm{Aff}(p_e, \widehat{p}_e)\right) \\ &= (1 - d_{\mathrm{edge}})^{Z_t} \prod_{e \in \mathcal{M} \setminus \mathcal{E}_t^{(2)}} \mathrm{Aff}(p_e, \widehat{p}_e). \end{aligned}$$

Every factor in the remaining product lies in $[0, 1]$, so

$$a_t(I, h) \le (1 - d_{\mathrm{edge}})^{Z_t}. \tag{97}$$

4. Multiply along the adaptive path and average. The preceding bound holds at each reached history, including histories under the midpoint law. Multiplying over rounds and using $Z = \sum_t Z_t$ yields

$$\prod_t a_t(I, H_{t-1}) \le (1 - d_{\mathrm{edge}})^Z.$$

Using the adaptive identity equation 60 first at fixed I, w and then averaging over I gives

$$\begin{aligned}
&\mathbb{E}_I h^2(P^I, \widehat{P}_w^I) \\
&\quad = 1 - \mathbb{E}_I \mathbb{E}_{H\sim\mathbb{M}_w^I} \prod_t a_t(I, H_{t-1}) \\
&\quad \geq 1 - \mathbb{E}_{I,H}(1 - d_{\mathrm{edge}})^Z \\
&\quad = \mathbb{E}_{I,H}[1 - (1 - d_{\mathrm{edge}})^Z],
\end{aligned}$$

where the joint expectation uses the conditional midpoint law given I, w. This proves equation 94. □

C.9 Nonlinear finite minimax bound

Intuition. When the query budget is a small enough fraction of the bank size, the preceding two results give

$$\text{risk} \;\gtrsim\; 1 - \exp\{-cN\eta^2/\overline{R}\} \simeq \min\{1, N\eta^2/\overline{R}\},$$

where $c > 0$ is a fixed numerical constant. The constant-probability collision event supplies order $N/\overline{R}$ missed edges; each contributes order η^2. For a small target tolerance ε, this forces $\overline{R} \gtrsim N\eta^2/\varepsilon$. The proof retains the finite constants and checks the prior-to-minimax step explicitly.

Proof of Theorem B.2. 1. Obtain the nonlinear bound at a fixed seed. Fix an arbitrary algorithm in $\mathfrak{A}(\overline{Q}, \overline{R})$ and a decision-rule seed w. Suppose $\overline{Q} \leq m/8$ and $1 \leq \overline{R} \leq N/16384$. By Lemma C.9 and Proposition C.8,

$$\begin{aligned}
&\mathbb{E}_{I\sim\Pi} h^2(P^I, \widehat{P}_w^I) \\
&\quad \geq \mathbb{E}_{I,H}\left[1 - (1 - d_{\mathrm{edge}})^Z\right] \\
&\quad \geq \frac{1}{4}\left[1 - (1 - d_{\mathrm{edge}})^{N/(8192\overline{R})}\right].
\end{aligned} \tag{98}$$

The exponent in the last display need not be an integer: on the event $Z \geq N/(8192\overline{R})$, monotonicity of $c \mapsto 1 - (1 - d_{\mathrm{edge}})^c$ gives the displayed lower bound. Using $1 - d \leq e^{-d}$ for $d \in [0, 1]$ and the one-edge estimate $d_{\mathrm{edge}} \geq \eta^2/12$ from equation 49, we obtain

$$\mathbb{E}_{I\sim\Pi} h^2(P^I, \widehat{P}_w^I) \geq \frac{1}{4}\left[1 - \exp\left\{-\frac{N\eta^2}{98304\overline{R}}\right\}\right]. \tag{99}$$

2. Average over seeds and pass from the prior to minimax risk. Finally, $d_{\mathrm{TV}} \geq h^2$ by Lemma H.1. The decision-rule seed W is independent of $I \sim \Pi$, so Tonelli's theorem gives

$$\begin{aligned}
\mathbb{E}_I \mathcal{R}_{\mathrm{TV}}(\mathcal{A}; P^I, q^I) &= \mathbb{E}_W \mathbb{E}_I d_{\mathrm{TV}}(P^I, \widehat{P}_W^I) \\
&\geq \mathbb{E}_W \mathbb{E}_I h^2(P^I, \widehat{P}_W^I) \\
&\geq \frac{1}{4}\left[1 - \exp\left\{-\frac{N\eta^2}{98304\overline{R}}\right\}\right].
\end{aligned} \tag{100}$$

At least one fixed instance has seed-averaged TV risk at least this prior average. Taking the supremum over the hard family and then the infimum over $\mathfrak{A}(\overline{Q}, \overline{R})$ proves equation 43. □

Proof of Corollary B.3. Suppose the first two alternatives in equation 45 fail. Then Theorem B.2 and equation 44 imply

$$\varepsilon \geq \frac{1}{4}\left[1 - \exp\left\{-\frac{N\eta^2}{98304\overline{R}}\right\}\right].$$

Since $\varepsilon \leq 1/8$, rearrangement gives

$$\begin{aligned}
\frac{N\eta^2}{98304\overline{R}} &\leq -\log(1 - 4\varepsilon) \\
&\leq \frac{4\varepsilon}{1 - 4\varepsilon} \leq 8\varepsilon.
\end{aligned}$$

Here $-\log(1-u) \le u/(1-u)$ for $u \in [0,1)$, obtained by integrating $(1-x)^{-1} \le (1-u)^{-1}$ over $x \in [0,u]$. Thus

$$\overline{R} \ge \frac{N\eta^2}{786432\varepsilon},$$

which is the third alternative in equation 45. □

D ALGORITHMIC CONSTRUCTION

We specify the randomized sampler $\mathcal{A}^{\mathrm{rnd}}_{\mathrm{pack}}$ of Theorem E.1, also denoted $\mathcal{A}_{\mathrm{pack}} := \mathcal{A}^{\mathrm{rnd}}_{\mathrm{pack}}$. All operations use only public parameters, the realized commit history, and replies of the single frozen oracle. The hidden forest, exact marginals, endpoint kernels, ranks, and response witnesses are never read.

The sampler flow in Figure 2 is illustrated concretely by a 15-vertex construction in Examples D.1, D.2 and E.2.

D.1 COMPLETE SAMPLER AND PUBLIC PARAMETERS

We assemble the sampler from read-only screens and commits, with public caps that bound its execution on every path.

Fix $N \ge 10$. The two resource controls are the degree cutoff $d \in \{9, \dots, N-1\}$ and readout-chunk parameter $J \in [N]$: the former sets the peeling threshold, while the latter sets the maximum readout size $B^{\mathrm{rd}} := \lceil N/J \rceil$. The accuracy inputs are a tail tolerance $\delta_{\mathrm{tail}} \in (0,1]$ and a screen-failure budget $\delta^{\mathrm{fail}} \in (0,1)$. The threshold floor $\underline{t}$ uses the standard choice equation 107. The draft generator is the zero-query choice equation 114 unless otherwise stated. The standard row selector keeps the highest-vote candidates on overflow. More generally, fix any admissible public selector S from Definition D.3; denote the resulting decision rule by $\mathcal{A}^{\mathrm{rnd}}_{\mathrm{pack}}[\mathsf{S}]$. The unadorned sampler uses the standard selector. The row-error radius and resulting screen resolution are derived quantities:

$$\varepsilon_0 := \frac{\varepsilon^{\mathrm{H}}}{\sqrt{N}}, \tag{101}$$

$$\delta := 4\varepsilon_0 + \delta_{\mathrm{tail}}. \tag{102}$$

The oracle condition equation 25 and Lemma H.1 imply, for every valid (y, j),

$$d_{\mathrm{TV}}\big(\mu_j(\cdot \mid y), q_j(\cdot \mid y)\big) \le \varepsilon_0. \tag{103}$$

The preprocessing and screen certificates use the row-TV bound equation 103 in both cases. In case (ii), equation 195 supplies this bound from the uniform forward row-KL condition. The forward-KL output guarantee is proved in Theorem F.3 using the joint-reference identity of Section F.2. The theorem requires feasible preprocessing and $\delta < \omega$; neither condition asks the sampler to inspect the hidden forest.

Define the public peel-phase cap

$$T_{\mathrm{peel}} := \left\lceil \frac{[\log(2N/(d+1))]_+}{\log(d/8)} \right\rceil. \tag{104}$$

Set

$$T_{\mathrm{scr}} := T_{\mathrm{peel}} + 1, \qquad \overline{R} := \left\lceil \frac{4NT_{\mathrm{peel}}}{d} \right\rceil + \lceil \log_2(N+1) \rceil + 2. \tag{105}$$

The caps are fixed before the first commit; they are not additional tuning parameters. Algorithm 2 is the full decision rule. Its one-time preprocessing, per-screen draft, and read-only screen are specified in the next three subsections. The named steps agree with Section 4 and Figure 2. Their identifiers name recurring operations, not commit rounds or oracle stages. Step 3 is expanded into Step 3.1, Step 3.2, and Step 3.3 in Algorithm 3. Proofs refer to these named steps as well as the algorithm's line numbers. In Algorithm 2, r counts completed nonempty commit rounds and k counts completed peel phases.

Algorithm 2 Randomized complete packed-screen sampler

Input: the public setup and parameters $\delta_{\mathrm{tail}}, \underline{t}, d, J, \delta^{\mathrm{fail}}$, an admissible Draft (standard by default), an admissible public row selector S (top votes by default), and access to q

Output: $\widehat{X} \in \mathcal{V}^N$ when preprocessing is feasible; otherwise INFEASIBLE

Step 1. Preprocess & initialize

Construct Prep by Definition D.1
if preprocessing reports infeasibility **then**
  return INFEASIBLE
end if
Set $T_{\mathrm{scr}}, \overline{R}$ by 105 and set $G \leftarrow \emptyset$, $x_G \leftarrow \emptyset$, $k \leftarrow 0$, and $r \leftarrow 0$
loop
  if $U_G = \emptyset$ **then**
    return the resulting assignment $\widehat{X}$
  end if
  Step 2. Draft & color
  Fix $f \leftarrow$ Draft using the current history and past replies
  Draw the coloring family by equation 118, with parameters from equation 117
  Step 3. Packed screen
  $A(H_G) \leftarrow \textsc{PackedScreen}(H_G; \mathsf{Prep}, f, \mathcal{C}^{\mathrm{val}}, (h_\tau(\cdot))_{\tau \in \mathcal{I}^{\mathrm{col}}}, d, J, q, \mathsf{S}, \mathcal{T})$ ▷ Algorithm 3
  Step 4. Claim & peel
  Form $\mathrm{cl.}(H_G)$ and $\mathcal{P}(H_G)$ by 106
  if $\mathcal{P}(H_G) = \emptyset$ or $k = T_{\mathrm{peel}}$ **then**
    break
  end if
  Freeze $\overline{\mathcal{P}} \leftarrow \mathcal{P}(H_G)$
  for $v \in \overline{\mathcal{P}}$ in increasing position order **do**
    if $r = \overline{R} - 1$ **then** ▷ GUARD
      COMMIT(U_G) and $r \leftarrow r + 1$ **return** the resulting assignment $\widehat{X}$
    end if
    COMMIT$(\{v\})$
    $r \leftarrow r + 1$
  end for
  $k \leftarrow k + 1$
end loop
Step 5. Terminal graph
$H_\star \leftarrow H_G$, $U_\star \leftarrow U_G$, and
  $\widehat{E}_\star \leftarrow \{\{u, v\} \subseteq U_\star : v \in A_u(H_\star) \text{ or } u \in A_v(H_\star)\}$
$\widehat{F} \leftarrow (U_\star, \widehat{E}_\star)$
while $\widehat{F}$ contains a cycle **do**
  if $r = \overline{R} - 1$ **then** ▷ GUARD
    COMMIT(U_G) and $r \leftarrow r + 1$; **return** $\widehat{X}$
  end if
  Choose a maximum-degree cycle vertex v; break ties by smallest index
  COMMIT$(\{v\})$; $r \leftarrow r + 1$; delete v and its incident edges from $\widehat{F}$
end while
Step 6. Centroid commits
while $\widehat{F}$ has a vertex **do**
  In each component choose its smallest-label centroid; call their set B
  if $r = \overline{R} - 1$ **then** ▷ GUARD
    COMMIT(U_G) and $r \leftarrow r + 1$ **return** the resulting assignment $\widehat{X}$
  end if
  COMMIT(B)
  $r \leftarrow r + 1$
  Delete B and its incident edges from $\widehat{F}$
end while
return the resulting assignment $\widehat{X}$

Given the screen output at H_G, recall the incoming claim count and peel set of Equation (7):

$$\mathrm{cl}_v(H_G) := \big|\{u \in U_G : v \in A_u(H_G)\}\big|, \qquad \mathcal{P}(H_G) := \{v \in U_G : \mathrm{cl}_v(H_G) > d/2\}. \tag{106}$$

The terminal row graph on $U_\star$ is constructed in Step 5 of Algorithm 2.

At any history, $\mathrm{Commit}(B)$ means the product-commit operation of Definition A.9: submit the current history-compatible commit state 27, read $q_j(\cdot \mid y)$ for $j \in B$, draw independently from the corresponding product law, and append the realized values to the history. This is the sampler's only history-changing operation. The batch B is fixed before the reply; this submission adds one to R and nothing to Q_{cf}.

1. *Screen, then test whether to stop.* Each pass first computes rows at the current history (Step 3) and only then tests whether the peel set is empty or the phase cap has been reached (Step 4). In particular, the last permitted peel phase is followed by another screen: the terminal graph never uses rows from before those commits. A nonempty peel set is frozen before its singleton commits; its members are committed in position order, without an intervening screen or recomputation of that set.
2. *Repair cycles in the terminal estimate.* In Step 5, the working graph $\widehat{F}$ initially equals $(U_\star, \widehat{E}_\star)$. A cycle vertex belongs to at least one simple cycle of this graph. The selection rule is
$$v = \min \quad \operatorname*{arg\,max}_{\substack{u \in V(\widehat{F}) \\ u \text{ lies on a cycle of } \widehat{F}}} \quad d_{\widehat{F}}(u).$$
The degree is measured in the whole current working graph. Commit v using the current history, delete v and its incident edges, and recompute cycle membership and degrees in the remaining graph. No new screen is run: the graph changes only by vertex deletion, whereas each commit queries fresh rows at the updated history. Once it is a forest, proceed to Step 6.
3. GUARD*: Round-cap fallback*. Before each peel, repair, or centroid commit, the check $r = \overline{R} - 1$ reserves the last permitted round for one product commit of all remaining positions. This ends the execution even if repair is incomplete. The cap bounds the round count on every path; this final product law need not approximate the true joint conditional.
4. *Separate the observable past from the random sources.* All ties use the public position or token order. The public transcript $\mathcal{T}$ starts with preprocessing and is updated after each query, coloring draw, selector output, and commit. It records only information already available to the decision rule, not unrevealed future randomness. The decision-rule seed W contains the coloring blocks and auxiliary draft randomness. Product-commit draws use a separate random source: even after conditioning on $W = w$, those draws remain random and are integrated into $\widehat{P}^q_w$.

D.2 ONE-TIME ALL-MASK PREPROCESSING

In Step 1 (Preprocess & initialize), one all-mask submission supplies the local vocabulary banks $\widehat{\mathcal{B}}_i$ and tail representatives b_i reused by every later screen.

Definition D.1 (All-mask preprocessing)**.** Use the standard floor

$$\underline{t} := 4 \max\{\varepsilon_0, C/V\}. \tag{107}$$

Here $C \geq 1$ is the public RF (Equation (21)) constant when an envelope is specified; otherwise use $C = 1$. This keeps the threshold above both the row-noise and frequency-envelope scales. If $\underline{t} > 1$, return INFEASIBLE. Otherwise put $J^{\mathrm{grid}} := \lceil \log_2(1/\underline{t}) \rceil$ and set

$$\mathcal{G} := \{\max\{\underline{t}, 2^{-m}\} : 0 \leq m \leq J^{\mathrm{grid}}\}. \tag{108}$$

Form the feasible set

$$\mathcal{G}^{\mathrm{feas}} := \{t \in \mathcal{G} : \min\{1, Lt^\alpha\} \leq \delta_{\mathrm{tail}}\}. \tag{109}$$

If it is empty, return INFEASIBLE before any submission or commit. Otherwise set $t := \max \mathcal{G}^{\mathrm{feas}}$.

Submit $y^{\perp} = (\mathsf{MASK}, \ldots, \mathsf{MASK})$ once and, for each position i, form its noisy marginal and *local vocabulary bank*:

$$\widetilde{\pi}_i := q_i(\cdot \mid y^{\perp}), \tag{110}$$

$$\widehat{\mathcal{B}}_i := \{a \in \mathcal{V} : \widetilde{\pi}_i(a) \geq t - \varepsilon_0\}. \tag{111}$$

Here $\widehat{\mathcal{B}}_i \subseteq \mathcal{V}$ is the subset of vocabulary tokens selected by the buffered all-mask marginal threshold $t - \varepsilon_0$ for explicit source-value probing. Tokens outside this set are represented in the response test by b_i below; the set does not restrict commit outputs. Enumerate the vocabulary bank $\widehat{\mathcal{B}}_i$ as $\widehat{\mathcal{B}}_i = \{\widehat{a}_{i,1}, \ldots, \widehat{a}_{i,\ell_i}\}$ in the public token order, with $\ell_i := |\widehat{\mathcal{B}}_i|$ and $\ell_{\max} := \max_i \ell_i$, and choose

$$b_i := \min_{\prec}\big(\mathcal{V} \setminus \widehat{\mathcal{B}}_i\big). \tag{112}$$

The floor ensures that this complement is nonempty; Lemma E.3 proves both existence and its exact-tail certificate. Return the reusable tuple

$$\mathsf{Prep} := \Big(t, (\widetilde{\pi}_i, \widehat{\mathcal{B}}_i, b_i, \ell_i)_{i\in[N]}, \ell_{\max}\Big). \tag{113}$$

Apart from the all-mask submission, preprocessing is deterministic and uses no commit draw.

D.3 Per-screen draft and source columns

We first fix the common background for one screen, then define the source assignments whose oracle replies will be compared.

Example (Standard zero-query draft)**.** Unless stated otherwise, Step 2 uses the standard draft

$$f_i := \arg\max_{a\in\mathcal{V}} \widetilde{\pi}_i(a), \qquad i \in U_G, \tag{114}$$

with ties resolved by $\prec$. It reuses the initial all-mask rows, works even for empty vocabulary banks, and requires no additional submission or adaptive stage.

Source columns. For the source assignments in Step 3.2, recall that $\widehat{\mathcal{B}}_i = \{\widehat{a}_{i,1}, \ldots, \widehat{a}_{i,\ell_i}\}$ lists the local vocabulary bank in the public token order, not in marginal-probability order. Here $\ell_i = |\widehat{\mathcal{B}}_i|$ and $\ell_{\max} = \max_{i\in[N]} \ell_i$ were fixed by preprocessing. For a fixed screen draft, set

$$\mathcal{I}^{\mathrm{src}} := [\ell_{\max}] \cup \{\mathsf{T}\}, \qquad \varphi_\kappa(i; f) := \begin{cases} \widehat{a}_{i,\kappa}, & \kappa \in [\ell_i], \\ f_i, & \kappa \in [\ell_{\max}] \setminus [\ell_i], \\ b_i, & \kappa = \mathsf{T}. \end{cases} \tag{115}$$

The column order is $1, \ldots, \ell_{\max}, \mathsf{T}$. Thus a bank-column index denotes a position-dependent assignment, not one common vocabulary token. All screen quantities below depend on the fixed f; this dependence is suppressed in row-family, vote, and graph notation. When $\ell_{\max} = 0$, the screen returns empty rows without submitting even the tail column.

For resource accounting, the number of submitted columns is

$$\Lambda := \begin{cases} \ell_{\max} + 1, & \ell_{\max} \geq 1, \\ 0, & \ell_{\max} = 0. \end{cases} \tag{116}$$

Example D.1 (A column indexes local vocabulary banks, not a common token)**.** Take $N = 15$, $G = \emptyset$, and a vocabulary whose first three tokens in the public order are $a \prec b \prec c$ (further tokens may follow). Suppose the local vocabulary banks $\widehat{\mathcal{B}}_i$ from Step 1 (Definition D.1) and the fixed draft are

$$\widehat{\mathcal{B}}_1 = \{a, c\}, \quad f_1 = a, \qquad \widehat{\mathcal{B}}_i = \{b\}, \quad f_i = b \quad (i \in [15] \setminus \{1\}).$$

This specifies the screen inputs, not a target distribution or an oracle. In particular, $\ell_1 = 2$, $\ell_i = 1$ for $i \neq 1$, and $\ell_{\max} = 2$. The first token outside each bank is $b_1 = b$ and $b_i = a$ for $i \neq 1$. Substituting into equation 115 gives

position	$\kappa = 1$	$\kappa = 2$	$\kappa = \mathsf{T}$
1	a	c	b
$i \neq 1$	b	$f_i = b$	a

$$\mathcal{I}^{\mathrm{src}} = \{1, 2, \mathsf{T}\}, \quad \Lambda = 3.$$

Thus column 1 assigns a at position 1 but b at position 4. Column 2 uses the second bank token at position 1 and draft padding at position 4, whose bank has no second token. The tail column uses the representative outside each bank. The same three columns are reused for every source color and readout chunk.

Three different indices. The screen uses the following roles; their domains and sampling law are specified in Section D.4.

Symbol	What it selects	What it does not select
τ	One entire coloring $h_\tau : [N] \to \mathcal{C}^{\mathrm{val}}$	A color within that coloring
$c^{\mathrm{src}}, c^{\mathrm{rd}}$	Source and readout colors under the chosen h_τ	Vocabulary tokens
$\kappa \in \mathcal{I}^{\mathrm{src}}$	One slot in the vocabulary bank $\widehat{\mathcal{B}}_i$ or its tail representative, separately at each source i	A color or one common token

Thus fixing τ fixes all position colors; fixing c^{src} then fixes which residual positions are sources. Varying κ changes the tokens assigned to those sources, without recoloring them. The concrete substitution into the submitted-state definition is in Example D.2.

Definition D.2 (Zero-query screen draft)**.** Before each screen, Draft fixes a token vector $f \in \mathcal{V}^{U_G}$ using only public data, the realized history, cached oracle replies, and auxiliary seed randomness independent of the coloring and product-commit random sources. It terminates without a submission or commit and without observing current or future colors. The draft is fixed across all colorings, chunks, source colors, and columns of this screen; it may be reused or changed at the next screen. A draft token f_i need not belong to the vocabulary bank $\widehat{\mathcal{B}}_i$ and is never committed directly.

D.4 Packed randomized-color screen

We generate one color family, pack its readouts into chunks, and compare source-column replies to form the screen's neighbor claims.

Random color generation. For Step 2, set

$$p_{\mathrm{rnd}} := 8(d+1), \qquad M_{\mathrm{rnd}} := \left\lceil 8 \log \frac{T_{\mathrm{scr}} N^2}{\delta^{\mathrm{fail}}} \right\rceil. \tag{117}$$

Write $p = p_{\mathrm{rnd}}$, $M = M_{\mathrm{rnd}}$, $\mathcal{C}^{\mathrm{val}} = [p]$, and $\mathcal{I}^{\mathrm{col}} = [M]$. These are deterministic public parameters. At each reached screen s, conditional on the realized past including its completed draft, the decision rule reveals a fresh block W_s with law

$$h_\tau(i) \overset{\text{i.i.d.}}{\sim} \mathrm{Unif}([p]), \qquad (\tau, i) \in [M] \times [N]. \tag{118}$$

The blocks $(W_s)_{s \le T_{\mathrm{scr}}}$ are mutually independent and independent of the auxiliary draft random source and primitive commit randomness; no current or future block is revealed during draft generation. The same family is used for all chunks, source colors, and columns of this screen: Algorithm 3 does not draw or refresh colors.

Accuracy calibration for the main benchmarks. The choices $\delta^{\mathrm{fail}} = \varepsilon^{\mathrm{H}}/2$ in case (i) and $\delta^{\mathrm{fail}} = \varepsilon^{\mathrm{KL}}/(2N \log V)$ in case (ii) specialize equation 117 to

$$p_{\mathrm{rnd}} = 8(d+1), \qquad M_{\mathrm{rnd}} := \begin{cases} \left\lceil 8 \log \dfrac{2T_{\mathrm{scr}} N^2}{\varepsilon^{\mathrm{H}}} \right\rceil, & \text{case (i)}, \\ \left\lceil 8 \log \dfrac{2T_{\mathrm{scr}} N^3 \log V}{\varepsilon^{\mathrm{KL}}} \right\rceil, & \text{case (ii)} \end{cases} \tag{119}$$

In case (ii), all radius-dependent preprocessing and screen quantities use $\sqrt{\varepsilon^{\mathrm{KL}}/2}$ in place of ε^{H}, as in Corollary F.4. The decision rule and commit rule are unchanged.

Readout chunks. Given the colors, Step 3.1 partitions each readout class deterministically so that each probe masks at most B^{rd} positions. For every $\tau \in \mathcal{I}^{\mathrm{col}}$ and $c^{\mathrm{rd}} \in \mathcal{C}^{\mathrm{val}}$, let

$$\mathcal{C}_{\tau,c^{\mathrm{rd}}}(H_G) := \{j \in U_G : h_\tau(j) = c^{\mathrm{rd}}\}, \qquad n_{\tau,c^{\mathrm{rd}}}(H_G) := |\mathcal{C}_{\tau,c^{\mathrm{rd}}}(H_G)|. \tag{120}$$

For a nonempty class, put $n := n_{\tau,c^{\mathrm{rd}}}(H_G)$, list its elements in increasing position order as $j_{\tau,c^{\mathrm{rd}},1} < \cdots < j_{\tau,c^{\mathrm{rd}},n}$, and partition it into

$$\mathcal{C}_{\tau,c^{\mathrm{rd}},m}(H_G) := \left\{j_{\tau,c^{\mathrm{rd}},\ell} : (m-1)B^{\mathrm{rd}} < \ell \le \min\{mB^{\mathrm{rd}}, n\}\right\}, \quad 1 \le m \le \left\lceil \frac{n}{B^{\mathrm{rd}}} \right\rceil. \tag{121}$$

Empty color classes create no chunk. The chunks depend on the current residual set U_G, form a disjoint cover of each color class, and have size at most B^{rd}. Write $\mathcal{C}_\tau(j)$ for the unique chunk containing j.

Submitted states. In Step 3.2, each probe masks one readout chunk and changes one source color's assignment, while keeping the remaining background fixed. For a nonempty readout chunk $\mathcal{C} = \mathcal{C}_{\tau,c^{\mathrm{rd}},m}(H_G)$, $\kappa \in \mathcal{I}^{\mathrm{src}}$, and a source color $c^{\mathrm{src}} \in \mathcal{C}^{\mathrm{val}} \setminus \{c^{\mathrm{rd}}\}$, define the probe state

$$(y_{\kappa,\tau,c^{\mathrm{src}}}(H_G, f, \mathcal{C}))_k := \begin{cases} x_k, & k \in G, \\ \mathsf{MASK}, & k \in \mathcal{C}, \\ \varphi_\kappa(k; f), & k \in U_G \setminus \mathcal{C} \text{ and } h_\tau(k) = c^{\mathrm{src}}, \\ f_k, & \text{otherwise.} \end{cases} \tag{122}$$

Here c^{rd} is the readout color, c^{src} is the source color, and τ selects the entire coloring. Since $c^{\mathrm{src}} \neq c^{\mathrm{rd}}$, the source class and readout chunk are disjoint. A missing bank entry leaves its source coordinate at the draft value. Column T instead assigns b_k to the source class only; all other unmasked residual coordinates stay at f_k. There is no separate all-draft baseline submission, and the tail-column state depends on the source color.

Example D.2 (From one coloring to three submitted states)**.** Continue Example D.1. Set $d = J = 9$, so $p = 8(d+1) = 80$ and $B^{\mathrm{rd}} = \lceil 15/9 \rceil = 2$. Fix one index $\tau \in [M]$ and suppose Step 2 draws the coloring

$$h_\tau(i) = \begin{cases} 1, & i \in \{1, 4\}, \\ 2, & i \in \{10, 13\}, \\ 3, & i \in [15] \setminus \{1, 4, 10, 13\}. \end{cases}$$

These are three of the 80 available colors, not a change to the color-count parameter. With $c^{\mathrm{src}} = 1$ and $c^{\mathrm{rd}} = 2$, Step 3.1 applies equation 120–equation 121 to obtain

$$\mathcal{C}_{\tau,1}(H_G) = \{1, 4\}, \qquad \mathcal{C}_{\tau,2}(H_G) = \{10, 13\}, \qquad \mathcal{C} = \mathcal{C}_{\tau,2,1}(H_G) = \{10, 13\}.$$

For this example only, write $y^\kappa := y_{\kappa,\tau,1}(H_G, f, \{10, 13\})$. Because $G = \emptyset$, substituting the bank columns into equation 122 yields

	src (source)		target (readout)	
κ	y^κ_1	y^κ_4	y^κ_{10}	y^κ_{13}
1	a	b	MASK	MASK
2	c	b	MASK	MASK
T	b	a	MASK	MASK

$$y^\kappa_k = f_k = b \quad (k \in [15] \setminus \{1, 4, 10, 13\}).$$

These remaining positions are neither sources nor masked readouts. Since $G = \emptyset$, they take the *otherwise* branch of equation 122 and retain their draft values f_k, which equal b by Example D.1. In Step 3.2, each of the three submissions returns both $q_{10}(\cdot \mid y^\kappa)$ and $q_{13}(\cdot \mid y^\kappa)$. Their role is to test dependence, not to commit the source tokens: G remains empty throughout. In this particular example, column 1 happens to agree with the draft at all unmasked positions; it is still one of the three columns, not an extra baseline submission. Figure 5 shows these exact assignments.

