SEMANTIC COVERAGE FOR TEST-TIME REASONING: THE VALUE AND COST OF EARLY INFORMATION

Anonymous authors
Paper under double-blind review

ABSTRACT

An additional reasoning trace can be individually valuable while adding little to the work already done. We study test-time allocation through semantic coverage, an objective that values the useful support a set of outputs provides jointly. Predicting this support before completion may avoid redundant work, but consumes computation that could otherwise be spent completing another trace. In a structured overlap model, we show that optimal coordination need not require a full model of possible outputs: learning which branches share useful contributions can suffice. We characterize how many observations are needed to approach optimal coverage in the worst case. In a screening model, we determine exactly when checking whether a candidate will add something new reduces the average computation needed to obtain a useful contribution. The check need not be perfect, and we derive how much it is worth paying for its ability to distinguish new contributions from duplicates. We then show how repeated checking in a discovery model can obtain every distinct useful contribution with high probability using only one full completion for each. Duplicates still have to be encountered, but need not be fully generated; we characterize the minimum total cost of checks and completions up to constant factors. In multi-answer question answering, output format changes the benefit of coordinated generation: naming answers before explaining them increases its coverage gain. In a controlled study using the same preview-based predictions, selecting unfinished traces for their predicted joint contribution recovers more distinct correct answers on average than individual ranking. In a separate comparison of complete procedures with all online work charged, agreement-filtered answer lists achieve higher mean F1 at lower mean token cost than iterative preview selection on retrieved evidence. These results distinguish improvements in selection from improvements in the overall quality–cost tradeoff.

1 INTRODUCTION

Sampling-based test-time computation gives a language model multiple attempts at a task (Wei et al., 2022; Wang et al., 2023; Yao et al., 2023; Snell et al., 2025). An attempt can be useful on its own yet add little to existing work, repeating an answer or argument without adding evidence. Useful allocation therefore considers both individual promise and joint contribution (Stouras et al., 2026; Zhao et al., 2025; Tu et al., 2026). The completed outputs form a *reasoning portfolio*. Its *semantic coverage* measures how much useful material they add beyond existing work, without counting the same contribution twice.

Because discarding a completed output cannot recover its generation cost, avoiding redundant work requires choosing while attempts are still unfinished. Short previews can help, but consume computation that could instead complete another attempt. We ask when early information can improve the tradeoff between semantic coverage and computational cost. The answer depends on three connected questions: **what information is enough to choose which attempts to complete together, when is that information worth buying, and which expensive work can it replace as discoveries accumulate?** We study these requirements in three models, illustrated in Fig. 1 and developed in Sec. 3.

Learning enough to choose well. A full probabilistic model of what the attempts might produce would let us evaluate which ones to complete together. Learning such a model is a high bar.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

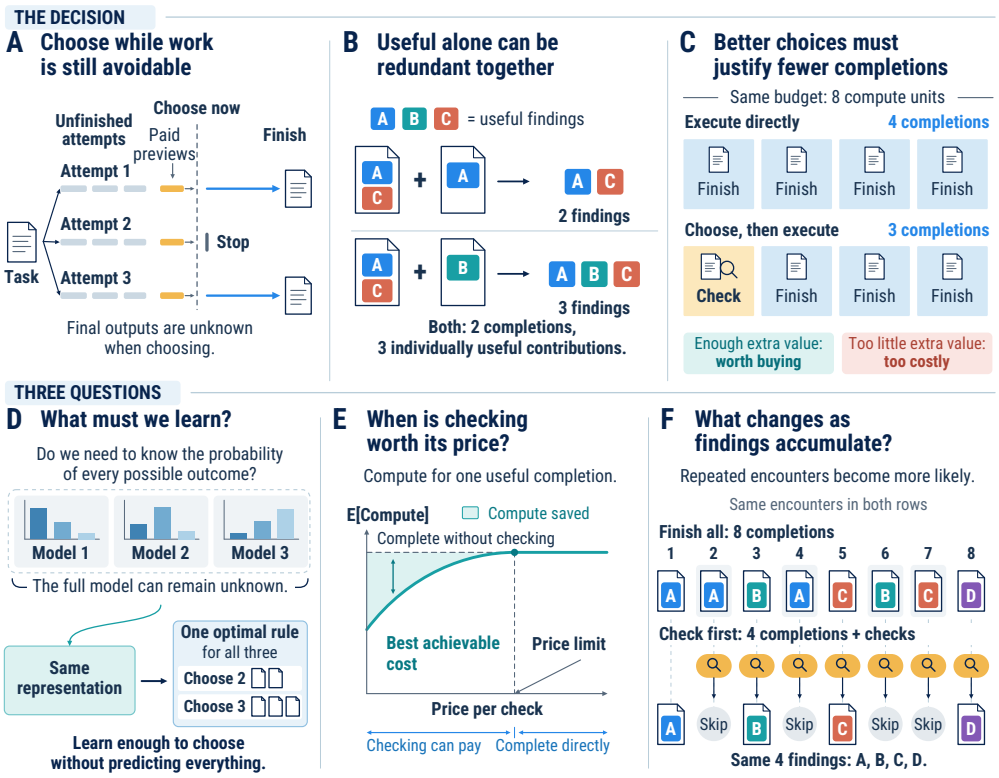


Figure 1: **Useful selection and worthwhile information. A–C: Why choose early?** Previews can guide which attempts to finish (A), and judging their contributions together avoids rewarding the same finding twice (B). Information has an opportunity cost (C): with two compute units per block, checking a larger candidate pool leaves room for three completions instead of four. Better choices must repay that lost completion. **D–F: What can early information achieve?** Under the models in Sec. 3, an overlap order can suffice for optimal joint selection while preserving individual selection probabilities (D). E sketches the price threshold below which checking saves computation. F illustrates a successful path collecting the same four findings with four rather than eight completions. Each magnifier denotes one or more paid, noisy checks; high-probability success requires sufficient budget.

We show that much less information can suffice: in a structured overlap model, an ordering that captures how the attempts overlap enables optimal joint selection without changing how often each attempt is selected. We bound the coverage lost when learning this order and show that the worst-case rate cannot generally be improved. The order can be reused for different batch sizes and selection probabilities, provided the distribution of which attempts contribute is unchanged.

Knowing when checking is worth its cost. Even an informative check can cost more than the work it saves. For noisy checks of a candidate's fixed usefulness, we derive the exact price below which checking reduces the expected cost of obtaining a useful completion. The price depends on how common duplicates are, how well the check distinguishes them from useful candidates, and how expensive completion is. In this model, repeating or combining individually overpriced checks cannot rescue them: more checking helps only when the information is cheap enough to begin with.

Saving generation as discoveries accumulate. As more useful outcomes are found, further attempts increasingly repeat them. If d outcomes are equally likely, completing every attempt takes order $d \log d$ completions on average to find them all. When a candidate's outcome remains fixed during checking, cheap informative but noisy checks can find all d with high probability, given sufficient budget, while using at most d completions. Our cost bounds explain what is saved: duplicates must still be encountered and checked, but need not be fully generated.

Our experiments connect these questions to two practical decisions: how to obtain complementary answers and how much computation to spend predicting them. In multi-answer question answering, naming answers before explaining them helps coordinated generation avoid repeats, showing that the benefit of coordination depends on output format. In a controlled comparison using the same preview-based predictions, selecting unfinished attempts for their combined contribution recovers more distinct correct answers on average than ranking them individually. A separate comparison evaluates complete procedures, charging all online work, including the acquisition of previews. On retrieved evidence, agreement-filtered lists achieve higher mean F1 at lower mean token cost than iterative preview selection.

