When a Known Answer Unlocks Search: Split–Localize Control for Structured One-Sided Search

Anonymous authors
Paper under double-blind review

Abstract

A query whose answer is already known can still be essential for efficient search. We study structured one-sided search in which targets must be explicitly localized before they leave the active query system. On an anchor family, exact one-step information gain and exact discovery gain require $\Theta(n)$ expected queries, while executing a zero-information localization first enables $\Theta(\log n)$ search. This motivates *split–localize control*, whose exact-posterior form admits an $O(H(\mu_0) + \bar{k}_0)$ scored-query bound and a constant-factor total-query guarantee after certification costs are included. We introduce Fractional-SL as a scalable realization based on entropy-regularized action weights. Under fixed nested-prefix geometry, exact optimization, and order-aligned proposal spread $R = \exp(\Delta/\tau)$, Fractional-SL uses $O_\lambda(RK \log(en))$ total queries. A one-target family forces Fractional-SL to use $\Omega(K \log(n/K))$ queries, exposing its sensitivity to a loose target-count upper bound. With 10,000 tail candidates, Fractional-SL requires 19.63 queries on average versus 5,004.5 for the exact myopic policies. In informative-score multi-target experiments, enforcing the certified positive-set constraint saves 3.55–7.75 queries on average under a same-selector ablation.

1 Introduction

A query with a deterministic response has zero one-step information gain. In a stateful search process, executing it can nevertheless change which future queries are useful. This distinction matters whenever sensing and operational state are coupled: learning that an object is present is not the same action as removing, isolating, or otherwise changing its role in subsequent tests.

We study this effect in structured one-sided search. A query returns a single existential bit indicating whether its set contains at least one target. Targets must eventually be localized by positive singleton queries, and localization changes the active system on which later structured queries are formed. Hence logical knowledge and operational state need not coincide. A target can already be certain under the posterior while remaining inside every large admissible query until an explicit localization removes it.

The distinction has a concrete analogue in sequential fault isolation. One failure can mask evidence about another, and diagnose–repair–retest procedures can expose faults that were previously hidden (Simpson & Sheppard, 1992). Our prefix access model formalizes this distinction through an abstract serial-access protocol: identifying a target leaves the active query system unchanged, whereas executing its singleton localization removes it from subsequent pools. Related structured-test restrictions arise in graph-constrained group testing and other settings with access-dependent pools (Cheraghchi et al., 2012; Karbasi & Zadimoghaddam, 2012; Sihag et al., 2022).

A minimal anchor construction makes the consequence exact. One active target a is known with probability one and one additional target is uniform among n tail candidates. The active order places a first and permits active singletons and prefixes. While a remains active, every nontrivial prefix is deterministically positive, so it cannot separate the tail. Exact one-step information gain and exact discovery gain therefore test uncertain tail singletons and require $\Theta(n)$ expected queries. Querying $\{a\}$ first has zero immediate information and zero logical discovery value, but it removes the mask. Balanced tail prefixes then become informative, yielding a $\Theta(\log n)$ continuation. The gap persists with exact posterior scores; it comes from the state transition induced by the query.

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.

This example separates three notions of progress. *Information value* measures posterior resolution. *Discovery value* measures growth of the logically certified target set. *Operational localization value* measures progress that changes the active search problem. The third can be positive even when the next observation is already known. We call the obstruction created by an active certain target *positive masking*, and the localization that removes it *operational unblocking*.

The mechanism motivates *split–localize control*. At a presence-certified state, follow an admissible chain of batches that adds one active candidate at a time. Either some batch has a balanced response probability, or the chain crosses the uncertain region by adding a candidate with substantial target probability. Exact-SL chooses between these two modes. Under a singleton-increment chain condition, informative batches are charged to posterior entropy and successful singleton queries are charged to the expected number of residual targets. This yields an expected $O(H(\mu_0) + \bar{k}_0)$ scored-query bound. Because identifying the residual hypothesis and explicitly localizing realized targets are both unavoidable at our operational endpoint, the same accounting gives a constant-factor comparison with the optimal admissible policy after certification queries are included.

For scale, we introduce *Fractional-SL*. It keeps the residual constraints symbolically but replaces posterior enumeration by an entropy-regularized fractional cover whose coordinates are action weights. Exact symbolic updates preserve the feasible family, while the fractional proposal chooses between structured splits and singletons. In the fixed nested-prefix geometry used by our multi-target experiments, exact optimization and order-aligned proposal spread $R = \exp(\Delta/\tau)$ yield a pathwise $O_\lambda(RK \log(en))$ bound on total oracle calls. A one-target construction requires $\Omega(K \log(n/K))$ queries for the unchanged controller, exposing a concrete supplied-budget slackness mechanism. The remaining-budget-one case admits the sharper geometric-contraction analysis used for the anchor continuation.

Experiments test both the mechanism and the cost of this approximation. On the anchor family, Fractional-SL localizes the known target first and follows the logarithmic regime: with 10,000 tail candidates it uses 19.63 queries on average including setup, compared with an analytic expectation of 5,004.5 for the exact myopic policies. Enumerable reachable states compare it with optimal sequential planning and posterior-aware baselines. Multi-target experiments then isolate the contribution of score information and the certified positive set, and a scale sweep reports the resulting query–computation trade-off.

Contributions. First, we establish a linear–logarithmic query-complexity separation showing that exact information- and discovery-greedy policies can miss a zero-information action that unlocks efficient search. Second, we develop Exact-SL for state-dependent query families with mandatory singleton localization and prove an entropy-plus-expected-localization bound together with a constant-factor total-query guarantee under the chain condition. Third, we introduce Fractional-SL and prove a pathwise $O_\lambda(RK \log(en))$ total-query bound under nested prefix geometry. We also give a one-target $\Omega(K \log(n/K))$ construction, showing that dependence on the supplied budget reflects a genuine controller mechanism. We evaluate the method against exact planning, score-matched splitting, and a strict no-cover ablation.

The anchor exposes a distinction between uncertainty and access. Executing the known singleton leaves the posterior over the original target set unchanged, yet enables informative batches that were previously masked. This suggests a design principle for structured search: evaluate an action both by what its response reveals and by which future tests its execution makes useful. Split–localize control implements this principle through local action scores, without constructing a lookahead tree.

Relation to prior work. Classical group testing and generalized binary search identify an unknown configuration through informative tests under a fixed or externally specified query interface (Aldridge et al., 2019; Nowak, 2011; Mussmann & Liang, 2018). Our analysis builds on the informative-test versus individual-test argument of Nikpey et al. (2024), which uses entropy and expected infections as separate accounting resources. We extend that argument to a state-dependent query family with mandatory singleton localization, where a singleton can be posterior-certain yet operationally necessary and where the expected residual target count also lower-bounds the optimal operational cost. Active search, decision-aware experimental design, and dynamically constrained design provide complementary views of nonmyopic query value and changing feasibility (Garnett et al., 2012; Jiang et al., 2017; Javdani et al., 2014; Huang et al., 2024; Nguyen et al., 2024; Huang et al., 2025; Guo et al., 2026). Appendix G gives a fuller comparison.

2 Problem Formulation and Query Value

Let V be a finite ground set and let the unknown target set $A^\star \subseteq V$ satisfy $|A^\star| \le K$. At round t, a policy chooses an admissible query $Q_t \in \mathcal{Q}(\mathcal{H}_t)$ as a function of the transcript and observes

$$Y_t = \mathbf{1}\{A^\star \cap Q_t \neq \varnothing\}. \tag{1}$$

The set-valued map $\mathcal{Q}(\mathcal{H}_t)$ is part of the problem instance. It may encode prefixes of an active ordering, graph-supported pools, or other structured access rules, and it may change after localization.

A positive singleton *operationally localizes* its element. Let L_t be the localized targets, N_t the certified non-targets, and $C_t = V \setminus (L_t \cup N_t)$ the active set. The endpoint requires $L_T = A^\star$ and $C_T = \varnothing$: every realized target has been explicitly localized and the absence of any residual target has been certified. This is stronger than merely identifying $A^\star$ in a posterior or version space. The distinction is essential for the anchor example: a target may be logically certain while still remaining in C_t and therefore in future structured queries.

Each unexplained positive response induces a residual set that must contain at least one active target. With $\mathcal{P}_t$ denoting these residual positive edges, the exact residual family is

$$\mathcal{F}_t = \{A \subseteq C_t : |A| \le K - |L_t|,\ A \cap P \neq \varnothing\ \forall P \in \mathcal{P}_t\}. \tag{2}$$

A negative answer removes the queried active elements and restricts every residual edge. A positive non-singleton adds a new edge. A positive singleton moves its target from C_t to L_t and removes every edge it explains. These updates preserve the exact transcript-consistent family; approximation enters only in action selection.

The scalable controller uses a small set of sound local closures rather than a general integer feasibility oracle. The one used directly in our Fractional-SL theory occurs when the remaining target budget is one. If $\mathcal{P}_t \neq \varnothing$, then every feasible residual target must lie in every positive edge, so

$$C_t \leftarrow \bigcap_{P \in \mathcal{P}_t} P, \qquad N_t \leftarrow N_t \cup \left(C_t^{\mathrm{old}} \setminus C_t\right). \tag{3}$$

Appendix A states the full residual invariant and the closure rules used in the experiments.

Let μ_t be the exact posterior on $\mathcal{F}_t$. For a query $Q \subseteq C_t$ and active candidate v, write $p_t(Q) = \mathbb{P}_{A\sim\mu_t}(A \cap Q \neq \varnothing)$ and $\rho_t(v) = \mathbb{P}_{A\sim\mu_t}(v \in A)$. The logically certified target set is

$$D_t = L_t \cup \bigcap_{A \in \mathcal{F}_t} A, \tag{4}$$

where the residual intersection lies in C_t. For deterministic feedback, $\mathrm{IG}_t(Q) = h_2(p_t(Q))$ and

$$\mathrm{DG}_t(Q) = \mathbb{E}[|D_{t+1}| - |D_t| \mid \mathcal{H}_t, Q_t = Q].$$

A singleton $\{v\}$ has one-step localization value $\rho_t(v)$. Entropies H and h_2 use natural logarithms. The stopping time T counts every oracle call through the operational endpoint, including guards. These quantities refer to different state changes. A balanced batch can carry $\log 2$ nats of information and localize nothing; a posterior-certain singleton can have zero information gain and zero discovery gain while having localization value one. Executing the latter can nevertheless change C_{t+1} and hence the next admissible query family.

Finally, residual presence is not always known when only an upper bound K is specified. If the empty residual hypothesis remains feasible, we allow the full active set C_t as a *presence guard*. A negative guard certifies completion; a positive guard creates the edge C_t and enters a presence-certified episode. Guard calls are counted in total query cost and are included explicitly in the constant-factor comparison below.

3 A Known Answer Can Unlock Logarithmic Search

Let $B_n = \{b_1, \ldots, b_n\}$. The anchor state has active set $\{a\} \cup B_n$, residual budget two, residual edges

$$\mathcal{P}_n^{\mathrm{anc}} = \{\{a\}, B_n\}, \tag{5}$$

and hence exact residual family

$$\mathcal{F}_n^{\mathrm{anc}} = \{\{a, b_i\} : 1 \le i \le n\}, \tag{6}$$

with a uniform posterior over the tail index. The active order is $a \succ b_1 \succ \cdots \succ b_n$, and admissible actions are active singletons and nontrivial prefixes.

Theorem 1 (Known-answer localization yields a linear–logarithmic separation)**.** *For $n \ge 3$, let $C^\star_{\mathrm{op}}(n)$ be the minimum expected query cost on Eqs. 5–6. Then*

$$\log_2 n \le C^\star_{\mathrm{op}}(n) \le 2 + \lceil \log_2 n \rceil, \tag{7}$$

while, under the stated score-order tie-breaking,

$$\mathbb{E}[T_{\mathrm{MaxIG}}(n)] = \mathbb{E}[T_{\mathrm{MaxDG}}(n)] = \frac{n+3}{2}. \tag{8}$$

Thus efficient search is $\Theta(\log n)$ while both exact myopic policies use $\Theta(n)$ expected queries.