TV tests and aggregation. Within each coloring, Step 3.3 compares all pairs of source-column replies at the same readout; a strict majority across colorings determines each candidate row. For distinct $i, j \in U_G$, when $h_\tau(i) \neq h_\tau(j)$, define the observable rows

$$q^{\mathrm{scr}}_{\tau,\kappa}(i \to j \mid H_G, f) := q_j(\cdot \mid y_{\kappa,\tau,h_\tau(i)}(H_G, f, \mathcal{C}_\tau(j))), \quad \kappa \in \mathcal{I}^{\mathrm{src}}, \tag{123}$$

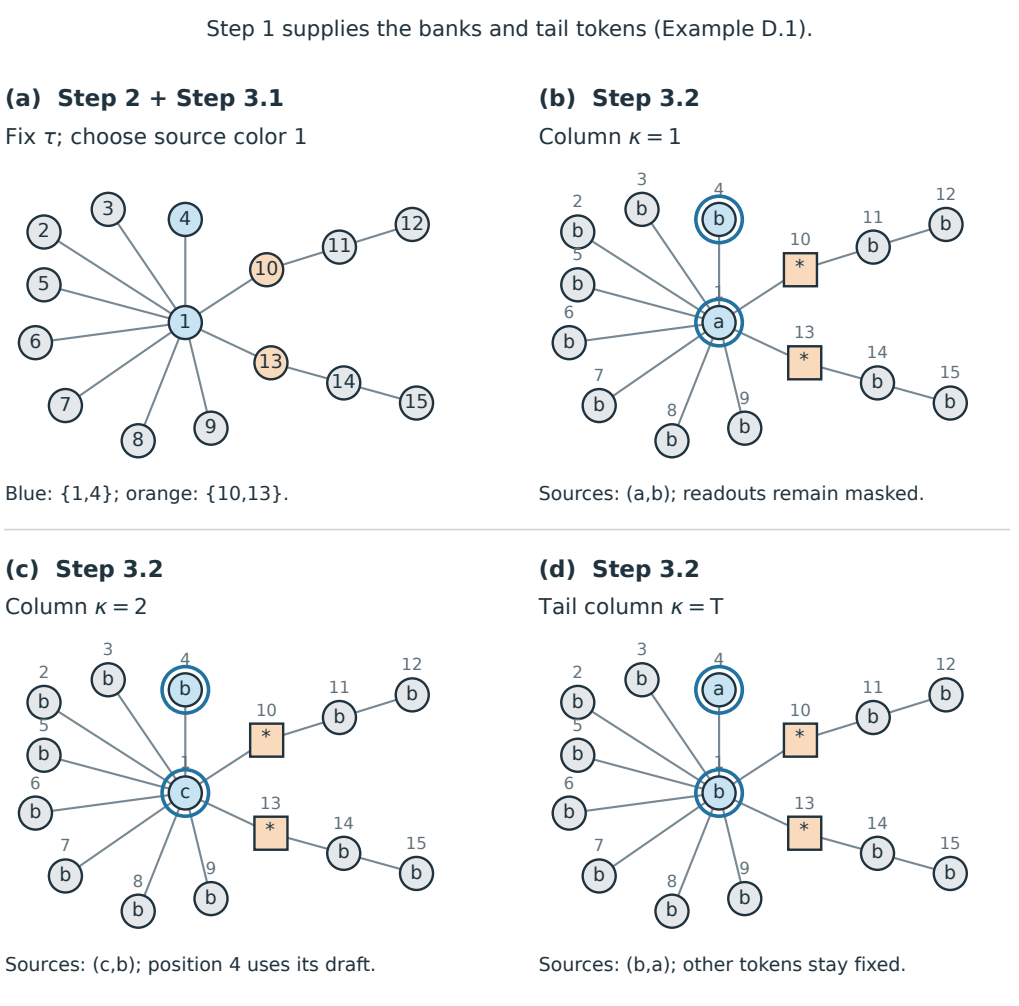


Figure 5: The coloring and three submitted states of Example D.2, labeled by the steps of Algorithms 2–3. All 15 positions are drawn separately at fixed locations. Fill denotes the color under τ; square outlines mark masked readouts and double outlines mark sources. Small outer numbers are position indices, and inner symbols in (b)–(d) are submitted tokens. No vertex is committed by these probes. The hidden edges are shown only to explain the construction.

and the observable row family

$$\mathcal{Q}_\tau(i \to j \mid H_G) := \{q^{\mathrm{scr}}_{\tau,\kappa}(i \to j \mid H_G, f) : \kappa \in \mathcal{I}^{\mathrm{src}}\}. \tag{124}$$

Its TV diameter and the corresponding vote are

$$D_\tau(i, j \mid H_G) := \max_{q,q' \in \mathcal{Q}_\tau(i \to j \mid H_G)} d_{\mathrm{TV}}(q, q'), \tag{125}$$

$$Y_\tau(i, j \mid H_G) := \begin{cases} \mathbf{1}\{D_\tau(i, j \mid H_G) > 2\varepsilon_0\}, & h_\tau(i) \neq h_\tau(j), \\ 0, & h_\tau(i) = h_\tau(j). \end{cases} \tag{126}$$

When the colors coincide, the row family and diameter need not be formed. The vote is set to zero and never refers to an unsubmitted probe. If $\ell_{\max} = 0$, the screen returns empty rows before forming these objects.

Computing the diameter requires no additional submissions. It compares every pair of bank-plus-tail rows, not just each bank row against the tail column.

For each readout j, define the candidate row by

$$\widetilde{A}_j(H_G) := \left\{ i \in U_G \setminus \{j\} : \sum_{\tau \in \mathcal{I}^{\mathrm{col}}} Y_\tau(i, j \mid H_G) > \frac{M}{2} \right\}, \tag{127}$$

Definition D.3 (Admissible public row selector)**.** Let $\mathcal{T}$ contain the public transcript available when the screen has finished, including its colors and votes and any retained public decision-rule state. A selector S is a fixed, terminating, deterministic measurable rule. Given j, a candidate set $C \subseteq U_G \setminus \{j\}$, the cutoff d, and $\mathcal{T}$, its output has the form

$$\mathsf{S}(j, C, d; \mathcal{T}) = \begin{cases} C, & |C| \leq d, \\ C \setminus C_{\mathrm{drop}}, & |C| > d. \end{cases}$$

On overflow, $C_{\mathrm{drop}} \subseteq C$ is the set of positions that the rule chooses to remove from these inputs, with $|C_{\mathrm{drop}}| \geq |C| - d$. Thus $|C \setminus C_{\mathrm{drop}}| = |C| - |C_{\mathrm{drop}}| \leq d$; the rule may return fewer than d positions, including none. It makes no oracle submission or commit and reads neither hidden target information nor unobserved future randomness or oracle replies.

The returned row $A_j(H_G)$ is the output of this selector applied to $C = \widetilde{A}_j(H_G)$:

$$A_j(H_G) := \mathsf{S}\big(j, \widetilde{A}_j(H_G), d; \mathcal{T}\big). \tag{128}$$

Example (Standard top-vote row selector)**.** The standard choice of S for returning $A_j(H_G)$ in equation 128 orders the positions $i_1, \ldots, i_{|C|}$ of C by decreasing total vote $\sum_{\tau \in \mathcal{I}^{\mathrm{col}}} Y_\tau(i, j \mid H_G)$, breaking ties by increasing position index. It returns

$$\mathsf{S}(j, C, d; \mathcal{T}) = \{i_\ell : 1 \leq \ell \leq \min\{d, |C|\}\}.$$

Thus it keeps all of C when $|C| \leq d$ and otherwise removes the positions ranked after d. The empty input gives the empty set. Taking $C = \widetilde{A}_j(H_G)$ gives the standard returned row $A_j(H_G)$.

The size cap alone would not suffice: deleting a candidate from a nonoverflowing row can destroy low-degree exactness. When $\ell_{\max} = 0$, every returned row is defined to be empty.

Remark (Alternative overflow rule). Returning $\emptyset$ on overflow is another admissible selector. The standard sampler uses top-vote selection; the invariance proposition compares the guarantees of these rules, not their realized performance.

Given its inputs, public transcript, and the frozen oracle, Algorithm 3 is deterministic; the full decision rule in Algorithm 2 supplies its color family in Step 2. It expands Step 3 of Algorithm 1. Within Step 3, Step 3.3 computes votes after each chunk's probes, then forms and selects rows after all colorings.

Algorithm 3 Packed color screen with a fixed draft

Input: H_G, Prep, f, a color set $\mathcal{C}^{\mathrm{val}}$, and a family $(h_\tau(\cdot))_{\tau\in\mathcal{I}^{\mathrm{col}}}$, d, J, read access to q, S, and the current public transcript $\mathcal{T}$
Output: $(A_j(H_G))_{j\in U_G}$; the history is unchanged

if $\ell_{\max}=0$ or $U_G=\emptyset$ **then**
  return $(\emptyset)_{j\in U_G}$ without a counterfactual submission
end if
Initialize all votes in 126 to zero
for $\tau\in\mathcal{I}^{\mathrm{col}}$ **do**
  Step 3.1. Readout chunks
  Form the color classes and current readout chunks by 120–121
  for each nonempty readout chunk $\mathcal{C}=\mathcal{C}_{\tau,c^{\mathrm{rd}},m}(H_G)$ **do**
    Step 3.2. Packed probes
    for $\kappa\in\mathcal{I}^{\mathrm{src}}$ **do**
      for $c^{\mathrm{src}}\in\mathcal{C}^{\mathrm{val}}\setminus\{c^{\mathrm{rd}}\}$ **do**
        Submit $y_{\kappa,\tau,c^{\mathrm{src}}}(H_G,f,\mathcal{C})$ and read all rows $j\in\mathcal{C}$
      end for
    end for
    Step 3.3. Votes & row selection
    Form the row-family diameters and votes for all $i\in U_G\setminus\{j\}$ and $j\in\mathcal{C}$ using 124–126
  end for
end for
Step 3.3. Votes & row selection (continued)
Form the candidate rows by 127
Append the current screen's replies, colors, and votes to $\mathcal{T}$
Apply the public selector by 128
return $(A_j(H_G))_{j\in U_G}$

Every bank- or tail-column state is a noncommit counterfactual submission, and the submitted states agree with H_G on G. One submitted state returns all rows indexed by its readout chunk and contributes one to Q_{cf}, regardless of $|\mathcal{C}|$ (Definition A.11). The tail-column reply for a fixed $(\tau,\mathcal{C},c^{\mathrm{src}})$ is reused in the diameter computation for that source color, not across different source colors.

D.5 Roles of the construction and proof route

The overview in Section E.2 maps the steps above to their structural, resource, and error guarantees. It also gives the proof routes for both accuracy cases and their specialization to Theorem 3.1.

Remark (Scope of cycle repair). Cycle repair avoids an immediate all-residual product commit merely because the terminal estimate has a cycle. A repaired forest need not contain every true residual edge. The guarantees still charge failed-screen paths to the screen-failure budget; they do not assert an accuracy improvement over the one-batch alternative.

E CASE (I): HELLINGER-TO-TV UPPER BOUND

We prove the finite TV guarantee for the sampler in Appendix D; Section E.2 gives the proof map. Screen certification, peeling, and resource counts support both accuracy cases via equation 195. The error analysis here proves case (i); Appendix F proves case (ii). Appendix G specializes both to Theorem 3.1.

E.1 FINITE UPPER BOUND

The finite theorem uses the standard preprocessing floor of Appendix D.2; its query bound uses the realized vocabulary-bank sizes $\ell_i = |\widehat{\mathcal{B}}_i|$. Together with Lemma G.1, it yields case (i) of Theorem 3.1 through Corollary G.3.

Theorem E.1 (Case (i): finite randomized Hellinger-to-TV upper bound)**.** *We work in case (i), with the oracle condition of Assumption A.7. Fix $N \geq 10$ and the objects of Appendix A. Let $\mathcal{A}^{\mathrm{rnd}}_{\mathrm{pack}}$ be Algorithm 2 with the standard zero-query draft of equation 114 and $\delta^{\mathrm{fail}} \in (0,1)$. Suppose preprocessing is feasible and*

$$\delta < \omega. \tag{129}$$

Here $p_{\mathrm{rnd}}, M_{\mathrm{rnd}}$ are defined in equation 117, and $T_{\mathrm{scr}}, \overline{R}$ in equation 105; Λ is $\ell_{\max} + 1$ when $\ell_{\max} \geq 1$ and zero when $\ell_{\max} = 0$, as in equation 116. Then, for every admissible frozen oracle, pathwise over the random colors and product-commit draws,

$$Q_{\mathrm{cf}} \leq \Lambda T_{\mathrm{scr}} M_{\mathrm{rnd}} p_{\mathrm{rnd}} (p_{\mathrm{rnd}} + J), \tag{130}$$

$$R \leq \overline{R}. \tag{131}$$

Moreover,

$$\mathcal{R}_{\mathrm{TV}}(\mathcal{A}^{\mathrm{rnd}}_{\mathrm{pack}}; P, q) \leq \varepsilon^{\mathrm{H}} + \delta^{\mathrm{fail}} \quad \textit{for every } q \in \mathcal{Q}^{\mathrm{Hel}}(P; \varepsilon^{\mathrm{H}}). \tag{132}$$

Consequently,

$$\overline{\mathcal{R}}_{\mathrm{TV}}(\mathcal{A}^{\mathrm{rnd}}_{\mathrm{pack}}; P) \leq \varepsilon^{\mathrm{H}} + \delta^{\mathrm{fail}}. \tag{133}$$

In particular, if $\varepsilon^{\mathrm{H}} > 0$ and

$$\delta^{\mathrm{fail}} := \frac{\varepsilon^{\mathrm{H}}}{2}, \tag{134}$$

then $\overline{\mathcal{R}}_{\mathrm{TV}}(\mathcal{A}^{\mathrm{rnd}}_{\mathrm{pack}}; P) \leq 3\varepsilon^{\mathrm{H}}/2$. If the distinguished all-mask readout is charged, then $Q_{\mathrm{nc}} = Q_{\mathrm{cf}} + 1$ by equation 34. Thus the bound for Q_{nc} is the right-hand side of equation 130 plus one; Q_{cf} still excludes that readout. Including that initial stage, the adaptive oracle depth is at most $D \leq 1 + R + T_{\mathrm{scr}}$.

E.2 PROOF OVERVIEW

The proof turns local screen certificates into safe parallel commits: accurate low-degree rows make peeling contract the high-degree core, after which the remaining forest is known and its components can be sampled in parallel. We then bound resources and output error separately. Table 3 follows this chain under the finite hypotheses above. The structural certificates hold when every reached screen succeeds; the resource caps hold on every path. The same construction supports both accuracy cases, with different output-error arguments.

Proof task	Key argument and conclusion	Algorithm steps	References
Certify the vocabulary banks	Preprocessing gives a nonempty complement of $\widehat{\mathcal{B}}_i$, whose tokens have true marginal mass below t, certifying the tail representative. Under the standard floor, (RF) bounds $\ell_i \le (4C/t)^{1/s}$.	Step 1	Lemma E.3
Recover low-degree neighborhoods	Separating colors reduce packed probes to one-source comparisons at a fixed boundary. RT–UEN and the screening margin make their votes correct; fresh-color majorities give $A_j(H_G) = \mathcal{N}_{F_G}(j)$ for degree at most d, except on reached-screen failures of total probability at most δ^{fail}.	Step 2–Step 3; Step 3.1–Step 3.3	Lemmas E.2, E.4 and E.5; Lemma E.6
Peel the high-degree core	Let n_k^{hi} count vertices of degree above d before phase k. On successful paths, the forest degree sum gives $n_{k+1}^{\mathrm{hi}} \le (4/d) n_k^{\mathrm{hi}}$. Either stopping test leaves maximum degree at most d, certifying the true residual forest.	Step 4–Step 5	Lemma E.7
Complete safe parallel sampling	Singleton peels and one centroid per true residual component are safe. Exact-row products then equal joint batch conditionals. Centroid deletion halves component sizes, giving logarithmically many terminal rounds on successful paths.	Step 4, Step 6	Lemmas E.2, E.8 and E.9
Bound all-path resources	Count screen loops for Q_{cf}, and peel and centroid commits for rounds. GUARD caps rounds even on failed-screen paths. Each screen is one parallel stage: $D \le 1 + R + T_{\mathrm{scr}}$. Q_{cf} excludes preprocessing and commits.	All: Step 1–Step 6, plus GUARD	Lemma E.8
Case (i): bound output TV	Each coordinate is committed once, so adaptive Hellinger composition bounds an analysis-only safe rule's oracle error. Coupling to that rule adds at most δ^{fail}, giving seed-averaged TV at most $\varepsilon^{\mathrm{H}} + \delta^{\mathrm{fail}}$.	All: Step 1–Step 6; output-law analysis	Lemmas E.9 to E.11; Theorem E.1
Case (ii): bound output KL	With forward row KL at most κ, the joint-reference identity separates row error from batch total correlation. The latter vanishes on successful reference paths, giving seed-averaged KL at most $N\kappa + N \log V \, \delta^{\mathrm{fail}}$.	All: Step 1–Step 6; output-law analysis	Lemma F.2 and Theorem F.3
Recover the main theorem	Verify preprocessing feasibility and screen separation for the main-text scaling, then specialize the finite bounds and choose the degree cutoff. Both cases yield Theorem 3.1.	All: Step 1–Step 6; parameter choices	Lemma G.1 and Corollary G.3

Table 3: Proof steps for the upper bound. Step labels link to Algorithms 2–3.

Two distinctions guide the proof:

- High-degree rows need only satisfy the size cap $|A_j(H_G)| \le d$. Accurate low-degree rows suffice to force contraction, even when high-degree rows are incorrect.
- For case (i), an analysis-only safe rule switches to singleton completion at the first failed screen (and before any cap fallback). We compare its full output law with the implemented sampler's law; no conditioning of the output on screen success is needed.

Combining these bounds gives Equation (192):

$$\mathbb{E}_W d_{\mathrm{TV}}\big(P, \widehat{P}_W^q\big) \le \underbrace{\varepsilon^{\mathrm{H}}}_{\text{oracle error at commits}} + \underbrace{\delta^{\mathrm{fail}}}_{\text{failed-screen coupling}}.$$

E.3 Steps 1–3: Preprocessing and exact low-degree certification

We first establish the residual Markov and tail certificates, then prove the fresh-color success bound and the low-degree screen contract. These certify the vocabulary banks from Step 1 (Preprocess & initialize), the fresh colors from Step 2 (Draft & color), and the rows returned by Step 3 (Packed screen).

The next lemma derives the residual Markov property from Equation (1). Recall from equation 14 that $U_G = [N] \setminus G$ and $F_G = F[U_G]$. Its row identity is used in Step 3.2 (Packed probes), and its component independence justifies Step 6 (Centroid commits).

Lemma E.2 (Residual Markov property)**.** *Fix a history $H_G = (G, x_G)$. Conditional on $X_G = x_G$, the variables in distinct connected components of F_G are independent. Moreover, fix $j \in U_G$ and the values of all residual neighbors of j. The exact conditional row of X_j is unchanged if any residual nonneighbor is revealed, masked, changed, or marginalized.*

Proof. 1. Factor the conditional law over residual components. Recall the forest factorization equation 12: $P(x) = \prod_{u \in \mathrm{Roots}(F)} \pi_u(x_u) \prod_{j \notin \mathrm{Roots}(F)} K_{\mathrm{pa}(j)\to j}(x_j \mid x_{\mathrm{pa}(j)})$. Here $\mathrm{Roots}(F)$ and $\mathrm{pa}(\cdot)$ are the original reference roots and parent map. Substitute the fixed committed values x_G. For each connected component T of F_G, identified with its vertex set, define

$$\begin{aligned} g_T(x_T) := & \prod_{u \in \mathrm{Roots}(F) \cap T} \pi_u(x_u) && \text{roots in } T \\ & \times \prod_{\substack{j \in T \setminus \mathrm{Roots}(F) \\ \mathrm{pa}(j) \in T}} K_{\mathrm{pa}(j)\to j}(x_j \mid x_{\mathrm{pa}(j)}) && \text{edges within } T \\ & \times \prod_{\substack{j \in T \setminus \mathrm{Roots}(F) \\ \mathrm{pa}(j) \in G}} K_{\mathrm{pa}(j)\to j}(x_j \mid x_{\mathrm{pa}(j)}) && \text{edges } G \to T \\ & \times \prod_{\substack{j \in G \setminus \mathrm{Roots}(F) \\ \mathrm{pa}(j) \in T}} K_{\mathrm{pa}(j)\to j}(x_j \mid x_{\mathrm{pa}(j)}) && \text{edges } T \to G. \end{aligned}$$

The fixed x_G dependence is suppressed in g_T. In particular, the last product retains factors for committed children of vertices in T: their parent values still vary with x_T. Every edge with a residual endpoint is either internal to one such T or crosses between T and G; no edge joins distinct residual components. Thus, with products over T ranging over these connected components,

$$P(x_G, x_{U_G}) = \left[\prod_{u \in \mathrm{Roots}(F) \cap G} \pi_u(x_u) \prod_{\substack{j \in G \setminus \mathrm{Roots}(F) \\ \mathrm{pa}(j) \in G}} K_{\mathrm{pa}(j)\to j}(x_j \mid x_{\mathrm{pa}(j)}) \right] \prod_T g_T(x_T).$$

The bracket is positive and depends only on x_G. It therefore cancels when dividing this joint probability by $P(X_G = x_G) = \sum_{z_{U_G}} P(x_G, z_{U_G})$. Since the connected components partition U_G, finite distributivity gives

$$\sum_{x_{U_G}} \prod_T g_T(x_T) = \prod_T \left(\sum_{x_T} g_T(x_T) \right).$$

Consequently

$$\begin{aligned} P(X_{U_G} = x_{U_G} \mid X_G = x_G) &= \frac{\prod_T g_T(x_T)}{\sum_{z_{U_G}} \prod_T g_T(z_T)} \\ &= \prod_T \frac{g_T(x_T)}{\sum_{z_T} g_T(z_T)}. \end{aligned}$$

All sums range over the vocabulary assignments on their indicated vertex sets; their denominators are positive by strict positivity of the factors. This proves conditional independence of the residual components.

2. Cancel all factors not incident to the readout. For the second claim, root the original connected component at j, using (C0). Every original neighbor v of j is observed: either $v \in G$, or its value is fixed in the lemma. Call this value x_v. The factors involving $X_j = a$ are exactly

$$\pi_j(a) \prod_{v \in \mathcal{N}_F(j)} K_{j \to v}(x_v \mid a).$$

All other factors are independent of a once these neighbor values are fixed. Revealing, changing, or summing out nonneighbors only changes a multiplicative factor independent of a. It cancels in the normalized row

$$\frac{\pi_j(a) \prod_{v \in \mathcal{N}_F(j)} K_{j \to v}(x_v \mid a)}{\sum_{b \in \mathcal{V}} \pi_j(b) \prod_{v \in \mathcal{N}_F(j)} K_{j \to v}(x_v \mid b)}.$$

Thus the history and fixed neighbor values determine the row. □

The next lemma certifies the omitted tokens for screening and, under (RF; Equation (21)) with the standard floor, bounds the number of source columns.

Intuition. The local vocabulary bank $\widehat{\mathcal{B}}_i$ from equation 111 includes every token whose true marginal mass exceeds t. Conversely, under the standard floor, every included token has true mass at least $t/2$. The rank envelope then bounds how many such tokens exist:

$$\frac{t}{2} \le C(\ell_i^{-s} + V^{-1}), \quad CV^{-1} \le \frac{t}{4} \quad \Longrightarrow \quad \ell_i \le (4C/t)^{1/s}.$$

This is an upper bound on bank size; (RF; Equation (21)) does not require the envelope to be attained. The complementary tokens can all be represented by one certified tail token in the response test.

Lemma E.3 (Preprocessing certificates; Step 1)**.** *Suppose preprocessing in Definition D.1 is feasible, and let t be the threshold it selects. Every local vocabulary bank $\widehat{\mathcal{B}}_i$ has a nonempty complement in $\mathcal{V}$, so each b_i is well-defined. For every $i \in [N]$ and $a \in \mathcal{V}$,*

$$\left|\pi_i(a) - \widetilde{\pi}_i(a)\right| \le \varepsilon_0, \qquad a \notin \widehat{\mathcal{B}}_i \Longrightarrow a \in \mathcal{T}_i(t). \tag{135}$$

Under (RF; Equation (21)*), the standard floor equation 107 additionally gives*

$$\ell_i \le \left(\frac{4C}{t}\right)^{1/s} \quad \textit{for every } i. \tag{136}$$

Proof. 1. Establish a nonempty vocabulary-bank complement. In Step 1 (Preprocess & initialize), preprocessing fixes $t = \max \mathcal{G}^{\mathrm{feas}}$ once from the public parameters, before the all-mask submission; this threshold is reused by every screen. Since $C \ge 1$, the standard floor satisfies $\underline{t} = 4\max\{\varepsilon_0, C/V\} \ge 2(\varepsilon_0 + 1/V)$. Together with $t \ge \underline{t}$, this gives

$$t - \varepsilon_0 \ge \varepsilon_0 + \frac{2}{V} > \frac{1}{V} \ge \min_{a \in \mathcal{V}} \widetilde{\pi}_i(a). \tag{137}$$

Thus $\widehat{\mathcal{B}}_i \ne \mathcal{V}$, proving tail-representative existence.

2. Certify every omitted token using the row-TV bound. At the all-mask state, $\mu_i(\cdot \mid y^{\perp}) = \pi_i$ and $q_i(\cdot \mid y^{\perp}) = \widetilde{\pi}_i$. Thus equation 103 and the singleton-event bound for TV give, for every $a \in \mathcal{V}$,

$$\left|\pi_i(a) - \widetilde{\pi}_i(a)\right| \le \sup_{A \subseteq \mathcal{V}} \left|\pi_i(A) - \widetilde{\pi}_i(A)\right| = d_{\mathrm{TV}}(\pi_i, \widetilde{\pi}_i) \le \varepsilon_0.$$

This bounds both signs of the difference, whether or not a belongs to $\widehat{\mathcal{B}}_i$. For a token outside the vocabulary bank $\widehat{\mathcal{B}}_i$, definition equation 111 then gives

$$a \notin \widehat{\mathcal{B}}_i \quad \Longrightarrow \quad \pi_i(a) \le \widetilde{\pi}_i(a) + \varepsilon_0 < t. \tag{138}$$

This proves equation 135, including the tail certificate for b_i.

3. Apply (RF; Equation (21)*) only for the bank-size estimate.* For the vocabulary-bank size $\ell_i = |\widehat{\mathcal{B}}_i|$, the standard floor gives $t \geq \underline{t} = 4\max\{\varepsilon_0, C/V\}$, hence both $t \geq 4\varepsilon_0$ and $t \geq 4C/V$. Every $a \in \widehat{\mathcal{B}}_i$ therefore satisfies

$$\pi_i(a) \geq \widetilde{\pi}_i(a) - \varepsilon_0 \geq t - 2\varepsilon_0 \geq \frac{t}{2}. \tag{139}$$

If $\ell_i \geq 1$, at least ℓ_i tokens have exact mass at least $t/2$; hence the exact rank-ℓ_i mass has the same lower bound. Assumption (RF; Equation (21)) and $t \geq 4C/V$ yield

$$\frac{t}{2} \leq C\left(\ell_i^{-s} + V^{-1}\right) \leq C\ell_i^{-s} + \frac{t}{4}.$$

Rearranging proves equation 136. The inequality is trivial when $\ell_i = 0$. □

Screen-success event (analysis only). The colors drawn in Step 2 (Draft & color) determine the following event for Step 3 (Packed screen). Fix a realized history $H_G = (G, x_G)$. For distinct $i, j \in U_G$ with $d_{F_G}(j) \leq d$, call τ (i,j)-*separating* when

$$\begin{aligned} &h_\tau(v) \neq h_\tau(j) \quad \text{for every } v \in \mathcal{N}_{F_G}(j), \\ &h_\tau(v) \neq h_\tau(i) \quad \text{for every } v \in \mathcal{N}_{F_G}(j) \setminus \{i\}. \end{aligned} \tag{140}$$

The screen family is *successful at* H_G, denoted $\mathsf{Iso}(H_G)$, if

$$\left|\left\{\tau \in \mathcal{I}^{\mathrm{col}} : \tau \text{ is } (i,j)\text{-separating}\right\}\right| > \frac{M}{2} \tag{141}$$

for every such ordered pair (i, j). This event is defined using the hidden residual forest only for analysis; the implemented screen neither observes nor tests it. For every separating τ, the two properties used by the screen proof are

$$\mathcal{N}_{F_G}(j) \cap \mathcal{C}_{\tau,h_\tau(j)}(H_G) = \emptyset, \qquad \mathcal{N}_{F_G}(j) \cap \mathcal{C}_{\tau,h_\tau(i)}(H_G) \subseteq \{i\}. \tag{142}$$

Example E.1 (Checking which colors separate a pair)**.** Use the coloring drawn in Step 2 from Example D.2 and the forest in Example E.2. For the tested pair $(i, j) = (1, 10)$, the readout has neighbors $\mathcal{N}_{F_G}(10) = \{1, 11\}$. The two tests in equation 142 are therefore

$$\{1, 11\} \cap \mathcal{C}_{\tau,2}(H_G) = \emptyset, \qquad \{1, 11\} \cap \mathcal{C}_{\tau,1}(H_G) = \{1\} \subseteq \{i\}.$$

Both hold: $h_\tau(1) = 1$, $h_\tau(10) = 2$, and $h_\tau(11) = 3$. If instead $h_\tau(11) = 2$, then neighbor 11 shares the readout color and the first intersection contains 11. If instead $h_\tau(11) = 1$, then an additional neighbor shares the source color and the second intersection is $\{1, 11\} \not\subseteq \{1\}$. This checks one pair and one coloring. It does not assert the simultaneous strict-majority event equation 141, which requires the full family of M colorings.

Call index, draft, and information before fresh colors. Fix the target, its forest, and the frozen oracle throughout. The index $s \in [T_{\mathrm{scr}}]$ counts screen calls (Step 3), not individual colorings or commit rounds. Define the *reach event*

$$R_s := \{\text{Algorithm 2 reaches its } s\text{th screen call}\}.$$

Thus R_s is an event, not the round count R. Recall that $f = (f_i)_{i \in U_G} \in \mathcal{V}^{U_G}$ is the background token vector fixed by Draft before fresh colors are drawn (Definition D.2). Its call-indexed version is

$$f^{(s)} := \begin{cases} f \text{ fixed for call } s \text{ in Step 2}, & \text{on } R_s, \\ \dagger, & \text{on } R_s^{\mathsf{c}}. \end{cases}$$

The symbol † means *no sth call*: it is a formal marker, not a vocabulary token, MASK, or a draft vector. Different executions can stop after different numbers of screens. This marker defines $f^{(s)}$ on every execution, so the proof can sum over the fixed range $s = 1, \ldots, T_{\mathrm{scr}}$ without adding any call.

On R_s, let $\mathcal{T}_{s-}$ record the past colors, replies, commits, and decision-rule state just after $f^{(s)}$ is fixed and before the current color block is drawn. On R_s^{c}, use the terminal transcript instead. Define

$$\mathcal{F}_{s-1} := \sigma(\mathcal{T}_{s-}, f^{(s)}), \qquad \mathcal{F} := \mathcal{F}_{s-1} \quad \text{at the call under consideration.}$$

On a reached call, this is precisely the information available between draft generation and fresh color generation: the history and completed draft are fixed under this conditioning, but the current colors are not. It excludes current or future colors and the full seed W; fixing W would fix those colors. No union over possible histories or drafts is needed.