2 USEFUL ALLOCATION AND THE PRICE OF INFORMATION

2.1 WHAT AN UNFINISHED ACTION CAN CONTRIBUTE

Suppose six analyses are under way for a report, and the remaining budget can finish any two. The report still lacks three equally valuable findings, A, B, and C. To see what makes one choice better, first imagine knowing what each completed analysis would add:

Analysis	1	2	3	4	5	6
Findings added	A, C	A	A, B	B	B, C	C

Finishing analyses 1 and 2 adds A and C, with A repeated. Finishing 1 and 4 adds all three findings. Both choices complete two analyses, but the second supplies more useful material (Fig. 1B).

An *action* is work we could complete, such as finishing an analysis or continuing a language-model prefix. Write $E = \{1, \ldots, M\}$ for the available actions and $S \subseteq E$ for the *batch* we choose within the budget. We choose before knowing the outputs, so we compare batches by their expected contribution given early information, such as previews (Fig. 1A).

Let $\mathcal{U}$ contain the missing useful contributions. The evaluator assigns each u a value $w_u \geq 0$. These weights let some findings matter more than others; all equal one in the example. Let $N_S(u)$ count how many actions in the chosen batch would supply u. Expected *semantic coverage* is

$$G(S) = \sum_{u \in \mathcal{U}} w_u \Pr\{N_S(u) \geq 1\}. \tag{1}$$

The probability asks whether the batch supplies u at least once; further copies earn no extra credit. Thus the example gives $G(\{1, 2\}) = 2$ and $G(\{1, 4\}) = 3$. An action adds new material only where its contributions are not already covered by the batch.

These comparisons keep each action's continuation rule and resource allowance fixed: selecting one action must not change what another would produce. Outputs can still be uncertain and correlated. All probabilities use the information available when choosing the batch. App. A.1 states the full planning assumptions and extends coverage to degrees of support for a finding.

2.2 INDIVIDUAL VALUE, DUPLICATION, AND DISPLACED WORK

A better batch can contain more useful material, repeat less of it, or do both. In the report example, analyses 1 and 2 supply three useful hits—A, C, and A—but cover only two findings because A is repeated. Separating useful hits from repeats explains a selection rule's coverage gain.

Let Q denote a selection rule, which may randomize over batches that meet the budget, and let $J_G(Q) = \mathbb{E}_Q G(S)$ be its expected coverage. Write $v_i = G(\{i\})$ for action i's individual value and $x_i(Q) = \Pr_Q(i \in S)$ for its probability of being selected. The expected value of all useful hits, before removing repeats, is then $\sum_i v_i x_i(Q)$. Let $\mathcal{D}(Q)$ be the expected value of the repeats: for each finding, count every hit after the first and multiply by its weight. Comparing Q with a reference rule Q_0, such as individual ranking or uniform selection, gives

$$\boxed{J_G(Q) - J_G(Q_0) = v^\top [x(Q) - x(Q_0)] + \mathcal{D}(Q_0) - \mathcal{D}(Q).} \tag{2}$$

The first term measures the change in individual useful value; the remaining terms measure repeats avoided. App. A.1 gives the derivation.

This distinction matters even when every action is selected equally often. In our example, choose $\{1, 4\}$, $\{2, 5\}$, or $\{3, 6\}$ with equal probability. Each choice covers all three findings, whereas choosing uniformly from all pairs covers 2.4 findings on average. Both rules select each action with probability $1/3$, so the individual-value term cancels: choosing complementary actions together explains the entire gain. Sec. 3.1 studies this form of coordination. Preview-based selectors (Sec. 4.2) can also change how often each action is selected.

Choosing well from available information is only part of the problem: buying previews leaves less budget for completion (Fig. 1C). Let $J_0(B)$ be a reference's expected final utility at budget B. A procedure that pays a fixed $c_P \leq B$ for information and then executes with $B - c_P$ has expected utility $J_P(B)$, averaging over observations and execution. Its advantage over the reference at that same remaining budget is $\Delta_P = J_P(B) - J_0(B - c_P)$. Comparing complete procedures at B must also account for the reference's value at B versus $B - c_P$:

$$\underbrace{J_P(B) - J_0(B)}_{\text{net acquisition value}} = \underbrace{\Delta_P}_{\text{remaining-budget advantage}} - \underbrace{[J_0(B) - J_0(B - c_P)]}_{\text{displaced reference work}}. \tag{3}$$

Paying for information helps when its remaining-budget advantage exceeds the reference-value difference. That advantage includes the actual procedure's generation, selection, and aggregation. App. G.6 gives a sufficient condition based on a lower bound on Δ_P.

Final utility can be coverage or a metric such as F1, which also penalizes incorrect answers; the preferred output may depend on this choice. Here the total budget is fixed. The next section asks instead how much computation is needed for a target such as the first useful completion.

3 What early information enables

3.1 Learning enough to coordinate

Can we coordinate unfinished work without predicting every possible output? We show that, in a structured model, an ordering that describes overlap suffices for optimal coordination (Fig. 1D).

Return to the six analyses in Sec. 2.1. The contributors are $\{1, 2, 3\}$ for A, $\{3, 4, 5\}$ for B, and $\{5, 6, 1\}$ for C. Each group is consecutive around the circle $1, \ldots, 6$, allowing wraparound. Each group is an *arc*. An order making every group consecutive is a *common circular order*.

To isolate coordination from individual ranking, fix each action's selection probability x_i, with $x \in [0, 1]^M$ and $\sum_i x_i = k$ for an integer batch size k. The decision is which actions to select together. Given the order, systematic sampling spreads the k selections around the circle while preserving each x_i (Madow, 1949). It balances selections within every contributor group: if a group's probabilities sum to at most one, at most one of its contributors is selected; if they sum to at least one, at least one contributor is selected. In particular, a sum of one gives exactly one selection, avoiding both a missed finding and a repeat. App. B.1 gives the construction.

This balance explains why the order is enough. A group's chance of supplying a finding cannot exceed either one or the sum of its members' selection probabilities. The sampler reaches this upper bound for every group at once, so no other rule with the same probabilities can improve coverage. How frequently each group occurs need not be known to coordinate optimally.

The next question is how much information is needed to learn a suitable order. Here each paid observation reveals the *complete contributor set* $B \subseteq E$ of a sampled useful contribution: an observation of finding A reveals all of $\{1, 2, 3\}$. Sampling contributions proportionally to their weights gives $G(S) = W \Pr(S \cap B \neq \varnothing)$, where W is their total weight; for uncertain outputs, the probability also averages over their joint randomness (Tang et al., 2014). An order is *consistent* when it keeps every observed contributor set consecutive. These full observations must be charged at their actual cost; a short text preview does not automatically supply one.

Theorem 1 (learnable structural sufficiency). Suppose an unknown circular order makes B an arc with probability one. For prescribed x with integer sum k, let OPT_x be the best coverage among *all* randomized rules selecting k actions with selection probabilities x, and let $D_M = \max(0, 2M-6)$. The sampler using the true order attains $\mathrm{OPT}_x = W\mathbb{E}_B \min\{1, \sum_{i\in B} x_i\}$.

After $n \geq 1$ independent complete contributor observations from that same distribution, any consistent order yields a sampler $\widehat{Q}_x$ with

$$\mathbb{E}[\mathrm{OPT}_x - J_G(\widehat{Q}_x)] \leq W \min\{1, D_M/(n+1)\}. \tag{4}$$

A corresponding high-probability bound holds simultaneously for all consistent orders and all integer-sum x (App. B.3). For infinitely many M, there are problems satisfying these assumptions and fractional selection probabilities for which every learner incurs expected coverage loss $\Omega(W \min\{1, M/n\})$, even if it may return any randomized rule with those probabilities.