Proof sketch. Every transcript must identify the uniform tail index, giving the lower bound. For the upper bound, query $\{a\}$ first; its deterministic positive response removes the anchor, after which prefix bisection identifies the tail in at most $\lceil \log_2 n \rceil$ answers and at most one final singleton localizes it. Before a is removed, every nontrivial prefix is deterministically positive. The anchor singleton has zero IG and DG, whereas an unresolved tail singleton has positive IG and DG, so both myopic policies scan tails in score order. Averaging the resulting $j+1$ cost over a uniform tail rank j yields Eq. 8. Appendix C gives the complete argument and a prescribed three-query construction whose all-positive branch reaches the anchor state; charging this common setup preserves both rates.

The separation uses two protocol features: posterior-certain targets are not removed for free, and batches cannot bypass the active anchor to query arbitrary tail subsets. These are precisely the state changes represented by operational localization and structured access.

The myopic failure is therefore present even with exact posterior scores: the zero-information anchor query is valuable because it changes the continuation problem.

4 Exact Split–Localize Control

The anchor suggests a general rule: when a useful split is unavailable, look for the individual candidate responsible for crossing the uncertain region and localize it instead. Exact-SL formalizes this idea with the exact posterior. Fix $\alpha \in (0, 1/2)$ and $\lambda > 0$. At a presence-certified nonterminal state, assume active singletons and a singleton-increment chain are admissible:

$$\varnothing = R_{t,0} \subset R_{t,1} \subset \cdots \subset R_{t,m_t} = C_t, \qquad |R_{t,i} \setminus R_{t,i-1}| = 1. \tag{9}$$

Let $\mathcal{B}_t = \{R_{t,2}, \ldots, R_{t,m_t-1}\}$. Exact-SL chooses a maximizer of

$$G_t(Q) = \begin{cases} h_2(p_t(Q)), & Q \in \mathcal{B}_t, \\ \lambda \rho_t(v), & Q = \{v\},\ v \in C_t, \end{cases} \tag{10}$$

Define $c_{\alpha,\lambda} = \min\{h_2(\alpha), \lambda\alpha, \lambda(1-2\alpha)\} > 0$.

Lemma 2 (A chain forces a split or a likely target)**.** *At every presence-certified state satisfying Eq. 9, Exact-SL selects an action of value at least $c_{\alpha,\lambda}$.*

Reason. Along the chain, the positive-response probability starts at zero and ends at one. If a nontrivial chain batch enters $[\alpha, 1-\alpha]$, its binary entropy is at least $h_2(\alpha)$. Otherwise the chain crosses this interval when a single candidate is added. If the first singleton is already balanced, that candidate has marginal at least α; if the chain jumps from below α to above $1-\alpha$, the added candidate has marginal greater than $1-2\alpha$. In either case a singleton has score at least the corresponding term in $c_{\alpha,\lambda}$. The split-or-high-marginal alternative is closely related to correlated group testing (Nikpey et al., 2024); here it is embedded in a stateful query family in which executing the singleton can change the next admissible batches.

Admissible chains beyond a fixed order. The chain condition is local to each reached state. An ordering may be recomputed after every response or localization, provided its nontrivial active prefixes remain queryable. The score-prefix family used in our experiments is one example. The condition also holds for connected-subgraph queries whenever the active induced graph remains connected at every reached state: a rooted connected expansion then provides the required chain. The guarantee therefore follows from the available query geometry and continues to apply when the ordering changes across states.

Theorem 3 (Entropy plus expected localization controls scored queries)**.** *Suppose every presence-certified nonterminal state reached by Exact-SL admits Eq. 9. Let $\bar{k}_0 = \mathbb{E}_{\mu_0}|A^\star \setminus L_0|$, and let T_+ count scored batch and singleton queries until every residual target is localized. Then*

$$\mathbb{E}[T_+] \le \frac{H(\mu_0) + \lambda \bar{k}_0}{c_{\alpha,\lambda}}. \tag{11}$$

Proof sketch. Separate the transcript into batch and singleton rounds. On a selected batch, the deterministic oracle transmits $h_2(p_t(Q_t)) \ge c_{\alpha,\lambda}$ nats of conditional information, so the expected number of batch rounds is at most $H(\mu_0)/c_{\alpha,\lambda}$. On a selected singleton, the success probability is at least $c_{\alpha,\lambda}/\lambda$, so the expected singleton count is at most $\lambda\bar{k}_0/c_{\alpha,\lambda}$. Successful singleton queries localize distinct realized targets and therefore occur at most $|A^\star \setminus L_0|$ times pathwise. Taking expectations charges the singleton rounds to $\bar{k}_0$. Appendix B gives the stopped-transcript argument needed to justify the accounting without assuming bounded stopping time in advance.

A lower bound matched to the operational endpoint. Both terms in Theorem 3 correspond to unavoidable costs at our endpoint. Any admissible policy must identify the residual hypothesis from binary observations and execute one positive singleton for every realized residual target. Hence

$$\mathrm{OPT}_{\mathrm{total}} \ge \max\left\{\frac{H(\mu_0)}{\log 2},\, \bar{k}_0\right\}. \tag{12}$$

Thus $\bar{k}_0$ is itself a lower bound on optimal operational cost.

Corollary 4 (Total-query guarantee)**.** *Under the assumptions of Theorem 3, let $\mathrm{OPT}_{\mathrm{total}}$ be the minimum expected number of oracle calls among admissible policies that identify the residual target set, explicitly localize every realized target, and certify completion. If every reached nonterminal state is presence-certified, then $\mathbb{E}[T] \le (\log 2 + \lambda)c_{\alpha,\lambda}^{-1}\mathrm{OPT}_{\mathrm{total}}$. With full-active presence guards in the upper-bound-only model,*

$$\mathbb{E}[T] \le \bar{k}_0 + 1 + \frac{H(\mu_0) + \lambda\bar{k}_0}{c_{\alpha,\lambda}} \le \left(1 + \frac{\log 2 + \lambda}{c_{\alpha,\lambda}}\right)\mathrm{OPT}_{\mathrm{total}} + 1. \tag{13}$$

For $\alpha = 1/3$ and $\lambda = 1$, the factors are below 5.08 without guards and below 6.08 up to one additive query in the general guarded model.

Applying Eq. 12 to Theorem 3 gives the guard-free factor. The guarded bound adds at most one terminal negative guard and charges positive guards to distinct localizations. Appendix B gives the full guard accounting and a uniform score-error extension.

5 Fractional Split–Localize

Exact-SL evaluates posterior response probabilities, which can require enumerating a large transversal family. Fractional-SL avoids this enumeration by computing convex action weights while retaining the exact symbolic residual constraints. At a presence-certified state, it solves

$$\begin{aligned} \max_{x\in[0,1]^{C_t}} \quad & \sum_{v\in C_t} \theta_t(v)x_v + \tau \sum_{v\in C_t} h_2(x_v) \\ \text{s.t.} \quad & \sum_{v\in P} x_v \ge 1 \quad (P \in \mathcal{P}_t), \\ & \sum_{v\in C_t} x_v \le K - |L_t|, \end{aligned} \tag{14}$$

where θ_t incorporates the external score signal and the entropy term prevents the proposal from collapsing onto a single explanation when the constraints allow several. The cover inequalities ensure that every unresolved positive set receives at least unit proposal mass, while the final constraint encodes the residual target budget. The resulting coordinates serve as action weights for the split–localize selector.

Budget-one form and computation. After the budget-one intersection closure, if one active candidate remains it is queried directly. For $m \geq 2$, the feasible region is the simplex $\sum_v x_v = 1$, and the unique interior optimizer has the one-parameter KKT form

$$x_v = \mathrm{sigmoid}((\theta_t(v) - \nu)/\tau), \qquad \sum_{v \in C_t} x_v = 1, \tag{15}$$

so ν is obtained by scalar bisection. The score θ_t tilts mass toward externally ranked candidates, while τ limits concentration. Once x is available, all score-prefix response surrogates can be evaluated in one linear scan using cumulative log-products. General residual covers retain the sparse convex program in Eq. 14.

For a prefix Q, the controller uses the batch-response surrogate

$$\widehat{p}_t(Q) = 1 - \prod_{v \in Q} (1 - x_v), \tag{16}$$

and scores it by $h_2(\widehat{p}_t(Q))$; the best singleton receives score λx_v. Non-singleton queries that already contain a certified positive edge are filtered because their response is known, but singleton queries are retained because a forced element may still require operational localization. After each response, exact symbolic updates and sound local closures are applied.

In the upper-bound-only interface, after closure residual presence is certified exactly when $\mathcal{P}_t \neq \varnothing$; fixed-cardinality diagnostics may supply presence separately. Algorithm 1 summarizes the complete control loop.

Algorithm 1 Fractional-SL control loop.

while true **do**
  Apply sound local closures to $(L_t, N_t, \mathcal{P}_t, C_t)$.
  if $C_t = \varnothing$ **then**
    return
  end if
  if residual presence is uncertified **then**
    Query the full active set C_t as a presence guard.
    Update the residual state and **continue**.
  end if
  Solve program 14 to obtain action weights x.
  Generate score-order prefixes and discard any non-singleton containing a certified positive edge.
  if no non-singleton prefix remains **then**
    Choose the highest-scoring singleton.
  else
    Choose the higher-scoring action between the best prefix $h_2(\widehat{p}_t(Q))$ and the best singleton λx_v.
    Break equal batch–singleton scores in favor of localization.
  end if
  Among equally scored singletons, choose the earliest candidate in the fixed active order.
  Execute the chosen action, apply the oracle response and exact residual updates.
end while

Nested-prefix regime. Assume a fixed total order induces the active order after deletions, every active singleton and every proper active prefix is queryable, and the full active set is available as a presence guard. Suppose θ_t is nonincreasing in that order at every reached state and $\max_v \theta_t(v) - \min_v \theta_t(v) \leq \Delta$. Set $R = \exp(\Delta/\tau)$. Positive prefix queries remain prefixes after negative deletions and singleton localizations, so the residual positive edges are nested. When nonempty, let B_t be the smallest such edge, $b_t = |B_t|$, $r_t = K - |L_t|$, and $S_t = \sum_{v \in B_t} x_v$.

Theorem 5 (Nested-prefix multi-target guarantee)**.** *Under the nested-prefix conditions above, assume Eq. 14 is solved exactly. If $|V| = n$ and $|A^\star| = k \le K$, then starting from the empty transcript Fractional-SL uses*

$$T = O_\lambda(RK \log(en)) \tag{17}$$

total oracle calls pathwise, including presence guards and explicit singleton localizations. No probabilistic calibration assumption on x is required.

Proof idea. KKT conditions separate binding and slack cover states. If $S_t = 1$, then $x_{\max} \le R/b_t$; positive batches and binding-cover negative batches can therefore be charged to geometric shrinkage of B_t. If $S_t > 1$, the cover multiplier vanishes and all active coordinates have ratio at most R, giving $x_{\max} \le Rr_t/|C_t|$; slack-cover negative batches are charged instead to global shrinkage of C_t. Certified-prefix shrinkage is charged across at most k localization phases, whereas global active-set shrinkage is charged only once. Large-coordinate states contribute only a bounded additional term. Appendix D gives the explicit pathwise bound and proof.

Proposition 6 (Supplied-budget dependence can be realized)**.** *Let $\lambda = 1$ and $\theta_t \equiv 0$, and place the sole realized target last in the fixed prefix order. For $n > 10K$, Fractional-SL makes at least*

$$\frac{\log(n/(10K))}{-\log(1 - 0.8/K)}$$

negative scored queries before the active pool falls below $10K$. Consequently, $T = \Omega(K \log(n/K))$ along families with $n/K \to \infty$, whereas a valid prefix-bisection policy uses $O(\log n)$ queries on the same instance.

What determines the budget dependence. On the lower-bound family, the optimizer fills the supplied budget, $x_v = K/m$, while the active pool is large. Balancing the product surrogate then selects prefixes of length $\Theta(m/K)$, although only one target is present. The relevant quantity is therefore the supplied bound K, which can substantially exceed the realized count k. Together, Theorem 5 and Proposition 6 establish logarithmic dependence on n for fixed K, R, and λ, and identify a concrete mechanism through which a loose budget slows Fractional-SL. The valid prefix-bisection comparison locates this additional cost in the controller's action selection.