Intuition. A low-degree readout creates at most $2d$ forbidden color equalities. Thus $p_{\mathrm{rnd}} = 8(d+1) \simeq d$ makes one coloring good with probability greater than $3/4$. Repeating independently M_{rnd} times makes a wrong majority exponentially unlikely:

$$\Pr(\text{wrong majority for one pair} \mid \mathcal{F}) \le e^{-M_{\mathrm{rnd}}/8}, \qquad T_{\mathrm{scr}} N^2 e^{-M_{\mathrm{rnd}}/8} \le \delta^{\mathrm{fail}}.$$

A union bound over pairs yields Equation (6). The chosen M_{rnd} is the ceiling of the logarithm obtained by solving the second inequality. Conditioning is on the past before the fresh colors.

Lemma E.4 (Fresh randomized screens succeed adaptively; Step 2)**.** *Suppose Algorithm 2 reaches a screen call with realized past $\mathcal{F}$, including the completed generation of its fixed draft but preceding the current coloring block. Conditional on $\mathcal{F}$, the fresh family in equation 117 satisfies*

$$\Pr\big(\mathsf{Iso}(H_G)^{\mathsf{c}} \mid \mathcal{F}\big) \le \frac{\delta^{\mathrm{fail}}}{T_{\mathrm{scr}}}. \tag{143}$$

Consequently, over all decision-rule randomness and commit draws,

$$\Pr\{\textit{some reached screen is unsuccessful}\} \le \delta^{\mathrm{fail}}. \tag{144}$$

Proof. 1. Separate one ordered pair with one fresh coloring. Conditional on $\mathcal{F}$, the history, residual forest, and draft are fixed, while the current colors drawn in Step 2 are mutually independent and uniform. Fix an ordered pair (i, j) with $d_{F_G}(j) \le d$. The two lines of equation 140 contain at most $2d_{F_G}(j) \le 2d$ forbidden color equalities. Each has probability $1/p_{\mathrm{rnd}}$, so one coloring is nonseparating with conditional probability at most

$$\frac{2d}{p_{\mathrm{rnd}}} = \frac{d}{4(d+1)} < \frac{1}{4}. \tag{145}$$

2. Amplify to a strict majority for that pair. Let X_τ indicate that coloring τ separates this pair. Conditional on $\mathcal{F}$, the X_τ are independent and have mean at least $3/4$. Hoeffding's inequality therefore gives

$$\Pr\left\{ \sum_{\tau=1}^{M_{\mathrm{rnd}}} X_\tau \le \frac{M_{\mathrm{rnd}}}{2} \,\middle|\, \mathcal{F} \right\} \le \exp\{-2M_{\mathrm{rnd}}(3/4 - 1/2)^2\} = \exp\{-M_{\mathrm{rnd}}/8\}. \tag{146}$$

3. Take a union bound over ordered pairs at the current history. There are fewer than N^2 relevant ordered pairs. A union bound and equation 117 give

$$N^2 e^{-M_{\mathrm{rnd}}/8} \le \delta^{\mathrm{fail}}/T_{\mathrm{scr}},$$

which proves equation 143.

4. Sum over adaptively reached calls by conditional expectation. Recall that R_s means the algorithm reaches its sth screen call, and $f^{(s)}$ is that call's draft; † marks an unreached call. On R_s, let $H^{(s)}$ be the committed history at that call, and define

$$E_s := \{R_s \text{ occurs and } \mathsf{Iso}(H^{(s)}) \text{ fails}\}.$$

Thus E_s means *call s is reached and fails*; it is false on R_s^{c}, with no history or screen evaluated there. The draft rule terminates before the fresh colors (Definition D.2), so reaching the call is decided before those colors. In the notation above, $R_s = \{f^{(s)} \neq \dagger\} \in \mathcal{F}_{s-1} = \sigma(\mathcal{T}_{s-}, f^{(s)})$. On R_s apply equation 143; on R_s^{c} the conditional failure probability is zero. Hence

$$\Pr(E_s \mid \mathcal{F}_{s-1}) \le \begin{cases} \delta^{\mathrm{fail}}/T_{\mathrm{scr}}, & \text{on } R_s, \\ 0, & \text{on } R_s^{\mathsf{c}} \end{cases} = \mathbf{1}_{R_s} \frac{\delta^{\mathrm{fail}}}{T_{\mathrm{scr}}}.$$

Taking expectations gives

$$\Pr(E_s) = \mathbb{E}[\Pr(E_s \mid \mathcal{F}_{s-1})] \le \mathbb{E}\left[\mathbf{1}_{R_s} \frac{\delta^{\mathrm{fail}}}{T_{\mathrm{scr}}}\right] = \Pr(R_s) \frac{\delta^{\mathrm{fail}}}{T_{\mathrm{scr}}} \le \frac{\delta^{\mathrm{fail}}}{T_{\mathrm{scr}}}.$$

The failed-screen event is the union of these reached-call failures, so

$$\Pr\left(\bigcup_{s=1}^{T_{\mathrm{scr}}} E_s\right) \le \sum_{s=1}^{T_{\mathrm{scr}}} \Pr(E_s) \le T_{\mathrm{scr}} \frac{\delta^{\mathrm{fail}}}{T_{\mathrm{scr}}} = \delta^{\mathrm{fail}}.$$

This proves equation 144. No independence between different screen-success events is asserted or needed. The same separating coloring isolates every packed column for a fixed ordered pair, so neither the event nor this union bound requires an additional union over bank values, columns, or possible drafts. Draft generation is allowed to depend on the past oracle transcript; independence of that draft from oracle error is not used. □

Intuition (Step 3.1, Step 3.2). By Lemma E.4, with high probability every reached screen has a strict separating majority for each pair with readout degree at most d. Fix one such coloring with different source and readout colors. The readout chunk contains no neighbor of j, and the probed source class contains no neighbor of j other than possibly i (Equation (142)). Thus every neighbor is observed, and only i can change among them. By Lemma E.2, the exact row equals the row obtained by varying only i at the common draft boundary in Equation (147); this is the boundary at which (RT; Equation (18)) and (UEN; Equation (19)) apply.

Lemma E.5 (Exact rows at a fixed draft boundary; Step 3.1, Step 3.2)**.** *Suppose preprocessing is feasible and $\ell_{\max} \ge 1$. Fix $H_G = (G, x_G)$, $f \in \mathcal{V}^{U_G}$, and distinct residual positions i, j. Define the complete boundary z on $[N] \setminus \{i, j\}$ by $z_k = x_k$ for $k \in G$ and $z_k = f_k$ otherwise. For an (i, j)-separating coloring with different source and readout colors, every $\kappa \in \mathcal{I}^{\mathrm{src}}$ satisfies*

$$\mu_j(\cdot \mid y_{\kappa,\tau,h_\tau(i)}(H_G, f, \mathcal{C}_\tau(j))) = \mu_{i\to j}(\cdot \mid \varphi_\kappa(i; f), z). \tag{147}$$

In particular, the exact row family contains all source values in the local vocabulary bank $\widehat{\mathcal{B}}_i$ and its tail representative b_i, at this same boundary. Padding may additionally contribute the row for f_i.

Proof. By Step 3.1 (Readout chunks), the readout chunk is contained in the readout color class. Thus the two isolation clauses in equation 142 give, respectively,

$$\mathcal{N}_{F_G}(j) \cap \mathcal{C}_\tau(j) = \varnothing, \qquad \mathcal{N}_{F_G}(j) \cap \mathcal{C}_{\tau,h_\tau(i)}(H_G) \subseteq \{i\}.$$

All residual neighbors of j are therefore observed. Since the source and readout colors differ, i is also outside the masked readout chunk. In the state submitted by Step 3.2 (Packed probes), the relevant observed coordinates are exactly

$$\big(y_{\kappa,\tau,h_\tau(i)}(H_G, f, \mathcal{C}_\tau(j))\big)_v = \begin{cases} x_v, & v \in G, \\ \varphi_\kappa(i; f), & v = i, \\ f_v, & v \in \mathcal{N}_{F_G}(j) \setminus \{i\}. \end{cases}$$

These are the same values as in the complete boundary with source value $\varphi_\kappa(i; f)$ and remaining values z. Conditional on $X_G = x_G$, Lemma E.2 says that the row for j depends only on its residual neighbors. Completing or changing all other coordinates to z therefore leaves it unchanged, proving the identity. The bank columns and column T give the asserted subfamily by equation 115. The boundary is independent of the coloring because one draft is fixed across the entire screen. □

Lemma E.4 bounds the probability that any reached screen fails. Under the screen-success event, Lemma E.6 formalizes equation 5 and identifies the output of Step 3.3 (Votes & row selection) with the true neighbor set for every readout of residual degree at most d. The target obeys the forest and RT–UEN (Equations (18) and (19)) assumptions (Assumptions A.2 and A.4), and oracle rows satisfy the TV bound equation 103. Recall the general resolution $\delta = 4\varepsilon_0 + \delta_{\mathrm{tail}}$ from equation 102; the main text uses $\varepsilon_0 = \varepsilon/(2\sqrt{N})$. The lemma's screening-margin hypothesis $\delta < \omega$ means that the target signal exceeds the row-noise and omitted-response allowances. This is a sufficient screening condition, not a consequence of RT–UEN (Equations (18) and (19)) alone. Preprocessing feasibility is specified in Definition D.1, and $\mathsf{Iso}(H_G)$ is defined in equation 141.

Lemma E.6 (Low-degree screen contract; Step 3.3)**.** *Suppose preprocessing is feasible and $\delta < \omega$ holds. At any history H_G, with any draft fixed throughout the screen, $\mathsf{Iso}(H_G)$ implies*

$$A_j(H_G) = \mathcal{N}_{F_G}(j) \quad \textit{for every } j \in U_G \textit{ with } d_{F_G}(j) \le d.$$

High-degree rows have only the construction's size cap; no subset or correctness guarantee is asserted for them.

Intuition for the screen contract. At a separating coloring, all exact rows for a nonedge coincide, so its observed diameter is at most $2\varepsilon_0$. For an edge, restricting source values to the vocabulary bank plus tail representative, $\widehat{\mathcal{B}}_i \cup \{b_i\}$, loses at most δ_{tail} of the full response diameter, and row noise loses at most another $2\varepsilon_0$. Thus the edge's observed diameter is at least

$$\omega - \delta_{\mathrm{tail}} - 2\varepsilon_0 > 2\varepsilon_0.$$

The strict inequality is exactly $4\varepsilon_0 + \delta_{\mathrm{tail}} < \omega$. The proof below verifies the bank/tail comparison, all empty-bank cases, and the strict-majority step.

Proof of Lemma E.6. 1. Fix a successful screen and the hidden exact rows. We analyze the votes and selected rows of Step 3.3 (Votes & row selection). If $\ell_{\max} = 0$, use the empty-bank argument in the final paragraph below. Otherwise assume $\ell_{\max} \geq 1$. Fix a history H_G and a screen draft f for which $\mathsf{lso}(H_G)$ holds, a readout $j \in U_G$ with $d_{F_G}(j) \leq d$, and a candidate source $i \in U_G \setminus \{j\}$. By equation 141, more than $M/2$ evaluations are (i,j)-separating.

Recall the binary dependence-test vote $Y_\tau(i,j \mid H_G)$ in equation 126: it is zero if $h_\tau(i) = h_\tau(j)$, and otherwise $Y_\tau(i,j \mid H_G) = \mathbf{1}\{D_\tau(i,j \mid H_G) > 2\varepsilon_0\}$. Here $D_\tau(i,j \mid H_G)$ is the observed TV diameter in equation 125. By equation 127, $i \in \widetilde{A}_j(H_G)$ exactly when $\sum_{\tau \in \mathcal{I}^{\mathrm{col}}} Y_\tau(i,j \mid H_G) > M/2$.

For a separating evaluation whose source and readout colors differ, define the hidden exact rows

$$\mu^{\mathrm{scr}}_{\tau,\kappa}(i \to j \mid H_G, f) := \mu_j(\cdot \mid y_{\kappa,\tau,h_\tau(i)}(H_G, f, \mathcal{C}_\tau(j))), \qquad \kappa \in \mathcal{I}^{\mathrm{src}}, \tag{148}$$

$$\mathcal{P}_\tau(i \to j \mid H_G) := \{\mu^{\mathrm{scr}}_{\tau,\kappa}(i \to j \mid H_G, f) : \kappa \in \mathcal{I}^{\mathrm{src}}\}. \tag{149}$$

2. No false positives. Assume $\{i,j\} \notin E(F_G)$ and fix a separating τ. If $h_\tau(i) = h_\tau(j)$, then $Y_\tau(i,j \mid H_G) = 0$ by equation 126. Otherwise, the probe state in equation 122 masks $\mathcal{C}_\tau(j)$, so the other masked positions are exactly $\mathcal{C}_\tau(j) \setminus \{j\}$. Since $\mathcal{C}_\tau(j) \subseteq \mathcal{C}_{\tau,h_\tau(j)}(H_G)$, the first relation in equation 142 gives $\mathcal{N}_{F_G}(j) \cap \mathcal{C}_\tau(j) = \varnothing$. The positions that can change across columns lie in the source class $\mathcal{C}_{\tau,h_\tau(i)}(H_G)$. The second relation in equation 142 bounds its intersection with $\mathcal{N}_{F_G}(j)$ by $\{i\}$; this intersection is empty because $i \notin \mathcal{N}_{F_G}(j)$. Thus all bank- and tail-column states agree on the committed history and reveal every residual neighbor of j at its fixed draft value. Lemma E.2 gives

$$\mu^{\mathrm{scr}}_{\tau,\mathsf{T}}(i \to j \mid H_G, f) = \mu^{\mathrm{scr}}_{\tau,\kappa}(i \to j \mid H_G, f) \quad \text{for every } \kappa \in \mathcal{I}^{\mathrm{src}}. \tag{150}$$

Every observed row is within ε_0 of this common exact row by equation 103; hence every pair of observed rows is within $2\varepsilon_0$, and $Y_\tau(i,j \mid H_G) = 0$. A strict majority of all evaluations are separating, so $\sum_{\tau \in \mathcal{I}^{\mathrm{col}}} Y_\tau(i,j \mid H_G) < M/2$. Thus $i \notin \widetilde{A}_j(H_G)$ by equation 127, proving

$$\widetilde{A}_j(H_G) \subseteq \mathcal{N}_{F_G}(j). \tag{151}$$

3. No false negatives when the local vocabulary bank is nonempty. Now assume $\{i,j\} \in E(F_G)$ and $\ell_i \geq 1$. For every separating τ, the first relation in equation 142 implies $h_\tau(i) \neq h_\tau(j)$. The second relation in equation 142 makes i the only residual neighbor of j in the probed source class. Fix a complete boundary $z \in \mathcal{V}^{[N]\setminus\{i,j\}}$ that agrees with the history and assigns f_v to every uncommitted coordinate $v \neq i,j$. Lemma E.5 gives

$$\begin{aligned} \mu^{\mathrm{scr}}_{\tau,\mathsf{T}}(i \to j \mid H_G, f) &= \mu_{i\to j}(\cdot \mid b_i, z), \\ \mu^{\mathrm{scr}}_{\tau,\kappa}(i \to j \mid H_G, f) &= \begin{cases} \mu_{i\to j}(\cdot \mid \widehat{a}_{i,\kappa}, z), & \kappa \leq \ell_i, \\ \mu_{i\to j}(\cdot \mid f_i, z), & \ell_i < \kappa \leq \ell_{\max}. \end{cases} \end{aligned} \tag{152}$$

Every separating τ therefore contains the same bank-plus-tail exact subfamily. Its diameter is

$$D^{\mathrm{ex}}_{H_G}(i,j) := \max_{a,b \in \widehat{\mathcal{B}}_i \cup \{b_i\}} d_{\mathrm{TV}}\big(\mu_{i\to j}(\cdot \mid a, z), \mu_{i\to j}(\cdot \mid b, z)\big). \tag{153}$$

Choose columns κ, κ' whose exact rows attain this finite bank-plus-tail maximum. Padding can add rows but cannot remove these columns. Applying equation 103 to both rows gives

$$\begin{aligned} D_\tau(i,j \mid H_G) &\geq d_{\mathrm{TV}}\big(q^{\mathrm{scr}}_{\tau,\kappa}(i\to j \mid H_G,f), q^{\mathrm{scr}}_{\tau,\kappa'}(i\to j \mid H_G,f)\big) \\ &\geq d_{\mathrm{TV}}\big(\mu^{\mathrm{scr}}_{\tau,\kappa}(i\to j \mid H_G,f), \mu^{\mathrm{scr}}_{\tau,\kappa'}(i\to j \mid H_G,f)\big) \\ &\quad - d_{\mathrm{TV}}\big(q^{\mathrm{scr}}_{\tau,\kappa}(i\to j \mid H_G,f), \mu^{\mathrm{scr}}_{\tau,\kappa}(i\to j \mid H_G,f)\big) \\ &\quad - d_{\mathrm{TV}}\big(q^{\mathrm{scr}}_{\tau,\kappa'}(i\to j \mid H_G,f), \mu^{\mathrm{scr}}_{\tau,\kappa'}(i\to j \mid H_G,f)\big) \\ &\geq D^{\mathrm{ex}}_{H_G}(i,j) - 2\varepsilon_0. \end{aligned}$$

If $D^{\mathrm{ex}}_{H_G}(i,j) > 4\varepsilon_0$, this is larger than $2\varepsilon_0$, so $Y_\tau(i,j \mid H_G) = 1$ for every separating τ. The strict majority then puts i in $\widetilde{A}_j(H_G)$ by equation 127. Taking the contrapositive,

$$i \notin \widetilde{A}_j(H_G) \quad\Longrightarrow\quad D^{\mathrm{ex}}_{H_G}(i,j) \leq 4\varepsilon_0. \tag{154}$$

To rule out this missed edge, it suffices to show $\Omega_{i\to j}(z) \leq \delta < \omega$, contradicting (UEN; Equation (19)). We establish this full-response bound next.

Recovering the full response diameter. The tail representative b_i and every token outside the vocabulary bank $\widehat{\mathcal{B}}_i$ lie in $\mathcal{T}_i(t)$ by Lemma E.3. Feasibility of t and (RT; Equation (18)) therefore give

$$\max_{a,b\notin\widehat{\mathcal{B}}_i} d_{\mathrm{TV}}\big(\mu_{i\to j}(\cdot \mid a,z), \mu_{i\to j}(\cdot \mid b,z)\big) \leq \min\{1, Lt^\alpha\} \leq \delta_{\mathrm{tail}}. \tag{155}$$

Suppose $i \notin \widetilde{A}_j(H_G)$. In the mixed case $a \in \widehat{\mathcal{B}}_i$ and $b \notin \widehat{\mathcal{B}}_i$, the triangle inequality through the tail representative gives

$$\begin{aligned} &d_{\mathrm{TV}}\big(\mu_{i\to j}(\cdot \mid a,z), \mu_{i\to j}(\cdot \mid b,z)\big) \\ &\quad\leq d_{\mathrm{TV}}\big(\mu_{i\to j}(\cdot \mid a,z), \mu_{i\to j}(\cdot \mid b_i,z)\big) + d_{\mathrm{TV}}\big(\mu_{i\to j}(\cdot \mid b_i,z), \mu_{i\to j}(\cdot \mid b,z)\big) \\ &\quad\leq 4\varepsilon_0 + \delta_{\mathrm{tail}}. \end{aligned}$$

The first distance uses equation 154; the second uses equation 155. Interchanging a, b handles the other mixed case. Together with the bank–bank and tail–tail bounds, this gives, for arbitrary $a, b \in \mathcal{V}$,

$$d_{\mathrm{TV}}\big(\mu_{i\to j}(\cdot \mid a,z), \mu_{i\to j}(\cdot \mid b,z)\big) \leq \begin{cases} 4\varepsilon_0, & a,b \in \widehat{\mathcal{B}}_i, \\ 4\varepsilon_0 + \delta_{\mathrm{tail}}, & \text{exactly one of } a,b \text{ lies in } \widehat{\mathcal{B}}_i, \\ \delta_{\mathrm{tail}}, & a,b \notin \widehat{\mathcal{B}}_i. \end{cases} \tag{156}$$

The maximum of the three upper bounds in equation 156 is δ by equation 102. Hence a missed true edge would have directed response at most δ, contradicting (UEN; Equation (19)) and $\delta < \omega$.

The same three comparisons also give the quantitative link used in the intuition, without assuming that the edge was missed:

$$\begin{aligned} \Omega_{i\to j}(z) &\leq \max\{D^{\mathrm{ex}}_{H_G}(i,j), D^{\mathrm{ex}}_{H_G}(i,j) + \delta_{\mathrm{tail}}, \delta_{\mathrm{tail}}\} \\ &= D^{\mathrm{ex}}_{H_G}(i,j) + \delta_{\mathrm{tail}}, \\ D_\tau(i,j \mid H_G) &\geq D^{\mathrm{ex}}_{H_G}(i,j) - 2\varepsilon_0 \\ &\geq \Omega_{i\to j}(z) - \delta_{\mathrm{tail}} - 2\varepsilon_0 \\ &\geq \omega - \delta_{\mathrm{tail}} - 2\varepsilon_0 > 2\varepsilon_0. \end{aligned}$$

Here the three entries are the bank–bank, mixed, and tail–tail cases; the last line uses (UEN; Equation (19)) for this true edge.

4. Empty vocabulary banks and row selection. If $\ell_i = 0$, every source token lies in the certified tail, so equation 155 bounds the entire directed response by $\delta_{\mathrm{tail}} \leq \delta < \omega$. Such an i cannot be adjacent to j by (UEN; Equation (19)). This also covers $\ell_{\max} = 0$: then every local vocabulary bank $\widehat{\mathcal{B}}_i$ is empty, every returned row is defined to be empty, and no residual edge incident to a low-degree readout can exist.

5. Preserve the exact candidate row under public selection. We have proved that $\widetilde{A}_j(H_G)$ equals the true neighbor set. Its size is at most d, so every admissible selector in Definition D.3, as applied in Step 3.3, preserves it by the nonoverflow identity condition. Therefore $A_j(H_G) = \mathcal{N}_{F_G}(j)$. □

E.4 Steps 4–6: Peeling, terminal certification, and resources

The screen contract now yields the contraction in Equation (8) and sound stopping in Step 4 (Claim & peel), certifying the graph in Step 5 (Terminal graph). We then count resources across the complete sampler, including Step 6 (Centroid commits) and GUARD, distinguishing successful-path estimates from unconditional implementation caps.

Example E.2 (One peel commit, then two centroid batches)**.** Let $U_G = [15]$, $d = 9$, and let the current residual forest have edge set

$$E = \{\{1, i\} : 2 \le i \le 9\} \cup \{\{1, 10\}, \{10, 11\}, \{11, 12\}, \{1, 13\}, \{13, 14\}, \{14, 15\}\}.$$

Vertex 1 has degree 10; every other vertex has degree at most 2. Suppose the reached screens are successful. By Lemma E.6, $A_u(H_G) = \mathcal{N}_{F_G}(u)$ for $u \neq 1$, whereas the row $A_1(H_G)$ need not be exact. Nevertheless, equation 106 gives

$$\begin{aligned} \mathrm{cl}_1(H_G) &= \sum_{u \neq 1} \mathbf{1}\{1 \in \mathcal{N}_{F_G}(u)\} = 10 > 9/2, \\ \mathrm{cl}_v(H_G) &= |\mathcal{N}_{F_G}(v) \setminus \{1\}| + \mathbf{1}\{v \in A_1(H_G)\} \le 2 + 1 = 3 < 9/2 \quad (v \neq 1). \end{aligned}$$

Hence $\mathcal{P}(H_G) = \{1\}$, regardless of the high-degree row's contents. In Step 4, the sampler freezes this set and commits vertex 1 alone. The residual components are then

$$\{2\}, \ldots, \{9\}, \qquad 10\text{–}11\text{–}12, \qquad 13\text{–}14\text{–}15.$$

The next pass through Step 3 uses the new history, before the stopping test in Step 4. Every residual degree is now at most 2, so all its rows are exact. Every incoming count is at most $2 < 9/2$, giving an empty peel set and the exact terminal forest in Step 5. The successive batches of Step 6 are

$$B_1 = \{2, \ldots, 9, 11, 14\}, \quad |B_1| = 10, \qquad B_2 = \{10, 12, 13, 15\}, \quad |B_2| = 4.$$

The eight isolates and the two path centers form B_1; deleting them leaves the four singleton components in B_2. Thus the example uses $1 + 2 = 3$ commit rounds. This count excludes screen submissions and is not the total oracle depth. Figure 6 keeps the vertex positions fixed throughout these steps.

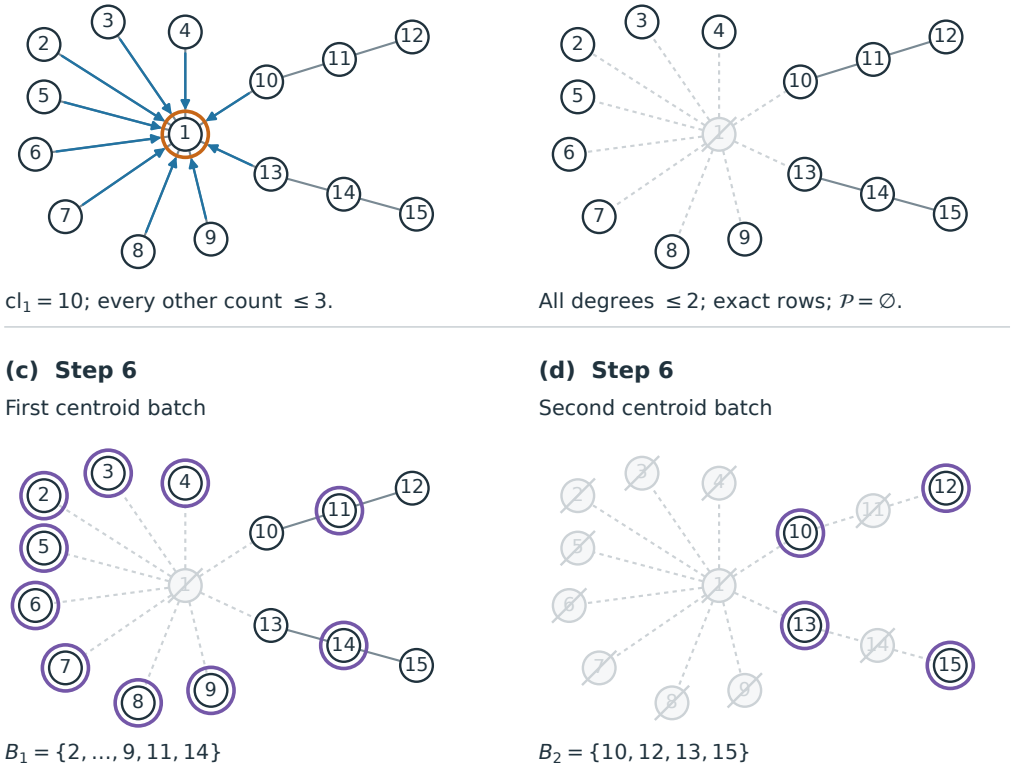


Figure 6: The successful-screen path of Example E.2, with $N = 15$ and $d = 9$. Orange and violet rings denote the next peel or centroid commit; pale crossed vertices are already committed. The fills here are neutral, not random colors. The step labels refer to Algorithm 2. A fresh screen separates (a) from (b); the terminal forest then needs no further discovery. All vertices remain individually visible.

Intuition. A high-degree vertex that receives at most $d/2$ claims has at most $d/2$ low-degree neighbors, because those neighbors report it exactly. It must therefore have more than $d/2$ neighbors inside the high-degree subforest. A forest has fewer than twice as many total degrees as vertices, so only a fraction at most $4/d$ of high-degree vertices can survive one phase. When $d = N^{\gamma_{\mathrm{deg}}+o(1)}$ with fixed $\gamma_{\mathrm{deg}} > 0$, the required number of phases is $O_{\gamma_{\mathrm{deg}}}(1)$.

Lemma E.7 (Successful-path contraction and sound stopping; Step 4, Step 5)**.** *Under the setup of Appendix A, suppose preprocessing is feasible and $\delta < \omega$. Consider an execution path on which every reached screen is successful in the sense of equation 141. Let n_k^{hi} be the number of vertices whose residual degree exceeds d at the beginning of peel phase k. After a nonempty peel phase,*

$$n_{k+1}^{\mathrm{hi}} \le \frac{4}{d} n_k^{\mathrm{hi}}, \tag{157}$$

with strict inequality whenever $n_k^{\mathrm{hi}} > 0$. Moreover, either peel-loop stopping test in Step 4 of Algorithm 2—an empty peel set or the peel-phase cap—implies that the terminal residual maximum degree is at most d. The terminal certified graph is therefore the true residual forest, so no cycle repair is performed.

Proof. 1. High-degree neighbors of an unpeeled vertex. Fix a phase and define its high-degree set by

$$\mathcal{V}^{\mathrm{hi}}(H_G) := \{v \in U_G : d_{F_G}(v) > d\}. \tag{158}$$

At the beginning of phase k, $n_k^{\mathrm{hi}} = |\mathcal{V}^{\mathrm{hi}}(H_G)|$. Recall from Step 4 (Claim & peel) that the incoming claim count is $\mathrm{cl}_v(H_G) = \sum_{u \in U_G} \mathbf{1}\{v \in A_u(H_G)\}$ and the peel threshold is $\mathrm{cl}_v(H_G) > d/2$. For $v \in \mathcal{V}^{\mathrm{hi}}(H_G) \setminus \mathcal{P}(H_G)$, every low-degree neighbor reports v exactly by Lemma E.6. Hence

$$\begin{aligned}
|\mathcal{N}_{F_G}(v) \setminus \mathcal{V}^{\mathrm{hi}}(H_G)| &= \sum_{u \in U_G \setminus \mathcal{V}^{\mathrm{hi}}(H_G)} \mathbf{1}\{v \in \mathcal{N}_{F_G}(u)\} \\
&= \sum_{u \in U_G \setminus \mathcal{V}^{\mathrm{hi}}(H_G)} \mathbf{1}\{v \in A_u(H_G)\} \le \mathrm{cl}_v(H_G) \le d/2, \\
d_{F_G[\mathcal{V}^{\mathrm{hi}}(H_G)]}(v) &= d_{F_G}(v) - |\mathcal{N}_{F_G}(v) \setminus \mathcal{V}^{\mathrm{hi}}(H_G)| \\
&\ge d_{F_G}(v) - \mathrm{cl}_v(H_G) > d/2.
\end{aligned}$$

The first equality uses the undirected nature of the forest. Claims from high-degree rows can only increase the incoming count, so they cannot invalidate the bound for an unpeeled vertex. We have shown that

$$d_{F_G[\mathcal{V}^{\mathrm{hi}}(H_G)]}(v) > \frac{d}{2}. \tag{159}$$

2. Contraction after one peel phase. Empty survivor set. If $\mathcal{V}^{\mathrm{hi}}(H_G) \setminus \mathcal{P}(H_G) = \emptyset$, no current high-degree vertex survives the phase. Deleting vertices cannot increase residual degrees, so $n_{k+1}^{\mathrm{hi}} = 0$; this proves equation 157, strictly when $\mathcal{V}^{\mathrm{hi}}(H_G) \neq \emptyset$.