The key bound is structural: at most D_M observations can strictly reduce the family of consistent circular orders (App. B.2). This gives Eq. (4): expected coverage loss from learning rather than knowing the order is $O(WM/n)$, with a matching worst-case rate even for unrestricted randomized rules with those selection probabilities. Once the bound is below W, doubling the observations roughly halves it. The learned order serves different batch sizes and selection probabilities under the same contributor distribution; choosing x additionally requires contribution values and frequencies (App. B.1).

App. B.5 bounds coverage loss for chosen imperfect orders and shows how to assess it using independent held-out complete observations. Fig. 3 in App. B.6 illustrates learning an order in this model.

What simpler summaries miss. Pairwise summaries can miss how three or more actions overlap. In a separate example, two outcome models agree on every individual and pairwise probability yet favor different batches; every rule based only on those summaries loses at least $1/8$ coverage relative to the model-specific optimum in one of the two models (App. A.4).

3.2 When is a diagnostic worth buying?

The ordering result identifies information sufficient for coordination. We now ask when acquiring information saves computation, using a separate screening model. Suppose we need one more finding, but many candidate analyses would repeat material already collected. Before paying to finish one, we can buy a noisy check of whether it would add something new (Fig. 1E). How much is that information worth, and can repeated checking make a costly diagnostic worthwhile?

Consider an unlimited supply of independent candidates, each useful with known probability p. A candidate's usefulness stays fixed throughout checking, and its identifier gives no advance clue to that usefulness. Completion costs $C > 0$, reveals whether the candidate is useful, and supplies a new useful unit if it is. A diagnostic costs $c > 0$ and supplies information only: it produces no usable output and leaves the completion charge unchanged. We seek the lowest expected cost to the first useful completion, among procedures that eventually succeed with probability one. Completing candidates without checks costs C/p on average, because success takes $1/p$ completions on average.

A check's value depends on how well it distinguishes useful candidates from duplicates. Reports take finitely many values, with known distributions P_1 for useful candidates and P_0 for duplicates. Repeated reports are independent draws from the same distribution once the candidate's usefulness is fixed. The report lets us accept a candidate for completion or reject it. The largest gap between useful and duplicate candidates' acceptance probabilities is the *total variation* $\mathrm{TV}(P_1, P_0)$. For a binary flag oriented toward usefulness, this gap is its true-positive rate minus its false-positive rate.

Theorem 2 (exact diagnostic profitability). For $0 < p < 1$, let $T^*(p)$ be the lowest expected cost over all permitted procedures. Then

$$\boxed{T^*(p) < C/p \quad \Longleftrightarrow \quad c < C(1-p)\,\mathrm{TV}(P_1, P_0).} \tag{5}$$

The price boundary has three factors: how expensive completion is (C), how often it would be wasted on a duplicate ($1-p$), and how well the check separates useful candidates from duplicates.

Below the boundary, one check per candidate already suffices to save cost, and repeated checks may save more. At or above it, no strategy can improve on blind completion, even by repeating checks, randomizing decisions, or setting partly checked candidates aside and returning to them. Thus repeated checking can increase savings but cannot extend the profitable price range.

For example, take $p = 0.2$, a true-positive rate of 0.8, and a false-positive rate of 0.3. The boundary is $C \times 0.8 \times (0.8 - 0.3) = 0.4C$. At $c = 0.2C$, checking once and completing positive flags costs $3.75C$ per useful completion on average, versus $5C$ blindly; at or above $0.4C$, no permitted strategy saves cost. App. C.1 derives this cost. Classification accuracy alone misses this distinction: when only one in twenty candidates is useful, always flagging "duplicate" is 95% accurate but supplies no information for selecting one.

The one-check calculation is a special case of Aouad et al. (2026); App. C proves impossibility even for repeated checks and buffered candidates. With a finite choice of diagnostics, an adaptive combination beats blind completion exactly when at least one diagnostic does so alone, provided each report follows its diagnostic's known distribution conditional on usefulness, regardless of earlier checks (App. C.4).

The boundary also gives a purchasing rule: with an independently calibrated binary diagnostic, buy one check only when a lower confidence bound on its discrimination justifies the price; otherwise complete directly (App. C.5). Its guarantee concerns expected cost to the next useful completion under the stated screening model. Usable preview output and reduced completion costs provide additional value (App. C.4).

3.3 Which work can checking replace during discovery?

Imagine collecting one completed analysis for each of d useful findings. Candidates that supply the same finding form a *class*. As the collection grows, more completions repeat findings. Fig. 1F illustrates the opportunity: can cheap checks reject duplicates before we pay for full generation?

Consider $d \geq 2$ classes and an unlimited supply of independent candidates, each equally likely to belong to any class. A candidate keeps its class throughout checking, and its identifier gives no clue to that class. Completion costs C, reveals the class, and supplies its finding. A check costs c and reports only whether the class appears new to the collection. Its known binary report has true-positive rate $a = \Pr(Y = 1 \mid \text{new})$ and false-positive rate $b = \Pr(Y = 1 \mid \text{duplicate})$, with $0 < b < a < 1$. Each check draws fresh noise, independently of earlier reports given the candidate's class and current collection. Checks cannot name or target missing classes.

With r classes missing, a new candidate helps with probability $p_r = r/d$. Blind discovery therefore takes d/r completions on average for the next class and dH_d overall, where $H_d = \sum_{r=1}^{d} 1/r$ grows like $\log d$. Can most of this repeated work instead be paid for at the check price c?

Thm. 2 tells us when checking starts to help. At a stage beginning with fresh, uninspected candidates and $1 \leq r < d$, it reduces expected cost to the next discovery exactly when $c < C(1 - r/d)(a - b)$. Checking can thus become worthwhile as redundancy grows. Initially $r = d$: every candidate is new, so complete directly. Collecting every class also requires controlling total cost and mistakes.

One check can make screening worthwhile while still sending many duplicates to completion as useful candidates become rare. Repeated fresh checks can reduce these costly mistakes (App. D.4). Our construction rejects after modest evidence of duplication but requires stronger evidence before completion (Wald & Wolfowitz, 1948; Malloy et al., 2013). Thm. 3 bounds expected cost and gives simultaneous caps on checks and completions.

Theorem 3 (discovery with two operation costs). Let T_d^* be minimum expected full-coverage cost and $B^*(d, \delta)$ the smallest fixed budget attaining full coverage with probability at least $1 - \delta$, where $0 < \delta < 1/2$. Then

$$\boxed{\begin{array}{c} T_d^* \asymp_{a,b} Cd + \min(c, C) dH_d, \\ B^*(d, \delta) \asymp_{a,b} Cd + \min(c, C) d \log(d/\delta). \end{array}} \tag{6}$$

Here $\asymp_{a,b}$ means matching upper and lower bounds up to constants that depend only on the fixed report rates. Each bound separates two necessary costs: Cd to complete one output per class, and the cost of encountering missing classes among repeated draws. This search can use whichever

operation is cheaper, priced by $\min(c, C)$. The lower bounds hold for every policy in this model. Blind completion costs exactly CdH_d in expectation and order $Cd\log(d/\delta)$ at high confidence.

For $c \leq C$, the high-confidence construction uses at most d completions and $O_{a,b}(d\log(d/\delta))$ checks on every path, including failures. With sufficient budget it collects all classes with probability at least $1-\delta$; on success, completions supply one finding per class. The separate expected-cost construction has no such completion cap. App. D proves both constructions and the lower bounds. Constants worsen as the check loses its ability to distinguish new findings from duplicates. Checks replace expensive duplicate completions, but repeated encounters remain: at fixed positive c/C, total cost still grows as $d\log d$.