The bound keeps R explicit. The z-score transform used in the multi-target experiments preserves the fixed score order but does not by itself imply a spread bound uniform in n. The equally spaced anchor scores do satisfy such a uniform bound. The anchor's initial edges are not nested; its continuation after localizing a is covered by the sharper remaining-budget-one result in Appendix D.

6 Experiments

All reported costs include operational localization and certified termination. Unless stated otherwise, $\lambda = 1$, $\tau = 0.05$, and $\theta_t(v) = 0.05\,\mathrm{zscore}_{C_t} s(v)$. Random seeds are fixed before aggregation, and raw trajectories are written before summary tables are computed. Appendix E gives complete protocols, pointwise 95% Student-t intervals, timing diagnostics, and source-data references. We use four sets of experiments to examine the unblocking mechanism, compare Fractional-SL with exact continuation planning, test its large-scale anchor behavior, and isolate the contribution of the certified positive-set constraint.

Mechanism and exact anchor control. A balanced single-target batch carries $\log 2$ nats of information while localizing nothing; the known anchor has zero IG and DG but localization value one. On enumerable anchor instances with $n \in \{4, 6, 8, 10, 12\}$, dynamic programming computes the optimal sequential policy exactly. Exact-SL matches the optimal operational cost at two tested sizes and remains within one third of a query at the others. The large-scale anchor sweep uses 30 matched targets for each $n \in \{32, 64, 128, 256, 512, 1000, 2000, 5000, 10000\}$. Fractional-SL localizes the anchor first in every run; subsequent prefixes retain only about 0.56–0.57 of the active tails on average. At $n = 10{,}000$, the mean total is 19.63 queries including the common setup, versus 18.36 for an ideal balanced reference and 5,004.5 analytically for MaxIG/MaxDG. Figure 1 (a,c) shows the geometric contraction and linear–logarithmic scaling gap.

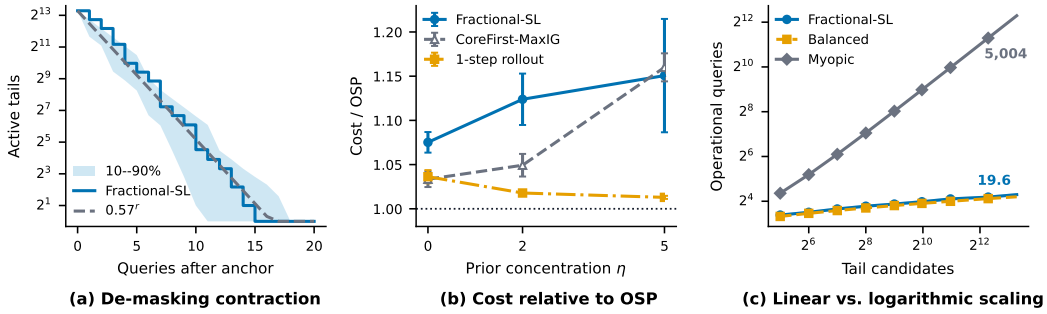


Figure 1: Evidence for split–localize control. (a) After anchor localization, Fractional-SL contracts the active tail set geometrically (10th–90th percentiles over 30 runs). (b) Reachable-state continuation cost relative to optimal sequential planning (OSP), with pointwise 95% Student-t intervals; costs begin after the common reachable-state initialization. (c) Fractional-SL and an ideal balanced reference remain logarithmic while the exact myopic expectation grows linearly; panel (c) includes the common three-query anchor setup.

Reachable-state comparison with exact planning. Figure 1(b) comes from a separate small-state benchmark rather than the anchor sweep. Each state has $n = 8$ and exactly two targets. A concentration parameter $\eta \in \{0, 2, 5\}$ changes the prior preference for one distinguished element; for each η, 30 states are generated by a common positive full-set query followed by one sampled progress-making prefix and its realized response. The evaluator then enumerates every future response branch, so OSP is the exact optimal expected continuation cost from that reachable state. The fixed-cardinality Fractional-SL variant uses symbolic residual constraints and external scores for action selection, with residual presence supplied by the known count. It attains mean continuation-cost ratios of 1.075–1.151 relative to OSP. One-step rollout is closer to optimal but requires exact posterior evaluation plus an extra planning step at every decision. Panel (b) reports these continuation costs after the common initialization; unlike panel (c), it does not include the three-query anchor setup.

Multi-target search and score-matched baselines. We next start from an empty transcript at $n = 128$, use supplied bounds $K \in \{2, 4, 8\}$ and true counts $k \in \{K - 1, K\}$, and generate scores as

$$s_v = \epsilon_v + \delta\, \mathbf{1}\{v \in A^\star\}, \qquad \epsilon_v \overset{\text{iid}}{\sim} N(0, 1), \qquad \delta \in \{0, 2\}.$$

Thus $\delta = 0$ removes score information and $\delta = 2$ supplies a moderate target shift. The fixed score order and complete prefix family satisfy the nested geometry of Theorem 5; in this protocol all cover inequalities reduce to the constraint on the smallest certified positive prefix. With informative scores, Fractional-SL improves over count-based prefix bisection in all six cells. With $\delta = 0$, prefix bisection has lower mean cost in all six cells, showing that the multi-target advantage depends on a useful score signal. Weighted-Split uses the same standardized scores, target budget, active order, local closures, and stopping rule but performs a direct half-mass split of the shortest certified bracket. Weighted-Split has slightly lower mean cost at $K = 2$. Fractional-SL gains are small at $K = 4$ and increase to 18–20% fewer queries at $K = 8$.

Strict positive-set ablation. No-Cover Fractional-SL retains the score-and-entropy objective but removes the positive-edge constraints. The product surrogate, prefix candidates, certified-positive filtering, singleton competition, local closures, and stopping rule remain unchanged. Omitting these constraints increases mean query count in all six informative-score cells by 3.55–7.75 queries (Table 1). At $K = 8$, the paired increases are 6.80 ± 1.75 and 7.75 ± 1.45 queries for $k = 7$ and $k = 8$.

In this nested regime the smallest certified positive edge is the effective constraint: it forces at least unit proposal mass into a set already known to contain a residual target. Without that constraint, the score-and-entropy proposal can allocate substantially less mass to the certified region, changing which prefix competes with localization. Table 1 directly measures the query-cost effect of the constraint; the mass-redistribution argument above provides a mechanism-level interpretation.

Table 1: Informative-score multi-target search at $n = 128$ (20 matched runs per cell). "No cover" removes only the positive-edge constraints from Fractional-SL. Δ_{NC} is No-cover minus Fractional-SL with a pointwise 95% paired Student-t interval.

K	k	Fractional	No cover	Weighted	Δ_{NC}
2	1	7.60	11.15	7.35	$+3.55 \pm 1.76$
2	2	12.95	17.80	12.50	$+4.85 \pm 1.08$
4	3	19.35	25.35	20.40	$+6.00 \pm 1.86$
4	4	24.05	31.40	24.70	$+7.35 \pm 1.33$
8	7	35.50	42.30	44.20	$+6.80 \pm 1.75$
8	8	40.95	48.70	49.75	$+7.75 \pm 1.45$

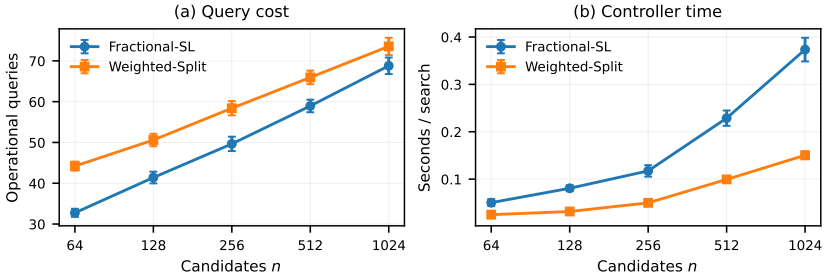


Figure 2: Query–computation scaling at $K = k = 8$ and $\delta = 2$ (20 matched instances per size). Error bars are pointwise 95% Student-t intervals over instances. Fractional-SL retains a query advantage across the sweep, while the simpler Weighted-Split controller is faster and the query gap decreases at fixed K.

Query–computation scaling. We fix $K = k = 8$ and $\delta = 2$ and sweep $n \in \{64, 128, 256, 512, 1024\}$ with 20 matched instances per size. All 200 policy runs terminate successfully. Fractional-SL uses fewer queries throughout, from 32.75 versus 44.20 at $n = 64$ to 68.80 versus 73.55 at $n = 1024$, while Weighted-Split requires less controller time (Figure 2). At $n = 1024$, the mean controller times are 0.373 and 0.150 seconds, respectively: Fractional-SL saves 4.75 queries at an additional controller cost of 0.223 seconds per search. This trade-off favors Fractional-SL when oracle calls dominate the additional computation, and the simpler controller when query latency is negligible.

Response-surrogate diagnostic. A remaining-budget-one ablation isolates the product surrogate itself. Under uniform single-target symmetry, $x_v = 1/m$ and a prefix of size q receives

$$\widehat{p}_{\mathrm{prod}}(q) = 1 - (1 - 1/m)^q.$$

Balancing this surrogate at $1/2$ gives $q = \log(1/2)/\log(1 - 1/m) \sim m \log 2$, whereas cumulative mass selects the exact half split $q = m/2$. At $m = 10{,}000$, cumulative mass reduces mean query count from 15.83 to 14.13 in the uniform experiment. Under the anchor score gradient, product and mass instead require 19.63 and 23.03 queries, respectively, including the common setup. Cumulative mass gives the true response probability under uniform single-target symmetry; with proposal-driven weights, the two surrogates induce different search trajectories and their ordering reverses. Appendix E reports the complete comparison and sensitivity checks.

Conclusion. A known answer can be worth executing because it changes the future query family. The anchor construction makes this operational value precise: exact information- and discovery-greedy policies require linear expected cost, while a zero-information localization enables logarithmic search. Under the chain condition, Exact-SL combines entropy reduction and mandatory localization into a constant-factor total-query guarantee. For Fractional-SL, the nested-prefix analysis establishes a pathwise $O_\lambda(RK \log(en))$ bound and explains how loose supplied budgets can slow the controller. Incomparable positive-edge geometries and finite optimization error remain open extensions. These results motivate search policies that account for both information acquired and the future queries made useful by execution.

AI USE STATEMENT

Generative AI tools were used in a limited supporting role during manuscript preparation. Their primary use was editorial: improving wording, readability, transitions, section organization, and terminology in material already developed by the authors. This included clarifying the distinction between information gain, discovery gain, and operational localization; improving the presentation of the anchor example and the progression from Exact-SL to Fractional-SL; and making the scope of the nested-prefix analysis and supplied-budget example easier to follow. These uses concerned presentation rather than the development of the underlying mathematical results.

AI tools were also used as a non-authoritative critical reader of already written technical material, for example to flag ambiguous wording, unclear qualifications, inconsistent notation, or places where the scope of a claim might be misunderstood. Any such issue was independently re-examined and resolved by the authors. AI output was not treated as evidence of mathematical correctness.

For experiments, AI assistance was limited to improving the presentation of already completed analyses, including descriptions of the reachable-state comparison, multi-target experiments, positive-set ablation, query–computation trade-off, response-surrogate diagnostic, and figure/table captions. AI tools were also used for literature retrieval and bibliographic assistance, such as suggesting search terms or potentially relevant work and helping check or format metadata; all retained references were verified against original papers, proceedings, or publisher records.

Generative AI was not relied upon to originate the research question, the split–localize framework, the anchor construction, the mathematical guarantees, the experimental data, or the scientific conclusions. The final theorem statements, proofs, assumptions, experimental results, citations, and conclusions were determined and manually reviewed and verified by the authors. AI-assisted material was incorporated only after human evaluation. The authors take full responsibility for the final content of the paper.

REFERENCES

Matthew Aldridge, Oliver Johnson, and Jonathan Scarlett. Group testing: An information theory perspective. *Foundations and Trends in Communications and Information Theory*, 15(3–4):196–392, 2019. doi: 10.1561/0100000099.