Nonempty survivor set. Suppose instead that $\mathcal{V}^{\mathrm{hi}}(H_G) \setminus \mathcal{P}(H_G) \neq \emptyset$. Summing equation 159 over these unpeeled high-degree vertices and using the forest degree-sum identity in Lemma H.4 gives

$$\frac{d}{2}|\mathcal{V}^{\mathrm{hi}}(H_G) \setminus \mathcal{P}(H_G)| < \sum_{v \in \mathcal{V}^{\mathrm{hi}}(H_G) \setminus \mathcal{P}(H_G)} d_{F_G[\mathcal{V}^{\mathrm{hi}}(H_G)]}(v) \le 2|E(F_G[\mathcal{V}^{\mathrm{hi}}(H_G)])| < 2|\mathcal{V}^{\mathrm{hi}}(H_G)|. \tag{160}$$

Let $H' = (G', x_{G'})$ be the history after all singleton commits in the frozen peel set of this phase, and recall the induced residual forest $F_{G'} = F[U_{G'}]$ from equation 14. Define $\mathcal{V}^{\mathrm{hi}}(H') := \{v \in U_{G'} : d_{F_{G'}}(v) > d\}$. Deleting vertices cannot increase residual degrees:

$$\mathcal{V}^{\mathrm{hi}}(H') \subseteq \mathcal{V}^{\mathrm{hi}}(H_G) \setminus \mathcal{P}(H_G), \qquad n_{k+1}^{\mathrm{hi}} = |\mathcal{V}^{\mathrm{hi}}(H')| \le |\mathcal{V}^{\mathrm{hi}}(H_G) \setminus \mathcal{P}(H_G)|.$$

Dividing equation 160 by $d/2$ now gives $|\mathcal{V}^{\mathrm{hi}}(H_G) \setminus \mathcal{P}(H_G)| < (4/d)|\mathcal{V}^{\mathrm{hi}}(H_G)|$, and hence the strict form of equation 157. If $\mathcal{V}^{\mathrm{hi}}(H_G) = \emptyset$, then the first case above gives equality $n_{k+1}^{\mathrm{hi}} = n_k^{\mathrm{hi}} = 0$.

3. Sound stopping at the phase cap. Initially, Lemma H.4 makes the forest degree sum smaller than $2N$, while each high-degree vertex has degree at least $d + 1$. Hence

$$n_0^{\mathrm{hi}} < \frac{2N}{d+1}. \tag{161}$$

If the right-hand side is at most one, integrality already gives $n_0^{\mathrm{hi}} = 0$. Otherwise, once $n_k^{\mathrm{hi}} = 0$, deleting further vertices keeps every later high-degree count equal to zero. If this never occurs before the phase cap, then $n_k^{\mathrm{hi}} > 0$ in every preceding phase, so the strict part of equation 157 may be iterated. In either case, after T_{peel} completed phases,

$$n_{T_{\mathrm{peel}}}^{\mathrm{hi}} < \left(\frac{4}{d}\right)^{T_{\mathrm{peel}}} \frac{2N}{d+1} \le \left(\frac{8}{d}\right)^{T_{\mathrm{peel}}} \frac{2N}{d+1} \le 1, \tag{162}$$

To check the last inequality, $d \ge 9$ gives $\log(d/8) > 0$, and the ceiling in equation 104 gives

$$\begin{aligned}
T_{\mathrm{peel}} \log(d/8) &\ge [\log(2N/(d+1))]_+, \\
\left(\frac{8}{d}\right)^{T_{\mathrm{peel}}} \frac{2N}{d+1} &= \exp\left\{\log \frac{2N}{d+1} - T_{\mathrm{peel}} \log(d/8)\right\} \le 1.
\end{aligned}$$

The integer in equation 162 is therefore zero.

4. Sound stopping with an empty peel set. If instead $\mathcal{P}(H_G) = \emptyset$ while $\mathcal{V}^{\mathrm{hi}}(H_G)$ is nonempty, then equation 160 holds with $\mathcal{V}^{\mathrm{hi}}(H_G) \setminus \mathcal{P}(H_G) = \mathcal{V}^{\mathrm{hi}}(H_G)$. It would imply $d/2 < 2$, contradicting $d \geq 9$. Hence an empty peel set also certifies the absence of high-degree vertices.

5. Bound peel-set size and certify the terminal graph. We first check that the hard round cap cannot interrupt peeling. Fix a phase at history H_G with $n := |U_G| \geq 1$; if no vertex remains, the algorithm has already returned. Low-degree exactness (Lemma E.6) and the row-size cap (Definition D.3) imply $|A_u(H_G)| \leq d_{F_G}(u)$ for every u: there is equality at low degree, while $|A_u(H_G)| \leq d < d_{F_G}(u)$ at high degree. Counting each reported ordered pair once, first at its recipient and then in its reporting row, gives

$$\begin{aligned}\sum_{v \in U_G} \mathrm{cl}_v(H_G) &= \sum_{v \in U_G} \sum_{u \in U_G} \mathbf{1}\{v \in A_u(H_G)\} \\ &= \sum_{u \in U_G} \sum_{v \in U_G} \mathbf{1}\{v \in A_u(H_G)\} \\ &= \sum_{u \in U_G} |A_u(H_G)| \\ &\leq \sum_{u \in U_G} d_{F_G}(u) = 2|E(F_G)| < 2n. \end{aligned} \tag{163}$$

The second equality interchanges finite sums; the last line uses the forest degree sum in Lemma H.4. Recall $\mathcal{P}(H_G) = \{v \in U_G : \mathrm{cl}_v(H_G) > d/2\}$ from Equation (106). If $\mathcal{P}(H_G) = \emptyset$, then $|\mathcal{P}(H_G)| = 0 < 4n/d$. Otherwise its strict threshold gives

$$\begin{aligned}\frac{d}{2}|\mathcal{P}(H_G)| &< \sum_{v \in \mathcal{P}(H_G)} \mathrm{cl}_v(H_G) \leq \sum_{v \in U_G} \mathrm{cl}_v(H_G) < 2n, \\ |\mathcal{P}(H_G)| &< \frac{4n}{d} \leq \frac{4N}{d}. \end{aligned} \tag{164}$$

Step 4 (Claim & peel) freezes this set before its singleton commits, so the phase uses exactly $|\mathcal{P}(H_G)|$ rounds if completed and no more if interrupted. With at most T_{peel} phases, immediately before any planned peel commit the completed-round counter satisfies

$$r < \frac{4NT_{\mathrm{peel}}}{d} < \overline{R} - 1, \tag{165}$$

by equation 105. Thus GUARD cannot interrupt peeling. If all positions have already been committed, the residual graph is empty. Otherwise peeling reaches one of the two tests identified in the statement. Each pass executes Step 3 before either stopping test in Step 4. At the terminal history every residual vertex is consequently covered by Lemma E.6; the terminal screen is successful by hypothesis, so all terminal rows are exact. The symmetrized edge set in Step 5 (Terminal graph) is therefore exactly $E(F_\star)$. In particular, the cycle-repair loop is skipped on the successful path. □

Example E.3 (A high-degree vertex can survive a peel phase)**.** This separate example illustrates the row contract used in Lemma E.7, rather than specifying a target–oracle pair. Let $d = 9$ and take the 65 vertices

$$U = \{z\} \cup \{h_i : 1 \leq i \leq 6\} \cup \{u_{i,j} : 1 \leq i \leq 6,\ 1 \leq j \leq 9\} \cup \{s_1, s_2, s_3, s_4\},$$

with edges

$$E = \{\{z, h_i\} : 1 \leq i \leq 6\} \cup \{\{h_i, u_{i,j}\} : 1 \leq i \leq 6,\ 1 \leq j \leq 9\} \cup \{\{z, s_j\} : 1 \leq j \leq 4\}.$$

At the history under consideration, take $U_G = U$ and $F_G = F = (U, E)$. The center z and all six hubs have degree 10; all other vertices are leaves. Take the following row outputs, which obey the contract *exact at degree at most d, size at most d elsewhere*:

$$A_z(H_G) = A_{h_i}(H_G) = \emptyset, \qquad A_{u_{i,j}}(H_G) = \{h_i\}, \qquad A_{s_j}(H_G) = \{z\}.$$

Computing the incoming counts, rather than reading a high-degree row's size as its degree, gives

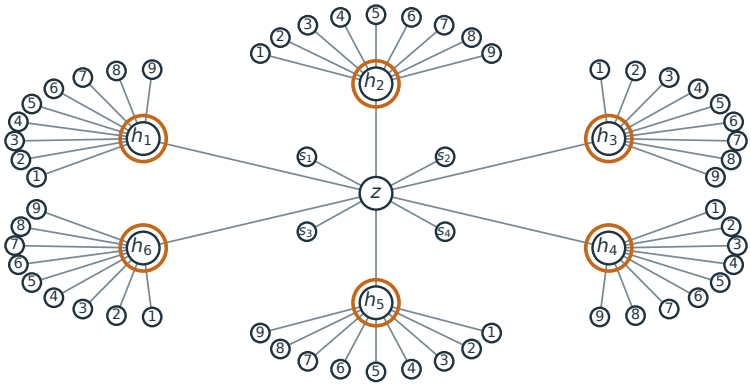
$$\begin{aligned}\mathrm{cl}_z(H_G) &= 4 \leq 9/2, & \mathrm{cl}_{h_i}(H_G) &= 9 > 9/2, \\ \mathrm{cl}_{u_{i,j}}(H_G) &= \mathrm{cl}_{s_j}(H_G) = 0, & \mathcal{P}(H_G) &= \{h_1, \ldots, h_6\}. \end{aligned}$$

Thus z is not peeled. Before the commits its high-degree neighbors are exactly $\{h_1, \ldots, h_6\}$, so their number is $6 > 9/2$, as required by equation 159. After the six frozen singleton commits in Step 4, its remaining neighbors are $\{s_1, \ldots, s_4\}$:

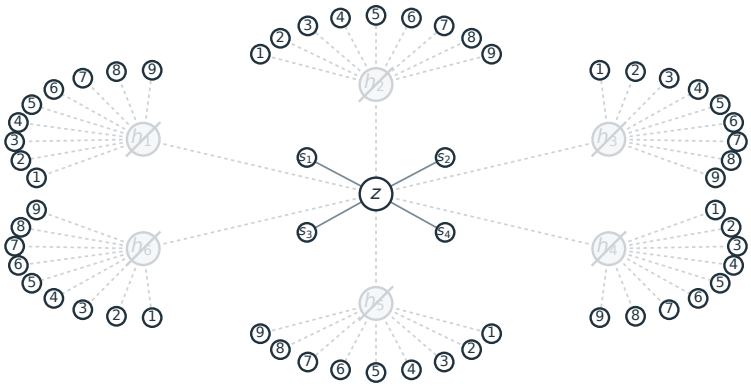
$$d_{F[U \setminus \mathcal{P}(H_G)]}(z) = 4,$$
$$|\{v \in U : d_F(v) > 9\}| = 7,$$
$$|\{v \in U \setminus \mathcal{P}(H_G) : d_{F[U \setminus \mathcal{P}(H_G)]}(v) > 9\}| = 0.$$

The center survives as a vertex but ceases to be high-degree. The 54 hub leaves become isolated; none was committed in this phase. Figure 7 draws each of them separately.

(a) Step 4 - Before: peel the six hubs



(b) Step 4 - After: the center survives, with degree 4



A leaf marked j next to hub h_i is the vertex $u_{i,j}$.
The two panels use identical locations; no leaf groups are collapsed.

Figure 7: The separate contraction example of Example E.3. The six orange-ringed hubs are committed one at a time in Step 4, with no intervening screen or peel-set recomputation. Their 54 leaves are drawn individually, not bundled. The unpeeled center z changes from degree 10 to degree 4. The lower panel keeps the same locations and crosses out only the committed hubs.

Intuition. The total counterfactual-submission count is bounded by the product of the screen-loop bounds:

$$Q_{\mathrm{cf}} \le \underbrace{\Lambda}_{\text{source columns}} \quad \underbrace{T_{\mathrm{scr}}}_{\text{screen calls}} \quad \underbrace{M_{\mathrm{rnd}}}_{\text{colorings/screen}} \quad \underbrace{p_{\mathrm{rnd}}}_{\text{source colors}} \quad \underbrace{(p_{\mathrm{rnd}}+J)}_{\text{chunks/coloring}}.$$

For rounds, Equation (164) bounds each peel set by $4N/d$; after the peel phases, centroid recursion needs only $\lceil \log_2(N+1) \rceil$ layers. The public cap extends the round bound to failed-screen paths. Total masked-state submissions, including preprocessing and commits, are $Q_{\mathrm{cf}} + R + 1$.

Lemma E.8 (Finite query and round counts; Step 1–Step 6)**.** *Suppose preprocessing is feasible and $\delta < \omega$. On every execution path of $\mathcal{A}^{\mathrm{rnd}}_{\mathrm{pack}}[\mathsf{S}]$ with any admissible row selector and the standard draft generator,*

$$Q_{\mathrm{cf}} \le \Lambda T_{\mathrm{scr}} M_{\mathrm{rnd}} p_{\mathrm{rnd}} (p_{\mathrm{rnd}} + J), \tag{166}$$

$$R \le \overline{R}. \tag{167}$$

where $\overline{R} = \lceil 4NT_{\mathrm{peel}}/d \rceil + \lceil \log_2(N+1) \rceil + 2$ is the public commit-round cap and $T_{\mathrm{scr}} = T_{\mathrm{peel}} + 1$ is the screen-call cap from equation 105; T_{peel} is the peel-phase cap in equation 104. In equation 166, p_{rnd} and M_{rnd} are the fixed numbers of colors and independent colorings per screen from equation 117. The subscript rnd *labels the randomized-color construction, not a round index. On a path on which every reached screen is successful, the sharper bound*

$$R < \frac{4NT_{\mathrm{peel}}}{d} + \lceil \log_2(N+1) \rceil \tag{168}$$

holds, with no cycle repair or round-cap fallback. The adaptive oracle depth, including the initial all-mask preprocessing stage, satisfies

$$D \le 1 + R + T_{\mathrm{scr}}. \tag{169}$$

Proof. 1. Count submissions over chunks, columns, and screens. Fix one screen execution at history H_G. Recall the quantities counted in its loops:

- $Q_{\mathrm{scr}}(H_G)$ denotes the number of masked-state submissions in this one call to Step 3 (Packed screen), all made in Step 3.2; preprocessing and commit submissions are excluded.
- $\ell_{\max} = \max_{i \in [N]} |\widehat{\mathcal{B}}_i|$ is the largest local vocabulary-bank size fixed in Definition D.1. The number of submitted source columns is $\Lambda = \ell_{\max} + 1$ when $\ell_{\max} \ge 1$, and $\Lambda = 0$ otherwise, as in equation 116.
- $p = p_{\mathrm{rnd}} = 8(d+1)$ is the number of colors, and $M = M_{\mathrm{rnd}}$ is the number of independent colorings supplied by Step 2 (Draft & color); their choices are equation 117.
- $J \in [N]$ is the readout-chunk parameter from Section D.1: it sets $B^{\mathrm{rd}} = \lceil N/J \rceil$, the maximum chunk size. It is not the actual number of chunks.

For one coloring τ, write $n_c := |\mathcal{C}_{\tau,c}(H_G)|$ and let J_τ be the total number of readout chunks across its color classes. The deterministic partition equation 121 in Step 3.1 creates exactly $\lceil n_c/B^{\mathrm{rd}} \rceil$ chunks for every nonempty color class. Therefore

$$\begin{aligned} J_\tau &= \sum_{c: n_c > 0} \left\lceil \frac{n_c}{B^{\mathrm{rd}}} \right\rceil \\ &\le \sum_{c: n_c > 0} \left(1 + \frac{n_c}{B^{\mathrm{rd}}}\right) \\ &\le p + \frac{|U_G|}{B^{\mathrm{rd}}} \le p + J. \end{aligned} \tag{170}$$

For each chunk, the bank columns and source-specific tail column in Step 3.2 use $(\ell_{\max}+1)(p-1)$ submissions. When $\ell_{\max} \ge 1$, this is at most Λp. Thus Step 3.1 supplies at most $p + J$ chunks per

coloring, and Step 3.2 submits at most Λp states per chunk. Summing over the M colorings from Step 2 gives

$$Q_{\mathrm{scr}}(H_G) \le \underbrace{\Lambda}_{\text{source columns}} \quad \underbrace{M}_{\text{colorings/screen}} \quad \underbrace{p}_{\text{source colors}} \quad \underbrace{(p+J)}_{\text{chunks per coloring}} . \tag{171}$$

When $\ell_{\max} = 0$, the algorithm submits no screen state and both sides are zero. The screen runs initially and once after each of at most T_{peel} completed peel phases, hence at most T_{scr} times. Substituting the randomized values of M and p proves equation 166. The standard draft reuses preprocessing and adds no query. Cycle repair runs no further screen: each singleton repair is one commit submission, charged to R, not to Q_{cf}.

Returned oracle rows. For the standard zero-query draft, each counterfactual submission reads only its chunk, of size at most $B^{\mathrm{rd}} = \lceil N/J \rceil$. Preprocessing reads N rows, and the disjoint commit batches $B_1, \dots, B_R$ partition $[N]$ (Definition A.9), so $\sum_{t=1}^{R} |B_t| = N$. The main construction takes $J = d$. Consequently the total number of returned oracle rows is at most

$$\underbrace{\lceil N/J \rceil Q_{\mathrm{cf}}}_{\text{screen submissions}} + \underbrace{N}_{\text{preprocessing}} + \underbrace{\sum_{t=1}^{R} |B_t|}_{\text{commits}} = \lceil N/J \rceil Q_{\mathrm{cf}} + 2N.$$

2. Count adaptive oracle stages. Once the history, draft, and colors are fixed, all submissions in a screen can be issued in one parallel oracle stage in Step 3.2, because their states do not depend on replies from that screen. Each commit requires one stage, and preprocessing in Step 1 requires one. This proves equation 169.

3. Bound peeling rounds on a successful path. On a path where every reached screen succeeds, equation 164 in Lemma E.7 shows that every nonempty peel phase in Step 4 (Claim & peel) commits fewer than $4N/d$ vertices, one per round. At most T_{peel} such phases therefore use fewer than $4NT_{\mathrm{peel}}/d$ rounds. The bound equation 165 also shows that GUARD cannot interrupt these commits.

4. Bound terminal centroid rounds and exclude repair and fallback. By Lemma E.7, the terminal certified graph is the true residual forest. Lemma E.2 makes its connected components conditionally independent. At any layer of Step 6 (Centroid commits), let the current components be $T_1, \dots, T_m$, with selected centroids $v_1^{\mathrm{cen}}, \dots, v_m^{\mathrm{cen}}$, and set $B = \{v_1^{\mathrm{cen}}, \dots, v_m^{\mathrm{cen}}\}$. Then

$$P(X_B = x_B \mid X_G = x_G) = \prod_{\ell=1}^{m} P\big(X_{v_\ell^{\mathrm{cen}}} = x_{v_\ell^{\mathrm{cen}}} \mid X_G = x_G\big). \tag{172}$$

Thus their exact singleton rows define the exact joint batch conditional. By Lemma H.4, removing a centroid leaves components of size at most half the parent size, so after ℓ terminal batches every remaining component has size at most $N/2^\ell$. For $\ell = \lceil \log_2(N+1) \rceil$ this is smaller than one. Adding the peeling and terminal rounds proves equation 168. Its right-hand side is smaller than $\overline{R} - 1$ by equation 105; hence the hard-cap check is never triggered during the terminal stage either. The terminal graph is already a forest, so the cycle-repair loop adds no rounds.

5. Enforce the round cap on every path. It remains to verify the unconditional randomized round bound enforced by GUARD. Initially $r = 0 \le \overline{R} - 1$. Before every peel, repair, or centroid commit, the algorithm checks whether $r = \overline{R} - 1$. If equality holds, it commits all remaining positions in the reserved final round and returns. Otherwise the integer counter satisfies $r \le \overline{R} - 2$, and the next commit leaves $r + 1 \le \overline{R} - 1$. Every such commit is nonempty and irrevocable, so it removes at least one remaining position; cycle repair cannot continue indefinitely. If it ends without the cap fallback, the remaining working graph is a forest and centroid completion applies. Induction over commit operations therefore gives $R \le \overline{R}$ on every path, proving equation 167. □

E.5 EXACT-COMMIT HYBRID AND ADAPTIVE ERROR COMPOSITION

This subsection completes the statistical part of the upper bound. Lemma E.9 identifies a structurally safe exact-row hybrid with the target law. Separately, Lemma E.10 bounds the distance between any product-commit decision rule and its exact-row hybrid, whether or not its batches are structurally safe. A safe-decision-rule comparison then accounts for failed screens in the implemented randomized sampler.

Structural safety. At a history $H_G = (G, x_G)$, let F_G be the true residual forest on $U_G = [N]\setminus G$. A complete decision rule is *structurally safe* if, at every realized history, each batch $B \subseteq U_G$ it commits satisfies

$$|B \cap V(T)| \le 1 \qquad \text{for every connected component } T \text{ of } F_G.$$

In particular, every singleton batch is safe. This is a property used in the analysis, not permission for the implemented sampler to inspect the hidden forest.

The two executions and their output laws. Fix one complete decision rule, the frozen oracle q, and a decision-rule seed $W = w$, including the colors and all auxiliary draft randomness. The product-commit draws remain random after this conditioning. For a chosen batch B and the history-compatible commit state y in equation 27, compare:

- *Oracle execution.* A batch assignment $z \in \mathcal{V}^B$ has mass $\prod_{j\in B} q_j(z_j \mid y)$. Recall from Definition A.10 that its full output law is $\widehat{P}^q_w(x) = \Pr(\widehat{X} = x \mid W = w)$, with q fixed and all product-commit draws integrated out.
- *Exact-commit hybrid.* Replace only this commit mass by $\prod_{j\in B} \mu_j(z_j \mid y)$, using the exact rows in equation 23. Keep the decision rule, draft generator, query rules, and all frozen-oracle replies unchanged. Denote this execution's full output law by $P^{\otimes}_w$.

Both are laws on $\mathcal{V}^N$; $\otimes$ refers to the product within each commit, not to independence of all output coordinates. The hybrid can still depend on q through its decisions; that dependence is suppressed in $P^{\otimes}_w$. The shared rules do not imply shared realized transcripts: different commit outcomes can lead to different later queries and batches.

Structural safety identifies the hybrid's product at each commit with the target's joint batch conditional; for an unsafe batch they need not agree. Thus Lemma E.9 uses safety to prove $P^{\otimes}_w = P$, whereas Lemma E.10 compares the two output laws without requiring safety. Both lemmas also apply to the analysis-only decision rule introduced below.

Intuition. A safe batch takes at most one vertex from each independent residual component. Its product of exact rows is therefore the exact joint conditional. Multiplying these joint conditionals along the adaptive partition is the ordinary chain rule, so the output law is exactly P.

Lemma E.9 (Exact hybrid equals the target law)**.** *For every structurally safe complete decision rule, the exact-commit hybrid satisfies*

$$P^{\otimes}_w = P \qquad \textit{for every decision-rule seed } w. \tag{173}$$

Proof. 1. Identify the exact joint law of a safe batch. For a singleton batch the product of exact singleton rows is trivially its exact joint conditional. For a nonsingleton safe batch, its vertices lie in distinct true residual components. Lemma E.2 therefore identifies their exact joint conditional with the product of their exact singleton conditionals, as in equation 172.

2. Apply the chain rule along each adaptive path. For fixed w, draft generation is deterministic as a function of the observed past, even when its rule is randomized before conditioning on w. For a full assignment $x \in \mathcal{V}^N$, let $B_t(x)$ be the batch selected along the unique hybrid path consistent with x, and let $G_{t-1}(x)$ be the set committed before that batch, with $G_0(x) = \varnothing$ and $G_t(x) = G_{t-1}(x) \cup B_t(x)$. The batches form an adaptive ordered partition of $[N]$. The preceding paragraph identifies every hybrid batch law with the corresponding exact joint conditional, so the conditional chain rule gives

$$\begin{aligned} P^{\otimes}_w(x) &= \prod_t P\big(X_{B_t(x)} = x_{B_t(x)} \mid X_{G_{t-1}(x)} = x_{G_{t-1}(x)}\big) \\ &= \prod_t \frac{P(X_{G_t(x)} = x_{G_t(x)})}{P(X_{G_{t-1}(x)} = x_{G_{t-1}(x)})} \\ &= P(x). \end{aligned} \tag{174}$$

The batch partition may depend on earlier committed values, but for each fixed x the displayed factors are precisely the chain-rule factors along that path. Consecutive numerators and denominators cancel; the first denominator is the probability of the empty constraint, hence one, and the

last numerator is $P(x)$ because every coordinate has been committed. This proves the equality pointwise. □

We next compare the full output laws $P_w^{\otimes}$ (exact-row commits) and $\widehat{P}_w^{q}$ (oracle-row commits) for the same decision rule and fixed seed. For probability masses P, Q on a finite set, write

$$\mathrm{Aff}(P,Q) := \sum_x \sqrt{P(x)Q(x)} = 1 - h^2(P,Q). \tag{175}$$

Intuition. The error budget is attached to committed coordinates, not to probes or rounds. Each coordinate is committed once, so the total budget is

$$N \cdot \frac{(\varepsilon^{\mathrm{H}})^2}{2N} = \frac{(\varepsilon^{\mathrm{H}})^2}{2}.$$

For adaptive batches, backward induction on the execution tree justifies this accounting. At each node, charge the current batch and then use the budget of the remaining coordinates.

Lemma E.10 (Adaptive squared-Hellinger composition)**.** *Assume Assumption A.7 (A2): every valid row pair obeys $h^2(\mu_j(\cdot \mid y), q_j(\cdot \mid y)) \le (\varepsilon^{\mathrm{H}})^2/(2N)$. For every complete decision rule using the product-commit rule and every fixed decision-rule seed w, regardless of structural safety,*

$$h^2\big(P_w^{\otimes}, \widehat{P}_w^{q}\big) \le \frac{(\varepsilon^{\mathrm{H}})^2}{2}. \tag{176}$$

Proof. 1. Compare the two kernels at a common node. Fix q and w throughout the proof. An *execution-tree node* is a stored algorithm state immediately before a commit, or a terminal state after completion; it is not a vertex of the hidden forest. A pre-commit state records

$$\mathfrak{h} = (H_G, \mathcal{T}, \text{decision-rule state}), \qquad H_G = (G, x_G), \quad U_G = [N] \setminus G.$$

Here $\mathcal{T}$ contains the preceding query/reply and commit record, and the internal state includes any current draft and counters. We use this complete stored state to define continuation kernels directly, without replaying the decision rule from its initial state. With q and w fixed, all operations between commits are deterministic. Thus, from the *same stored state* $\mathfrak{h}$, both executions receive the same frozen-oracle replies and select the same batch $B(\mathfrak{h}) \subseteq U_G$ and commit state $y(\mathfrak{h})$ from equation 27. Comparing kernels at this common state does not assert that two separately sampled executions have identical paths. For a batch assignment $z \in \mathcal{V}^{B(\mathfrak{h})}$, their kernels are

$$\begin{aligned} P^{\otimes}(z \mid \mathfrak{h}) &:= \prod_{j \in B(\mathfrak{h})} \mu_j(z_j \mid y(\mathfrak{h})), \\ \widehat{P}^{\otimes}(z \mid \mathfrak{h}) &:= \prod_{j \in B(\mathfrak{h})} q_j(z_j \mid y(\mathfrak{h})). \end{aligned} \tag{177}$$

Both kernels are probability masses on $\mathcal{V}^{B(\mathfrak{h})}$:

$$\begin{aligned} \sum_{z \in \mathcal{V}^{B(\mathfrak{h})}} P^{\otimes}(z \mid \mathfrak{h}) &= \prod_{j \in B(\mathfrak{h})} \sum_{a \in \mathcal{V}} \mu_j(a \mid y(\mathfrak{h})) = 1, \\ \sum_{z \in \mathcal{V}^{B(\mathfrak{h})}} \widehat{P}^{\otimes}(z \mid \mathfrak{h}) &= \prod_{j \in B(\mathfrak{h})} \sum_{a \in \mathcal{V}} q_j(a \mid y(\mathfrak{h})) = 1. \end{aligned}$$

The finite sums factor by distributivity over the Cartesian product. The first kernel is a product of exact rows, whether or not it equals the target's joint batch conditional. Hence this comparison needs no structural safety condition.

2. Charge the current batch to its coordinates. Apply the product identity in Lemma H.1 with index set $B(\mathfrak{h})$, each coordinate space equal to $\mathcal{V}$, and component laws $\mu_j(\cdot \mid y(\mathfrak{h}))$ and $q_j(\cdot \mid y(\mathfrak{h}))$. This gives

$$\mathrm{Aff}(P^{\otimes}(\cdot \mid \mathfrak{h}), \widehat{P}^{\otimes}(\cdot \mid \mathfrak{h})) = \prod_{j \in B(\mathfrak{h})} \left[1 - h^2\big(\mu_j(\cdot \mid y(\mathfrak{h})), q_j(\cdot \mid y(\mathfrak{h}))\big)\right]. \tag{178}$$

Using $1 - \prod_j (1 - u_j) \le \sum_j u_j$ for $u_j \in [0, 1]$ and equation 25,

$$1 - \mathrm{Aff}(P^{\otimes}(\cdot \mid \mathfrak{h}), \widehat{P}^{\otimes}(\cdot \mid \mathfrak{h})) = h^2(P^{\otimes}(\cdot \mid \mathfrak{h}), \widehat{P}^{\otimes}(\cdot \mid \mathfrak{h})) \tag{179}$$

$$\le \sum_{j \in B(\mathfrak{h})} h^2\big(\mu_j(\cdot \mid y(\mathfrak{h})), q_j(\cdot \mid y(\mathfrak{h}))\big)$$

$$\le |B(\mathfrak{h})| \frac{(\varepsilon^{\mathrm{H}})^2}{2N}. \tag{180}$$

3. Derive the suffix-affinity recursion. Take a stored execution state $\mathfrak{h}$ from part 1 of this proof, with committed history $H_G = (G, x_G)$ and remaining set $U_G = [N] \setminus G$. Here a *suffix* is the assignment of all still-uncommitted coordinates, not necessarily a contiguous block of positions. Define:

- $P^{\mathrm{suf}}(\cdot \mid \mathfrak{h})$: the law of those remaining values when execution resumes from this entire stored state using exact-row commits.
- $\widehat{P}^{\mathrm{suf}}(\cdot \mid \mathfrak{h})$: the corresponding law when it resumes from the same state using oracle-row commits.