The numerical illustration in App. D.5 shows why freshness and budget allocation both matter: reusing one noisy report cannot replace fresh checks, and a policy that caps expensive completions need not minimize total spending. At the minimum budget dC, any check displaces a necessary completion. Savings therefore require enough budget for both information and useful work.

4 Empirical Consequences

Our experiments examine how to obtain complementary answers and how much computation to spend predicting them. We first study how output format affects coordinated generation, then compare selection rules given the same predictions, and finally compare complete procedures with all online work charged. QAMPARI (Amouyal et al., 2023) asks questions with multiple correct entity answers. Coverage counts distinct correct entities; F1 also penalizes errors. We compare F1 within each protocol because the studies score aliases differently (Apps. F.1 and G.2).

4.1 When does coordination improve coverage?

We compare naming an entity before explaining it (*entity first*) with explaining before naming it (*rationale first*). Within each format, QuasiMoTTo lattice sampling (Li et al., 2026) links the token-sampling randomness across outputs while preserving the distribution of each individual output; independent draws in the same implementation and at the same completion cap provide the control. The study pairs 164 QAMPARI questions and eight seeded repetitions on three backbones, with retrieved or answer-bearing passages (*oracle evidence*). Reference answers are used only for evaluation (App. E).

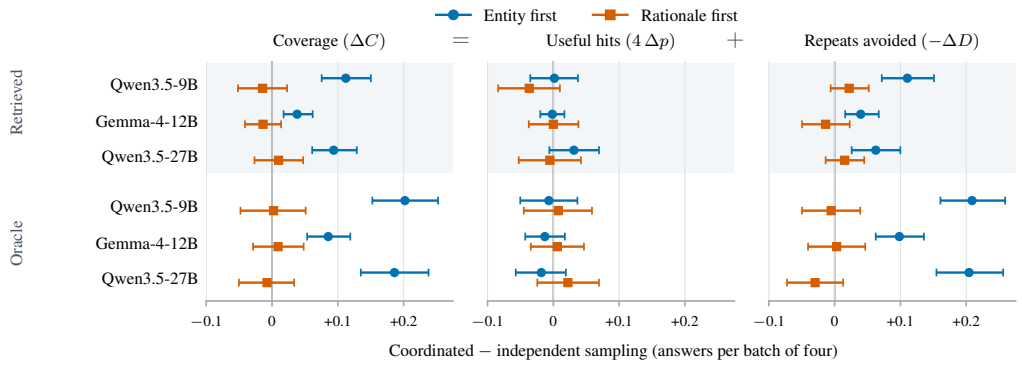


Figure 2: **Naming the answer first helps coordination avoid repeats.** For four outputs, the coordinated-minus-independent coverage change separates into additional useful hits ($4\Delta p$, where p is the fraction matching a gold entity) and repeats avoided. Bars show paired pointwise 95% intervals for these secondary, unadjusted contrasts. Primary tests compare gains between formats; App. E gives their multiplicity adjustment and accounting sensitivity.

Entity-first output increases the coordination gain by 0.052–0.200 distinct correct entities per batch relative to rationale-first output; all six interactions remain significant after Holm correction across these six tests (accounting sensitivity in App. E.3). Fig. 2 shows fewer repeated correct answers in every entity-first condition, while the intervals for changes in total correct-answer counts and for rationale-first coverage include zero.

On 256 AmbigQA training questions (Min et al., 2020), with answer-bearing evidence, entity-first coordination adds 0.111 correct entities per four-output batch [0.073, 0.149] on Qwen3.5-9B and 0.016 [0.000, 0.031] on Gemma-4-12B. These secondary, unadjusted 95% intervals compare coordinated with independent sampling; the Gemma effect does not reach the 5% significance threshold (App. E.5).

4.2 How should shared predictions guide selection?

Given predictions of what unfinished attempts might contribute, does selecting them together recover more distinct correct answers than ranking them individually? On 256 QAMPARI official-test questions and three seeds, each rule selects four of the same eight unfinished attempts (*prefixes*) generated by Qwen3.5-9B. All rules share answer-bearing evidence and two short entity forecasts (*previews*) per prefix (Fig. 1A–B). A predictor fitted on separate development questions combines preview agreement, answer confidence, and verifier judgments of evidence support. Test answers are used only for evaluation. Preview selection uses stochastic text forecasts rather than complete contributor observations or fixed candidate usefulness.

We weight each answer's preview frequency by its estimated correctness to predict its contribution. Additive ranking sums these contributions for each prefix. *Intent deduplication* instead prioritizes prefixes with different greedy answer predictions and fills remaining places. Semantic-Coverage Portfolio Search (SCPS) uses *product-union selection*, adding the prefix predicted to contribute most beyond those already selected. A weaker attempt can help by supplying a missing answer (Eq. (2)).

Each selected prefix supplies four continuations under shared caps. The score approximates expected distinct-answer coverage rather than F1; it does not jointly model a prefix's continuations or shared uncertainty about an entity's correctness (Apps. A.5 and F.1).

In this shared pool, product-union selection recovers a mean of 2.418 distinct correct entities per batch: 0.279 more than additive ranking, 0.096 more than fixed order (the first four prefixes), and 0.040 more than intent deduplication (Tab. F1). Against additive ranking, F1 rises by 3.11 percentage points while precision over individual emitted names, counting repeats, falls by 1.91 points. Tab. F1 reports precision after deduplicating names. Less individually reliable outputs together recover more correct answers, the tradeoff in Eq. (2).

On 78 separate evaluation questions, pairwise overlap correction improves coverage over additive ranking from the same information; both backbone contrasts survive their two-test Holm correction (App. F.1).

Two further controls examine whether predicted overlap is informative and whether selection gains persist across fresh completions. Reassigning predicted answers while preserving individual candidate scores reduces oracle-evidence coverage on two of three backbones; retrieved-evidence intervals include zero (App. F.2.1). A separate 96-question study evaluates sixteen-prefix selection on fresh terminal draws, with one continuation per selected prefix in each draw. Across the tested rules and conditions, exploratory paired 95% intervals for coverage relative to uniform selection include zero or favor uniform (App. F.2.3).

4.3 How should the full budget be spent?

A separate study asks how to spend the entire online budget, including the cost of obtaining early information. A *single answer list* uses greedy generation. *Agreement-filtered lists* retain names appearing in at least two of sixteen independent lists, a threshold selected on development questions. *Memory-conditioned lists* condition further batches on answers already found and return their unfiltered union of distinct names. *Prefix-text diversity* selects four textually dissimilar prefixes without entity previews. *Iterative preview selection* buys two previews for each of sixteen prefixes per admitted round, selects four by product union, and completes each once. Completions update memory; the shared-pool study instead uses eight prefixes, four continuations per selected prefix, and a different predictor (App. G.1).

All procedures receive the same allowance, a ceiling rather than a spending target. We charge uncached input and all output tokens, including acquisition and discarded work; offline calibration and evidence preparation are outside this online allowance.

Table 1: **Coverage, F1, and token expenditure under a shared token ceiling.** Descriptive means on 164 retrieved-evidence QAMPARI questions and three seeded repetitions. Cov. counts distinct correct targets; target F_1 also penalizes errors. Charged tokens are in thousands under a 65,536-token ceiling. Apps. G.2 and G.3 give metric definitions and paired uncertainty.