Mahdi Cheraghchi, Amin Karbasi, Soheil Mohajer, and Venkatesh Saligrama. Graph-constrained group testing. *IEEE Transactions on Information Theory*, 58(1):248–262, 2012. doi: 10.1109/TIT.2011.2169535.

Vašek Chvátal. A greedy heuristic for the set-covering problem. *Mathematics of Operations Research*, 4(3):233–235, 1979. doi: 10.1287/moor.4.3.233.

Sanjoy Dasgupta. Analysis of a greedy active learning strategy. In *Advances in Neural Information Processing Systems 17*, pp. 337–344, 2004. URL https://proceedings.neurips.cc/paper/2004/hash/c61fbef63df5ff317aecdc3670094472-Abstract.html.

Thomas G. Dietterich, Richard H. Lathrop, and Tomás Lozano-Pérez. Solving the multiple instance problem with axis-parallel rectangles. *Artificial Intelligence*, 89(1–2):31–71, 1997. doi: 10.1016/S0004-3702(96)00034-3.

Robert Dorfman. The detection of defective members of large populations. *The Annals of Mathematical Statistics*, 14(4):436–440, 1943. doi: 10.1214/aoms/1177731363.

Miroslav Fiedler. Algebraic connectivity of graphs. *Czechoslovak Mathematical Journal*, 23(2):298–305, 1973. doi: 10.21136/CMJ.1973.101168.

Roman Garnett, Yamuna Krishnamurthy, Xuehan Xiong, Jeff Schneider, and Richard P. Mann. Bayesian optimal active search and surveying. In *Proceedings of the 29th International Conference on Machine Learning*, pp. 843–850, 2012. URL https://icml.cc/2012/papers/618.pdf.

Daniel Golovin and Andreas Krause. Adaptive submodularity: Theory and applications in active learning and stochastic optimization. *Journal of Artificial Intelligence Research*, 42:427–486, 2011. doi: 10.1613/jair.3278.

Daniel Golovin, Andreas Krause, and Debajyoti Ray. Near-optimal bayesian active learning with noisy observations. In *Advances in Neural Information Processing Systems 23*, pp. 766–774, 2010. URL https://proceedings.neurips.cc/paper/2010/hash/1e6e0a04d20f50967c64dac2d639a577-Abstract.html.

Mira Gonen, Michael Langberg, and Alex Sprintson. Group testing on general set-systems. In *2022 IEEE International Symposium on Information Theory (ISIT)*, pp. 874–879, 2022. doi: 10.1109/ISIT50566.2022.9834789.

Andrew Guillory and Jeff Bilmes. Interactive submodular set cover. In *Proceedings of the 27th International Conference on Machine Learning*, pp. 415–422, 2010. URL https://icml.cc/Conferences/2010/papers/436.pdf.

Andrew Guillory and Jeff Bilmes. Simultaneous learning and covering with adversarial noise. In *Proceedings of the 28th International Conference on Machine Learning*, pp. 369–376, 2011. URL https://icml.cc/Conferences/2011/papers/261_icmlpaper.pdf.

Yujia Guo, Daolang Huang, Xinyu Zhang, Sammie Katt, Samuel Kaski, and Ayush Bharti. Constrained bayesian experimental design via online planning. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026. URL https://arxiv.org/abs/2605.26990. Accepted at ICML 2026.

Daolang Huang, Yujia Guo, Luigi Acerbi, and Samuel Kaski. Amortized bayesian experimental design for decision-making. In *Advances in Neural Information Processing Systems 37*, volume 37, pp. 109460–109486, 2024. doi: 10.52202/079017-3475. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/c59f05d7ab3638b138cc61f32e1a7cd1-Abstract-Conference.html.

Daolang Huang, Xinyi Wen, Ayush Bharti, Samuel Kaski, and Luigi Acerbi. ALINE: Joint amortization for bayesian inference and active data acquisition. In *Advances in Neural Information Processing Systems 38*, volume 38, 2025. doi: 10.52202/085713-1805. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/4df9a5e6bad9e64ebcea453e031142bb-Abstract-Conference.html.

Frank K. Hwang. A method for detecting all defective members in a population by group testing. *Journal of the American Statistical Association*, 67(339):605–608, 1972. doi: 10.1080/01621459.1972.10481257.

Laurent Hyafil and Ronald L. Rivest. Constructing optimal binary decision trees is np-complete. *Information Processing Letters*, 5(1):15–17, 1976. doi: 10.1016/0020-0190(76)90095-8.

Shervin Javdani, Yuxin Chen, Amin Karbasi, Andreas Krause, J. Andrew Bagnell, and Siddhartha S. Srinivasa. Near optimal bayesian active learning for decision making. In *Proceedings of the Seventeenth International Conference on Artificial Intelligence and Statistics*, volume 33 of *Proceedings of Machine Learning Research*, pp. 430–438. PMLR, 2014. URL https://proceedings.mlr.press/v33/javdani14.html.

Bruno Jedynak, Peter I. Frazier, and Raphaël Sznitman. Twenty questions with noise: Bayes optimal policies for entropy loss. *Journal of Applied Probability*, 49(1):114–136, 2012. doi: 10.1239/jap/1331216837.

Shali Jiang, Gustavo Malkomes, Geoff Converse, Alyssa Shofner, Benjamin Moseley, and Roman Garnett. Efficient nonmyopic active search. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1714–1723. PMLR, 2017. URL https://proceedings.mlr.press/v70/jiang17d.html.

Shali Jiang, Gustavo Malkomes, Matthew Abbott, Benjamin Moseley, and Roman Garnett. Efficient nonmyopic batch active search. In *Advances in Neural Information Processing Systems 31*, pp. 1107–1117, 2018. URL https://proceedings.neurips.cc/paper/2018/hash/a7aeed74714116f3b292a982238f83d2-Abstract.html.

Shali Jiang, Roman Garnett, and Benjamin Moseley. Cost effective active search. In *Advances in Neural Information Processing Systems 32*, pp. 4881–4890, 2019. URL https://proceedings.neurips.cc/paper/2019/hash/df0e09d6f25a15a815563df9827f48fa-Abstract.html.

Kirthevasan Kandasamy, Willie Neiswanger, Reed Zhang, Akshay Krishnamurthy, Jeff Schneider, and Barnabás Póczos. Myopic posterior sampling for adaptive goal oriented design of experiments. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 3222–3232. PMLR, 2019. URL https://proceedings.mlr.press/v97/kandasamy19a.html.

Amin Karbasi and Morteza Zadimoghaddam. Sequential group testing with graph constraints. In *2012 IEEE Information Theory Workshop (ITW)*, pp. 292–296, 2012. doi: 10.1109/ITW.2012.6404678.

Andreas Krause and Carlos Guestrin. Optimal value of information in graphical models. *Journal of Artificial Intelligence Research*, 35:557–591, 2009. doi: 10.1613/jair.2737.

László Lovász. On the ratio of optimal integral and fractional covers. *Discrete Mathematics*, 13(4): 383–390, 1975. doi: 10.1016/0012-365X(75)90058-8.

Yifei Ma, Roman Garnett, and Jeff Schneider. Active search for sparse signals with region sensing. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, pp. 2315–2321, 2017. doi: 10.1609/AAAI.V31I1.10865.

Stephen Mussmann and Percy Liang. Generalized binary search for split-neighborly problems. In *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, volume 84 of *Proceedings of Machine Learning Research*, pp. 1561–1569. PMLR, 2018. URL https://proceedings.mlr.press/v84/mussmann18a.html.

Quan Nguyen, Anindya Sarkar, and Roman Garnett. Amortized nonmyopic active search via deep imitation learning, 2024. URL https://arxiv.org/abs/2405.15031.

Hesam Nikpey, Saswati Sarkar, and Shirin Saeedi Bidokhti. Group testing with general correlation using hypergraphs. In *2024 IEEE International Symposium on Information Theory (ISIT)*, pp. 3225–3230, 2024. doi: 10.1109/ISIT57864.2024.10619244. URL https://arxiv.org/abs/2412.17751.

Robert D. Nowak. The geometry of generalized binary search. *IEEE Transactions on Information Theory*, 57(12):7893–7906, 2011. doi: 10.1109/TIT.2011.2169298.

Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000. doi: 10.1109/34.868688.

Saurabh Sihag, Ali Tajer, and Urbashi Mitra. Adaptive graph-constrained group testing. *IEEE Transactions on Signal Processing*, 70:381–396, 2022. doi: 10.1109/TSP.2021.3137026.

William R. Simpson and John W. Sheppard. System testability assessment for integrated diagnostics. *IEEE Design & Test of Computers*, 9(1):40–54, 1992. doi: 10.1109/54.124516.

Bruce Spang and Mary Wootters. Unconstraining graph-constrained group testing. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2019)*, volume 145 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pp. 46:1–46:20. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2019. doi: 10.4230/LIPIcs.APPROX-RANDOM.2019.46.

A Exact Residual Semantics and Stopping

Write $A_t^\star = A^\star \setminus L_t$ for the true residual target set. The state $(L_t, N_t, \mathcal{P}_t)$ exactly represents the hypotheses in Eq. 2. The following update identity records this invariant.

Lemma 7 (Exact residual invariant)**.** *Assume $\mathcal{F}_t$ is the feasible residual family before querying $Q \subseteq C_t$, and consider a response branch with positive posterior probability. After the state update in Section 2, the new support is*

$$\mathcal{F}_{t+1} = \begin{cases} \{A \in \mathcal{F}_t : A \cap Q = \varnothing\}, & Y_t = 0, \\ \{A \in \mathcal{F}_t : A \cap Q \neq \varnothing\}, & Y_t = 1,\ |Q| > 1, \\ \{A \setminus \{v\} : A \in \mathcal{F}_t,\ v \in A\}, & Y_t = 1,\ Q = \{v\}. \end{cases} \tag{18}$$

The posterior μ_{t+1} is obtained by conditioning in the first two cases and by the corresponding pushforward after conditioning in the third. The sound local closures may shrink the ambient active set but leave this support unchanged.

Proof. A negative response imposes $A \cap Q = \varnothing$. Removing Q from the active set and from every positive edge produces exactly this restriction of Eq. 2. A positive non-singleton contributes the new constraint $A \cap Q \neq \varnothing$, which is represented by adjoining Q to $\mathcal{P}_t$. Finally, a positive singleton proves $v \in A_t^\star$ and moves v to L_{t+1}. Every edge containing v is thereby explained; the remaining edges constrain $A \setminus \{v\}$. The residual budget also decreases by one, giving the last line of Eq. 18. These operations are reversible at the level of feasible residual sets, so they neither add nor discard a hypothesis beyond conditioning on the observed response. □

Budget-one closure. Suppose $K - |L_t| = 1$ and a query Q is positive. Every surviving $A \in \mathcal{F}_t$ then satisfies $|A| \leq 1$ and $A \cap Q \neq \varnothing$, hence $A \subseteq Q$. No surviving hypothesis contains an element of $C_t \setminus Q$, proving Eq. 3. Restricting all residual edges to Q therefore preserves the exact support in Lemma 7.

Why certified stopping needs a guard. If $\mathcal{P}_t \neq \varnothing$, the empty residual set violates at least one edge, so $p_t(C_t) = 1$. When $\mathcal{P}_t = \varnothing$ under the upper-bound model, the empty set satisfies Eq. 2 and may retain posterior mass; localizing every realized target is not, by itself, an observable stopping rule. Querying C_t closes this gap exactly: a negative response leaves only the empty residual hypothesis and moves all of C_t to N_{t+1}, while a positive response adds the edge C_t and begins a new presence-certified episode.

B Exact Split–Localize Proofs

We now justify the two charges used in the proof sketch of Theorem 3 without assuming in advance that the stopping time is bounded. Let $Z \sim \mu_0$ denote the initial residual hypothesis and let $\mathcal{G}_t$ contain the transcript and any policy randomness revealed before round t.