Both are probability laws on $\mathcal{V}^{U_G}$, with q and w still fixed. The recursion below defines these continuation laws from the batch kernels in equation 177; at a terminal state $U_G = \varnothing$, each puts mass one on the empty assignment. Set

$$\mathrm{Aff}_{\mathrm{suf}}(\mathfrak{h}) := \mathrm{Aff}(P^{\mathrm{suf}}(\cdot \mid \mathfrak{h}), \widehat{P}^{\mathrm{suf}}(\cdot \mid \mathfrak{h})). \tag{181}$$

After a batch value $z \in \mathcal{V}^{B(\mathfrak{h})}$, write $\mathfrak{h} \oplus z$ for the next pre-commit node (or terminal node), including all intervening deterministic operations. It has remaining coordinates $U_G \setminus B(\mathfrak{h})$. For $x' \in \mathcal{V}^{U_G \setminus B(\mathfrak{h})}$, the suffix assignment (z, x') has masses

$$P^{\mathrm{suf}}((z, x') \mid \mathfrak{h}) = P^{\otimes}(z \mid \mathfrak{h})\, P^{\mathrm{suf}}(x' \mid \mathfrak{h} \oplus z),$$
$$\widehat{P}^{\mathrm{suf}}((z, x') \mid \mathfrak{h}) = \widehat{P}^{\otimes}(z \mid \mathfrak{h})\, \widehat{P}^{\mathrm{suf}}(x' \mid \mathfrak{h} \oplus z).$$

These are normalized: summing first over x' gives one for each child law, and summing over z then gives one by the batch normalization above. There are at most N nonempty commits, so backward induction starts at the empty suffix and defines every law. Summing the geometric means first over x' now yields

$$\mathrm{Aff}_{\mathrm{suf}}(\mathfrak{h}) = \sum_z \sqrt{P^{\otimes}(z \mid \mathfrak{h}) \widehat{P}^{\otimes}(z \mid \mathfrak{h})} \sum_{x'} \sqrt{P^{\mathrm{suf}}(x' \mid \mathfrak{h} \oplus z) \widehat{P}^{\mathrm{suf}}(x' \mid \mathfrak{h} \oplus z)}$$

$$= \sum_z \sqrt{P^{\otimes}(z \mid \mathfrak{h}) \widehat{P}^{\otimes}(z \mid \mathfrak{h})}\, \mathrm{Aff}_{\mathrm{suf}}(\mathfrak{h} \oplus z). \tag{182}$$

At a terminal node the suffix is empty and its affinity is one.

4. Bound the remaining loss by backward induction. At a node with committed set G, every complete continuation partitions U_G into disjoint future batches. Thus its total coordinate budget is

$$\sum_{\text{future } t} |B(\mathfrak{h}_t)| \frac{(\varepsilon^{\mathrm{H}})^2}{2N} = |U_G| \frac{(\varepsilon^{\mathrm{H}})^2}{2N}. \tag{183}$$

We prove by backward induction that this budget bounds the suffix loss:

$$1 - \mathrm{Aff}_{\mathrm{suf}}(\mathfrak{h}) \le |U_G| \frac{(\varepsilon^{\mathrm{H}})^2}{2N}. \tag{184}$$

The terminal case has both sides zero. At a nonterminal node, subtract the recursion from one and add and subtract the batch affinity:

$$\begin{aligned}1-\mathrm{Aff}_{\mathrm{suf}}(\mathfrak{h}) &= 1-\mathrm{Aff}(P^{\otimes}(\cdot\mid\mathfrak{h}),\widehat{P}^{\otimes}(\cdot\mid\mathfrak{h}))\\ &\quad+\sum_z\sqrt{P^{\otimes}(z\mid\mathfrak{h})\widehat{P}^{\otimes}(z\mid\mathfrak{h})}\big(1-\mathrm{Aff}_{\mathrm{suf}}(\mathfrak{h}\oplus z)\big)\\ &\le |B(\mathfrak{h})|\frac{(\varepsilon^{\mathrm{H}})^2}{2N}+\big(|U_G|-|B(\mathfrak{h})|\big)\frac{(\varepsilon^{\mathrm{H}})^2}{2N}\sum_z\sqrt{P^{\otimes}(z\mid\mathfrak{h})\widehat{P}^{\otimes}(z\mid\mathfrak{h})}\\ &\le |U_G|\frac{(\varepsilon^{\mathrm{H}})^2}{2N}.\end{aligned}$$

The first inequality uses the batch bound and the induction hypothesis: every child has exactly $|U_G|-|B(\mathfrak{h})|$ uncommitted coordinates. In the last inequality, the remaining sum is the batch affinity, at most $\sqrt{1\cdot 1}=1$ by Cauchy–Schwarz and the two kernel normalizations in part 1. The resulting coordinate count is $|B(\mathfrak{h})|+(|U_G|-|B(\mathfrak{h})|)=|U_G|$.

5. Evaluate the budget at the root. The root $\mathfrak{h}_\emptyset$ is the common state before the first commit, after the initial deterministic operations. It has $G=\varnothing$ and $U_G=[N]$, so its continuation laws are the full output laws recalled above:

$$P^{\mathrm{suf}}(\cdot\mid\mathfrak{h}_\emptyset)=P_w^{\otimes},\qquad \widehat{P}^{\mathrm{suf}}(\cdot\mid\mathfrak{h}_\emptyset)=\widehat{P}_w^{q}.$$

In particular, $|U_G|=N$. Equivalently, along every complete path the disjoint commit batches satisfy

$$\sum_t |B(\mathfrak{h}_t)|=N. \tag{185}$$

Therefore

$$h^2(P_w^{\otimes},\widehat{P}_w^{q})=1-\mathrm{Aff}_{\mathrm{suf}}(\mathfrak{h}_\emptyset)\le N\frac{(\varepsilon^{\mathrm{H}})^2}{2N}=\frac{(\varepsilon^{\mathrm{H}})^2}{2}. \tag{186}$$

Counterfactual operations affect which common node is reached; they do not generate committed coordinates and add no term to this budget. □

We now separate oracle error from screen failures by comparing with an analysis-only safe decision rule, whose seed-averaged discrepancy from the sampler is charged to the failed-screen probability.

Recall the reached-call event $E_s=\{R_s \text{ occurs and } \mathsf{Iso}(H^{(s)}) \text{ fails}\}$: R_s means the implemented sampler reaches call s, and $H^{(s)}$ is its history. Failure means that the separating-color majority condition equation 141 is violated. Thus $\mathsf{Fail}=\bigcup_{s=1}^{T_{\mathrm{scr}}}E_s$.

Intuition. The safe decision rule follows the sampler until a screen fails and then finishes by singleton commits. Its batches are always safe, so the previous two lemmas bound its TV error by ε^{H}. A common-path coupling charges the discrepancy between the two outputs only to screen failure:

$$\mathbb{E}_W\mathrm{TV}(\text{target},\text{sampler})\le\varepsilon^{\mathrm{H}}+\delta^{\mathrm{fail}}.$$

The first contribution is controlled for every fixed seed; the failure contribution is bounded only after averaging over the colors and commit draws, as verified below. We compare the full output laws: conditioning the implemented output on screen success would also condition its reached histories.

Lemma E.11 (Safe decision rule and failed-screen coupling)**.** *Fix a target–oracle pair satisfying the finite hypotheses of Theorem E.1. There is an analysis-only structurally safe decision rule whose conditional output law $\widehat{P}_w^{\mathrm{safe},q}$ satisfies*

$$d_{\mathrm{TV}}\big(P,\widehat{P}_w^{\mathrm{safe},q}\big)\le\varepsilon^{\mathrm{H}} \qquad \textit{for every } w, \tag{187}$$

$$\mathbb{E}_W d_{\mathrm{TV}}\big(\widehat{P}_W^{q},\widehat{P}_W^{\mathrm{safe},q}\big)\le\delta^{\mathrm{fail}}. \tag{188}$$

Proof. 1. Define the analysis-only safe decision rule. The safe decision rule simulates Algorithm 2 with the same fixed public row selector, draft generator, auxiliary seed, and fresh colorings while

every reached screen satisfies equation 141. It is additionally allowed to inspect the hidden residual forest. Immediately after the first unsuccessful execution of Step 3, and also before any GUARD fallback, it stops using the screen output and commits all remaining coordinates as singletons in increasing position order. More formally, number the successive simulation checkpoints (screen returns and pre-fallback checks) by n, and define

$$\tau_{\mathrm{sw}} := \inf\left\{ n : \begin{array}{l} \text{an unsuccessful screen has just returned, or} \\ \text{the simulated sampler is about to activate a fallback} \end{array} \right\}, \qquad \inf \varnothing := \infty.$$

Before this checkpoint the two rules agree; from this checkpoint onward the safe rule uses only the singleton completion just specified. The checkpoint index is not the peel-phase index. Hidden-forest inspection defines only this comparison law; it is not available to the implemented sampler. Each singleton commit uses the same frozen oracle q at the updated committed history. Writing $\widehat{X}^{\mathrm{safe}}$ for its output, define

$$\widehat{P}_w^{\mathrm{safe},q}(x) := \Pr(\widehat{X}^{\mathrm{safe}} = x \mid W = w),$$

with the target and oracle fixed and all product-commit draws integrated out.

2. Check structural safety in all three stages.

- Before a switch, the peel commits in Step 4 are singletons.
- If the terminal stage is reached without a switch, Lemma E.7 identifies the graph constructed in Step 5 with the true residual forest, so every centroid batch in Step 6 contains one vertex from each true residual component.
- After a switch, all batches are singletons.

Hence the safe decision rule is structurally safe on every path.

3. Bound oracle error for each fixed seed. Apply Lemmas E.9 and E.10 to this safe rule and its own product-of-exact-rows hybrid. The first lemma identifies that hybrid with P; the second compares it with the safe rule's oracle output. Thus, for each fixed w,

$$h^2\big(P, \widehat{P}_w^{\mathrm{safe},q}\big) \le \frac{(\varepsilon^{\mathrm{H}})^2}{2}. \tag{189}$$

By equation 214,

$$d_{\mathrm{TV}}\big(P, \widehat{P}_w^{\mathrm{safe},q}\big) \le \sqrt{2h^2\big(P, \widehat{P}_w^{\mathrm{safe},q}\big)} \le \varepsilon^{\mathrm{H}}, \tag{190}$$

which proves equation 187.

4. Couple the sampler to its own safe decision rule. For equation 188, fix the full decision-rule seed w and couple the two conditional executions as follows. Before the safe decision rule switches, whenever their histories agree, use the same draw from their common product-commit law. Thus their histories and all drafts and deterministic frozen-oracle replies remain identical until the safe decision rule switches. By Lemma E.8, in the implemented execution,

$$\{\text{cycle repair or a hard round-cap fallback occurs}\} \subseteq \mathsf{Fail}.$$

An empty peel set or the peel-phase cap triggers the normal loop exit in Step 4. On $\mathsf{Fail}^{\mathsf{c}}$ no switch occurs and the outputs coincide; repair can start only after the safe rule has switched to singleton completion. For the coupled outputs $\widehat{X}$ and $\widehat{X}^{\mathrm{safe}}$, the common prefix gives the pathwise inclusions

$$\{\widehat{X} \neq \widehat{X}^{\mathrm{safe}}\} \subseteq \{\tau_{\mathrm{sw}} < \infty\} \subseteq \mathsf{Fail}.$$

Indeed, a first switch caused by a failed screen occurs in that common prefix. A first switch caused by a fallback with no preceding failure is excluded by Lemma E.8. After a switch the two executions may be completed with any coupling of their respective laws. The first common-path failure is exactly the first failure of the implemented execution, since the coupled histories agree up to that point. For any coupling (X, Y) of laws P, Q, the coupling inequality is $d_{\mathrm{TV}}(P, Q) \le \Pr(X \neq Y)$. Applying it to these two outputs gives

$$d_{\mathrm{TV}}\big(\widehat{P}_w^{q}, \widehat{P}_w^{\mathrm{safe},q}\big) \le \Pr(\widehat{X} \neq \widehat{X}^{\mathrm{safe}} \mid W = w) \le \Pr(\mathsf{Fail} \mid W = w), \tag{191}$$

where the probability integrates the coupled commit draws.

5. Average the failed-screen probability over seeds. Averaging equation 191 over the seed gives

$$\begin{aligned}&\mathbb{E}_W d_{\mathrm{TV}}\big(\widehat{P}_W^q, \widehat{P}_W^{\mathrm{safe},q}\big)\\ &\quad\le \mathbb{E}_W \Pr(\mathsf{Fail} \mid W) = \Pr(\mathsf{Fail}) = \Pr\left(\bigcup_{s=1}^{T_{\mathrm{scr}}} E_s\right) \le \delta^{\mathrm{fail}}.\end{aligned}$$

The equality averaging conditional probabilities is the law of total expectation over colors, draft randomness, and commit draws; the final inequality is equation 144. This proves equation 188, without a seedwise failure bound. □

Proof of Theorem E.1. 1. Account for submissions, commits, and depth. With the standard zero-query generator, the pathwise query and round bounds are equation 166–equation 167 from Lemma E.8. The all-mask preprocessing state is excluded from Q_{cf} by Definition A.11. Charging it gives $Q_{\mathrm{nc}} = Q_{\mathrm{cf}} + 1$, as in equation 34. The depth bound is equation 169.

2. Combine oracle error and failed-screen error. For every target–oracle pair, the TV triangle inequality followed by Lemma E.11 gives

$$\begin{aligned}&\mathbb{E}_W d_{\mathrm{TV}}\big(P, \widehat{P}_W^q\big)\\ &\quad\le \mathbb{E}_W d_{\mathrm{TV}}\big(P, \widehat{P}_W^{\mathrm{safe},q}\big) + \mathbb{E}_W d_{\mathrm{TV}}\big(\widehat{P}_W^{\mathrm{safe},q}, \widehat{P}_W^q\big)\\ &\quad\le \varepsilon^{\mathrm{H}} + \delta^{\mathrm{fail}}.\end{aligned} \tag{192}$$

This proves equation 132 for each fixed oracle. Taking the supremum over $q \in \mathcal{Q}^{\mathrm{Hel}}(P;\varepsilon^{\mathrm{H}})$ gives equation 133. With the choice equation 134, its right-hand side is $3\varepsilon^{\mathrm{H}}/2$. □

F Case (ii): forward-KL upper bound

We prove case (ii) of Theorem 3.1, with the finite guarantee in Theorem F.3. The same sampler uses the row-KL and screen-failure budgets defined below. Its analysis uses a reference process that draws each batch from the true *joint* conditional, even for unsafe batches. This differs from the product-of-exact-rows hybrid used for TV.

Remark (Relation to the TV formulation). The Hellinger/TV and forward-KL guarantees use the same product-commit rule with different public accuracy calibrations. The Hellinger condition (A2) applies to the TV case.

F.1 Oracle condition and output objective

We use the same frozen row family, probe/commit operations, and resource counts as in Definitions A.6, A.9 and A.11, with the following forward-KL accuracy condition and output risk, corresponding to Assumption 2.7 and Definition 2.6.

Definition F.1 (Uniform forward-KL oracle and seed-averaged KL risk)**.** For a strictly positive target law and $\kappa \ge 0$, let $\mathcal{Q}^{\mathrm{KL}}(P;\kappa)$ consist of the strictly positive deterministic frozen maps of Definition A.6 such that

$$\sup_{(y,j):\, j \notin O(y)} D_{\mathrm{KL}}(\mu_j(\cdot \mid y) \| q_j(\cdot \mid y)) \le \kappa. \tag{193}$$

For a complete admissible decision rule, use the output law of Definition A.10 and define

$$\mathcal{R}_{\mathrm{KL}}(\mathcal{A}; P, q) := \mathbb{E}_W D_{\mathrm{KL}}\left(P \middle\| \widehat{P}_W^q\right). \tag{194}$$

The decision-rule seed excludes the primitive randomness used to draw commit values. All KL divergences use natural logarithms and the displayed forward direction: target or exact conditional first, generated law or oracle second.

The bound in equation 193 is uniform over *all* masked states, including counterfactual and draft-filled inputs, not merely an average under a data–mask or generated-trajectory law. Lemma H.2 implies

$$\kappa \le \frac{(\varepsilon^{\mathrm{H}})^2}{N} \quad \Longrightarrow \quad \mathcal{Q}^{\mathrm{KL}}(P;\kappa) \subseteq \mathcal{Q}^{\mathrm{Hel}}(P;\varepsilon^{\mathrm{H}}). \tag{195}$$

Thus all existing preprocessing and screen certificates apply with this public Hellinger radius. Neither the converse inclusion nor an output-KL bound from (A2) alone is asserted. Reverse row KL is sufficient for (A2), but is not the assumption used for the forward-KL result here.

F.2 Adaptive joint-reference identity

Learning error and conditional total correlation also appear in the adaptive planner decomposition of Lavenant & Zanella (2025, Proposition 2) (arXiv:2510.25544v2). For completeness, we prove the fixed-seed equality for our frozen-oracle interface, keeping counterfactual operations in the adaptive execution path.

The joint-reference process. Fix the frozen oracle and a decision-rule seed w. Keep the decision rule, counterfactual queries, oracle replies, public row selector, and draft-generation rule fixed, but replace each commit kernel by the *true joint conditional* of that batch given the committed history. Denote this analysis-only process by $\mathbb{P}_w^{\mathrm{joint}}$; its dependence on the target, oracle, and decision rule is suppressed. Unlike $P_w^{\otimes}$, it draws a dependent batch jointly when the decision rule selects an unsafe batch. It need not be implementable by the available oracle.

Its conditional batch law and total correlation. Here $\mathfrak{h}$ is the stored pre-commit execution-tree state from the proof of Lemma E.10: it includes the committed history, transcript, and internal decision-rule state. At this node, let $G(\mathfrak{h})$ and $x_{G(\mathfrak{h})}$ be the committed set and values, $B(\mathfrak{h})$ the next batch, and $y(\mathfrak{h})$ the corresponding commit state. Put

$$p(\cdot \mid \mathfrak{h}) := \mathcal{L}_P(X_{B(\mathfrak{h})} \mid X_{G(\mathfrak{h})} = x_{G(\mathfrak{h})}).$$

Its singleton marginals are $\mu_j(\cdot \mid y(\mathfrak{h}))$. The conditional batch's total correlation is

$$\mathrm{TC}(p(\cdot \mid \mathfrak{h})) := D_{\mathrm{KL}}\left(p(\cdot \mid \mathfrak{h}) \middle\| \bigotimes_{j \in B(\mathfrak{h})} \mu_j(\cdot \mid y(\mathfrak{h})) \right).$$

Intuition. For one batch, insert the product of its true singleton marginals into the likelihood ratio:

$$\log \frac{p(X_B)}{\prod_j q_j(X_j)} = \log \frac{p(X_B)}{\prod_j p_j(X_j)} + \sum_j \log \frac{p_j(X_j)}{q_j(X_j)}.$$

Expectation under the true joint law gives the batch's total correlation plus its row-KL errors. The adaptive chain rule then adds these quantities along the true-joint reference path.

Lemma F.2 (Joint-reference chain rule and KL decomposition (cf. Ben-Hamu et al., 2025, Eq. (7) and Appendix A.1))**.** *For every complete admissible decision rule and every fixed seed w, the output law under $\mathbb{P}_w^{\mathrm{joint}}$ equals P, even if its batches are not structurally safe. Moreover,*

$$\begin{aligned} &D_{\mathrm{KL}}\Big(P \Big\| \widehat{P}_w^{q}\Big) \\ &\quad = \mathbb{E}_{\mathbb{P}_w^{\mathrm{joint}}} \sum_t \left[\mathrm{TC}(p(\cdot \mid \mathfrak{h}_t)) + \sum_{j \in B(\mathfrak{h}_t)} D_{\mathrm{KL}}(\mu_j(\cdot \mid y(\mathfrak{h}_t)) \| q_j(\cdot \mid y(\mathfrak{h}_t))) \right]. \end{aligned} \tag{196}$$

Here t runs over the nonempty commits on the reference path. Under equation 193, the sum of row-KL terms is at most $N\kappa$ on every such path. The sum of total-correlation terms is at most $N \log V$ on every path and is zero whenever all batches are structurally safe for the target forest.

Proof. 1. Identify the reference output law along each adaptive path. With w and the frozen map fixed, all operations between commits are deterministic functions of the observed past. Every full

assignment x therefore determines a unique execution path. For a reachable node $\mathfrak{h}$, the full assignments that visit it form the cylinder

$$\{u \in \mathcal{V}^N : \mathfrak{h} \text{ lies on the fixed-seed path of } u\} = \{u \in \mathcal{V}^N : u_{G(\mathfrak{h})} = x_{G(\mathfrak{h})}\}.$$

For the forward inclusion, visiting the node requires its committed values. For the reverse inclusion, induct along the path to $\mathfrak{h}$: every earlier committed set is a subset of $G(\mathfrak{h})$, so these assignments produce the same earlier commit values, states, frozen-oracle replies, and next actions. Counterfactual replies are functions of the submitted states, not of the uncommitted realization of u. Thus the transcript imposes no further restriction on that realization.

For the ordered partition $(B_t(x))_t$ chosen along this path, the adaptive chain rule gives $\prod_t P\big(X_{B_t(x)} = x_{B_t(x)} \mid X_{G_{t-1}(x)} = x_{G_{t-1}(x)}\big) = P(x)$. The product telescopes through successive committed sets exactly as in equation 174. This is the reference output mass, so the reference output has law P. The cylinder identity then gives $\mathbb{P}_w^{\text{joint}}(\text{reach } \mathfrak{h}) = P(X_{G(\mathfrak{h})} = x_{G(\mathfrak{h})}) > 0$ and the node conditional $\mathbb{P}_w^{\text{joint}}(X_{B(\mathfrak{h})} = z \mid \text{reach } \mathfrak{h}) = P(X_{B(\mathfrak{h})} = z \mid X_{G(\mathfrak{h})} = x_{G(\mathfrak{h})}) = p(z \mid \mathfrak{h})$, where $p(z \mid \mathfrak{h})$ is the mass of $p(\cdot \mid \mathfrak{h})$. The actual output mass along the same path is $\widehat{P}_w^q(x) = \prod_t \prod_{j \in B_t(x)} q_j(x_j \mid y(\mathfrak{h}_t))$.

2. Group the pathwise log ratios by reached nodes. Strict positivity makes both masses positive. Since the reference output has law P and its conditional at each reached node $\mathfrak{h}$ is $p(\cdot \mid \mathfrak{h})$, grouping paths by these nodes gives

$$\begin{aligned}
&D_{\mathrm{KL}}\Big(P\Big\|\widehat{P}_w^q\Big) \\
&= \sum_x P(x) \sum_{\mathfrak{h} \text{ on the path of } x} \log \frac{p(x_{B(\mathfrak{h})} \mid \mathfrak{h})}{\prod_{j \in B(\mathfrak{h})} q_j(x_j \mid y(\mathfrak{h}))} \\
&= \sum_{\mathfrak{h}} \Pr_{\mathbb{P}_w^{\text{joint}}}\{\text{reach } \mathfrak{h}\} \sum_z p(z \mid \mathfrak{h}) \log \frac{p(z \mid \mathfrak{h})}{\prod_{j \in B(\mathfrak{h})} q_j(z_j \mid y(\mathfrak{h}))} \\
&= \mathbb{E}_{\mathbb{P}_w^{\text{joint}}} \sum_t D_{\mathrm{KL}}\left(p(\cdot \mid \mathfrak{h}_t)\middle\|\bigotimes_{j \in B(\mathfrak{h}_t)} q_j(\cdot \mid y(\mathfrak{h}_t))\right).
\end{aligned}$$

Here $\mathfrak{h}$ ranges over nonterminal nodes of the fixed-seed execution tree. Every path has at most N nonempty commits and each batch has finitely many outcomes, so the sums are finite. Each node contributes exactly when the path visits it; no common path length is required.

3. Decompose the batch KL. For a positive batch law p with singleton marginals p_j, recall $\mathrm{TC}(p) = \mathbb{E}_p \log \frac{p(X_B)}{\prod_j p_j(X_j)}$. Inserting the product of these marginals into the likelihood ratio gives $D_{\mathrm{KL}}\Big(p\Big\|\bigotimes_j q_j\Big) = \mathrm{TC}(p) + \sum_j \mathbb{E}_p \log \frac{p_j(X_j)}{q_j(X_j)}$. For each j, expanding the expectation gives $\mathbb{E}_p \log \frac{p_j(X_j)}{q_j(X_j)} = \sum_{a \in \mathcal{V}} \big(\sum_{x_B : x_j = a} p(x_B)\big) \log \frac{p_j(a)}{q_j(a)}$. Since $\sum_{x_B : x_j = a} p(x_B) = p_j(a)$, this equals $\sum_{a \in \mathcal{V}} p_j(a) \log \frac{p_j(a)}{q_j(a)} = D_{\mathrm{KL}}(p_j \| q_j)$. Thus $D_{\mathrm{KL}}\Big(p\Big\|\bigotimes_j q_j\Big) = \mathrm{TC}(p) + \sum_j D_{\mathrm{KL}}(p_j \| q_j)$, which proves equation 196 after substitution into Step 2.

4. Bound the pathwise sums. Uniform row accuracy and $\sum_t |B(\mathfrak{h}_t)| = N$ give the $N\kappa$ bound. For a finite probability mass p, write $H(p) := -\sum_z p(z) \log p(z)$, with $0 \log 0 := 0$. For a batch law on $\mathcal{V}^B$ with marginals p_j, expanding gives $\mathrm{TC}(p) = \sum_z p(z) \log p(z) - \sum_{j \in B} \sum_{a \in \mathcal{V}} \big(\sum_{z : z_j = a} p(z)\big) \log p_j(a)$. Substituting $\sum_{z : z_j = a} p(z) = p_j(a)$ yields $\mathrm{TC}(p) = \sum_{j \in B} H(p_j) - H(p)$. Here $H(p) \geq 0$ because $p(z) \leq 1$, and $H(p_j) \leq \log V$ follows from $D_{\mathrm{KL}}(p_j \| \mathrm{Unif}(\mathcal{V})) = \log V - H(p_j) \geq 0$. Apply this calculation with $p = p(\cdot \mid \mathfrak{h})$ and $p_j = \mu_j(\cdot \mid y(\mathfrak{h}))$ to obtain $0 \leq \mathrm{TC}(p(\cdot \mid \mathfrak{h})) \leq |B(\mathfrak{h})| \log V$. Summing over the disjoint batches gives $N \log V$. Finally, a structurally safe batch contains at most one vertex per true residual component (or is a singleton). Lemma E.2 makes its true conditional a product, so its total correlation is zero. □

F.3 FINITE BOUND AND THE EPSILON CALIBRATION

As in the finite case (i) bound of Theorem E.1, preprocessing uses the standard floor; feasibility and screen separation remain explicit hypotheses. With the accuracy calibration in Corollary F.4, the theorem below yields case (ii) of Theorem 3.1 through Corollary G.3.

Intuition. Every path commits N coordinates, so the row-error sum is at most $N\kappa$. Successful reference paths use only safe batches and have zero batch total correlation. On all other paths the total correlation is at most $N \log V$. Thus

$$\text{expected output KL} \le N\kappa + N \log V \Pr_{\text{joint reference and seed}}(\mathsf{Fail}).$$

The fresh-color argument bounds this last probability by δ^{fail} under the reference process as well.

Theorem F.3 (Case (ii): finite randomized forward-KL upper bound)**.** *We work in case (ii), with the oracle class and output risk of Definition F.1. Fix $N \ge 10$, $V \ge 2$, and the target, response, and algorithmic setup of Appendices A and D. Let $\varepsilon^{\mathrm{H}} \in [0,1]$ and $0 \le \kappa \le (\varepsilon^{\mathrm{H}})^2/N$. Use Algorithm 2 with the standard zero-query draft, the public radius ε^{H}, and $\delta^{\mathrm{fail}} \in (0,1)$. Suppose preprocessing is feasible and $\delta < \omega$. For every $q \in \mathcal{Q}^{\mathrm{KL}}(P;\kappa)$,*

$$\mathcal{R}_{\mathrm{KL}}(\mathcal{A}^{\mathrm{rnd}}_{\mathrm{pack}}; P, q) \le N\kappa + N \log V \, \delta^{\mathrm{fail}}. \tag{197}$$

On every path, including failed-screen paths, the resources satisfy

$$\begin{aligned} Q_{\mathrm{cf}} &\le \Lambda T_{\mathrm{scr}} M_{\mathrm{rnd}} p_{\mathrm{rnd}} (p_{\mathrm{rnd}} + J), \\ R \le \overline{R}, &\qquad D \le 1 + R + T_{\mathrm{scr}}. \end{aligned} \tag{198}$$

The parameters on the right are the original finite parameters in equation 116, equation 117, and equation 105. Charging the initial all-mask state gives $Q_{\mathrm{nc}} = Q_{\mathrm{cf}} + 1$ by equation 34, so its submission bound is the displayed bound for Q_{cf} plus one. If feasibility and separation hold for every oracle in this KL class, the supremum of the risk in equation 197 over that class has the same bound.

Proof. 1. Transfer the oracle and resource guarantees. By Lemma H.2 and equation 193, every valid (y, j) satisfies

$$h^2(\mu_j(\cdot \mid y), q_j(\cdot \mid y)) \le \tfrac{1}{2} D_{\mathrm{KL}}(\mu_j(\cdot \mid y) \| q_j(\cdot \mid y)) \le \tfrac{1}{2}\kappa \le \frac{(\varepsilon^{\mathrm{H}})^2}{2N}.$$

Thus the oracle meets Assumption A.7 (A2), as recorded in equation 195, and Lemma E.8 transfers the preprocessing, screen, and resource guarantees. The output-KL bound instead uses the preceding Lemma F.2.