	Qwen3.5-9B			Gemma-4-12B			Qwen3.5-27B		
Procedure	Cov.	F_1	Tokens	Cov.	F_1	Tokens	Cov.	F_1	Tokens
Single answer list	1.555	.204	3.2	1.518	.212	3.1	2.045	.246	3.2
Agreement-filtered lists	1.955	.207	8.1	1.638	.220	4.3	2.467	.239	10.6
Memory-conditioned lists	2.764	.081	47.2	2.134	.194	11.1	3.793	.071	55.0
Prefix-text diversity	2.169	.174	59.7	2.167	.185	45.7	2.400	.217	57.9
Iterative preview selection	1.596	.189	53.7	1.876	.194	43.9	1.557	.199	49.4

Agreement filtering has higher mean F1 at lower mean spend than preview selection on all three backbones (Tab. 1); Gemma's preview selection covers more targets. The F1 deficits survive Holm adjustment in Gemma's and Qwen27B's prespecified families, but not Qwen9B's post-hoc family (App. G.3). Single lists cost least, with the highest mean F1 on Qwen27B. Memory-conditioned lists have the highest mean coverage on Qwen, and prefix-text diversity on Gemma.

Filtering uses recurrence to retain answers: on the same lists, it preserves 88.0–98.6% of mean correct-target coverage while removing 20.2–67.5% of unmatched names (Tab. G3). Retaining previewed answers can also improve output: at caps of four and eight names, Qwen27B gains F_1 over equally capped completion-only output under both evidence conditions (Tab. G4). Across the tested retention rules and conditions, filtered lists have higher mean F_1 at lower mean cost. Retention changes only the final output; prefixes and previews still account for about 80–85% of mean spend under retrieved evidence.

In executable test generation, a call can start only if the remaining budget covers its maximum charge, even when it later uses fewer tokens. Buying a check can therefore block a later call despite tokens left unused at the end (Fig. 1C; App. H).

5 Related work

Retriever Portfolios learns complementary retrievers, DynaAct combines action utility with embedding diversity, and DeepPrune predicts answer equivalence from prefixes to select traces for voting (Stouras et al., 2026; Zhao et al., 2025; Tu et al., 2026). When overlapping contributor groups can be arranged consecutively in a common circular order, knowing that order supports optimal joint selection at prescribed selection probabilities (Thm. 1). Exact pairwise answer-equivalence comparisons identify answer classes; our obstruction in Sec. 3.1 concerns uncertain, overlapping contributions. Arithmetic Sampling and QuasiMoTTo coordinate generation while preserving each output's distribution (Vilnis et al., 2023; Li et al., 2026); our structural result instead chooses actions from a pool.

Adaptive inference and plan selection guide computation using difficulty estimates, verification, or generated plans (Snell et al., 2025; Cobbe et al., 2021; Heuillet & Peddiraju, 2026). Our acquisition comparison asks whether better decisions repay the execution displaced by obtaining information (Eq. (3) and Sec. 4.3). App. I develops these comparisons and the inspection and sequential-search foundations of our theorems.

6 Discussion and conclusion

Early information must improve which work gets completed enough to repay its cost. The theory identifies sufficient overlap structure and diagnostic conditions; the experiments distinguish using predicted overlap from buying it. Joint selection from shared previews improves mean coverage over individual ranking. On retrieved evidence, agreement-filtered lists have higher mean F1 and lower mean cost than iterative preview selection. Repeating a correct answer adds no coverage, but recurrence helps decide which uncertain answers to retain.

ETHICS STATEMENT

Improving semantic coverage does not ensure that generated content is correct. Errors and biases in the extraction or evaluation of useful contributions can affect what the method selects, and greater coverage may include more unsupported claims. Our evaluation includes incorrect and invalid outputs and measures distinct correct-answer coverage, answer precision, and citation quality. Applications in scientific research or decision support would require further validation against appropriate evidence. Reproduction and deployment must also respect the licenses of the datasets and models used.

REPRODUCIBILITY STATEMENT

Apps. A–D provide the assumptions, proofs, and numerical methods for the theoretical results. Apps. E–H describe the datasets, experimental procedures, evaluation metrics, statistical comparisons, and computational budgets. Code will be released upon acceptance.

AI-USE STATEMENT

We used generative AI for writing and editing, developing and refining results and proofs, literature search, coding assistance, and preliminary manuscript review. The authors take responsibility for the final content of the paper.

REFERENCES

Juan Altmayer Pizzorno and Emery D. Berger. CoverUp: Effective High Coverage Test Generation for Python. *Proceedings of the ACM on Software Engineering*, 2:FSE128:2897–FSE128:2919, 2025. doi: 10.1145/3729398. URL https://dl.acm.org/doi/10.1145/3729398.

Samuel Amouyal, Tomer Wolfson, Ohad Rubin, Ori Yoran, Jonathan Herzig, and Jonathan Berant. QAMPARI: A Benchmark for Open-domain Questions with Many Answers. In Sebastian Gehrmann, Alex Wang, João Sedoc, Elizabeth Clark, Kaustubh Dhole, Khyathi Raghavi Chandu, Enrico Santus, and Hooman Sedghamiz (eds.), *Proceedings of the Third Workshop on Natural Language Generation, Evaluation, and Metrics (GEM)*, pp. 97–110, Singapore, 2023. Association for Computational Linguistics. URL https://aclanthology.org/2023.gem-1.9/.

Ali Aouad, Jingwei Ji, and Yaron Shaposhnik. The Pandora's Box Problem with Sequential Inspections. *Operations Research*, 74(4):2179–2195, 2026. ISSN 0030-364X. doi: 10.1287/opre.2024.0733. URL https://pubsonline.informs.org/doi/10.1287/opre.2024.0733.

Jaime Carbonell and Jade Goldstein. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '98, pp. 335–336, New York, NY, USA, 1998. Association for Computing Machinery. ISBN 978-1-58113-015-7. doi: 10.1145/290941.291025. URL https://dl.acm.org/doi/10.1145/290941.291025.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training Verifiers to Solve Math Word Problems. arXiv:2110.14168, 2021. URL http://arxiv.org/abs/2110.14168.

Gordon V. Cormack, Charles L A Clarke, and Stefan Buettcher. Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '09, pp. 758–759, New York, NY, USA, 2009. Association for Computing Machinery. ISBN 978-1-60558-483-6. doi: 10.1145/1571941.1572114. URL https://dl.acm.org/doi/10.1145/1571941.1572114.

Arghavan Moradi Dakhel, Amin Nikanjam, Vahid Majdinasab, Foutse Khomh, and Michel C. Desmarais. Effective test generation using pre-trained Large Language Models and mutation testing. *Information and Software Technology*, 171:107468, 2024. ISSN 0950-5849. doi: 10.1016/j.infsof.2024.107468. URL https://www.sciencedirect.com/science/article/pii/S0950584924000739.

Marina Drygala, Silvio Lattanzi, Andreas Maggiori, Miltiadis Stouras, Ola Svensson, and Sergei Vassilvitskii. Data-Driven Solution Portfolios. In Raghu Meka (ed.), *16th Innovations in Theoretical Computer Science Conference (ITCS 2025)*, volume 325 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pp. 46:1–46:15, Dagstuhl, Germany, 2025. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. ISBN 978-3-95977-361-4. doi: 10.4230/LIPIcs.ITCS.2025.46. URL https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.ITCS.2025.46.

Bradley Efron and R. J. Tibshirani. *An Introduction to the Bootstrap*. Chapman and Hall/CRC, New York, 1994. ISBN 978-0-429-24659-3. doi: 10.1201/9780429246593.

Adam N. Elmachtoub and Paul Grigas. Smart "Predict, then Optimize". *Management Science*, 68 (1):9–26, 2022. ISSN 0025-1909. doi: 10.1287/mnsc.2020.3922. URL https://pubsonline.informs.org/doi/10.1287/mnsc.2020.3922.

Simon D. Fink, Matthias Pfretzschner, and Ignaz Rutter. Experimental Comparison of PC-Trees and PQ-Trees. *ACM Journal of Experimental Algorithmics*, 28:1.10:1–1.10:24, 2023. ISSN 1084-6654. doi: 10.1145/3611653. URL https://dl.acm.org/doi/10.1145/3611653.