Full proof of Theorem 3. Let $\tau = T_+$ and truncate at $\tau_m = \min\{\tau, m\}$. Write $N_{\mathrm{batch}}(m)$ and $N_{\mathrm{sing}}(m)$ for the numbers of batch and singleton rounds before τ_m. Because Q_t is $\mathcal{G}_t$-measurable and the oracle is deterministic conditional on Z,

$$I(Z; Y_t \mid \mathcal{G}_t) = H(Y_t \mid \mathcal{G}_t) = h_2(p_t(Q_t)). \tag{19}$$

Let $\mathcal{T}_m$ be the length-m transcript obtained by using a null symbol after stopping. The chain rule applies to $\mathcal{T}_m$. Singleton rounds contribute nonnegative information, while every batch round selected before τ_m has value at least $c_{\alpha,\lambda}$ by Lemma 2. Hence

$$c_{\alpha,\lambda}\mathbb{E}[N_{\mathrm{batch}}(m)] \leq I(Z; \mathcal{T}_m) \leq H(\mu_0). \tag{20}$$

For a singleton round, let S_t indicate that the queried element is a target. Conditional on $\mathcal{G}_t$, its success probability is $\rho_t(v_t) \geq c_{\alpha,\lambda}/\lambda$. Therefore

$$\frac{c_{\alpha,\lambda}}{\lambda}\mathbb{E}[N_{\mathrm{sing}}(m)] \leq \mathbb{E}\left[\sum_{t<\tau_m} \mathbf{1}\{Q_t = \{v_t\}\}\rho_t(v_t)\right] \tag{21}$$

$$= \mathbb{E}\left[\sum_{t<\tau_m} \mathbf{1}\{Q_t = \{v_t\}\}S_t\right] \leq \mathbb{E}_{\mu_0}[|A^\star \setminus L_0|] = \bar{k}_0. \tag{22}$$

The inequality follows pathwise before taking expectation: active singletons are never queried again, and each successful singleton localizes a distinct realized residual target.

Both counts increase monotonically with m. Monotone convergence in Eqs. 20–22 gives

$$\mathbb{E}[T_+] = \mathbb{E}[N_{\mathrm{batch}}] + \mathbb{E}[N_{\mathrm{sing}}] \leq \frac{H(\mu_0) + \lambda\bar{k}_0}{c_{\alpha,\lambda}}.$$

The finite right-hand side also implies $T_+ < \infty$ almost surely. □

Full proof of Corollary 4. Ignore guard rounds and apply the preceding two charges to the single global transcript: every scored round lies inside a presence-certified episode, so the same bounds on N_{batch} and N_{sing} apply across all episodes.

A negative guard terminates and can occur at most once. Each positive guard creates the residual edge C_t. Before another guard can be issued, every positive edge must have disappeared. Negative queries can only restrict this edge while preserving its residual target; a positive non-singleton adds an edge. Thus the guard edge can disappear only when a positive singleton localizes a target that explains it. Charge the guard to that target. The charged targets are distinct, so the positive-guard count is pathwise at most $|A^\star \setminus L_0|$ and has expectation at most $\bar{k}_0$. Adding at most one negative guard to Eq. 11 proves Eq. 13. □

Lemma 8 (Uniform score error preserves the query bound)**.** *Suppose the candidate set contains the chain batches and active singletons, and approximate scores satisfy $|\widehat{G}_t(Q) - G_t(Q)| \leq \delta$ for every candidate at every reached presence-certified state. If $2\delta < c_{\alpha,\lambda}$, the approximate maximizer satisfies*

$$\mathbb{E}[T_+] \leq \frac{H(\mu_0) + \lambda\bar{k}_0}{c_{\alpha,\lambda} - 2\delta}. \tag{23}$$

Proof. Let $Q_t^\star$ maximize the exact score and let $\widehat{Q}_t$ maximize the approximate score. Then

$$G_t(\widehat{Q}_t) \geq \widehat{G}_t(\widehat{Q}_t) - \delta \geq \widehat{G}_t(Q_t^\star) - \delta \geq G_t(Q_t^\star) - 2\delta \geq c_{\alpha,\lambda} - 2\delta.$$

Repeat the stopped information and localization charges above with this smaller progress threshold. □

C Anchor Reachability and Myopic Policies

Lemma 9 (Three positive batches realize the anchor family)**.** *Let c, d be two auxiliary elements and start from an empty positive-edge family on $\{a, c, d\} \cup B_n$ with budget two. Consider a staged protocol that issues the following three common non-singleton queries before opening the prefix/singleton continuation family:*

$$B_n, \qquad \{a, c\}, \qquad \{a, d\} \tag{24}$$

The queries, followed by the specialized reduction in the proof, produce the anchor state in Eq. 5 whenever all three responses are positive. A prior symmetric over the tails induces the uniform conditional tail distribution used by the anchor theorem. No target is operationally localized.

Proof. A positive answer on B_n forces every residual hypothesis to contain some b_i. If a size-two hypothesis hitting the other two positive edges omitted a, it would also have to contain both c and d, requiring at least three elements once b_i is included. Hence it contains a, and the budget forces

it to be $\{a, b_i\}$. Conversely, every $\{a, b_i\}$ hits all three queried sets. Thus the exact family after the transcript is $\{\{a, b_i\} : 1 \le i \le n\}$. Neither c nor d occurs in a feasible hypothesis. This argument gives a specialized linear-time reduction that moves them to N_t and restricts the auxiliary edges to $\{a\}$, without solving a general zero-support feasibility problem. The remaining distinct edges are exactly $\{a\}$ and B_n. □

Exact myopic path. Suppose $r > 1$ tail candidates remain while a is active. Every nontrivial prefix contains a, so its response is deterministic and its IG and DG are zero. The anchor singleton also has zero IG and zero DG because it only moves a from the residual core into L_t. A tail singleton has positive IG $h_2(1/r)$. Its DG is also positive: a positive answer certifies the queried tail, and when $r = 2$ a negative answer certifies the other tail by elimination. Thus both MaxIG and MaxDG select tails in score order until one tail remains.

If the true tail has rank $j < n$, the policy observes $j - 1$ negative tails, then localizes b_j, and finally localizes a, for $j + 1$ queries. If $j = n$, the first $n - 1$ tails are negative. Both scores are then zero on the known anchor and the logically forced final tail; fixed score-order tie-breaking queries a first and the tail second, again for $j + 1$ queries. Averaging over the uniform tail index gives

$$\frac{1}{n}\sum_{j=1}^{n}(j+1) = \frac{n+3}{2}.$$

The decision-tree lower bound and anchor-first upper bound in Theorem 1 then give the claimed $\Theta(n)$-versus-$\Theta(\log n)$ separation. Adding the three common queries in Eq. 24 preserves both rates.

D Fractional Nested-Prefix and Budget-One Analysis

We first prove the multi-target result for the nested-prefix regime of Theorem 5. All logarithms are natural. At a certified state let C be the active set, $m = |C|$, $r = K - |L|$, and let B be the smallest residual positive prefix, with $b = |B|$. Because the positive edges are nested, Eq. 14 reduces to

$$\max_{0\le x\le 1} \sum_{v\in C} \theta_v x_v + \tau \sum_{v\in C} h_2(x_v) \quad \text{s.t.} \quad \sum_{v\in B} x_v \ge 1, \qquad \sum_{v\in C} x_v \le r. \tag{25}$$

Every surviving batch is a prefix strictly contained in B.

Lemma 10 (Order and coordinate ratios)**.** *For $b \ge 2$, the optimizer is nonincreasing in the active order. Writing $S_B = \sum_{v\in B} x_v$ and $z = \max_{v\in C} x_v$,*

$$z = \max_{v\in B} x_v, \qquad \max_{v\in B} x_v \le R \min_{v\in B} x_v, \qquad z \le \frac{RS_B}{b}. \tag{26}$$

If $S_B > 1$, the cover multiplier vanishes and additionally

$$\max_{v\in C} x_v \le R \min_{v\in C} x_v, \qquad z \le \frac{Rr}{m}. \tag{27}$$

Under the fixed-order singleton tie-break, the selected best singleton belongs to B; this conclusion also holds when $b = 1$.

Proof. For $r > 1$ and $b \ge 2$, strict feasibility and the entropy term give an interior unique optimum with KKT form

$$x_v = \mathrm{sigmoid}((\theta_v + u\mathbf{1}\{v \in B\} - \beta)/\tau), \qquad u, \beta \ge 0.$$

Because B is a prefix and θ is order-aligned, the coordinates are nonincreasing. Since B is a prefix, the earliest singleton maximizer therefore lies in B. Within B, log-odds differ by at most Δ/τ; an odds ratio bounded by R gives the same coordinate-ratio bound. This proves Eq. 26. If $S_B > 1$, complementary slackness gives $u = 0$, so the ratio bound holds over all of C and summing yields Eq. 27. If $r = 1$, the stipulated closure gives $B = C$ and the program is the entropy-regularized simplex. If $b = 1$, its sole coordinate is forced to one and fixed-order tie-breaking selects it. □

Each localization inside B discharges every residual positive edge, so a certified search phase ends at its first positive singleton and there are at most k phases. Within a phase, b never increases. Set

$$a = \frac{2}{5}, \qquad c = 1 + \log a > 0, \qquad \varepsilon = \min\{1/10, h_2(a)/(2\lambda)\}, \qquad D = \lceil R/\varepsilon \rceil. \tag{28}$$

Lemma 11 (Small-coordinate progress)**.** *If $z \le \varepsilon$, the selected action is a proper non-singleton prefix $Q \subsetneq B$ with $\widehat{p}(Q) \in [a, 1-a]$. Moreover:*

1. *a positive response gives $|B_{\mathrm{new}}| \le (1 - c/R)b$;*
2. *a negative response with $S_B = 1$ gives $|B_{\mathrm{new}}| \le (1 - a/R)b$;*
3. *a negative response with $S_B > 1$ gives $|C_{\mathrm{new}}| \le (1 - a/(Rr))m$.*

Proof. The full B has surrogate at least $1 - e^{-S_B} \ge 1 - e^{-1}$. Take the first prefix whose surrogate reaches a. Each increment is at most $z \le 1/10$, so this prefix has surrogate in $[a, 1/2]$. It has at least two elements and is strictly shorter than B. Its batch score is at least $h_2(a)$, while every singleton score is at most $\lambda\varepsilon \le h_2(a)/2$, so the maximizing batch has surrogate in $[a, 1-a]$.

For the selected batch, $\widehat{p}(Q) \le 1 - a$ implies $\sum_Q x_v \le -\log a = 1 - c$, hence $\sum_{B\setminus Q} x_v \ge cS_B$. Combining this with $z \le RS_B/b$ gives $|B \setminus Q| \ge cb/R$, proving the positive-response bound. For a negative response, the union bound gives $a \le \widehat{p}(Q) \le \sum_Q x_v \le |Q|z$. Use $z \le R/b$ when $S_B = 1$ and $z \le Rr/m$ when $S_B > 1$ to obtain the two negative-update bounds. □

Lemma 12 (Large coordinates outside a small bracket)**.** *If $b > D$ and $z > \varepsilon$, then $S_B > 1$ and $m < KD$. If a batch is selected, it has size at most D.*

Proof. If $S_B = 1$, Eq. 26 would imply $z \le R/b < \varepsilon$. Thus $S_B > 1$, and Eq. 27 gives $m < Rr/\varepsilon \le KD$. Every coordinate in B is at least z/R. Let $j_0 = \lceil R \log 2/z \rceil \le D$. A prefix by that index has surrogate at least $1/2$; after the first crossing of $1/2$, prefix entropy decreases. Hence the best eligible prefix has size at most D. □

Full proof of Theorem 5. Classify scored rounds, giving priority to the first applicable class.

1. If $b \le D$, the bracket never grows within a phase and each query reduces it or ends the phase, contributing at most kD rounds.
2. If $b > D$ and $z \le \varepsilon$, positive batches and negative batches with $S_B = 1$ contract b by at least $1 - c/R$. There are at most $(R/c)\log n$ such rounds per phase.
3. If $b > D$, $z \le \varepsilon$, and a negative batch has $S_B > 1$, it contracts m by at least $1 - a/(RK)$. Since m never increases, these rounds contribute at most $(RK/a)\log n$ over the entire execution.
4. For $b > D$ and $z > \varepsilon$, the first negative such round occurs with $m < KD$ by Lemma 12; subsequent negative queries remove at least one active candidate, so there are at most KD in total.
5. A positive singleton ends the phase. A positive batch in the remaining large-coordinate case creates a bracket of size at most D, so there is at most one such round per phase.