2. Bound the KL cost at a fixed seed. Using the call notation of Lemma E.4 on the joint-reference execution, define the reached-screen failure event by

$$\mathsf{Fail} := \bigcup_{s=1}^{T_{\mathrm{scr}}} \{R_s \text{ occurs and } \mathsf{Iso}(H^{(s)}) \text{ fails}\}.$$

On $\mathsf{Fail}^{\mathsf{c}}$, the deterministic successful-path certificates in Lemmas E.7 and E.8 imply that neither cycle repair nor GUARD is activated, and every commit is a singleton peel in Step 4 or a safe centroid batch in Step 6. These statements hold for every sequence of realized values, so they also hold for reference commits. Thus the total-correlation cost vanishes on $\mathsf{Fail}^{\mathsf{c}}$, while the pathwise bound from Lemma F.2 applies on the failure event:

$$\sum_t \mathrm{TC}(p(\cdot \mid \mathfrak{h}_t)) = \mathbf{1}_{\mathsf{Fail}} \sum_t \mathrm{TC}(p(\cdot \mid \mathfrak{h}_t)) \le N \log V \, \mathbf{1}_{\mathsf{Fail}}.$$

Repair batches are singletons, so each has zero total correlation. Their coordinates still enter the row-KL sum only once, as for every other commit; on every path,

$$\sum_t \sum_{j \in B(\mathfrak{h}_t)} D_{\mathrm{KL}}(\mu_j(\cdot \mid y(\mathfrak{h}_t)) \| q_j(\cdot \mid y(\mathfrak{h}_t))) \le \kappa \sum_t |B(\mathfrak{h}_t)| = N\kappa.$$

Taking the reference expectation replaces the indicator by $\mathbb{P}_w^{\mathrm{joint}}(\mathsf{Fail})$. Adding the pathwise row-KL budget $N\kappa$ in Lemma F.2 therefore gives, for each fixed w,

$$D_{\mathrm{KL}}\Big(P\Big\|\widehat{P}_w^q\Big) \le N\kappa + N\log V\,\mathbb{P}_w^{\mathrm{joint}}(\mathsf{Fail}). \tag{199}$$

3. Average the screen failure under the joint-reference law. It remains to average the failure term over the decision-rule seed. Generate reference commit values with a fresh primitive random source independent of the color blocks and draft randomness. Just before each reached color draw in Step 2, condition on the reference history and completed draft generation, but not on the current or future colors. The residual forest and draft are then fixed and the new colors remain independent and uniform. The proof of Lemma E.4 uses only these facts, not the law of preceding commit values: for each ordered pair, at most $2d$ color equalities are forbidden, each with probability $1/p_{\mathrm{rnd}}$, and independent repetition gives the failure bound $\exp\{-M_{\mathrm{rnd}}/8\}$. Union bounds over fewer than N^2 pairs give the same conditional bound at a reached reference call. To pass from that conditional estimate to the seed average, define the joint probability of an event A by

$$\Pr_{\mathrm{joint}}(A) := \mathbb{E}_W\mathbb{P}_W^{\mathrm{joint}}(A).$$

Form $\mathcal{T}_{s-}, f^{(s)}, \mathcal{F}_{s-1}, R_s, E_s$ exactly as in Lemma E.4, now from the reference execution. In particular, $\mathcal{F}_{s-1} = \sigma(\mathcal{T}_{s-}, f^{(s)})$ excludes the current and future colors, and $\mathsf{Fail} = \bigcup_{s=1}^{T_{\mathrm{scr}}} E_s$. Then

$$\Pr_{\mathrm{joint}}(E_s \mid \mathcal{F}_{s-1}) \le \mathbf{1}_{R_s}\frac{\delta^{\mathrm{fail}}}{T_{\mathrm{scr}}}.$$

The union bound and conditional expectation give

$$\begin{aligned}\mathbb{E}_W\mathbb{P}_W^{\mathrm{joint}}(\mathsf{Fail}) &= \Pr_{\mathrm{joint}}\left(\bigcup_{s=1}^{T_{\mathrm{scr}}} E_s\right)\\ &\le \sum_{s=1}^{T_{\mathrm{scr}}}\mathbb{E}_{\mathrm{joint}}\left[\Pr_{\mathrm{joint}}(E_s \mid \mathcal{F}_{s-1})\right]\\ &\le \frac{\delta^{\mathrm{fail}}}{T_{\mathrm{scr}}}\sum_{s=1}^{T_{\mathrm{scr}}}\Pr_{\mathrm{joint}}(R_s) \le \delta^{\mathrm{fail}}.\end{aligned}$$

This sums conditional failure probabilities over adaptively reached calls; it does not require the calls to be independent. We have therefore proved

$$\mathbb{E}_W\mathbb{P}_W^{\mathrm{joint}}(\mathsf{Fail}) \le \delta^{\mathrm{fail}}. \tag{200}$$

This is a joint probability over fresh colors and reference commit draws; it is *not* a failure-probability bound conditional on the full seed w. Averaging equation 199 proves equation 197. In particular, failed-screen paths are charged through their finite total-correlation cost, not by attempting to convert a TV coupling bound into KL. No minimum atom size or bounded likelihood ratio is assumed beyond the existing strict positivity. □

Corollary F.4 (Case (ii) with output KL at most epsilon)**.** *Fix $0 < \varepsilon^{\mathrm{KL}} \le 1$ and use the preceding finite target and algorithmic setup. Choose*

$$\kappa := \frac{\varepsilon^{\mathrm{KL}}}{2N}, \qquad \delta^{\mathrm{fail}} := \frac{\varepsilon^{\mathrm{KL}}}{2N\log V}. \tag{201}$$

Run the existing algorithm with its public radius set to $\sqrt{\varepsilon^{\mathrm{KL}}/2}$. If its preprocessing is feasible and $\delta < \omega$ with this radius, then for every $q \in \mathcal{Q}^{\mathrm{KL}}(P;\kappa)$,

$$\mathcal{R}_{\mathrm{KL}}(\mathcal{A}_{\mathrm{pack}}^{\mathrm{rnd}}; P, q) \le \varepsilon^{\mathrm{KL}}, \tag{202}$$

with the pathwise bounds in equation 198. Thus the per-row budget scales as $\varepsilon^{\mathrm{KL}}/N$, not $(\varepsilon^{\mathrm{KL}})^2/N$.

Proof. The auxiliary radius is in $[0,1]$, $\kappa = (\sqrt{\varepsilon^{\mathrm{KL}}/2})^2/N$, and the displayed failure budget is in $(0,1)$ for $N \ge 10$, $V \ge 2$. Each term on the right of equation 197 is $\varepsilon^{\mathrm{KL}}/2$. □

The auxiliary radius $\sqrt{\varepsilon^{\mathrm{KL}}/2}$ is an input calibration for case (ii), not a redefinition of the case (i) parameter ε^{H}.

Remark (Additional metric consequences). Pinsker's inequality and Jensen's inequality imply from equation 202 that $\mathbb{E}_W d_{\mathrm{TV}}(P, \widehat{P}_W^q)^2 \le \varepsilon^{\mathrm{KL}}/2$ and the mean TV is at most $\sqrt{\varepsilon^{\mathrm{KL}}/2}$. This explains the KL versus squared-TV scaling convention in the comparison; it is not a two-sided equivalence of the metrics. Convexity also bounds the forward KL of the seed-marginalized output by equation 194; the latter, stronger objective is the one proved here.

G From finite bounds to the main theorems

We derive the main-text Theorems 3.1 and 3.2 directly from the finite results. Under the main-text scaling in Assumption 2.4 (Scaling), the row-TV allowance $\varepsilon/(2\sqrt{N})$ and vocabulary floor V^{-1} are $o(\omega)$. Lemma G.1 verifies the resulting finite upper-bound design conditions. Next, Corollary G.2 calibrates the finite lower bound with an independent Hellinger radius and checks the matching witness's class inclusion at the actual finite public values. With both ingredients in place, Corollary G.3 derives both main theorems with accuracy fixed before $N \to \infty$, using $\varepsilon^{\mathrm{H}} = \varepsilon/2$ in case (i) and $\varepsilon^{\mathrm{KL}} = \varepsilon^2/2$ in case (ii).

G.1 A finite design condition

Intuition. With response constant and exponent equal to one, choosing $\delta_{\mathrm{tail}} = \omega/2$ leaves the other half of the edge signal for row noise. If both $x/\sqrt{N}$ and V^{-1} are sufficiently below ω, the threshold grid has a feasible point with $t \simeq \omega$. The bank bound then gives a column upper bound of order $\omega^{-1/s}$. The following finite inequalities specify the margins and the grid rounding. Here x denotes the public radius supplied to preprocessing: use $x = \varepsilon^{\mathrm{H}}$ under (A2), or the Hellinger radius obtained from the KL inclusion equation 195. It is not a separate output-error tolerance.

Lemma G.1 (Public feasibility with a half-signal tail tolerance)**.** *Let $N \ge 10$, $V \ge 2$, $s > 1$, and $\alpha = L = C = 1$. Use a public preprocessing radius $x \in (0, 1]$, so the marginal and screen TV error bounds used by the construction are $x/\sqrt{N}$. Suppose*

$$16 \max\{x/\sqrt{N}, V^{-1}\} \le \omega \le \frac{1}{2}. \tag{203}$$

Set $\delta_{\mathrm{tail}} = \omega/2$ and use the standard threshold floor equation 107. Then preprocessing is feasible, the selected threshold satisfies

$$\frac{\omega}{4} \le t \le \frac{\omega}{2}, \qquad 4x/\sqrt{N} + \delta_{\mathrm{tail}} \le \frac{3}{4}\omega < \omega, \tag{204}$$

and, under (RF; Equation (21)*), the number of source columns obeys*

$$\Lambda \le \left(\frac{16}{\omega}\right)^{1/s} + 1. \tag{205}$$

Proof. 1. Exhibit a feasible grid point. Under the stated constants, the public floor is exactly

$$\underline{t} = 4 \max\{x/\sqrt{N}, V^{-1}\} \le \omega/4.$$

Specifically, choose

$$t' = 2^{-\lceil \log_2(2/\omega) \rceil}, \qquad \omega/4 < t' \le \omega/2.$$

The bracket follows by applying $a \le \lceil a \rceil < a + 1$ to $a = \log_2(2/\omega)$. Since $t' > \underline{t}$ and $\lceil \log_2(2/\omega) \rceil \le \lceil \log_2(1/\underline{t}) \rceil = J^{\mathrm{grid}}$, this is an unclipped point of the grid in equation 108. Every such point is feasible because its response bound is the threshold itself, at most $\delta_{\mathrm{tail}} = \omega/2 < 1$. The maximal feasible grid point lies in the same interval.

2. Verify separation and bound the bank. The screen error bound in equation 203 gives $4x/\sqrt{N} \le \omega/4$, proving strict separation. Lemma E.3 now bounds every bank by $(4/t)^{1/s} \le (16/\omega)^{1/s}$. Adding the tail column proves equation 205; if every bank is empty the column count is zero and the same bound holds. □

G.2 FINITE LOWER-BOUND CALIBRATION FOR THE MAIN TEXT

The next corollary supplies the finite lower bound used in Theorem 3.2. It places the matching witness in the normalized Hellinger/TV model of Section A.3, then calibrates the signal to the independently specified radius ε^{H}. The main-text class takes $\varepsilon^{\mathrm{H}} = \varepsilon/2$. The fixed-accuracy limit, including removal of the floor and the finite round minimum, is carried out in Corollary G.3.

Corollary G.2 (Signal-calibrated finite query–round lower bound)**.** *Let $N \geq 4$ be even, let $V \geq 2$ be an integer, and fix $s > 1$, $0 < \varepsilon^{\mathrm{H}} \leq 1$, and $\alpha = L = C = 1$. Suppose*

$$0 < \omega \leq \frac{1}{2}, \qquad V\omega > 1, \qquad \omega^3 \leq \frac{(\varepsilon^{\mathrm{H}})^2}{2N}. \tag{206}$$

Fix $0 < \varepsilon \leq 1/8$. Every admissible algorithm with integer worst-case pathwise budgets $\overline{Q} \geq 0$, $\overline{R} \geq 1$ and seed-averaged output-TV error at most ε uniformly over the target–oracle tuples satisfying the structural and response assumptions of Appendix A, the finite envelope equation 21, and Assumption A.7 (A2), with these public parameters satisfies

$$\overline{Q} > \frac{\lfloor \omega^{-1/s} \rfloor}{8} \quad \text{or} \quad \overline{R} \geq \min\left\{ \frac{N}{16384}, \frac{N\omega^2}{786432\varepsilon} \right\}. \tag{207}$$

For the benchmark choice $\omega = \frac{1}{2}\left(\frac{(\varepsilon^{\mathrm{H}})^2}{N}\right)^{1/3}$, the conditions in equation 206 reduce to $V > 2(N/(\varepsilon^{\mathrm{H}})^2)^{1/3}$, and the alternative becomes

$$\begin{aligned} &\overline{Q} > \frac{1}{8}\left\lfloor 2^{1/s}\left(\frac{N}{(\varepsilon^{\mathrm{H}})^2}\right)^{1/(3s)} \right\rfloor \quad \text{or} \\ &\overline{R} \geq \min\left\{ \frac{N}{16384}, \frac{N^{1/3}(\varepsilon^{\mathrm{H}})^{4/3}}{3145728\varepsilon} \right\}. \end{aligned} \tag{208}$$

Proof. 1. Construct the witness using the same public parameters. Keep the public vocabulary of size V, its order, s, ω, and ε^{H} from the statement, with $\alpha = L = C = 1$. In the matching family of Definition B.1, choose

$$\eta = \rho = \omega, \qquad m = \lfloor \omega^{-1/s} \rfloor. \tag{209}$$

The inequalities

$$\begin{aligned} 1 \leq m \leq \omega^{-1/s} < \omega^{-1} < V, \\ m\omega \leq \omega^{1-1/s} < 1, \qquad \omega \leq m^{-s} \end{aligned}$$

give $1 \leq m < V$ and $m\rho < 1$. Together with $0 < \eta = \rho = \omega \leq 1/2$, these verify equation 35. For the last rank inequality, explicitly, $m \leq \omega^{-1/s}$ implies $m^s \leq \omega^{-1}$, hence $\rho = \omega \leq m^{-s}$. The remaining hypotheses of Proposition C.2 are

$$V\rho = V\omega > 1, \qquad \omega = \eta, \qquad \eta^2\rho = \omega^3 \leq \frac{(\varepsilon^{\mathrm{H}})^2}{2N}.$$

Thus that proposition applies at the actual finite public values. It verifies Assumption 2.2 and the RT, UEN, and RF items of Assumption 2.4, and the Hellinger condition (A2) in Assumption A.7. At $\varepsilon^{\mathrm{H}} = \varepsilon/2$, the latter is precisely Assumption 2.7(i). The vocabulary-growth requirement is additionally supplied when taking the public sequence in Assumption 2.4 (Scaling).

2. Restrict the uniform guarantee and apply the finite tradeoff. Fix any admissible algorithm satisfying this corollary's risk and pathwise-budget hypotheses. Every target–oracle pair in the chosen matching family belongs to the stipulated finite class, so the same algorithm satisfies

$$\sup_{I \in \mathfrak{I}^{\mathrm{hard}}[V,m,\omega,\omega]} \mathcal{R}_{\mathrm{TV}}(\mathcal{A}; P^I, q^I) \leq \varepsilon.$$

Its deterministic budgets $\overline{Q}, \overline{R}$ still hold on every instance, seed, transcript, and path in this subfamily. Apply Corollary B.3 at tolerance ε, with $\eta = \omega$; the required tolerance range is precisely $0 < \varepsilon \leq 1/8$. If $\overline{Q} > m/8$, the query alternative in equation 207 already holds. Otherwise that corollary implies either $\overline{R} > N/16384$ or $\overline{R} \geq N\omega^2/(786432\varepsilon)$; either alternative implies $\overline{R} \geq \min\{N/16384, N\omega^2/(786432\varepsilon)\}$. Substituting $m = \lfloor \omega^{-1/s} \rfloor$ proves equation 207.

3. Substitute the signal calibration. Use the benchmark signal $\omega = \frac{1}{2}\left(\frac{(\varepsilon^{\mathrm{H}})^2}{N}\right)^{1/3} = \frac{1}{2}((\varepsilon^{\mathrm{H}})^2/N)^{1/3}$. It places the matching witness inside the public oracle-accuracy class while retaining the $\omega^{-1/s}$ query threshold. This is the signal choice in Assumption 2.4 (Scaling) at $\varepsilon^{\mathrm{H}} = \varepsilon/2$. Since $0 < \varepsilon^{\mathrm{H}} \le 1$ and $N \ge 4$, this gives $0 < \omega \le 1/2$, and

$$\omega^3 = \frac{1}{8}\,\frac{(\varepsilon^{\mathrm{H}})^2}{N} = \frac{(\varepsilon^{\mathrm{H}})^2}{8N} \le \frac{(\varepsilon^{\mathrm{H}})^2}{2N}.$$

The only remaining condition in equation 206 is the vocabulary condition:

$$V\omega > 1 \quad\Longleftrightarrow\quad \frac{V}{2}\left(\frac{(\varepsilon^{\mathrm{H}})^2}{N}\right)^{1/3} > 1 \quad\Longleftrightarrow\quad V > 2\left(\frac{N}{(\varepsilon^{\mathrm{H}})^2}\right)^{1/3}.$$

The two resource thresholds become

$$\omega^{-1/s} = \left[\frac{1}{2}\left(\frac{(\varepsilon^{\mathrm{H}})^2}{N}\right)^{1/3}\right]^{-1/s} = 2^{1/s}\left(\frac{N}{(\varepsilon^{\mathrm{H}})^2}\right)^{1/(3s)},$$

$$\frac{N\omega^2}{786432\varepsilon} = \frac{N}{786432\varepsilon}\,\frac{1}{4}\left(\frac{(\varepsilon^{\mathrm{H}})^2}{N}\right)^{2/3} = \frac{N^{1/3}(\varepsilon^{\mathrm{H}})^{4/3}}{3145728\varepsilon}.$$

Substitution into equation 207 proves equation 208, with its floor and minimum still intact. The lower-bound part of Corollary G.3 completes the passage from this finite statement to Theorem 3.2. □

G.3 Fixed-accuracy specialization for the main text

The main text fixes its accuracy parameter before taking $N \to \infty$. We first retain independent oracle and output budgets, then specialize them at the end of the proof. The cutoff d remains a direct parameter, and the logarithmic overhead is displayed explicitly.

The three main claims use the following proof routes; the final part of Corollary G.3 performs their common main-text specialization to the classes in Definition 2.8.

Main claim	Finite result and calibration
Theorem 3.1, case (i)	Theorem E.1 and Lemma G.1
Theorem 3.1, case (ii)	Theorem F.3, Corollary F.4, and Lemma G.1
Theorem 3.2	Theorem B.2, Corollaries B.3 and G.2

Corollary G.3 (Fixed-accuracy main-text bounds)**.** *Fix $s > 1$, $\nu > 1/3$, and a public $V = N^{\nu+o(1)}$. For simplicity, set $\alpha = L = C = 1$ and take all target–oracle tuples satisfying the forest, RT–UEN (Equations* (18) *and* (19)*), and RF (Assumption A.5) assumptions of Appendix A, with either calibration:*

- Case (i): *fix $0 < \varepsilon^{\mathrm{H}} \le 1$, use (A2), and set $\omega = \frac{1}{2}\left(\frac{(\varepsilon^{\mathrm{H}})^2}{N}\right)^{1/3}$.*
- Case (ii): *fix $0 < \varepsilon^{\mathrm{KL}} \le 1$, impose uniform forward row KL at most $\varepsilon^{\mathrm{KL}}/(2N)$, and set $\omega = \frac{1}{2}\left(\frac{\varepsilon^{\mathrm{KL}}}{2N}\right)^{1/3}$.*

The exponents s, ν and the chosen accuracy parameter (ε^{H} or $\varepsilon^{\mathrm{KL}}$) are fixed independently of N; the vocabulary size, signal floor, and per-row oracle-error allowance follow the displayed N-dependent calibrations. Here $\mathfrak{F}^{\mathrm{sh}}$ is the Hellinger class at these public values, and $\mathcal{R}^{\mathrm{sh}}_{\mathrm{TV}}(\mathcal{A}) := \sup_{I\in\mathfrak{F}^{\mathrm{sh}}} \mathcal{R}_{\mathrm{TV}}(\mathcal{A}; P^I, q^I)$ abbreviates its worst-case seed-averaged TV risk. For every public integer choice $9 \le d \le N-1$ and $J = d$, use the standard draft, the standard threshold floor, and $\delta_{\mathrm{tail}} = \omega/2$. Choose $\delta^{\mathrm{fail}} = \varepsilon^{\mathrm{H}}/2$ in case (i), or $\delta^{\mathrm{fail}} = \varepsilon^{\mathrm{KL}}/(2N\log V)$ in case (ii). For all sufficiently large N, uniformly over this cutoff range and the respective target–oracle class, the construction is feasible and

$$Q_{\mathrm{cf}} = O\Big(\omega^{-1/s} d^2 (\log N)^2\Big), \qquad R, D = O\left(\frac{N}{d}\log N\right). \tag{210}$$

These bounds hold on every path. The output guarantees are $\mathcal{R}^{\mathrm{sh}}_{\mathrm{TV}}(\mathcal{A}^{\mathrm{rnd}}_{\mathrm{pack}}) \le 3\varepsilon^{\mathrm{H}}/2$ *in case (i) and expected forward KL at most* $\varepsilon^{\mathrm{KL}}$ *in case (ii).*

For case (i), every admissible algorithm with uniform TV risk at most the fixed $0 < \varepsilon \le 1/8$ *satisfies, for all sufficiently large even* N*,*

$$\overline{Q} = \Omega\Big((N/(\varepsilon^{\mathrm{H}})^2)^{1/(3s)}\Big) \quad \textit{or} \quad \overline{R} = \Omega\Big(N^{1/3}(\varepsilon^{\mathrm{H}})^{4/3}/\varepsilon\Big). \tag{211}$$

Taking $\varepsilon^{\mathrm{H}} = \varepsilon/2$ *in case (i) and* $\varepsilon^{\mathrm{KL}} = \varepsilon^2/2$ *in case (ii) gives Theorems 3.2 and 3.1, including their balanced bounds.*

Proof. Upper bound: feasibility for Theorem 3.1. Set $x = \varepsilon^{\mathrm{H}}$ in case (i) and $x = \sqrt{\varepsilon^{\mathrm{KL}}/2}$ in case (ii). Both calibrations have $\omega = \frac{1}{2}(x^2/N)^{1/3}$, so

$$\frac{x/\sqrt{N}}{\omega} = 2x^{1/3}N^{-1/6} \longrightarrow 0, \qquad \frac{V^{-1}}{\omega} = 2x^{-2/3}N^{1/3-\nu-o(1)} \longrightarrow 0.$$

Also $\omega \to 0$. Lemma G.1 therefore supplies preprocessing, screen separation, and $\Lambda = O_s(\omega^{-1/s})$ for all sufficiently large N. This step does not depend on d.

Phase and coloring counts. The finite cap equation 104 satisfies

$$T_{\mathrm{peel}} \le 1 + \frac{\log(2N)}{\log(9/8)} = O(\log N), \qquad T_{\mathrm{scr}} = T_{\mathrm{peel}} + 1 = O(\log N),$$

uniformly for $9 \le d \le N-1$. The chosen failure budgets give respectively

$$M_{\mathrm{rnd}} = \begin{cases} \left\lceil 8 \log \dfrac{2T_{\mathrm{scr}}N^2}{\varepsilon^{\mathrm{H}}} \right\rceil, & \text{case (i)}, \\ \left\lceil 8 \log \dfrac{2T_{\mathrm{scr}}N^3 \log V}{\varepsilon^{\mathrm{KL}}} \right\rceil, & \text{case (ii)}. \end{cases}$$

Each is $O(\log N)$: $\log T_{\mathrm{scr}} = O(\log\log N)$, $\log\log V = O(\log\log N)$, and the logarithms of the inverse accuracy parameters are fixed constants. With $p_{\mathrm{rnd}} = 8(d+1) \le 16d$,

$$p_{\mathrm{rnd}}(p_{\mathrm{rnd}} + J) \le 16d(17d) = 272d^2.$$

Substituting these factors into equation 130 gives

$$\begin{aligned} Q_{\mathrm{cf}} &\le \Lambda\, T_{\mathrm{scr}}\, M_{\mathrm{rnd}}\, p_{\mathrm{rnd}}(p_{\mathrm{rnd}} + J) \\ &\le O_s(\omega^{-1/s})\, O(\log N)\, O(\log N)\, O(d^2) \\ &= O\Big(\omega^{-1/s} d^2 (\log N)^2\Big). \end{aligned}$$

This proves the query bound. The round cap and the depth inequality give

$$\begin{aligned} R &\le \left\lceil \frac{4NT_{\mathrm{peel}}}{d} \right\rceil + \lceil \log_2(N+1) \rceil + 2 \\ &= O\left(\frac{N}{d}\log N + \log N\right) = O\left(\frac{N}{d}\log N\right), \\ D &\le 1 + R + T_{\mathrm{scr}} = O\left(\frac{N}{d}\log N\right). \end{aligned}$$

The last equalities use $N/d \ge 1$. All inputs to these caps are public, so the estimates also cover fallback paths. The finite TV theorem gives $\varepsilon^{\mathrm{H}} + \delta^{\mathrm{fail}} = 3\varepsilon^{\mathrm{H}}/2$. The finite KL calibration gives

$$N\frac{\varepsilon^{\mathrm{KL}}}{2N} + N \log V \, \frac{\varepsilon^{\mathrm{KL}}}{2N \log V} = \varepsilon^{\mathrm{KL}}.$$

For either value of x chosen above,

$$\omega^{-1/s} = \left[\frac{1}{2}(x^2/N)^{1/3}\right]^{-1/s} = 2^{1/s}(N/x^2)^{1/(3s)}.$$

Thus suppressing the displayed powers of $\log N$ gives the resource bound before the final single-accuracy substitution below. The TV tolerance is met when $\varepsilon \geq 3\varepsilon^{\mathrm{H}}/2$; the KL guarantee uses the separate forward-KL hypothesis, not a TV-to-KL conversion. The constants and the sufficiently-large-N threshold depend only on the fixed public parameters and sequences, not on the cutoff, target, or frozen oracle.

Balancing and sublinearity. Put $d = J = \lceil (N\omega^{1/s})^{1/3} \rceil$. Using $\omega = \frac{1}{2}(x^2/N)^{1/3}$, the unrounded value is

$$(N\omega^{1/s})^{1/3} = 2^{-1/(3s)} x^{2/(9s)} N^{1/3-1/(9s)}.$$

Since $s > 1$ and $x > 0$ is fixed, this tends to infinity and is $o(N)$; the cutoff eventually belongs to $\{9, \dots, N-1\}$. For all large N, the unrounded value is at least one, so

$$(N\omega^{1/s})^{1/3} \leq d < (N\omega^{1/s})^{1/3} + 1 \leq 2(N\omega^{1/s})^{1/3}.$$

Consequently rounding changes only a constant factor, and

$$\omega^{-1/s} d^2 = \Theta\Big(N^{2/3}\omega^{-1/(3s)}\Big),$$
$$N/d = \Theta\Big(N^{2/3}\omega^{-1/(3s)}\Big).$$

The two calibrations yield

$$\omega^{-1/(3s)} = \begin{cases} 2^{1/(3s)}(N/(\varepsilon^{\mathrm{H}})^2)^{1/(9s)}, & \text{case (i)}, \\ 2^{4/(9s)}(N/\varepsilon^{\mathrm{KL}})^{1/(9s)}, & \text{case (ii)}. \end{cases}$$

Thus the balanced resources are $\widetilde{O}(N^{2/3+1/(9s)} x^{-2/(9s)})$. After division by N, their upper bounds are a fixed constant times at most $N^{-1/3+1/(9s)}(\log N)^2$, which tends to zero. The additional all-mask submission contributes only $1/N$.

Lower bound: deduction of Theorem 3.2. Work in the corollary's independently parameterized case (i). The public parameters $s > 1$, $0 < \varepsilon^{\mathrm{H}} \leq 1$, and $0 < \varepsilon \leq 1/8$ are fixed before $N \to \infty$, and $V = N^{\nu+o(1)}$ with $\nu > 1/3$. The finite conditions of Corollary G.2 hold for all sufficiently large even N, because

$$\omega = \frac{1}{2}(\varepsilon^{\mathrm{H}})^{2/3} N^{-1/3} \longrightarrow 0,$$
$$V\omega = \frac{1}{2}(\varepsilon^{\mathrm{H}})^{2/3} N^{\nu-1/3+o(1)} \longrightarrow \infty,$$
$$\omega^3 = \frac{(\varepsilon^{\mathrm{H}})^2}{8N} \leq \frac{(\varepsilon^{\mathrm{H}})^2}{2N}.$$

The matching witness in that corollary is therefore a subfamily of the Hellinger class $\mathfrak{F}^{\mathrm{sh}}$ at these public values. For any admissible $\mathcal{A}$ with $\mathcal{R}^{\mathrm{sh}}_{\mathrm{TV}}(\mathcal{A}) \leq \varepsilon$ and the stated deterministic budgets, equation 208 applies.

To remove the query threshold's floor, use

$$u_N := 2^{1/s}(N/(\varepsilon^{\mathrm{H}})^2)^{1/(3s)} \longrightarrow \infty, \qquad \lfloor u_N \rfloor \geq u_N - 1 \geq u_N/2 \quad (u_N \geq 2).$$

If the finite query alternative holds, then

$$\overline{Q} > \frac{\lfloor u_N \rfloor}{8} \geq \frac{u_N}{16} = \frac{2^{1/s}}{16}\left(\frac{N}{(\varepsilon^{\mathrm{H}})^2}\right)^{1/(3s)},$$

which is the query alternative in equation 211. Otherwise the finite corollary forces its round alternative. To identify the smaller term in that minimum,

$$\frac{N^{1/3}(\varepsilon^{\mathrm{H}})^{4/3}/(3145728\varepsilon)}{N/16384} = \frac{(\varepsilon^{\mathrm{H}})^{4/3}}{192\varepsilon N^{2/3}} \longrightarrow 0.$$

Because both accuracy parameters are fixed and positive, this ratio is at most one for all sufficiently large N. Thus the finite round alternative gives

$$\overline{R} \geq \min\left\{\frac{N}{16384}, \frac{N^{1/3}(\varepsilon^{\mathrm{H}})^{4/3}}{3145728\varepsilon}\right\} = \frac{N^{1/3}(\varepsilon^{\mathrm{H}})^{4/3}}{3145728\varepsilon}.$$

Together with the query case, this proves equation 211. The query coefficient $2^{1/s}/16$ depends only on s, and the round coefficient is $1/3145728$. The sufficiently-large-N threshold can depend on the fixed public parameters and vocabulary sequence, but none of these choices depends on the hidden instance or the algorithm.