Yichao Fu, Xuewei Wang, Hao Zhang, Yuandong Tian, and Jiawei Zhao. Deep Think with Confidence. *International Conference on Learning Representations*, 2026:94355–94377, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/9825728534085426218550 0e59bc0f28-Abstract-Conference.html.

Tianyu Gao, Howard Yen, Jiatong Yu, and Danqi Chen. Enabling Large Language Models to Generate Text with Citations. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 6465–6488, Singapore, 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.398. URL https://aclanthology.org/2023.emnlp-main.398/.

Gemma Team, Sherif El Abd, Vaibhav Aggarwal, Robin Algayres, Alek Andreev, Olivier Bachem, Ian Ballantyne, Cormac Brick, Victor Cărbune, Michelle Casbon, Mayank Chaturvedi, Aditya Chawla, Victor Cotruta, Alice Coucke, Phil Culliton, Robert Dadashi, Lucas Dixon, Mohamed Elhawaty, Utku Evci, Clément Farabet, Johan Ferret, Filippo Galgani, Sertan Girgin, Jean-Bastien Grill, Maarten Grootendorst, Jiaxian Guo, Cassidy Hardin, Yanzhang He, Steven M. Hernandez, Omri Homburger, Léonard Hussenot, Juyeong Ji, Armand Joulin, Aishwarya Kamath, Parnian Kassraie, Olivier Lacombe, Preethi Lahoti, Gaël Liu, Gus Martins, Luciano Martins, Tatiana Matejovicova, Ramona Merhej, Nikola Momchev, Sneha Mondal, Ryan Mullins, Sindhu Raghuram Panyam, Shreya Pathak, Sarah Perrin, André Susano Pinto, Etienne Pot, Angéline Pouget, Alexandre Ramé, Sabela Ramos, Douglas Reid, David Rim, Morgane Rivière, Karsten Roth, Louis Rouillard, Omar Sanseviero, Pier Giuseppe Sessa, Shane Settle, Danila Sinopalnikov, Sara Smoot, Piotr Stanczyk, Andreas Steiner, Lawrence Stewart, Ilya Tolstikhin, Michael Tschannen, Anton Tsitsulin, Nino Vieillard, Renjie Wu, Pingmei Xu, Haichuan Yang, Edouard Yvinec, Biao Zhang, Li Zhang, Joe Zou, Nicolas Aagnes, Abdelrahman Abdelhamed, Jakub Adamek, Shivani Agrawal, Shubham Agrawal, Ibrahim Alabdulmohsin, Jean Baptiste Alayrac, Uri Alon, Chandramouli Amarnath, Ankesh Anand, Chrysovalantis Anastasiou, Setareh Ariafar, François-Xavier Aubet, Kyriakos Axiotis, Federico Barbero, Joelle Barral, Alexei Bendebury, Urs Bergmann, Stanley Bileschi, Kat Black, Mathieu Blondel, Sebastian Borgeaud, Arthur Bražinskas, Ryan Burnell, Robert Busa-Fekete, Mu Cai, Daniele Calandriello, Glenn Cameron, Charlotte Caucheteux, Rahma Chaabouni, Garima Chadha, Jetha Chan, Blake Jianhang Chen, Jesse Chen, Lin Chen, Xu Chen, Derek Cheng, Tzu-hsiang Chien, Nikolai Chinaev, Yi Chou, Zhaohui Chu, Benjamin Coleman, Pooja Consul, Sam Conway-Rahman, Scott Crowell, Dylan Cutler, Vivek Dani, Samira Daruki, Anil Das, Daniel Deutsch, Nishanth Dikkala, Li Ding, Qiuhan Ding, Shenil Dodhia, Konstantin Donhauser, Tulsee Doshi, Anca Dragan, Alex Druinsky, Sahil Dua, Zoltan Egyed, Danielle Eisenbud, Daniel Eppens, Cindy Fan, Bahare Fatemi, Yassir Fathullah, Vlad Feinberg, Milen Ferev, Sebastian Flennerhag, Takumi Fujimoto, João Gabriel Oliveira, Isaac Galatzer-Levy, João Gante, Simon Geisler, Soham Ghosal, Antonious M. Girgis, Tamara von Glehn, Alec Go, Alhaad Gokhale, Alex Grills, Yiming Gu, Mayank Gupta, Pramod Gupta, Guru Guruganesh, Raia Hadsell, Hamza Harkous, Jitendra Harlalka, Demis Hassabis, Anja Hauth, Joe Heyward, Arian Hosseini, Chih-Yang Hsia, I.-Hung Hsu, Xiaopeng Huang, Yangsibo Huang, Kevin Hui, Adrian Hutter, Te I, Fotis Iliopoulos, Advait Jain, Ganesh Jawahar, Ziwei Ji, Qilin Jin, Melvin Johnson, Kandarp Joshi, Arun Kandoor, Wang-Cheng Kang, Koray Kavukcuoglu, Mehran Kazemi, Kathleen Kenealy, Amr Khalifa, Phoebe Kirk, Ivan Korotkov, Suraj Kothawade, Vitaly Kovalev, Neel Kovelamudi, Adam Kraft, Ravin Kumar, Vivek Kumar, Harish Kuppam, Justin Lannin, Chen-Yu Lee, Seungji Lee, Dmitry Lepikhin, Alon Levkovitch, Dongdong Li, Qiujia Li, Valentin Liévin, Ethan Lin, Ziqian Lin, Casper Liu, Tianlin Liu, Tianqi Liu, Xin Liu, Ivan Lobov, Mayank Lunayach, Min Ma, Gagan Madan, Andrii Maksai, Eric Malmi, Michal Matuszak, Daniel McDuff, Gaurav Menghani, Maciej Mikuła, Daniil Mirylenka, Karolis Misiunas, Vedant Misra, Andreea Mitran, Kareem Mohamed, Maksim Mukha, Eric Noland, James O'Donnell, Brendan O'Donoghue, Kate Olszewska, Bernett Orlando, Wanqiong Pan, Rina Panigrahy, Unnati Parekh, Nicolas Perez-Nieves, Chunjong Park, Eric Paskie, Liqian Peng, Bryce Petrini, Slav Petrov, Jonas Pfeiffer, Bilal Piot, Martyna Plomecka, Siim Poder, Octavio Ponce, Arijit Pramanik, David Racz, Anish Rajan, Michelle Ramanovich, Anand Rao, Marvin Ritter, Vitor Rodrigues, Evan Rosen, Mikołaj Rybiński, Noveen Sachdeva, Michaël E. Sander, Rohit Sathyanarayana, Sagar Savla, Samuel Schmidgall, Tal Schuster, George Scrivener, Benoit Seguin, Andrew Sellergren, Aliaksei Severyn, Izhak Shafran, Dhruv Shah, Bobak Shahriari, Yuan Shangguan, Ashish Shenoy, Pradeep Shenoy, Rakesh Shivanna, Pauline Sho, Lucas Spangher, Wojciech Stokowiec, Tim Strother, Yao Su, Yinghao Sun, Mukund Sundararajan, Andrea Tacchetti, Mor Hazan Taege,

Pouya Tafti, Jean Tarbouriech, Chetan Tekur, Shantanu Thakoor, Rahul Thapa, Madeleine Traverse, Lenart Treven, Tao Tu, Chien Te Tung, Çağlar Ünlü, Petar Veličković, Malini Pooni Venkat, Sagar Gubbi Venkatesh, Vidya Venkiteswaran, Francesco Visin, Alex Vitvitskyi, Kiran Vodrahalli, Weiyi Wang, Xin Wang, Tris Warkentin, Jan Wassenberg, John Wieting, Cindy Wu, Lechao Xiao, Hao Xu, Yuhui Xu, Fuzhao Xue, Arun Yadav, Jun Yan, Antoine Yang, Lin Yang, Ming-Hsuan Yang, Ziyu Ying, Jae Hyeon Yoo, Morteza Zadimoghaddam, Sajjad Zafar, Fred Zhang, Jiageng Zhang, Jianyi Zhang, Xiaofan Zhang, Chao Zhao, David Zhou, and Chen Zou. Gemma 4 Technical Report. arXiv:2607.02770, 2026. URL http://arxiv.org/abs/2607.02770.