Every positive guard starts a phase or itself localizes the final singleton; a subsequent positive guard requires an intervening localization. There are at most k positive guards and one terminal negative guard. Therefore

$$T \le 1 + 2k + (K + k)D + \left(\frac{kR}{c} + \frac{KR}{a}\right)\log n, \tag{29}$$

which implies Eq. 17 for fixed λ. The case $k = 0$ terminates with the first negative guard. □

Proof of Proposition 6. The first full-set guard is positive. Every queried proper prefix misses the last candidate, so the smallest residual positive edge remains the full active set throughout this negative-prefix trajectory. The supplied remaining budget stays K, even though $k = 1$. For $m \geq 10K$, strict concavity and symmetry give the exact optimizer $x_v = K/m$: the entropy objective is increasing up to $1/2$ and fills the available budget.

Writing $x = K/m \leq 0.1$, the continuous prefix length giving surrogate $1/2$ is $q_\star = \log(1/2)/\log(1-x)$. The entropy-maximizing integer prefix is one of its two neighboring integers. A balanced eligible batch has score at least $h_2(0.4) > 0.1$, so it beats all singleton scores. Its size obeys

$$q \leq q_\star + 1 \leq \frac{(\log 2)m}{K} + 1 \leq (\log 2 + 0.1)\frac{m}{K} < 0.8\frac{m}{K}.$$

All these answers are negative, so $m_{t+1} \geq (1 - 0.8/K)m_t$. Hence before the active pool falls below $10K$ the controller makes at least

$$\frac{\log(n/(10K))}{-\log(1 - 0.8/K)} \tag{30}$$

negative scored queries, which is $\Omega(K \log(n/K))$.

For comparison, a legal policy queries the full active set, bisects the current certified positive prefix until one target is identified, executes that singleton, and repeats with a full-active guard when needed. Positive bisection responses leave candidates outside the bracket active for later phases, so the policy remains valid for arbitrary target sets of size at most K. On the specified last-target instance all proper-prefix responses are negative, giving at most $\lceil \log_2 n \rceil + 2$ total queries. □

D.1 Budget-one special case and anchor application

The nested theorem is deliberately parameterized by the supplied budget. In the remaining-budget-one regime a sharper contraction argument is available. Define

$$b = \frac{2}{5}, \qquad c_b = 1 + \log b > 0, \qquad \gamma = \frac{1 + \log b}{R}, \qquad M = \left\lceil \max\left\{10R, \frac{2\lambda R}{h_2(b)}, 2\right\}\right\rceil.$$

Corollary 13 (Budget-one geometric contraction)**.** *Consider a state with remaining target budget one, nonempty residual positive edge family, and active size $m_0 \geq 2$ after the local intersection closure. Apply that closure after every update. Under exact optimization and proposal range at most Δ, Fractional-SL completes in*

$$\left\lceil \frac{[\log(m_0/M)]_+}{-\log(1-\gamma)} \right\rceil + M = O(\log m_0) \tag{31}$$

queries.

In a certified budget-one state every feasible residual hypothesis is a singleton in the intersection of the positive edges. The local intersection closure removes all other elements, after which Eq. 14 reduces to the simplex

$$x_i \in (0,1), \qquad \sum_{i=1}^{m} x_i = 1. \tag{32}$$

The entropy term makes the optimizer unique and interior whenever $m \geq 2$.

Lemma 14 (A fractional prefix contracts both branches)**.** *Consider a budget-one state with $m \geq M$ active candidates and proposal range $\max_i \theta_i - \min_i \theta_i \leq \Delta$. Fractional-SL selects a nontrivial prefix Q whose batch-response surrogate lies in $[b, 1-b]$. Both Q and its complement contain at least γm active candidates. Consequently either response leaves at most $(1-\gamma)m$ candidates.*

Proof. The KKT stationarity equations for Eq. 32 are

$$\theta_i + \tau \log \frac{1 - x_i}{x_i} = \nu. \tag{33}$$

Hence the odds of any two coordinates differ by at most R, which implies $x_{\max} \leq R x_{\min}$. Since the coordinates sum to one,

$$\frac{1}{mR} \leq x_i \leq \frac{R}{m} \qquad (1 \leq i \leq m). \tag{34}$$

Let $q_j = 1 - \prod_{i \le j}(1 - x_i)$ in score order. The full prefix satisfies $q_m \ge 1 - e^{-1} > 1 - b$. Let j be the first index with $q_j \ge b$. Since $x_i \le R/m \le 1/10$, this is a proper nontrivial prefix and $q_j \le b + R/m \le 1/2$. Its batch score is at least $h_2(b)$, while the best singleton score is at most $\lambda R/m \le h_2(b)/2$, so Fractional-SL selects a batch with surrogate in $[b, 1 - b]$.

The union bound and Eq. 34 give $|Q| \ge bm/R$. The upper surrogate bound gives $\sum_{i \in Q} x_i \le -\log b$, hence $\sum_{i \notin Q} x_i \ge 1 + \log b = c_b$, so $|C_t \setminus Q| \ge c_b m/R = \gamma m$. Either response therefore retains at most $(1 - \gamma)m$ candidates. □

Proof of Corollary 13. While the active size exceeds M, Lemma 14 contracts it by a factor at most $1 - \gamma$ per batch. At most the displayed number of rounds reaches size M; thereafter every progress-making prefix or positive-marginal singleton strictly shrinks the active set, so at most M further queries suffice. □

Application to the anchor family. On the anchor state in Eqs. 5–6, the constraints and residual budget imply $x_a = 1$ and $\sum_{i=1}^{n} x_{b_i} = 1$. Every nontrivial prefix contains the certified positive edge $\{a\}$ and is filtered, while the anchor has the largest singleton weight. Fractional-SL therefore localizes a first. The residual problem then has budget one on the n tails, so Corollary 13 gives an $O(\log n)$ continuation for fixed controller parameters and bounded proposal spread.

Bounded spread for the anchor scaling scores. In the scaling experiment the tail scores are equally spaced and $\theta_t(v) = c_s \operatorname{zscore}_{C_t} s(v)$ with fixed c_s. After the anchor is localized, every reachable tail set is a contiguous score interval. For $m \ge 2$ equally spaced scores, the standardized range is

$$\frac{m - 1}{\sqrt{(m^2 - 1)/12}} < \sqrt{12}.$$

Hence the proposal range is uniformly bounded by $\Delta < c_s\sqrt{12}$, independently of n and search depth. With the experimental $c_s = 0.05$, one may take $\Delta < 0.174$.

E Experimental Protocols and Additional Results

Common protocol. All experiments use the deterministic oracle in Eq. 1 and report operational query cost unless an endpoint is named explicitly. The primary candidate family contains every active singleton and every proper prefix of the decreasing score order of size at least two. A full-active query is used only as the presence guard. Entropies are in nats. Fractional-SL uses $\lambda = 1$, $\tau = 0.05$, and proposal scale $c_s = 0.05$ in the primary runs. Batch–singleton ties favor localization; among equally scored singletons, the earliest candidate in the fixed active order is selected. All remaining ties are deterministic. The full configuration fixes every random seed. Reported 95% confidence intervals are $t_{0.975,r-1}$ times the sample standard error for r runs and summarize variation over independently generated instances or hidden targets; exact-DP quantities have no sampling error.

Exact one-step values. Table 2 gives the source values behind Figure 1(a). The single-target examples are analytic. In the anchor rows, $n = 128$ and the tail is uniform. The saturated prefix contains the anchor, whereas the anchor singleton moves a certain target from the core to the operationally localized set. The two actions therefore have the same zero IG and DG, but opposite operational effects.

Exact policy evaluation. For the enumerable anchor experiment, a state stores the localized set and a weighted collection of residual target hypotheses. For a state s, query Q, and answer y, let $s_{Q,y}$ be the exact state update and $p(y \mid s, Q)$ its posterior probability. OSP is computed by the Bellman recurrence below. Optimal decision-tree construction is computationally hard in general (Hyafil & Rivest, 1976); enumeration is feasible here only because these instances are deliberately small, as in Bayesian active-search benchmarks (Garnett et al., 2012).

$$V_{\mathrm{OSP}}(s) = \begin{cases} 0, & s \text{ is terminal,} \\ 1 + \min\limits_{Q \in \mathcal{A}(s)} \sum\limits_{y \in \{0,1\}} p(y \mid s, Q) V_{\mathrm{OSP}}(s_{Q,y}), & \text{otherwise,} \end{cases} \tag{35}$$

Table 2: Exact one-step values. Here p is the positive-response probability and IG uses natural logarithms.

query state	p	IG	DG	localization
balanced single-target batch	.5000	.6931	.0000	.0000
singleton, $p = .9$	.9000	.3251	.9000	.9000
singleton, $p = .99$	.9900	.0560	.9900	.9900
singleton, $p = .999$	.9990	.0079	.9990	.9990
singleton, $p = .9999$	.9999	.0010	.9999	.9999
saturated anchor prefix	1.0000	.0000	.0000	.0000
anchor singleton	1.0000	.0000	.0000	1.0000
tail singleton	.0078	.0457	.0078	.0078

Table 3: Exact expected end-to-end cost on the anchor family, including the three common setup queries. The random policy chooses uniformly among progress-making structured actions.

endpoint	n	OSP	Exact-SL	MaxIG/MaxDG	Random
operational	4	6.500	6.500	6.500	6.531
	6	7.000	7.333	7.500	7.512
	8	7.500	7.500	8.500	8.449
	10	7.800	8.000	9.500	9.352
	12	8.000	8.333	10.500	10.229
logical	4	5.250	6.000	5.250	5.692
	6	6.333	6.667	6.333	6.756
	8	7.000	7.000	7.375	7.727
	10	7.400	7.400	8.400	8.648
	12	7.667	7.667	9.417	9.536

where $\mathcal{A}(s)$ contains all progress-making prefixes and singletons. Replacing the minimum by a fixed selector evaluates Exact-SL, MaxIG, MaxDG, or a uniform random structured policy. The logical endpoint stops when one hypothesis remains; the operational endpoint stops only when every surviving residual hypothesis is empty. Table 3 reports both recurrences.

Reachable-state bridge. Each instance has $n = 8$, exactly two targets, and all pairs initially feasible. For $\eta \in \{0, 2, 5\}$, pair weights satisfy

$$\mu(A) \propto \exp\{\eta \mathbf{1}\{0 \in A\} + \xi_A\}, \qquad \xi_A \overset{\text{iid}}{\sim} \mathcal{N}(0, 0.15^2). \tag{36}$$

Scores are independent standard Gaussians with a shift of two on element zero. A hidden pair is drawn from this prior. A positive full-set query is followed by a uniformly sampled progress-making proper prefix and its realized answer. There are 6–27 remaining hypotheses in the sampled states. Costs in this experiment begin after these two common setup calls; unlike the anchor scaling plot, they are continuation costs.

Thirty states per concentration use seeds $20260919 + 1000\,\mathrm{round}(10\eta + 10) + i$, $0 \le i < 30$. Exact baselines enumerate feasible pairs for probabilities, cores and branch integration. All selectors share the local active-set closure convention; posterior-zero actions may be ignored by exact selectors but do not silently prune the common active prefix family. Fractional-SL reads only the symbolic edges, active set, count bound and scores. Its exact selected-action probability is computed by the evaluation driver after selection, never used by the selector. The updated evaluator applies budget-one edge intersection also after singleton localization.

CoreFirst-Batch takes a core singleton whenever available, otherwise the maximum-entropy batch, and falls back to a singleton if no informative batch exists. CoreFirst-MaxIG differs by allowing every progress-making singleton in the entropy maximization. One-step rollout evaluates every candidate followed by Exact-SL, and repeats this improvement at later states. These are small-state posterior-informed comparisons, not implementations with the same computational cost as Fractional-SL.

Table 4: Mean operational continuation cost divided by fixed-cardinality OSP; parentheses give pointwise 95% Student-t confidence half-widths over 30 states. All policies share the local closure protocol. The guarded row is a semantic-interface diagnostic.