Single-accuracy substitution for both main theorems. Fix $0 < \varepsilon \le 1/8$. Set $\varepsilon^{\mathrm{H}} = \varepsilon/2$ in case (i) and $\varepsilon^{\mathrm{KL}} = \varepsilon^2/2$ in case (ii). The respective row assumptions become

$$\frac{(\varepsilon/2)^2}{2N} = \frac{\varepsilon^2}{8N}, \qquad \frac{\varepsilon^2/2}{2N} = \frac{\varepsilon^2}{4N},$$

as required in Assumption 2.7. Both cases use $x = \varepsilon/2$, so

$$\omega = \frac{1}{2}\left(\frac{(\varepsilon/2)^2}{N}\right)^{1/3} = \left(\frac{\varepsilon^2}{32N}\right)^{1/3}, \qquad \omega^{-1/s} = \left(\frac{32N}{\varepsilon^2}\right)^{1/(3s)}.$$

This is exactly the target-class calibration in Assumption 2.4 (Scaling). The output bounds are

$$\mathcal{R}^{\mathrm{sh}}_{\mathrm{TV}}(\mathcal{A}^{\mathrm{rnd}}_{\mathrm{pack}}) \le \frac{3\varepsilon^{\mathrm{H}}}{2} = \frac{3\varepsilon}{4} \le \varepsilon, \qquad \mathbb{E}_W D_{\mathrm{KL}}\Big(P\Big\|\widehat{P}^q_W\Big) \le \varepsilon^{\mathrm{KL}} = \frac{\varepsilon^2}{2} \le \varepsilon^2,$$

in cases (i) and (ii), respectively. Substituting $x = \varepsilon/2$ in the resource and balanced bounds proves Theorem 3.1. For the two lower alternatives,

$$(N/(\varepsilon^{\mathrm{H}})^2)^{1/(3s)} = 2^{2/(3s)}(N/\varepsilon^2)^{1/(3s)}, \qquad \frac{N^{1/3}(\varepsilon^{\mathrm{H}})^{4/3}}{\varepsilon} = 2^{-4/3}N^{1/3}\varepsilon^{1/3}.$$

Absorbing only these fixed factors into the constants proves Theorem 3.2. The independent finite statements remain available for other oracle and output budgets. □

Accuracy dependence in the abstract. Fix $0 < \varepsilon \le 1/8$ and take $\varepsilon^{\mathrm{H}} = \varepsilon/2$ in case (i). The output TV bound is then $3\varepsilon/4 \le \varepsilon$. The balanced bound equation 3 gives the abstract's upper form with $C = 2/3 + 1/(9s) < 1$ and $a = 2/(9s) > 0$. The lower alternatives become $\overline{Q} = \Omega(N^{1/(3s)}\varepsilon^{-2/(3s)})$ or $\overline{R} = \Omega(N^{1/3}\varepsilon^{1/3})$. Since $N \ge 1$, $0 < \varepsilon \le 1$, and $s > 1$,

$$N^{1/(3s)}\varepsilon^{-2/(3s)} \ge N^{1/(3s)}\varepsilon^{1/3}, \qquad N^{1/3}\varepsilon^{1/3} \ge N^{1/(3s)}\varepsilon^{1/3}.$$

Thus either counterfactual submissions or commit rounds have the common lower bound $\Omega(N^c\varepsilon^b)$ with $c = 1/(3s) > 0$ and $b = 1/3$. This is a coarser summary, not a matching lower bound for the upper rate.

H Auxiliary results

This appendix collects the classical finite facts invoked by the construction and the upper-bound proof, followed by the supplementary row-selector invariance result (Proposition H.5). The elementary finite facts are deterministic algebraic or graph statements and introduce no additional model assumptions.

H.1 Finite Hellinger-affinity calculus

The textbook cited below uses $H^2(p,q) = 2h^2(p,q)$ (author-hosted PDF). We include finite proofs for completeness.

Lemma H.1 (Product affinity and TV comparison; Polyanskiy & Wu, 2025, Eqs. (7.22) and (7.26))**.** *Let J be finite. For every $j \in J$, let p_j, q_j be probability masses on a finite set $\mathcal{X}_j$. Then*

$$\mathrm{Aff}\left(\bigotimes_{j\in J} p_j, \bigotimes_{j\in J} q_j\right) = \prod_{j\in J} \mathrm{Aff}(p_j, q_j), \tag{212}$$

and consequently

$$h^2\left(\bigotimes_{j\in J} p_j, \bigotimes_{j\in J} q_j\right) \le \sum_{j\in J} h^2(p_j, q_j). \tag{213}$$

For any two probability masses p, q on the same finite set,

$$h^2(p,q) \le d_{\mathrm{TV}}(p,q) \le \sqrt{2h^2(p,q)}. \tag{214}$$

Proof. 1. Products. For $f_j(x) := \sqrt{p_j(x)q_j(x)}$, distributivity gives $\mathrm{Aff}(\bigotimes_j p_j, \bigotimes_j q_j) = \sum_{(x_j)_j \in \prod_j \mathcal{X}_j} \prod_j f_j(x_j) = \prod_j (\sum_{x \in \mathcal{X}_j} f_j(x)) = \prod_j \mathrm{Aff}(p_j, q_j)$. Put $u_j := h^2(p_j, q_j) \in [0,1]$, enumerate J, and telescope: $1 - \prod_j (1-u_j) = \sum_j [\prod_{k<j}(1-u_k) - \prod_{k \le j}(1-u_k)] = \sum_j u_j \prod_{k<j}(1-u_k) \le \sum_j u_j$. Since $h^2 = 1 - \mathrm{Aff}$, this and equation 212 give equation 213.

2. TV comparison. The identity $\min\{a,b\} = (a+b-|a-b|)/2$ gives $\sum_x \min\{p(x), q(x)\} = (2 - \sum_x |p(x) - q(x)|)/2 = 1 - d_{\mathrm{TV}}(p,q)$. Since $\min\{p(x), q(x)\} \le \sqrt{p(x)q(x)}$, subtracting the summed inequality from one gives $h^2(p,q) \le d_{\mathrm{TV}}(p,q)$. For the other direction, Cauchy–Schwarz yields

$$\begin{aligned} 2d_{\mathrm{TV}}(p,q) &= \sum_x |\sqrt{p(x)} - \sqrt{q(x)}|(\sqrt{p(x)} + \sqrt{q(x)}) \\ &\le \left(\sum_x (\sqrt{p(x)} - \sqrt{q(x)})^2\right)^{1/2} \left(\sum_x (\sqrt{p(x)} + \sqrt{q(x)})^2\right)^{1/2} \\ &\le 2\sqrt{2h^2(p,q)}. \end{aligned}$$

Here the first squared sum is $\sum_x (\sqrt{p(x)} - \sqrt{q(x)})^2 = 2 - 2\,\mathrm{Aff}(p,q) = 2h^2(p,q)$, and the second is $\sum_x (\sqrt{p(x)} + \sqrt{q(x)})^2 = \sum_x p(x) + \sum_x q(x) + 2\sum_x \sqrt{p(x)q(x)} = 2 + 2\,\mathrm{Aff}(p,q) \le 4$. The last inequality uses Cauchy–Schwarz: $\mathrm{Aff}(p,q) \le \sqrt{\sum_x p(x) \sum_x q(x)} = 1$. Dividing by two proves equation 214. □

Lemma H.2 (KL-to-Hellinger comparison; Polyanskiy & Wu, 2025, Eq. (7.33))**.** *For strictly positive probability masses p, q on a finite set,*

$$h^2(p,q) \le 1 - \exp\{-D_{\mathrm{KL}}(p\|q)/2\} \le \frac{1}{2} D_{\mathrm{KL}}(p\|q)\,. \tag{215}$$

The same inequalities hold with the KL arguments reversed.

Proof. This is the order-$1/2$ Rényi/KL comparison in our Hellinger normalization (see also van Erven & Harremoës, 2014). Jensen's inequality for the convex exponential gives $\mathrm{Aff}(p,q) = \mathbb{E}_{X\sim p} \exp\{-\frac{1}{2}\log\frac{p(X)}{q(X)}\} \ge \exp\{-D_{\mathrm{KL}}(p\|q)/2\}$. Subtract from one and use $1 - e^{-x} \le x$ for $x \ge 0$. The reversed statement follows by exchanging p, q and using the symmetry of Hellinger distance. □

Remark H.3 (No upper KL bound from Hellinger alone). Strict positivity without a uniform lower bound on the masses does not give a converse upper bound on KL. For $0 < u < 1/4$, consider

$$p_u = (1-u, u), \qquad q_u = (1 - ue^{-1/u^2}, ue^{-1/u^2}).$$

Both laws are strictly positive, and

$$h^2(p_u, q_u) \le d_{\mathrm{TV}}(p_u, q_u) = u(1 - e^{-1/u^2}) \le u \longrightarrow 0.$$

Nevertheless,

$$D_{\mathrm{KL}}(p_u\|q_u) = \frac{1}{u} + (1-u)\log\frac{1-u}{1-ue^{-1/u^2}} \ge \frac{1}{u} - u \longrightarrow \infty,$$

For the last inequality, $1 - ue^{-1/u^2} \le 1$ implies $\log((1-u)/(1-ue^{-1/u^2})) \ge \log(1-u)$, while

$$-\log(1-u) = \int_0^u \frac{dt}{1-t} \le \frac{u}{1-u} \quad \Longrightarrow \quad (1-u)\log(1-u) \ge -u.$$

Swapping the two laws proves the corresponding failure for reverse KL.

H.2 Elementary forest facts

Lemma H.4 (Forest degree sum and tree centroid)**.** *Let $H = (W, E_H)$ be a finite nonempty forest with $c(H)$ connected components. Then*

$$|E_H| = |W| - c(H), \qquad \sum_{v \in W} d_H(v) = 2|E_H| < 2|W|. \tag{216}$$

Moreover, every finite nonempty tree T has a vertex v such that every component of $T \setminus \{v\}$ has at most $|V(T)|/2$ vertices.

Proof. Each tree component with n_ℓ vertices has $n_\ell - 1$ edges. Summing over the components gives the first identity; the degree-sum identity then gives the remaining claims in equation 216.

For the centroid claim, put $n := |V(T)|$ and choose v minimizing the largest component size after deleting v. If a component S of $T \setminus \{v\}$ had more than $|V(T)|/2$ vertices, let w be the neighbor of v in S. After deleting w, the component containing v has $|V(T)| - |S| < |V(T)|/2$ vertices, while every other component is a proper subset of S and hence has at most $|S| - 1$ vertices. Thus the largest component after deleting w has size at most

$$\max\{n - |S|,\ |S| - 1\} < |S|,$$

whereas the largest component after deleting v has size $|S|$ (all other components together have $n - 1 - |S| < |S|$ vertices). This contradicts the choice of v, proving the centroid claim. □

H.3 Public row-selector invariance

Intuition. The structural proof uses two selector properties: exact candidate rows of size at most d are preserved, and every returned row has size at most d. Consequently the same successful-path and all-path arguments apply to every public selector with this contract. The statistical comparisons are then constructed for that selector's own decision rule.

Proposition H.5 (Invariance under public row selection)**.** *Replace the standard top-vote selector by any rule in Definition D.3, keeping all other screen parameters, fresh-color rules, and decision-rule caps unchanged. Under the finite hypotheses of Theorem E.1, $\mathcal{A}^{\mathrm{rnd}}_{\mathrm{pack}}[\mathsf{S}]$ has the same low-degree screen contract, successful-path contraction and stopping guarantees, all-path resource bounds, and expected-TV bound $\varepsilon^{\mathrm{H}} + \delta^{\mathrm{fail}}$. Under the additional oracle hypotheses of Theorem F.3, its expected forward KL is at most $N\kappa + N \log V\, \delta^{\mathrm{fail}}$. Consequently both main-text benchmark guarantees retain their original calibrations and rates. This statement compares guarantees, not the realized paths or errors of two selectors.*

Proof. Recall the selector contract of Definition D.3:

$$\mathsf{S}(j, C, d; \mathcal{T}) \subseteq C, \qquad |\mathsf{S}(j, C, d; \mathcal{T})| \le d,$$
$$|C| \le d \quad \Longrightarrow \quad \mathsf{S}(j, C, d; \mathcal{T}) = C.$$

The selector is a deterministic function of the available public transcript, makes no submission or commit, and does not read future randomness. The dependencies of the proof are as follows.

Property retained	Argument using it	Guarantee retained
Identity on nonoverflowing rows	Lemma E.6	Low-degree exactness
Row-size cap and low-degree exactness	Lemmas E.7 and E.8	Contraction, stopping, resource bounds
Fresh colors conditional on the past	Lemma E.4	Failure budget
This selector's own safe rule	Lemmas E.9 to E.11	Expected TV bound
This selector's own joint reference	Lemma F.2 and Theorem F.3	Expected forward-KL bound

1. Preserve the low-degree screen contract. On a successful screen, the proof of Lemma E.6 identifies the candidate row with the true neighborhood before applying the selector in Step 3.3. Its size is at most d, so the nonoverflow identity preserves it. Every other returned row still has size at most d; no claim that it contains only true neighbors is needed.

2. Transfer the structural and resource bounds. These two properties are exactly those used in equation 159 and equation 163. Thus the contraction, terminal exactness, and successful-path round count above apply to the decision rule with the chosen selector. In particular, neither cycle repair nor the round-cap fallback activates on a successful path. The selector adds no submission or commit, and the unchanged caps bound resources on every path, including failed screens.

3. Retain the conditional screen-failure budget. Conditioning on this decision rule's realized past, including previous selector outputs, still leaves the next colors drawn in Step 2 independent and uniform. Lemma E.4 therefore gives the same failure budget without a union bound over selectors or histories.

4. Use the chosen selector's own safe decision rule for TV. For TV, construct the safe decision rule of Lemma E.11 using this same selector. The common-path coupling compares $\mathcal{A}^{\mathrm{rnd}}_{\mathrm{pack}}[\mathsf{S}]$ with its own safe decision rule using the same selector; it does not compare different selectors. Structural safety and the generic adaptive Hellinger bound give ε^{H}, and a failed screen contributes at most δ^{fail}.

5. Use the chosen selector's own joint-reference process for KL. For KL, instead construct the joint-reference process of Appendix F.2 for the decision rule with the chosen selector. Write Fail for the event that a reached screen fails in this selector's joint-reference process. On its reference paths, the costs in equation 196 satisfy

$$\sum_t \sum_{j \in B(\mathfrak{h}_t)} D_{\mathrm{KL}}(\mu_j(\cdot \mid y(\mathfrak{h}_t)) \| q_j(\cdot \mid y(\mathfrak{h}_t))) \le \kappa \sum_t |B(\mathfrak{h}_t)| = N\kappa,$$

$$\sum_t \mathrm{TC}(p(\cdot \mid \mathfrak{h}_t)) \le N \log V \, \mathbf{1}_{\mathsf{Fail}}.$$

The second line is zero on a successful reference path by the structural argument; on any path, the sum is at most $N \log V$. Conditioning before each fresh color block in this same reference process gives equation 200. Therefore

$$\begin{aligned} \mathbb{E}_W D_{\mathrm{KL}}\left(P \middle\| \widehat{P}^q_W\right) &\le N\kappa + N \log V \, \mathbb{E}_W \mathbb{P}^{\mathrm{joint}}_W(\mathsf{Fail}) \\ &\le N\kappa + N \log V \, \delta^{\mathrm{fail}}. \end{aligned}$$

All output and reference laws here belong to the chosen selector. No comparison with the standard top-vote selector, or conversion from the TV coupling, is used. □

Remark (Interpreting selector invariance). The resource guarantees do not bound a selector's local arithmetic or memory costs.

I Existing guarantees in the units of this paper

We separate two tasks: accounting for oracle submissions, and evaluating the dependence quantities in the representative schedule bounds of Table 1 and Section 5. The latter uses one explicit path, evaluated for TC, DTC, and effective TC. The comparison substitutes into published sufficient upper bounds; it is not a lower bound on another sampler or a tight worst-case complexity over forests. Appendix J records the source-specific oracle conditions and output objectives.

I.1 Submissions and parallel depth

One submitted masked state returns all requested singleton rows; repeated submissions and different inputs in a parallel batch are charged separately. A query for one conditional row can therefore be implemented by one such submission, so a published row-query upper bound is also a full-output submission upper bound. This conversion need not be tight. A product-unmasking schedule with K updates needs at most K submissions and depth K. Our completed feasible run uses $Q_{\mathrm{cf}} + R + 1$

submissions, including preprocessing. All these counts exclude local arithmetic and vocabulary readout costs.

Parallel verification does not require sequential prefix queries. Write μ for the target law used by the external verification sampler, h for its accepted history, and $z = (z_1, \dots, z_b)$ for an ordered candidate block, with $z_{<j} = (z_1, \dots, z_{j-1})$. Once z and h are available, every input in the chain-rule product

$$\mu(z \mid h) = \prod_j \mu(z_j \mid h, z_{<j})$$

is known and the conditional rows can be queried in parallel. This is consistent with the sublinear-depth Anari results summarized in the table; their exact resource statements are recorded only in Appendix J.1.

I.2 One path with linear TC and DTC

All entropies and mutual informations below are under $X \sim P$. Using the finite entropy $H(p) = -\sum_x p(x) \log p(x)$, write $H(A) := H(\mathcal{L}_P(A))$ and, for finite random variables A, B, C,

$$H(A \mid C) := \sum_c P(C = c) H(P_{A|C=c}),$$
$$I(A; B \mid C) := \sum_c P(C = c) D_{\mathrm{KL}}\big(P_{AB|C=c} \big\| P_{A|C=c} \otimes P_{B|C=c}\big)\,.$$

The sums run over positive-probability values of C; omitting C gives unconditional mutual information. Here $P_{AB|C=c}$ denotes the conditional joint law and $P_{A|C=c}, P_{B|C=c}$ its marginals. With $X_{-i} := (X_j)_{j \neq i}$, write

$$\mathrm{TC}(P) := \sum_i H(X_i) - H(X), \qquad \mathrm{DTC}(P) := H(X) - \sum_i H(X_i \mid X_{-i}).$$

For every forest law, choose a rooted ordering in which each parent precedes its children. In the factorization equation 12, a nonroot vertex j, conditional on earlier vertices, has law $K_{\mathrm{pa}(j)\to j}(\cdot \mid X_{\mathrm{pa}(j)})$: none of its descendants has appeared yet, and its parent separates it from the other earlier vertices. Roots belong to independent components and retain their marginal laws. The entropy chain rule therefore gives

$$H(X) = \sum_{r \in \mathrm{Roots}(F)} H(X_r) + \sum_{j \notin \mathrm{Roots}(F)} H(X_j \mid X_{\mathrm{pa}(j)}).$$

Subtracting from $\sum_i H(X_i)$ cancels the root entropies and yields

$$\begin{aligned} \mathrm{TC}(P) &= \sum_{j \notin \mathrm{Roots}(F)} [H(X_j) - H(X_j \mid X_{\mathrm{pa}(j)})] \\ &= \sum_{j \notin \mathrm{Roots}(F)} I(X_j; X_{\mathrm{pa}(j)}). \end{aligned} \tag{217}$$

A forest need not have large TC: the matching lower-bound witness becomes weakly coupled as accuracy changes. The following different witness has constant dependence per edge while retaining the response and tail assumptions. All background tokens share one type, making rare-token exchanges invisible to the conditional rows.

Intuition. The vocabulary has only two dependence-relevant types. The type chain has a fixed nonzero dependence between adjacent positions, while token variation within a type contributes only independent emissions. Consequently each edge contributes a positive constant to TC, and each interior position contributes a positive constant to DTC:

$$\mathrm{TC}(P) \simeq N, \qquad \mathrm{DTC}(P) \simeq N.$$

The background vocabulary can grow without changing these dependence scales. The proof checks both class membership and the entropy identities.

Proposition I.1 (A shared-class path with linear TC and DTC)**.** *Fix $s > 1$ and put $p = 2^{-s}$. For every $N, V \geq 3$, let $\mathcal{V} = \{a_1, a_2\} \sqcup \mathcal{B}$, $|\mathcal{B}| = V - 2$, with*

$$\pi(a_1) = 1 - p, \qquad \pi(a_2) = p/2, \qquad \pi(b) = \frac{p}{2(V-2)} \quad (b \in \mathcal{B}).$$

Define $\tau(a_1) = \mathsf{A}$ and $\tau(a) = \mathsf{B}$ for $a \neq a_1$. The type masses are $P_{\mathsf{A}} = 1 - p$, $P_{\mathsf{B}} = p$. In the type order (A, B), set

$$J = \begin{pmatrix} 1 - 3p/2 & p/2 \\ p/2 & p/2 \end{pmatrix}, \qquad g(t, t') = \frac{J(t, t')}{P_t P_{t'}}, \qquad K(y \mid x) = \pi(y) g(\tau(x), \tau(y)). \tag{218}$$

Then $P(x) = \pi(x_1) \prod_{j=2}^{N} K(x_j \mid x_{j-1})$ is a strictly positive path law with all marginals π. There is a constant $\omega_\star(p) > 0$, independent of N, V, such that this law satisfies (RT; Equation (18))–(UEN; Equation (19)) with $L = \alpha = 1$ and every $\omega \leq \omega_\star(p)$, and (RF; Equation (21)) with $C = 1$. Its exact oracle satisfies both oracle accuracy conditions. Moreover,

$$\mathrm{TC}(P) = \Theta(N), \qquad \mathrm{DTC}(P) = \Theta(N), \tag{219}$$

where the constants depend only on p. Consequently this path belongs to each main-text benchmark class for all sufficiently large N along that class's vocabulary and accuracy regime.

Proof. 1. Kernel normalization and stationary marginal. For a type t, the definition of its mass is $P_t = \sum_{x:\tau(x)=t} \pi(x)$. Here $0 < p < 1/2$, and the four entries of J are positive. Its row sums are

$$(1 - 3p/2) + p/2 = 1 - p = P_{\mathsf{A}}, \qquad p/2 + p/2 = p = P_{\mathsf{B}}.$$

Since J is symmetric, its column sums are the same. For a fixed token x of type $t = \tau(x)$, group the sum over y by type:

$$\begin{aligned} \sum_y K(y \mid x) &= \sum_{t' \in \{\mathsf{A},\mathsf{B}\}} \sum_{y:\tau(y)=t'} \pi(y) \frac{J(t, t')}{P_t P_{t'}} \\ &= \sum_{t' \in \{\mathsf{A},\mathsf{B}\}} P_{t'} \frac{J(t, t')}{P_t P_{t'}} = \frac{1}{P_t} \sum_{t'} J(t, t') = 1. \end{aligned}$$

Thus K is a probability kernel. For a fixed y of type t', the column-sum identity similarly yields

$$\begin{aligned} \sum_x \pi(x) K(y \mid x) &= \pi(y) \sum_{t \in \{\mathsf{A},\mathsf{B}\}} \sum_{x:\tau(x)=t} \pi(x) \frac{J(t, t')}{P_t P_{t'}} \\ &= \pi(y) \frac{1}{P_{t'}} \sum_t J(t, t') = \pi(y). \end{aligned}$$

Starting the path from π therefore gives marginal π at every position, by induction along the path. Also,

$$\pi(x) K(y \mid x) = \pi(x) \pi(y) \frac{J(\tau(x), \tau(y))}{P_{\tau(x)} P_{\tau(y)}} = \pi(y) K(x \mid y),$$

which verifies endpoint compatibility. Positivity of π and K gives strict positivity of the path law.

2. Boundary-conditioned rows and the edge signal. Substituting the kernel formula into the path law gives

$$P(x) = \left(\prod_{j=1}^{N} \pi(x_j) \right) \left(\prod_{j=2}^{N} g(\tau(x_{j-1}), \tau(x_j)) \right).$$

For fixed values outside readout j, all factors not incident to j cancel. Write $\mathcal{N}_F(j) = \{j - 1, j + 1\} \cap [N]$ for its path neighbors. Its normalized conditional row is

$$P(X_j = y \mid X_{-j} = x_{-j}) = \frac{\pi(y) \prod_{k \in \mathcal{N}_F(j)} g(\tau(y), \tau(x_k))}{\sum_z \pi(z) \prod_{k \in \mathcal{N}_F(j)} g(\tau(z), \tau(x_k))}.$$

Within type t, every factor g is constant, so the conditional token law given the type is always $\pi(y)/P_t$ for $\tau(y) = t$. Summing the numerator separately over the two types shows that the conditional odds of type A against type B are

$$\begin{aligned}
&\frac{P(\tau(X_j) = \mathsf{A} \mid X_{-j} = x_{-j})}{P(\tau(X_j) = \mathsf{B} \mid X_{-j} = x_{-j})} \\
&\qquad = \frac{P_{\mathsf{A}} \prod_{k \in \mathcal{N}_F(j)} g(\mathsf{A}, \tau(x_k))}{P_{\mathsf{B}} \prod_{k \in \mathcal{N}_F(j)} g(\mathsf{B}, \tau(x_k))} = \rho \prod_{k \in \mathcal{N}_F(j)} r_{\tau(x_k)}, \\
&\rho := \frac{P_{\mathsf{A}}}{P_{\mathsf{B}}} = \frac{1-p}{p}, \qquad r_t := \frac{g(\mathsf{A}, t)}{g(\mathsf{B}, t)} \quad (t \in \{\mathsf{A}, \mathsf{B}\}).
\end{aligned}$$

Thus ρ is the marginal type-odds ratio, and r_t is the multiplicative contribution of a neighbor of type t. Their values follow from the four entries

$$g(\mathsf{A}, \mathsf{A}) = \frac{1 - 3p/2}{(1-p)^2}, \quad g(\mathsf{A}, \mathsf{B}) = g(\mathsf{B}, \mathsf{A}) = \frac{1}{2(1-p)}, \quad g(\mathsf{B}, \mathsf{B}) = \frac{1}{2p} :$$

$$\begin{aligned}
r_{\mathsf{A}} &= \frac{g(\mathsf{A}, \mathsf{A})}{g(\mathsf{B}, \mathsf{A})} = \frac{(1 - 3p/2)/(1-p)^2}{1/[2(1-p)]} = \frac{2 - 3p}{1-p}, \\
r_{\mathsf{B}} &= \frac{g(\mathsf{A}, \mathsf{B})}{g(\mathsf{B}, \mathsf{B})} = \frac{1/[2(1-p)]}{1/(2p)} = \frac{p}{1-p}.
\end{aligned}$$

In particular, $r_{\mathsf{A}} - r_{\mathsf{B}} = 2(1-2p)/(1-p) > 0$.

Fix a directed edge $i \to j$ and a complete boundary $z \in \mathcal{V}^{[N] \setminus \{i,j\}}$. The two rows to compare are precisely

$$\begin{aligned}
\mu_{i \to j}(\cdot \mid a_1, z) &= \mathcal{L}_P(X_j \mid X_i = a_1, X_{[N] \setminus \{i,j\}} = z), \\
\mu_{i \to j}(\cdot \mid a_2, z) &= \mathcal{L}_P(X_j \mid X_i = a_2, X_{[N] \setminus \{i,j\}} = z).
\end{aligned}$$

Here $\tau(a_1) = \mathsf{A}$ and $\tau(a_2) = \mathsf{B}$. Put $f(o) = o/(1+o)$, the type-A probability at odds o, and define the contribution of the other neighbors by

$$q := \prod_{k \in \mathcal{N}_F(j) \setminus \{i\}} r_{\tau(z_k)} \in \{1, r_{\mathsf{A}}, r_{\mathsf{B}}\}.$$

The empty product is one; a path readout has at most one other neighbor. The type-A weights of these two rows are, respectively,

$$\begin{aligned}
u &:= P(\tau(X_j) = \mathsf{A} \mid X_i = a_1, X_{[N] \setminus \{i,j\}} = z) = f(\rho r_{\mathsf{A}} q), \\
u' &:= P(\tau(X_j) = \mathsf{A} \mid X_i = a_2, X_{[N] \setminus \{i,j\}} = z) = f(\rho r_{\mathsf{B}} q).
\end{aligned}$$

Within each type, both rows have the same token law $\pi(y)/P_t$. Expanding TV over the disjoint type supports therefore gives

$$\begin{aligned}
&d_{\mathrm{TV}}\big(\mu_{i \to j}(\cdot \mid a_1, z), \mu_{i \to j}(\cdot \mid a_2, z)\big) \\
&\quad = \frac{1}{2} \left(\sum_{\tau(y) = \mathsf{A}} |u - u'| \frac{\pi(y)}{P_{\mathsf{A}}} + \sum_{\tau(y) = \mathsf{B}} |(1-u) - (1-u')| \frac{\pi(y)}{P_{\mathsf{B}}} \right) \\
&\quad = \frac{1}{2} \big(|u - u'| + |(1-u) - (1-u')| \big) = u - u' =: D(q), \\
&D(q) = f(\rho r_{\mathsf{A}} q) - f(\rho r_{\mathsf{B}} q) = \frac{\rho q (r_{\mathsf{A}} - r_{\mathsf{B}})}{(1 + \rho r_{\mathsf{A}} q)(1 + \rho r_{\mathsf{B}} q)} > 0.
\end{aligned}$$

The two sums of $\pi(y)/P_t$ equal one by the definition of P_t. The conditional row depends on the source token only through its type, so a_1, a_2 attain the full directed response:

$$\Omega_{i \to j}(z) = \max_{c, c' \in \mathcal{V}} d_{\mathrm{TV}}\big(\mu_{i \to j}(\cdot \mid c, z), \mu_{i \to j}(\cdot \mid c', z)\big) = D(q).$$

Define $\omega_\star(p) := \min\{D(1), D(r_\mathsf{A}), D(r_\mathsf{B})\} > 0$. This minimum depends only on p, not on N, V or the boundary. The same calculation applies to either orientation of every path edge, so the required condition equation 19 is

$$\min\{\Omega_{i\to j}(z), \Omega_{j\to i}(z)\} \geq \omega_\star(p) \geq \omega \qquad \text{(UEN; Equation (19))}.$$

3. Tail response and the rank–frequency envelope. Recall that $\pi_i = \pi$ at every position, so the marginal tail in equation 15 is $\mathcal{T}_i(t) = \{c \in \mathcal{V} : \pi(c) \leq t\}$. For a nonedge $\{i, j\} \notin E$, the conditional row does not involve the source value, and hence for every $c, c' \in \mathcal{V}$,

$$\mu_{i\to j}(\cdot \mid c, z) = \mu_{i\to j}(\cdot \mid c', z), \qquad d_{\mathrm{TV}}\big(\mu_{i\to j}(\cdot \mid c, z), \mu_{i\to j}(\cdot \mid c', z)\big) = 0.$$

For an edge, the two threshold ranges are as follows.

- If $t < 1 - p = \pi(a_1)$, then $\mathcal{T}_i(t) \subseteq \{a_2\} \cup \mathcal{B} = \tau^{-1}(\mathsf{B})$. Thus all permitted source tokens have the same type and give the same readout row; their maximum TV distance is zero.
- If $t \geq 1 - p$, then $\mathcal{T}_i(t) = \mathcal{V}$, since $1 - p$ is the largest marginal mass. The full directed response is $D(q)$ from step 2, so it remains to prove $D(q) \leq 1 - p$.