D. Golovin and A. Krause. Adaptive Submodularity: Theory and Applications in Active Learning and Stochastic Optimization. *Journal of Artificial Intelligence Research*, 42:427–486, 2011. ISSN 1076-9757. doi: 10.1613/jair.3278. URL https://www.jair.org/index.php/jair/article/view/10731.

Anupam Gupta, Haotian Jiang, Ziv Scully, and Sahil Singla. The Markovian Price of Information. In Andrea Lodi and Viswanath Nagarajan (eds.), *Integer Programming and Combinatorial Optimization*, pp. 233–246, Cham, 2019. Springer International Publishing. ISBN 978-3-030-17953-3. doi: 10.1007/978-3-030-17953-3_18.

Steve Hanneke. Refined Error Bounds for Several Learning Algorithms. *Journal of Machine Learning Research*, 17(135):1–55, 2016. ISSN 1533-7928. URL http://jmlr.org/papers/v17/15-655.html.

Nicholas Hay, Stuart Russell, David Tolpin, and Solomon Eyal Shimony. Selecting Computations: Theory and Applications. arXiv:1207.5879, 2012. URL http://arxiv.org/abs/1207.5879.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. DEBERTA: DECODING-ENHANCED BERT WITH DISENTANGLED ATTENTION. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=XPZIaotutsD.

Maxime Heuillet and Sharadind Peddiraju. CIPHER: A Decoupled Exploration-Selection Framework for Test-Time Scaling of Data Science Agents. arXiv:2607.14386, 2026. URL http://arxiv.org/abs/2607.14386.

Wassily Hoeffding. Probability Inequalities for Sums of Bounded Random Variables. *Journal of the American Statistical Association*, 58(301):13–30, 1963. ISSN 0162-1459. doi: 10.1080/01621459.1963.10500830. URL https://doi.org/10.1080/01621459.1963.10500830.

Sture Holm. A Simple Sequentially Rejective Multiple Test Procedure. *Scandinavian Journal of Statistics*, 6(2):65–70, 1979. ISSN 0303-6898. URL https://www.jstor.org/stable/4615733.

Wen-Lian Hsu and Ross M. McConnell. PC trees and circular-ones arrangements. *Theoretical Computer Science*, 296(1):99–116, 2003. ISSN 0304-3975. doi: 10.1016/S0304-3975(02)00435-8. URL https://www.sciencedirect.com/science/article/pii/S0304397502004358.

Jingkai Huang, Will Ma, and Zhengyuan Zhou. Optimal Bayesian Stopping for Efficient Inference of Consistent LLM Answers. arXiv:2602.05395, 2026. URL http://arxiv.org/abs/2602.05395.

Kush Jain, Gabriel Synnaeve, and Baptiste Roziere. TestGenEval: A Real World Unit Test Generation and Test Completion Benchmark. *International Conference on Learning Representations*, 2025:14947–14999, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/26ded5c8ee8ec1bc4caced4e1c9b1584-Abstract-Conference.html.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. Semantic Uncertainty: Linguistic Invariances for Uncertainty Estimation in Natural Language Generation. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=VD-AYtP0dve.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. Efficient Memory Management for Large Language Model Serving with PagedAttention. In *Proceedings of the 29th Symposium on Operating Systems Principles*, SOSP '23, pp. 611–626, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 979-8-4007-0229-7. doi: 10.1145/3600006.3613165. URL https://dl.acm.org/doi/10.1145/3600006.3613165.

Philippe Laban, Alexander Fabbri, Caiming Xiong, and Chien-Sheng Wu. Summary of a Haystack: A Challenge to Long-Context LLMs and RAG Systems. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 9885–9903, Miami, Florida, USA, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.552. URL https://aclanthology.org/2024.emnlp-main.552/.

Caroline Lemieux, Jeevana Priya Inala, Shuvendu K. Lahiri, and Siddhartha Sen. CodaMosa: Escaping Coverage Plateaus in Test Generation with Pre-trained Large Language Models. In *2023 IEEE/ACM 45th International Conference on Software Engineering (ICSE)*, pp. 919–931, 2023. doi: 10.1109/ICSE48619.2023.00085. URL https://ieeexplore.ieee.org/document/10172800.

Michael Y. Li, Anthony Zhan, Kanishk Gandhi, Noah D. Goodman, and Emily B. Fox. Quasi-MoTTo: Quasi-Monte Carlo Test-Time Scaling. arXiv:2607.01179, 2026. URL http://arxiv.org/abs/2607.01179.

Hui Lin and Jeff Bilmes. A Class of Submodular Functions for Document Summarization. In Dekang Lin, Yuji Matsumoto, and Rada Mihalcea (eds.), *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pp. 510–520, Portland, Oregon, USA, 2011. Association for Computational Linguistics. URL https://aclanthology.org/P11-1052/.

Mingchen Ma and Christos Tzamos. Buying Information for Stochastic Optimization. In *Proceedings of the 40th International Conference on Machine Learning*, pp. 23388–23411. PMLR, 2023. URL https://proceedings.mlr.press/v202/ma23g.html.

William G. Madow. On the Theory of Systematic Sampling, II. *The Annals of Mathematical Statistics*, 20(3):333–354, 1949. ISSN 0003-4851, 2168-8990. doi: 10.1214/aoms/1177729988. URL https://projecteuclid.org/journals/annals-of-mathematical-statistics/volume-20/issue-3/On-the-Theory-of-Systematic-Sampling-II/10.1214/aoms/1177729988.full.

Matthew L. Malloy, Gongguo Tang, and Robert D. Nowak. The Sample Complexity of Search Over Multiple Populations. *IEEE Transactions on Information Theory*, 59(8):5039–5050, 2013. ISSN 1557-9654. doi: 10.1109/TIT.2013.2258071. URL https://ieeexplore.ieee.org/document/6497615.

Sewon Min, Julian Michael, Hannaneh Hajishirzi, and Luke Zettlemoyer. AmbigQA: Answering Ambiguous Open-domain Questions. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu (eds.), *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 5783–5797, Online, 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.466. URL https://aclanthology.org/2020.emnlp-main.466/.

G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher. An analysis of approximations for maximizing submodular set functions—I. *Mathematical Programming*, 14(1):265–294, 1978. ISSN 1436-4646. doi: 10.1007/BF01588971. URL https://doi.org/10.1007/BF01588971.

Jianmo Ni, Chen Qu, Jing Lu, Zhuyun Dai, Gustavo Hernandez Abrego, Ji Ma, Vincent Zhao, Yi Luan, Keith Hall, Ming-Wei Chang, and Yinfei Yang. Large Dual Encoders Are Generalizable Retrievers. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang (eds.), *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 9844–9855, Abu Dhabi, United Arab Emirates, 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.emnlp-main.669. URL https://aclanthology.org/2022.emnlp-main.669/.