Policy	$\eta = 0$	$\eta = 2$	$\eta = 5$
OSP	1.000 (0.000)	1.000 (0.000)	1.000 (0.000)
One-step rollout	1.036 (0.007)	1.018 (0.003)	1.013 (0.002)
Exact-SL	1.099 (0.011)	1.070 (0.008)	1.057 (0.006)
CoreFirst-Batch	1.115 (0.017)	1.181 (0.031)	1.261 (0.027)
CoreFirst-MaxIG	1.033 (0.009)	1.049 (0.013)	1.160 (0.016)
Fractional-SL, fixed cardinality	1.075 (0.012)	1.124 (0.029)	1.151 (0.064)
Fractional-SL, guarded	1.278 (0.014)	1.354 (0.029)	1.393 (0.070)
MaxIG	1.045 (0.007)	1.065 (0.019)	1.160 (0.016)
MaxDG	1.074 (0.013)	1.112 (0.013)	1.147 (0.011)

Table 5: Paired baseline cost minus fixed-cardinality Fractional-SL cost, in oracle calls (mean $\pm$ pointwise 95% Student-t half-width). Positive values favor Fractional-SL.

Baseline	$\eta = 0$	$\eta = 2$	$\eta = 5$
CoreFirst-Batch	$+0.183 \pm 0.066$	$+0.237 \pm 0.175$	$+0.490 \pm 0.254$
CoreFirst-MaxIG	-0.203 ± 0.070	-0.330 ± 0.155	$+0.074 \pm 0.259$
1-step rollout	-0.185 ± 0.067	-0.464 ± 0.128	-0.542 ± 0.219

The fixed-cardinality variant omits the presence guard because residual presence is known until both targets are localized; it still uses the stated upper-budget fractional relaxation, not an equality-constrained cardinality posterior. The guarded variant retains the full-active presence query whenever no positive edge remains. Both are evaluated on the same fixed-cardinality instances and normalized by the same OSP. Their difference measures the cost of this restricted information interface, not a comparison with an optimal policy under a genuinely unknown target count.

Fractional-SL improves on the batch-only core-first ablation in all three regimes. Posterior-aware CoreFirst-MaxIG and one-step rollout are lower-cost references in some regimes, as shown directly in Table 5. First-action agreement with Exact-SL is 46.7, 56.7, 70.0 percent, respectively; mean selected-action probability errors are 0.231, 0.196, 0.332, and mean exact-score regrets are 0.160, 0.150, 0.233. These diagnostics quantify the approximation gap between fractional action weights and exact posterior control.

Anchor scaling. For each $n \in \{32, 64, 128, 256, 512, 1000, 2000, 5000, 10000\}$, seeds 0–29 draw a uniform hidden tail. The anchor has score five and tail scores decrease linearly from one to zero. The compact initial edges are $\{a\}$ and B_n; every reported query total adds the same three prescribed setup calls. The continuation budget is $\max\{16, 4\lceil \log_2 n \rceil + 8\}$. All 270 runs succeed and localize the anchor first.

The balanced reference localizes the anchor, repeatedly halves the tail pool, and explicitly localizes the final tail. Reference columns below integrate all n hidden-tail positions; raw paired CSV rows instead evaluate the same 30 sampled positions as Fractional-SL. These are different estimands, particularly for the linear-scan baseline. Reference formulas carry no measured runtime; missing timings are recorded as missing values, not zeros. The split-only ablation applies the certified-positive filter and has no admissible batch at the initial anchor state; it terminates without a continuation query and is recorded as a failure.

The setup replay starts with auxiliary controls, executes all three positive queries, and applies the specialized reachability reduction. For four tail sizes and three seeds each, the tests compare every continuation query signature and response against compact initialization. Neither this reduction nor the subsequent budget-one intersection requires general integer closure.

Fractional solver and numerical checks. For disjoint positive edges partitioning the active set into exactly k_t blocks, the cover and upper-budget constraints imply mass one per block. For a

Table 6: Anchor query cost including three prescribed setup calls. Fractional-SL uses 30 sampled tails per size and Student-t intervals; reference columns are analytic expectations over all tail identities, not sample means. Retention is the run-averaged batch survivor fraction.

n	Fractional-SL	Balanced reference	Myopic	Retention
32	10.43 ± 0.44	10.00	20.5	0.568
128	12.60 ± 0.56	12.00	68.5	0.570
512	14.67 ± 0.73	14.00	260.5	0.564
2000	17.20 ± 0.75	15.98	1004.5	0.567
10000	19.63 ± 0.97	18.36	5004.5	0.564

nonsingleton block the unique solution satisfies

$$x_v = \mathrm{sigmoid}((\theta_v - \nu)/\tau), \qquad \sum_{v \text{ in block}} x_v = 1.$$

The implementation solves this scalar equation with 80 bisection steps; a singleton block has coordinate one. An independent constrained-primal test checks the objective against this reduction. This handles every anchor continuation and avoids a degenerate general dual solve. Prefix product probabilities are scanned with vectorized cumulative log-products, preserving the action family and entropy score.

For general overlapping covers, write A for the edge incidence matrix, $u \geq 0$ for cover multipliers, and $\beta \geq 0$ for the budget multiplier:

$$x_v = \mathrm{sigmoid}\left(\frac{\theta_v + (A^\top u)_v - \beta}{\tau}\right). \tag{37}$$

Sparse L-BFGS-B uses 500 iterations initially, function tolerance 10^{-12}, gradient tolerance 10^{-9} and feasibility threshold 5×10^{-6}. A stricter second pass is followed, if necessary, by a nearest-feasible LP and a feasible-start solve of the same convex primal. A directly accepted repair must move each coordinate by at most 10^{-3}. Feasibility checks do not certify optimization error; optimizer status and message are logged separately.

The current anchor sweep records 3198 continuation rounds with maximum cover and budget violations 3.00×10^{-15} and 1.33×10^{-15}. Reachable-state first-action maxima are 4.39×10^{-10} and 1.93×10^{-8}. At $n = 10{,}000$, mean continuation wall time is 0.209 seconds, including 0.0144 seconds for its first solve. This measurement includes action selection and state updates but excludes setup and instance construction. Timing values come from the released run manifest. Query counts are reported separately from this machine-dependent implementation cost.

Matched implementation timing. A separate same-host comparison uses two implementation variants on ten identical 10,000-tail instances (seeds 0–9), with one warm-up and single-thread BLAS settings. Every query/response trajectory is identical. Mean continuation time changes from 0.621 to 0.244 seconds (2.55 times faster on this host); both use 16.40 continuation queries on average. This timing comparison is separate from the 30-run scaling table and does not change query trajectories.

Parameter sensitivity. We vary $\tau \in \{0.02, 0.05, 0.1\}$ and $\lambda \in \{0.5, 1, 2\}$ on the $n = 512$ anchor state, with ten hidden-tail draws per cell and $c_s = 0.05$. Table 7 gives the rerun query counts. All cells terminate, but query cost depends materially on τ: the tested means are about 21–22 queries at $\tau = 0.02$ and about 14 at $\tau = 0.1$. We therefore treat this grid as a robustness check, not evidence of parameter insensitivity.

Full-transcript multi-target protocol. The complete multi-target experiment uses $n = 128$, $K \in \{2, 4, 8\}$, $k \in \{K, K-1\}$ and target-score shifts $\delta \in \{0, 2\}$. Hidden sets are drawn uniformly among size-k subsets. Scores are independent standard Gaussians plus δ on targets; policies receive scores but not membership. There are 20 seeds per cell, generated as $20260921 + i + 1000K + 100(K-k) + 10000\delta$, $0 \leq i < 20$. All policies start with $L = N = \varnothing$ and no positive edge, and share the local response and closure rules. The query cap is $n + 2K + 2$; we report query counts and failure rates. The protocol yields 720 policy runs with aggregate success 100.0%.

Table 7: Mean end-to-end queries at $n = 512$, ten runs per cell, including three setup calls.

$\tau\backslash\lambda$	0.5	1.0	2.0
0.02	20.9	20.6	22.3
0.05	14.6	14.6	14.9
0.10	13.9	13.9	13.9

Table 8: Exploratory full-transcript search with $n = 128$ and 20 independent instances per cell. Costs include all guards and explicit localization; intervals are pointwise Student-t intervals. K is the supplied bound, k the true count, and δ the target-score shift.

K	k	δ	Fractional-SL	Prefix-bisection SL	Localize-only
2	2	0	17.15 ± 1.19	16.10 ± 0.45	78.70 ± 15.86
2	2	2	12.95 ± 0.69	16.40 ± 0.35	15.55 ± 6.92
2	1	0	11.35 ± 1.67	9.40 ± 0.24	128.00 ± 0.00
2	1	2	7.60 ± 0.75	9.40 ± 0.24	128.00 ± 0.00
4	4	0	32.35 ± 1.77	31.25 ± 0.88	101.65 ± 11.26
4	4	2	24.05 ± 1.47	32.30 ± 0.63	33.95 ± 12.43
4	3	0	29.75 ± 2.69	24.95 ± 0.74	128.00 ± 0.00
4	3	2	19.35 ± 1.14	25.60 ± 0.47	128.00 ± 0.00
8	8	0	61.40 ± 2.08	58.75 ± 1.58	112.90 ± 6.40
8	8	2	40.95 ± 1.72	62.70 ± 0.91	48.60 ± 12.30
8	7	0	60.90 ± 3.12	52.95 ± 1.43	128.00 ± 0.00
8	7	2	35.50 ± 1.27	55.70 ± 0.70	128.00 ± 0.00

The prefix-bisection comparator uses a presence guard when necessary and localizes an explicit singleton positive edge when available. Otherwise it finds the shortest score prefix containing a positive edge, which certifies presence in that bracket, and queries its first half (a singleton if needed). It bisects the certified bracket rather than the entire active pool, avoiding a weak rule that can peel one element at a time after positive responses. Score-order singleton search uses the same certified stopping rule. Neither baseline computes a posterior or an exact core. Fractional-SL retains the primary parameters throughout; this sweep is not used for tuning. The maximum logged cover and budget violations are 6.00×10^{-7} and 4.56×10^{-6}; 2 fractional solves report an unsuccessful optimizer status despite the separate feasibility checks.

Score-matched baseline and strict cover ablation. Weighted-Split is a deliberately simple independent baseline. It solves the same entropy-regularized budget-only proposal, identifies the shortest score prefix that contains a residual positive edge, and queries the strict prefix whose cumulative surrogate mass is closest to one half; forced singleton edges are localized immediately. It shares score magnitudes, ordering, closure rules, and stopping semantics with Fractional-SL, but it also changes the action selector, so it is not a one-factor attribution test.

For that attribution test we additionally run No-Cover Fractional-SL. It keeps Fractional-SL's product proxy, score-prefix candidates, certified-positive filter, singleton competition, proposal, closures, and

Table 9: Informative-score multi-target comparison at $n = 128$ with 20 matched runs per cell. "No cover" changes only the positive-edge constraints in Fractional-SL; "Weighted" is the independent half-mass weighted-split baseline. Differences are baseline minus Fractional-SL with pointwise 95% paired Student-t intervals; positive values favor Fractional-SL.

K	k	Fractional	No cover	Δ_{NC}	Weighted	Δ_{W}
2	1	7.60	11.15	$+3.55 \pm 1.76$	7.35	-0.25 ± 0.43
2	2	12.95	17.80	$+4.85 \pm 1.08$	12.50	-0.45 ± 0.80
4	3	19.35	25.35	$+6.00 \pm 1.86$	20.40	$+1.05 \pm 0.54$
4	4	24.05	31.40	$+7.35 \pm 1.33$	24.70	$+0.65 \pm 0.93$
8	7	35.50	42.30	$+6.80 \pm 1.75$	44.20	$+8.70 \pm 1.23$
8	8	40.95	48.70	$+7.75 \pm 1.45$	49.75	$+8.80 \pm 1.32$

Table 10: Mean controller wall time (ms) for the matched runs in Table 9. Timings are same-process diagnostics, not hardware-independent benchmarks.