Here is the needed bound with its maximization made explicit. Set $x = \sqrt{r_\mathsf{A}/r_\mathsf{B}} > 1$ and $\zeta = \rho r_\mathsf{B} q > 0$. Then

$$D(q) = \frac{x^2\zeta}{1 + x^2\zeta} - \frac{\zeta}{1 + \zeta} = \frac{(x^2 - 1)\zeta}{(1 + x^2\zeta)(1 + \zeta)}.$$

The denominator is $1 + (1 + x^2)\zeta + x^2\zeta^2$. The quotient rule gives

$$\frac{d}{d\zeta}\frac{(x^2 - 1)\zeta}{(1 + x^2\zeta)(1 + \zeta)} = \frac{(x^2 - 1)(1 - x^2\zeta^2)}{(1 + x^2\zeta)^2(1 + \zeta)^2}.$$

The derivative is positive for $\zeta < 1/x$ and negative for $\zeta > 1/x$. Substituting $\zeta = 1/x$ therefore gives

$$\sup_{q>0} D(q) = \frac{(x^2 - 1)/x}{(1 + x)(1 + 1/x)} = \frac{x - 1}{x + 1}.$$

All quantities are positive, and

$$\frac{x - 1}{x + 1} \leq 1 - p \quad \Longleftrightarrow \quad px \leq 2 - p \quad \Longleftrightarrow \quad p^2x^2 \leq (2 - p)^2.$$

Finally,

$$p^2x^2 = p(2 - 3p), \qquad (2 - p)^2 - p(2 - 3p) = 4 - 6p + 4p^2 > 0 \quad (0 < p < 1/2).$$

Combining the nonedge and edge cases gives, for every distinct i, j, complete boundary z, and $t \in [0, 1]$,

$$\max_{c,c' \in \mathcal{T}_i(t)} d_{\mathrm{TV}}\big(\mu_{i\to j}(\cdot \mid c, z), \mu_{i\to j}(\cdot \mid c', z)\big)$$
$$\leq \begin{cases} 0, & 0 \leq t < 1 - p, \\ 1 - p, & 1 - p \leq t \leq 1 \end{cases} \leq t = Lt^\alpha \qquad \text{(RT; Equation (18))},$$

where $L = \alpha = 1$, and the maximum is zero for a tail with at most one token, as in equation 18.

For (RF; Equation (21)), a_1 has mass $1 - p > 1/2$ and is the largest atom. The mass of a_2 is $p/2 = 2^{-s-1} \leq 2^{-s}$. Each background mass obeys, since $V \geq 3$,

$$\frac{p}{2(V - 2)} \leq \frac{1}{4(V - 2)} \leq \frac{1}{V}; \qquad V \leq 4(V - 2) \iff 3V \geq 8.$$

The background masses are no larger than $p/2$. Therefore, with the ranked tokens of equation 20, every position i and rank $k \in [V]$ satisfy

$$\pi_i(a_{i,k}) \leq \begin{cases} 1, & k = 1, \\ 2^{-s}, & k = 2, \\ V^{-1}, & 3 \leq k \leq V \end{cases}$$
$$\leq k^{-s} + V^{-1} = C(k^{-s} + V^{-1}) \qquad \text{(RF; Equation (21))},$$

with $C = 1$. If $V = 3$, the possible tie at rank two has mass $p/2$ on either token, so the same bounds apply under the public tie-breaking order. This is the finite envelope equation 21; the asymptotic part of (RF; Assumption A.5) is supplied by the prescribed sequence $V = N^{\nu+o(1)}$. The exact oracle has zero Hellinger and KL error.

4. Total correlation. Let $T_j = \tau(X_j)$. Its transition probability follows by grouping K:

$$P(T_{j+1} = t' \mid T_j = t) = \sum_{\tau(y)=t'} \pi(y) g(t, t') = \frac{J(t, t')}{P_t}.$$

Since its marginal is P_t, the adjacent type joint law is J. The adjacent token joint law is

$$P(X_j = x, X_{j+1} = y) = J(t, t') \frac{\pi(x)}{P_t} \frac{\pi(y)}{P_{t'}}, \qquad t = \tau(x), \quad t' = \tau(y).$$

Its likelihood ratio relative to $\pi(x)\pi(y)$ is $J(t, t')/(P_t P_{t'})$, which depends only on the types. Grouping the mutual-information sum by type therefore gives

$$I(X_j; X_{j+1}) = \sum_{t,t'} J(t, t') \log \frac{J(t, t')}{P_t P_{t'}} = I(T_j; T_{j+1}) =: c.$$

To see why this constant is positive, the displayed sum is $D_{\mathrm{KL}}(J \| (P_t P_{t'})_{t,t'})$. KL is zero exactly when its two laws agree. Such equality would make J a product table and hence rank one, but

$$\det J = (1 - 3p/2)(p/2) - (p/2)^2 = \frac{p(1 - 2p)}{2} > 0.$$

It depends only on p, so equation 217 gives $\mathrm{TC}(P) = (N - 1)c = \Theta(N)$.

5. Dual total correlation. The chain rule and conditional Markov property give, respectively,

$$H(X) = H(X_1) + \sum_{i=2}^{N} H(X_i \mid X_{i-1})$$

and

$$\sum_i H(X_i \mid X_{-i}) = H(X_1 \mid X_2) + \sum_{i=2}^{N-1} H(X_i \mid X_{i-1}, X_{i+1}) + H(X_N \mid X_{N-1}).$$

Subtracting cancels the final-coordinate term and yields

$$\begin{aligned} \mathrm{DTC}(P) &= H(X_1) - H(X_1 \mid X_2) \\ &\quad + \sum_{i=2}^{N-1} \big[H(X_i \mid X_{i-1}) - H(X_i \mid X_{i-1}, X_{i+1}) \big] \\ &= I(X_1; X_2) + \sum_{i=2}^{N-1} I(X_i; X_{i+1} \mid X_{i-1}). \end{aligned} \tag{220}$$

To evaluate an interior term, fix the previous type t. The conditional joint table of (T_i, T_{i+1}) is

$$P(T_i = u, T_{i+1} = v \mid T_{i-1} = t) = \frac{J(t, u)}{P_t} \frac{J(u, v)}{P_u}.$$

This is obtained from J by multiplying row u by the positive factor $J(t, u)/(P_t P_u)$. Its determinant is consequently

$$\frac{J(t, \mathsf{A}) J(t, \mathsf{B})}{P_t^2 P_{\mathsf{A}} P_{\mathsf{B}}} \det J > 0.$$

The two conditional type variables are therefore dependent for either t, and their conditional mutual information is a positive constant depending only on p.

It remains to relate this type calculation to the token term. Conditioning on $X_{i-1} = x$ affects the future law only through $\tau(x) = t$. For this calculation write $\Pr_t(\cdot) = \Pr(\cdot \mid T_{i-1} = t)$. For tokens y, z of types u, v, the emission factors cancel as follows:

$$\begin{aligned}
&\frac{\Pr_t(X_i = y, X_{i+1} = z)}{\Pr_t(X_i = y)\Pr_t(X_{i+1} = z)} \\
&\quad = \frac{\Pr_t(T_i = u, T_{i+1} = v)\,[\pi(y)/P_u]\,[\pi(z)/P_v]}{\Pr_t(T_i = u)\,[\pi(y)/P_u]\;\Pr_t(T_{i+1} = v)\,[\pi(z)/P_v]} \\
&\quad = \frac{\Pr_t(T_i = u, T_{i+1} = v)}{\Pr_t(T_i = u)\Pr_t(T_{i+1} = v)}.
\end{aligned}$$

Grouping the token sum by (u, v) therefore gives

$$\begin{aligned}
&I(X_i; X_{i+1} \mid T_{i-1} = t) \\
&= \sum_{u,v} \Pr_t(T_i = u, T_{i+1} = v) \log \frac{\Pr_t(T_i = u, T_{i+1} = v)}{\Pr_t(T_i = u)\Pr_t(T_{i+1} = v)} \\
&\qquad \times \underbrace{\left(\sum_{y:\tau(y)=u} \frac{\pi(y)}{P_u}\right)}_{=1} \underbrace{\left(\sum_{z:\tau(z)=v} \frac{\pi(z)}{P_v}\right)}_{=1} \\
&= I(T_i; T_{i+1} \mid T_{i-1} = t).
\end{aligned}$$

The two sums equal one because $P_u = \sum_{\tau(y)=u} \pi(y)$ and similarly for P_v. The conditional future law is the same for every previous token of type t, so averaging over the previous token, or equivalently over its type with weights P_t, yields

$$I(X_i; X_{i+1} \mid X_{i-1}) = I(T_i; T_{i+1} \mid T_{i-1}).$$

Every interior term is the same positive constant by stationarity and is at most $\log 2$, since the types are binary. Together with $I(X_1; X_2) = c > 0$, the identity above proves $\mathrm{DTC}(P) = \Theta(N)$.

All constants used for the signal and dependence depend only on $p = 2^{-s}$. Each main benchmark has $\omega \to 0$, so eventually $\omega \le \omega_\star(p)$; its prescribed vocabulary sequence supplies the remaining asymptotic part of (RF; Assumption A.5). This proves membership in those benchmark classes. □

What the substitution establishes. For this path, the TC/DTC sufficient counts listed in Table 1 become $\widetilde{O}(1 + N/\varepsilon)$ at forward-KL tolerance ε. These displayed guarantees alone do not yield a sublinear count at fixed accuracy. They do not establish that those samplers require linear work: exact singleton updates already give an N-call cap. Our case-(ii) benchmark also applies, with the balanced count in equation 3 on its sublinear range. The calculation concerns one class member, not every forest or every dependence-adaptive schedule.

Remark (Other dependence measures). No corresponding quantitative comparison is made here for entropy-based bounds or refined unmasking growth complexity.

I.3 Effective total correlation on the same path

To compare with Dmitriev et al. (2026a), let $Y(t)$ be an independently masked copy of $X \sim P$: each $Y_i(t)$ equals X_i with probability e^{-t} and equals MASK otherwise. Write $Y_{-(i,j)}(t)$ for all coordinates except i, j. Their effective total correlation, written $\mathcal{D}_{\mathrm{eff}}$ here, is

$$\mathcal{D}_{\mathrm{eff}}(P) := \int_0^\infty \min\{1, t\}\mathcal{I}_P(t)\,dt, \qquad \mathcal{I}_P(t) := \sum_{i \neq j} I\big(Y_i(t); Y_j(t) \mid Y_{-(i,j)}(t)\big). \tag{221}$$

These are equation (16) of arXiv:2602.15008v2. Its Lemma 16 gives $\mathcal{D}_{\mathrm{eff}}(P) \le \min\{\mathrm{TC}(P), \mathrm{DTC}(P)\}$. Large TC and DTC alone do not imply large effective TC: that paper's Proposition 5 supplies a counterexample. We therefore evaluate $\mathcal{D}_{\mathrm{eff}}$ on our path directly.

Intuition. For an interior edge, reveal its two endpoints and their two outer neighbors. This occurs with probability e^{-4t}. Fixing those neighbors isolates a dependent two-vertex conditional law, regardless of other reveals. Summing over the $N-3$ interior edges gives $\mathcal{I}_P(t) \gtrsim_s Ne^{-4t}$; integrating over a fixed time interval keeps a linear contribution.

Corollary I.2 (Linear effective TC on the comparison path)**.** *For the path law in Proposition I.1, with fixed $s > 1$, $\mathcal{D}_{\text{eff}}(P) = \Theta(N)$ as $N \to \infty$. The constants depend only on s, not on V.*

Proof. 1. Expand over the visible coordinates. Take $N \geq 4$, fix $t > 0$, and put $\lambda = e^{-t}$. For a pair $i \neq j$, the visible set outside the pair is $S \subseteq [N] \setminus \{i, j\}$ with probability $\lambda^{|S|}(1-\lambda)^{N-2-|S|}$. Put $M_k := \mathbf{1}\{Y_k(t) \neq \mathsf{MASK}\}$, so the M_k are independent $\mathrm{Ber}(\lambda)$ variables, independent of X. Fix an exterior mask pattern with visible set S and revealed values $X_S = x_S$. The conditioning event is

$$\mathcal{C} = \{M_k = \mathbf{1}\{k \in S\} \text{ for every } k \notin \{i, j\},\ X_S = x_S\}.$$

For this calculation abbreviate $Y_k(t)$ to Y_k. Since M_i is determined by Y_i and M_j by Y_j, two applications of the conditional chain rule give

$$\begin{aligned} I(Y_i; Y_j \mid \mathcal{C}) &= I(M_i; Y_j \mid \mathcal{C}) + I(Y_i; Y_j \mid M_i, \mathcal{C}) \\ &= I(Y_i; M_j \mid M_i, \mathcal{C}) + I(Y_i; Y_j \mid M_i, M_j, \mathcal{C}) \\ &= \lambda^2 I(X_i; X_j \mid X_S = x_S). \end{aligned}$$

Here $I(M_i; Y_j \mid \mathcal{C}) = 0$ and $I(Y_i; M_j \mid M_i, \mathcal{C}) = 0$ by mask independence. The last conditional mutual information averages over (M_i, M_j). It is zero whenever either endpoint is masked, because that Y value is constant; when both are visible it equals $I(X_i; X_j \mid X_S = x_S)$, with weight λ^2. Averaging over x_S and then over exterior mask patterns gives

$$I\big(Y_i(t); Y_j(t) \mid Y_{-(i,j)}(t)\big) = \lambda^2 \sum_{S \subseteq [N] \setminus \{i,j\}} \lambda^{|S|}(1-\lambda)^{N-2-|S|} I(X_i; X_j \mid X_S). \tag{222}$$

2. Retain a positive dependence across each interior edge. Recall the two types $T_i = \tau(X_i)$ and their transition matrix $\bar{K}(a, b) = J(a, b)/P_a$ from the proof of Proposition I.1. All entries are positive, and $\det \bar{K} = \det J/(P_{\mathsf{A}} P_{\mathsf{B}}) > 0$. For an interior pair $i, i+1$, where $2 \leq i \leq N-2$, fix the outer types $T_{i-1} = a, T_{i+2} = b$. The conditional two-by-two table is

$$\Pr(T_i = u, T_{i+1} = v \mid T_{i-1} = a, T_{i+2} = b) = \frac{\bar{K}(a, u)\bar{K}(u, v)\bar{K}(v, b)}{Z_{a,b}},$$

$$Z_{a,b} := \sum_{u,v} \bar{K}(a, u)\bar{K}(u, v)\bar{K}(v, b) > 0.$$

Here and below type indices range over $\{\mathsf{A}, \mathsf{B}\}$. This table is obtained from $\bar{K}$ by multiplying row u by $\bar{K}(a, u)$, column v by $\bar{K}(v, b)$, and the whole table by $Z_{a,b}^{-1}$. Its determinant is

$$\frac{\bar{K}(a, \mathsf{A})\bar{K}(a, \mathsf{B})\bar{K}(\mathsf{A}, b)\bar{K}(\mathsf{B}, b)}{Z_{a,b}^2} \det \bar{K} > 0.$$

Thus the conditional pair has positive mutual information for each boundary type pair, by the KL and rank-one criterion used above. There are only four such pairs. By stationarity, the positive constant

$$c_{\text{pair}} := \min_{a,b} I(T_2; T_3 \mid T_1 = a, T_4 = b) > 0$$

depends only on $p = 2^{-s}$.

If $\{i-1, i+2\} \subseteq S \subseteq [N] \setminus \{i, i+1\}$, path factorization shows that other observed coordinates affect only factors outside this pair. Conditional on $X_S = x_S$, the type pair therefore has the displayed table with $a = \tau(x_{i-1})$, $b = \tau(x_{i+2})$. Applying data processing to the two type maps at each x_S and then averaging gives

$$\begin{aligned} I(X_i; X_{i+1} \mid X_S) &= \sum_{x_S} P(X_S = x_S) I(X_i; X_{i+1} \mid X_S = x_S) \\ &\geq \sum_{x_S} P(X_S = x_S) I(T_i; T_{i+1} \mid X_S = x_S) \\ &\geq \sum_{x_S} P(X_S = x_S) c_{\text{pair}} = c_{\text{pair}}. \end{aligned}$$

The total mask weight of these sets is

$$\sum_{\substack{S\subseteq[N]\setminus\{i,i+1\}\\ \{i-1,i+2\}\subseteq S}} \lambda^{|S|}(1-\lambda)^{N-2-|S|} = \lambda^2\big(\lambda+(1-\lambda)\big)^{N-4} = \lambda^2.$$

The first λ^2 requires the two outer neighbors to be visible; each remaining coordinate contributes $\lambda + (1-\lambda) = 1$. Substituting into equation 222 yields $I(Y_i(t); Y_{i+1}(t) \mid Y_{-(i,i+1)}(t)) \geq c_{\text{pair}}\lambda^4$.

3. Sum the edges and integrate. There are $N-3$ such edges and two orientations per edge in $\mathcal{I}_P(t)$. Nonnegativity of the other terms gives

$$\mathcal{I}_P(t) \geq 2(N-3)c_{\text{pair}}e^{-4t}.$$

Hence

$$\mathcal{D}_{\text{eff}}(P) \geq 2(N-3)c_{\text{pair}}\int_0^\infty \min\{1,t\}e^{-4t}\,dt = \Omega_s(N).$$

The integral is a positive finite absolute constant. The reverse bound follows from $\mathcal{D}_{\text{eff}}(P) \leq \text{DTC}(P) = O_s(N)$, using the cited Lemma 16 and Proposition I.1. □

J Source locators and conventions for Table 1

This appendix is a verification aid for Table 1, not a separate survey. For each external row, it records the linked version, the theorem or definition supporting the entry, and only the conversion needed to read the table. Numbering below refers to the linked version and may differ from another bibliographic version.

Common conventions. For an external randomized schedule M with conditional output law Q_M, *Mean KL/TV* denotes $\mathbb{E}_M D(P, Q_M)$, not $D(P, \mathbb{E}_M Q_M)$; an unqualified TV guarantee concerns the randomized output law. The marker *exp.* applies to both resource columns, while *same* means that the submission bound also bounds depth. Appendix I.1 gives the conversion from conditional-row queries or unmasking updates to our masked-state submissions. Fixed accuracy and polynomial vocabulary are imposed only when comparing growth in N. Uniform counterfactual-row error and an average under a data–mask or trajectory law are different oracle assumptions; Table 1 does not identify them.

J.1 Distribution-general sampler rows

Anari et al. (2024). arXiv:2408.09442v1, Theorem 2 and Algorithm 4: exact conditional rows and exact output, with $O(N)$ expected row queries and $\widetilde{O}(N^{2/3})$ expected depth for polynomial vocabulary. Appendix I.1 supplies the submission conversion used in the table.

Anari et al. (2026). arXiv:2511.07869v1, Theorem 27 and footnote 3: $O(N \log N)$ expected full-row queries and $O(\sqrt{N \log V}\,(\log N)^3)$ expected depth. Remark 32, equation (25), gives the noisy-row TV entry:

$$e \leq \min\{\tau/N, N^{-3/2}\} \quad \Longrightarrow \quad \text{TV}(P,Q) \leq \tau. \tag{223}$$

The condition is uniform TV accuracy of the normalized rows; the conclusion is TV for the output law. Exact rows give exact output.

J.2 Dependence-based schedule rows

Li & Cai (2025). arXiv:2505.21400v2, Definition 1, equation (7), defines the predictor error averaged over target data, random masks, and the weighted time index. Theorem 1, equation (8), and Corollary 1, equation (9), give for near-balanced blocks

$$\mathbb{E}_M \text{KL}(P\|Q_M) \leq \frac{C}{K}\big(\text{TC}(P) + \text{DTC}(P)\big) + \varepsilon_{\text{train}},$$

where M is the random schedule and K is its number of updates. The table allocates a constant accuracy budget to $\varepsilon_{\text{train}}$ and substitutes the path calculation in Proposition I.1; no uniform counterfactual-row condition is inferred from this average.

Chen et al. (2026). arXiv:2511.04647v2, Definition 1.1 specifies schedule-averaged forward KL, Definition 2.1 defines one partial-assignment query returning all remaining conditional rows, and Theorem 1.9 gives, for a supplied TC or DTC upper bound $\widehat{T}$,

$$\mathbb{E}_M \mathrm{KL}(P \| Q_M) \le \varepsilon, \qquad K \le 2 + (1 + \log N)(1 + \lceil \widehat{T}/\varepsilon \rceil). \tag{224}$$

The table uses exact rows and a constant-factor supplied bound. On the path of Proposition I.1, both TC and DTC are $\Theta(N)$.

Baselines and this paper. Exact singleton sampling uses N submissions and depth N by the chain rule. For marker a, one exact product batch has forward KL $\mathrm{TC}(P)$, which is linear on the path of Proposition I.1; it is only a resource baseline. Our two rows are the balanced bounds in Theorem 3.1 and equation 3: case (i) uses uniform row squared-Hellinger error and reports mean TV, while case (ii) uses uniform row forward KL and reports mean forward KL. Their exact finite statements are Theorems E.1 and F.3.

J.3 Supplementary citation map

The following citations supply context only; they are not used to derive a row of Table 1.

Modeling background.

- *Forest approximation.* Liu et al. (2011) study nonparametric density estimation using forests, allowing the true continuous distribution to lie outside that family (abstract). This motivates forests as an approximation class; our target instead satisfies exact forest factorization.
- *Prediction without exact structure recovery.* Bresler & Karzand (2020) study tree Ising prediction using *small-set TV*, which compares marginals on small subsets (abstract). This differs from the global TV objective of Daskalakis & Pan (2021) and our case (i).
- *Learned conditional distributions.* Heckerman et al. (2000) allow separately learned local conditionals that need not be compatible with a joint law (Section 3, pp. 55–56). Their inference procedure uses ordered pseudo-Gibbs updates; our interface supplies masked-state rows for irreversible commits.

Algorithmic antecedents. The construction in Section 4 connects the following design principles to hidden-forest sampling.

- *Shared evaluations (Steps 2–3).* Coleman & Moré (1983) use graph coloring to reduce function evaluations for sparse Jacobian estimation (see the paper's abstract). Our packed probes share evaluations across source–readout pairs; the separation conditions (R)/(S) in Section 4.2 make their conditional replies reproduce single-source tests.
- *Hashing and majority recovery (Step 3).* Bshouty & Costa (2018), arXiv v1, Section 4.3, Theorem 8, Figure 1, steps 1–3, hash variables, lift identified buckets back to variables, and aggregate by majority. Our bucket tests use conditional-row diameters; we recover low-degree neighborhoods rather than a Boolean function.
- *Peeling (Step 4).* Becker et al. (2010), arXiv v2, Section 3.1, p. 9, reconstruct a forest by decoding leaves from exact degrees and neighbor-ID sums, then removing them and updating these summaries. Here, low-degree reports instead select singleton commits that shrink the high-degree core (Equation (8)), without requiring complete neighborhoods for the committed vertices.
- *Separator layers (Steps 5–6).* Iyer et al. (1988) relate vertex ranking to separator-tree height (see their abstract). In our centroid schedule, earlier layers have higher ranks: their committed vertices separate same-layer vertices into distinct residual components. We use centroid recursion, not their optimal-ranking algorithm.

Additional context. Diffusion foundations and discrete or language formulations include Sohl-Dickstein et al. (2015); Ho et al. (2020); Song et al. (2021); Campbell et al. (2022); Li et al. (2022); He et al. (2023). Parallel-time and complementary complexity perspectives include Shih et al. (2023); Chen et al. (2024); Yao et al. (2026); Feng et al. (2025); Jiang et al. (2026); Zhang et al. (2026); Liang et al. (2026).

Adaptive order, lookahead, confidence, and schedule analyses include Li et al. (2024); Kim et al. (2025); Hayakawa et al. (2026); Lee et al. (2025); Fu et al. (2025); Lavenant & Zanella (2025); Zhao & Cai (2026); Cai & Li (2026); Park et al. (2025). For the specific round statement cited in the main text, the locator for Fu et al. (2025) is arXiv:2511.21103v1, Assumption 3.1 and Theorem 3.2. Dependence-based, remasking, and within-batch modeling perspectives include Dmitriev et al. (2026a;b); Wainwright (2026); Bansal et al. (2026); Hayakawa et al. (2025); Liu et al. (2025); Lezama et al. (2023).

Related oracle, learning, and convergence settings include approximate-density and conditional-sampling access (Golowich et al., 2026; Canonne et al., 2015; 2021; Chen et al., 2021; Blanca et al., 2023), distributed sampling (Feng et al., 2020), learned-denoiser error (Wakasugi & Suzuki, 2025), and discrete-diffusion convergence (Zhang et al., 2025; Ren et al., 2025; Liang et al., 2025). These works use objectives or access models different from the fixed singleton-conditional interface studied here; no quantitative conversion is claimed beyond the four source rows above.

J.4 Reading Table 1

- The external entries are sufficient upper bounds, not lower bounds on their samplers. The two Anari rows retain their distribution-general guarantees. For the TC/DTC rows, Appendix I.2 substitutes one path with $\mathrm{TC}(P) = \mathrm{DTC}(P) = \Theta(N)$; this does not imply that those samplers require linear work.
- Marker b is exactly the pair of row-precision requirements in equation 223. For Chen et al. (2026), $\widehat{T}$ is a supplied TC or DTC upper bound. For the Li–Cai row, $K = N$ removes the factorization term in its Theorem 1; for the Chen et al. row, exact singleton sampling gives the N-call cap.
- The scaling comparison fixes positive output tolerance and polynomial vocabulary. Constant allocations to source-specific learning or sampling error do not change the displayed powers of N.
- Our rows use $s > 1$ and the balanced examples of equation 3. Their common leading resource power is $2/3 + 1/(9s) < 1$; the target class and oracle assumptions remain part of the comparison.

K Ideal-target fixed-accuracy protocol

K.1 Targets

Let $\mathcal{V} = \{0, \dots, 2047\}$, with $\phi(0) = 1$, $\phi(1) = -1$, and $\phi(a) = 0$ otherwise. On a hidden forest $F = ([N], E)$, the target law is

$$P(x) = \frac{1}{Z_F} \prod_{i=1}^{N} \psi_i(x_i) \prod_{\{i,j\} \in E} \big(1 + w_{ij} \phi(x_i)\phi(x_j)\big), \qquad x \in \mathcal{V}^N, \tag{225}$$

where Z_F is the normalizer. For independent $h_i \sim \mathrm{Unif}[-1, 1]$, its unary factors are

$$\psi_i(a) = \frac{1}{0.4e^{h_i} + 0.4e^{-h_i} + 0.2} \begin{cases} 0.4e^{h_i}, & a = 0, \\ 0.4e^{-h_i}, & a = 1, \\ 0.2/2046, & a \in \{2, \dots, 2047\}. \end{cases} \tag{226}$$

For each target draw, a master field vector is shared by prefixes across sizes and forest families, and vertex labels are randomly permuted. Edge weights are given in Table 4. The frozen conditional oracle is exact, $q_j(\cdot \mid y) = \mu_j(\cdot \mid y)$, and the sampler is not given the edges. We test $N \in \{8192, 10240, 12288, 14336, 16384\}$.

Table 4: Ideal target families and fixed absolute accuracy thresholds. Here $\Delta_\star(N) = \max\{16, \lceil\sqrt{N}\rceil\}$, and $K_\star = 10^{-10}L_0$ uses the reference in Appendix K.2.

Family	Structure before relabeling	$w_F(N)$	$K_\star$
Matching	Disjoint pairs	0.5	6.375289×10^{-9}
Path	One path	0.25	2.801962×10^{-9}
Binary tree	Complete binary tree	$1/6$	1.226162×10^{-9}
Growing stars	At most $\Delta_\star(N)$ leaves per star	$0.5/\Delta_\star(N)$	5.138930×10^{-12}

K.2 Resources and accuracy

$$Q_{\mathrm{tot}} = Q_{\mathrm{pre}} + Q_{\mathrm{cf}} + R, \tag{227}$$

counts preprocessing submissions, counterfactual submissions, and nonempty commit rounds. Each submission is charged once, including repeated states; rows sharing a submission share its cost. For random sampling with B balanced batches, $Q_{\mathrm{tot}} = R = B$. The proposal has no external $Q_{\mathrm{tot}} \le N$ cap.

At history $H_{t-1} = (G_{t-1}, x_{G_{t-1}})$ before batch B_t, let y_t reveal $x_{G_{t-1}}$ and mask the other positions, and put $q_{t,i} = q_i(\cdot \mid y_t)$ and $P_t = P(X_{B_t} \in \cdot \mid X_{G_{t-1}} = x_{G_{t-1}})$, using the commit index of Definition A.9. We measure

$$K = \sum_{t=1}^{R} D_{\mathrm{KL}}\left(\bigotimes_{i\in B_t} q_{t,i} \,\middle\|\, P_t\right). \tag{228}$$

This is a product-to-joint sum along the sampled history, distinct from the target-to-output risk $\mathcal{R}_{\mathrm{KL}}$. Passive evaluation adds no oracle rounds: after conditioning and marginalization, boundary factors of arity at most four are enumerated exactly; larger factors use 128 independent Monte Carlo draws each. $\mathrm{SE}_{\mathrm{MC}}$ is the conditional standard error of the total estimate. A completed run passes the empirical criterion

$$K + 2\,\mathrm{SE}_{\mathrm{MC}} \le K_\star, \qquad K_\star = 10^{-10}L_0. \tag{229}$$

For each forest family, the reference target $P^{(0)}$ has $N = 2048$ and

$$L_0 = D_{\mathrm{KL}}\left(\bigotimes_{i=1}^{2048} P_i^{(0)} \,\middle\|\, P^{(0)}\right), \qquad P_i^{(0)} = \mathcal{L}_{P^{(0)}}(X_i). \tag{230}$$

Thus each absolute threshold in Table 4 is fixed across sizes and methods.

K.3 Sampling and evaluation

Proposal. We tune color counts, degree cutoffs, and thresholds empirically, without enforcing the theoretical parameter prescriptions (Section D.1 and Equation (117)); the target family also need not satisfy (RT; Equation (18)) with $L = \alpha = 1$. Within each (forest, N) cell, we screen candidate settings, then evaluate six low-cost diverse candidates on target draws $171, 172, 173$ crossed with sampler streams $271, 272$. We freeze the minimum-mean-Q_{tot} candidate that completes and passes on all six pairs at development threshold $10^{-3}L_0$. Screening and development use respectively 32 and 64 Monte Carlo draws per nonenumerated factor; evaluation uses 128. All 120 fresh proposal evaluation runs have structural exact zero discrepancy, $K = \mathrm{SE}_{\mathrm{MC}} = 0$.

Random baseline. For each evaluation run, fix a random permutation and commit random stream across budgets; budget B partitions the permutation into B balanced nonempty batches. Since adjacent partitions need not be nested, among observed budgets $b_1 < \cdots < b_m = N$ select the first passing suffix:

$$B_{\mathrm{safe}} = b_{k_\star}, \qquad k_\star = \min\{k : \text{every observed } b_\ell,\ \ell \ge k,\ \text{passes equation 229}\}. \tag{231}$$

The adjacent unsafe–safe bracket is refined to width at most $0.05N$ (observed maximum: $0.03125N$); $B = N$ is the exact singleton endpoint. This is per-run selection from each evaluation curve, not a globally frozen budget. Reported cost is B_{safe} and excludes search costs, just as proposal development is excluded.

Replicates. Each point uses target draws $196, 197, 198$, crossed with sampler streams $292, 293$ for the proposal and disjoint streams $294, 295$ for random. Means and min–max ranges use these same six runs.

Computation. Eight workers ran on an Apple-silicon Mac mini (macOS 15.7.7, Python 3.13.15, NumPy 2.2.2) and a Linux 5.4.0 host (Python 3.10.12, NumPy 1.26.4; one BLAS/OpenMP thread per worker). The proposal/random study took about 64.7 minutes; the per-run random budget searches took about 12.0 minutes in total.