K	k	Fractional-SL	No cover	Weighted-Split
2	1	18.2	9.0	6.0
2	2	23.3	15.4	6.7
4	3	54.1	21.2	11.0
4	4	46.1	29.5	14.9
8	7	84.0	38.2	29.1
8	8	83.1	43.1	32.3

Table 11: Multi-target scaling at $K = k = 8$ and $\delta = 2$, 20 matched instances per size. All 200 policy runs terminate successfully. Times are mean controller wall time per complete search.

n	Fractional queries	Weighted queries	Paired diff. $\pm 95\%$ CI	F time (s)	W time (s)
64	32.75	44.20	11.45 ± 1.08	0.050	0.025
128	41.40	50.60	9.20 ± 1.55	0.081	0.032
256	49.65	58.40	8.75 ± 1.17	0.117	0.050
512	58.95	65.95	7.00 ± 1.53	0.229	0.099
1,024	68.80	73.55	4.75 ± 2.39	0.373	0.150

stopping rule exactly unchanged; the solver alone receives an empty positive-edge constraint list. At $K = 8, \delta = 2$, 20 matched instances per cardinality give paired No-Cover-minus-Fractional increases of 6.80 ± 1.75 queries for $k = 7$ and 7.75 ± 1.45 for $k = 8$. All runs in the strict comparison terminate successfully; including Weighted-Split, all 120 $K = 8$ policy runs terminate. The released script `experiments/run_strict_cover_ablation.py` reproduces the comparison and records the edge-structure audit.

Because every batch query in this experiment is a prefix of a fixed score order restricted to the current active set, residual positive edges remain prefixes after negative deletions and singleton localizations. They are therefore nested by inclusion, so with $x_v \geq 0$ the full family of constraints $\sum_{v \in P_j} x_v \geq 1$ is equivalent to the constraint for the smallest positive edge. The 120 audited $K = 8$ runs across Fractional-SL, No-Cover Fractional-SL, and Weighted-Split contain 5,228 decision states and zero non-nested states; up to 13 positive edges are stored at once. Accordingly, the current prefix experiments establish the value of the effective certified-set constraint, not of general overlapping/non-laminar cover geometry. Table 9 reports the informative-score comparisons; the released CSVs retain the underlying runs.

Multi-target scaling. To test whether the comparison survives beyond $n = 128$, we fix $K = k = 8$ and $\delta = 2$ and sweep $n \in \{64, 128, 256, 512, 1024\}$ with 20 matched instances per size. Table 11 reports query cost, controller wall time, and success. Both methods terminate on every run. Fractional-SL's query advantage persists throughout the sweep, while Weighted-Split remains computationally cheaper; the gap narrows with n on this fixed-K sequence.

Budget-one response-proxy ablation. When the remaining target budget is one, we compare the product response proxy with $\widehat{p}_{\mathrm{mass}}(Q) = \sum_{v \in Q} x_v$ while changing no other part of the controller. Under constant scores and a uniform single target, the fractional solution is symmetric and the mass proxy produces exact half-mass splits. Under the anchor score gradient, the same coordinates are deliberately proposal-driven rather than calibrated, so half surrogate mass need not mean half true response probability. Table 12 reports both settings and makes this calibration–robustness tradeoff explicit.

Optional graph candidates. The auxiliary graph has 256 tails in two equal blocks, within-block edge probability 0.06, five between-block probabilities, and two anchor edges per block. Within each block a chain also ensures connectivity. Fiedler sweep prefixes and complements are scored by surrogate entropy minus 0.2 times conductance (Fiedler, 1973; Shi & Malik, 2000). The eigensolver start vector, eigenvector sign and candidate order are deterministic; sweep conductances are computed

Table 12: Budget-one response-proxy ablation. "Product" uses $1 - \prod_v (1 - x_v)$; "Mass" uses $\sum_v x_v$. Uniform rows have one uniformly distributed target and constant scores; anchor rows use the paper's score gradient and include the common three-query setup. All runs terminate successfully.

Setting	m	Product queries	Mass queries	Product first split	Mass first split
Uniform	128	8.11	7.48	0.688	0.500
Uniform	1,024	11.75	10.52	0.692	0.500
Uniform	10,000	15.83	14.13	0.693	0.500
Anchor	128	12.35	13.85	–	–
Anchor	1,000	15.59	18.11	–	–
Anchor	10,000	19.63	23.03	–	–

Table 13: Optional graph augmentation: mean calls include setup and are capped at 51, with 20 runs per cell. The last two columns show graph use and augmented-policy success. A capped failure is not a completed-query cost.

λ	p_{out}	Prefix	Prefix+graph	Graph use	Success
0.5	0.0005	13.85	14.85	1.00	100%
0.5	0.001	13.85	15.90	1.00	95%
0.5	0.003	13.85	14.80	1.00	100%
0.5	0.01	13.85	14.70	1.00	100%
0.5	0.03	13.85	14.35	1.00	100%
1.0	0.0005	13.85	13.85	0.00	100%
1.0	0.001	13.85	13.85	0.00	100%
1.0	0.003	13.85	13.85	0.00	100%
1.0	0.01	13.85	13.85	0.00	100%
1.0	0.03	13.85	13.85	0.00	100%

incrementally and checked against direct cuts. The one-step complementarity diagnostic and the full policy comparison are reported separately. Table 13 gives the latter, including graph use and success rates, at $\lambda \in \{0.5, 1\}$ over 20 graphs per cell. At $\lambda = 1$ the two variants have the same mean cost, 13.85 queries. At $\lambda = 0.5$, one of 100 graph-augmented runs fails to terminate within 48 continuation calls (51 with setup). Its recorded cost is capped, not a completed-run cost. A useful graph candidate alone is therefore not evidence of lower end-to-end complexity, and the optional family is not a headline gain.

Reproduction and provenance. The supplementary release separates experiment code and source data from the manuscript build. Claim-bearing data are organized under `results/paper/`; `reproduce_release.py` runs the release audit, regenerates paper artifacts, and exposes the reviewer-focused reruns. The release manifest records source and artifact hashes, and the audit recomputes the strict-cover paired statistics, nested-edge checks, proxy summaries, and scaling summaries from the released CSV files.

F Scope and Extensions

Structural condition. The singleton-increment chain in Theorem 3 is the mechanism that guarantees either a balanced batch or a high-probability singleton. Different restricted action families may admit analogous progress structures. When only an upper bound on target count is known, certified termination also requires an admissible absence certificate; the full-active presence query plays that role here.

Fractional guarantee. Fractional-SL preserves the exact symbolic feasible family while using local closures rather than exhaustive logical inference. The direct multi-target bound in Theorem 5 covers fixed nested-prefix geometry, order-aligned proposal spread, complete prefix candidates, and exact convex optimization. The explicit factor $R = \exp(\Delta/\tau)$ is retained; ordinary z-score normalization does not by itself make R independent of instance size. Appendix D gives the sharper budget-one

result and proves a uniform spread bound for the equally spaced anchor scores. The implementation uses finite solver tolerances; feasibility residuals and optimizer status are reported as numerical diagnostics. General incomparable positive-edge systems and finite optimization error remain outside the direct Fractional-SL theorem.

Setup and action protocol. The anchor constraints can be supplied directly or realized by the prescribed three-query staged construction in Appendix C. Reported anchor query totals charge all three setup calls. The setup is used to establish a reachable state that isolates the masking mechanism; the continuation family is then the common prefix/singleton protocol analyzed in the main text.

Noise. The current model uses deterministic existential feedback. Under noise, a positive observation no longer creates a hard residual edge and a positive singleton no longer justifies deterministic removal. Extending operational unblocking to that setting will require likelihood-weighted residual evidence and confidence-controlled localization. Noisy Bayesian querying and robust interactive cover provide natural starting points (Golovin et al., 2010; Jedynak et al., 2012; Guillory & Bilmes, 2011).

G Extended Related Work

Group testing and constrained pools. The oracle in Eq. 1 begins with classical group testing, from Dorfman's pooled screening to adaptive generalized binary splitting (Dorfman, 1943; Hwang, 1972); Aldridge et al. (2019) give a modern information-theoretic treatment. Graph-constrained group testing restricts pools to paths, connected subgraphs, or other graph-supported sets (Cheraghchi et al., 2012; Karbasi & Zadimoghaddam, 2012; Spang & Wootters, 2019; Sihag et al., 2022). General set-system models instead restrict the possible defective sets (Gonen et al., 2022). Most closely, Nikpey et al. (2024) place an arbitrary distribution over target sets, seek informative pooled tests, and switch to individual tests when such splits are unavailable; their analysis obtains bounds in terms of prior entropy and expected infections. Our Lemma 2 uses the same broad split-or-high-marginal proof motif. The technical distinction is that their pool family is not changed by identifying an infected node, whereas our admissible structured family is part of the state: a posterior-certain target can remain inside every useful batch until its singleton is executed. Moreover, our endpoint charges that execution for every realized target, making k_0 a universal lower bound and yielding the total-query guarantees in Corollary 4. Positive masking appears in this coupling of inference, action geometry, and operational completion.

Hypothesis identification and information gain. Generalized binary search maintains a version space and chooses tests that approximately balance it; logarithmic guarantees follow when the test geometry supports suitable splits (Dasgupta, 2004; Nowak, 2011; Mussmann & Liang, 2018). Adaptive-submodular analyses provide related greedy principles for stochastic partial realizations (Golovin et al., 2010; Golovin & Krause, 2011). These ideas explain the splitting half of our theory. Positive masking exposes the missing half: the version space may already identify a target with probability one while the current structured test family contains no informative batch. The useful action then changes the next test geometry rather than shrinking the current version space.

Decision-aware and constrained sequential design. A broad line of work values observations through downstream decisions rather than posterior entropy alone. Value-of-information methods, interactive submodular cover, decision-region determination, and goal-oriented experimental design all formalize this shift (Krause & Guestrin, 2009; Guillory & Bilmes, 2010; 2011; Javdani et al., 2014; Kandasamy et al., 2019). More recent work makes this perspective computationally practical at larger scale. Huang et al. (2024) amortize Bayesian experimental design for downstream decision utility; Huang et al. (2025) jointly amortize inference and active acquisition; and Guo et al. (2026) combine amortized models with online multistep planning under dynamically evolving design constraints. These works establish that the acquisition problem depends on downstream structure and feasibility. Our contribution isolates a complementary phenomenon with a sharp complexity consequence: an action can have a known response and zero immediate information value, yet be necessary because executing it changes which future structured queries are informative.

Active search and discovery. Active search maximizes the number of valuable points found within a budget; Bayesian dynamic programming yields optimal small-instance policies, while nonmyopic and batch approximations extend the horizon (Garnett et al., 2012; Jiang et al., 2017; 2018). Cost-effective active search studies the cost of reaching a discovery objective (Jiang et al., 2019), and region sensing uses aggregate spatial queries to localize sparse positives (Ma et al., 2017). Recent amortized nonmyopic active search learns to imitate an expensive planning policy and substantially reduces decision-time computation (Nguyen et al., 2024). Our endpoint is stricter than standard discovery-count objectives: every realized target must be operationally localized and absence must be certified. The anchor example shows why immediate discovery value can fail for the same structural reason as information gain.

Algorithmic ingredients. Fractional-SL uses a fractional transversal, drawing on the classical relation between fractional and integral covers (Lovász, 1975; Chvátal, 1979). Here the fractional solution is an action proposal, not a posterior approximation that is assumed calibrated. Exact symbolic constraints carry correctness, while the fractional weights choose between candidate splits and localizations. The optional graph candidates use Fiedler ordering and normalized-cut ideas (Fiedler, 1973; Shi & Malik, 2000). Exact OSP is a small decision-tree dynamic program, used only where enumeration is feasible in view of the hardness of optimal decision-tree construction (Hyafil & Rivest, 1976). Existential labels also connect formally to multiple-instance learning (Dietterich et al., 1997), though our goal is adaptive localization rather than bag-level classification.

Noisy feedback. Noisy twenty-questions policies, noisy Bayesian active learning, and robust interactive cover show how sequential splitting changes under unreliable answers (Jedynak et al., 2012; Golovin et al., 2010; Guillory & Bilmes, 2011). Operational localization adds a further issue: an incorrect singleton decision can alter the future action family. A noisy split–localize theory must therefore price both evidence accumulation and the risk of an incorrect operational transition.