OpenAI, Sandhini Agarwal, Lama Ahmad, Jason Ai, Sam Altman, Andy Applebaum, Edwin Arbus, Rahul K. Arora, Yu Bai, Bowen Baker, Haiming Bao, Boaz Barak, Ally Bennett, Tyler Bertao, Nivedita Brett, Eugene Brevdo, Greg Brockman, Sebastien Bubeck, Che Chang, Kai Chen, Mark Chen, Enoch Cheung, Aidan Clark, Dan Cook, Marat Dukhan, Casey Dvorak, Kevin Fives, Vlad Fomenko, Timur Garipov, Kristian Georgiev, Mia Glaese, Tarun Gogineni, Adam Goucher, Lukas Gross, Katia Gil Guzman, John Hallman, Jackie Hehir, Johannes Heidecke, Alec Helyar, Haitang Hu, Romain Huet, Jacob Huh, Saachi Jain, Zach Johnson, Chris Koch, Irina Kofman, Dominik Kundel, Jason Kwon, Volodymyr Kyrylov, Elaine Ya Le, Guillaume Leclerc, James Park Lennon, Scott Lessans, Mario Lezcano-Casado, Yuanzhi Li, Zhuohan Li, Ji Lin, Jordan Liss, Lily (Xiaoxuan) Liu, Jiancheng Liu, Kevin Lu, Chris Lu, Zoran Martinovic, Lindsay McCallum, Josh McGrath, Scott McKinney, Aidan McLaughlin, Song Mei, Steve Mostovoy, Tong Mu, Gideon Myles, Alexander Neitz, Alex Nichol, Jakub Pachocki, Alex Paino, Dana Palmie, Ashley Pantuliano, Giambattista Parascandolo, Jongsoo Park, Leher Pathak, Carolina Paz, Ludovic Peran, Dmitry Pimenov, Michelle Pokrass, Elizabeth Proehl, Huida Qiu, Gaby Raila, Filippo Raso, Hongyu Ren, Kimmy Richardson, David Robinson, Bob Rotsted, Hadi Salman, Suvansh Sanjeev, Max Schwarzer, D. Sculley, Harshit Sikchi, Kendal Simon, Karan Singhal, Yang Song, Dane Stuckey, Zhiqing Sun, Philippe Tillet, Sam Toizer, Foivos Tsimpourlas, Nikhil Vyas, Eric Wallace, Xin Wang, Miles Wang, Olivia Watkins, Kevin Weil, Amy Wendling, Kevin Whinnery, Cedric Whitney, Hannah Wong, Lin Yang, Yu Yang, Michihiro Yasunaga, Kristen Ying, Wojciech Zaremba, Wenting Zhan, Cyril Zhang, Brian Zhang, Eddie Zhang, and Shengjia Zhao. Gpt-oss-120b & gpt-oss-20b Model Card. arXiv:2508.10925, 2025. URL http://arxiv.org/abs/2508.10925.

Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026a. URL https://qwen.ai/blog?id=qwen3.5.

Qwen Team. Qwen3.6-35B-A3B: Agentic coding power, now open to all, April 2026b. URL https://qwen.ai/blog?id=qwen3.6-35b-a3b.

Stephen Robertson and Hugo Zaragoza. The Probabilistic Relevance Framework: BM25 and Beyond. *Foundations and Trends in Information Retrieval*, 3(4):333–389, 2009. ISSN 1554-0669. doi: 10.1561/1500000019. URL https://doi.org/10.1561/1500000019.

Sven Schmit, Virag Shah, and Ramesh Johari. Optimal Testing in the Experiment-rich Regime. In *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, pp. 626–633. PMLR, 2019. URL https://proceedings.mlr.press/v89/schmit19a.html.

Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling LLM Test-Time Compute Optimally Can be More Effective than Scaling Parameters for Reasoning. *International Conference on Learning Representations*, 2025:10131–10165, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/1b623663fd9b874366f3ce019fdfdd44-Abstract-Conference.html.

Miltiadis Stouras, Vincent Cohen-Addad, Silvio Lattanzi, and Ola Svensson. Retriever Portfolios: A Principled Approach to Adaptive RAG. In *Forty-Third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=NyGHaciFrK.

Youze Tang, Xiaokui Xiao, and Yanchen Shi. Influence maximization: Near-optimal time complexity meets practical efficiency. In *Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data*, SIGMOD '14, pp. 75–86, New York, NY, USA, 2014. Association for Computing Machinery. ISBN 978-1-4503-2376-5. doi: 10.1145/2588555.2593670. URL https://dl.acm.org/doi/10.1145/2588555.2593670.

Philip Thomas, Georgios Theocharous, and Mohammad Ghavamzadeh. High Confidence Policy Improvement. In *Proceedings of the 32nd International Conference on Machine Learning*, pp. 2380–2388. PMLR, 2015. URL https://proceedings.mlr.press/v37/thomas15.html.

Shangqing Tu, Yaxuan Li, Yushi Bai, Lei Hou, and Juanzi Li. DeepPrune: Parallel Scaling without Inter-trace Redundancy. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 13389–13403,

San Diego, California, United States, 2026. Association for Computational Linguistics. ISBN 979-8-89176-395-1. doi: 10.18653/v1/2026.findings-acl.656. URL https://aclanthology.org/2026.findings-acl.656/.

Luke Vilnis, Yury Zemlyanskiy, Patrick Murray, Alexandre Tachard Passos, and Sumit Sanghai. Arithmetic Sampling: Parallel Diverse Decoding for Large Language Models. In *Proceedings of the 40th International Conference on Machine Learning*, pp. 35120–35136. PMLR, 2023. URL https://proceedings.mlr.press/v202/vilnis23a.html.

Willem Waegeman, Krzysztof Dembczyński, Arkadiusz Jachnik, Weiwei Cheng, and Eyke Hüllermeier. On the Bayes-Optimality of F-Measure Maximizers. *Journal of Machine Learning Research*, 15(103):3513–3568, 2014. ISSN 1533-7928. URL http://jmlr.org/papers/v15/waegeman14a.html.

A. Wald and J. Wolfowitz. Optimum Character of the Sequential Probability Ratio Test. *The Annals of Mathematical Statistics*, 19(3):326–339, 1948. ISSN 0003-4851, 2168-8990. doi: 10.1214/aoms/1177730197. URL https://projecteuclid.org/journals/annals-of-mathematical-statistics/volume-19/issue-3/Optimum-Character-of-the-Sequential-Probability-Ratio-Test/10.1214/aoms/1177730197.full.

Xinglin Wang, Shaoxiong Feng, Yiwei Li, Peiwen Yuan, Yueqi Zhang, Chuyi Tan, Boyuan Pan, Yao Hu, and Kan Li. Make Every Penny Count: Difficulty-Adaptive Self-Consistency for Cost-Efficient Reasoning. In Luis Chiruzzo, Alan Ritter, and Lu Wang (eds.), *Findings of the Association for Computational Linguistics: NAACL 2025*, pp. 6919–6932, Albuquerque, New Mexico, 2025. Association for Computational Linguistics. ISBN 979-8-89176-195-7. doi: 10.18653/v1/2025.findings-naacl.383. URL https://aclanthology.org/2025.findings-naacl.383/.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-Consistency Improves Chain of Thought Reasoning in Language Models. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=1PL1NIMMrw.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. In *Advances in Neural Information Processing Systems*, volume 35, pp. 24824–24837. Curran Associates, Inc., 2022. doi: 10.52202/068431-1800. URL https://proceedings.neurips.cc/paper/2022/hash/9d5609613524ecf4f15af0f7b31abca4-Abstract-Conference.html.

Adina Williams, Nikita Nangia, and Samuel R. Bowman. A Broad-Coverage Challenge Corpus for Sentence Understanding through Inference. In Marilyn Walker, Heng Ji, and Amanda Stent (eds.), *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pp. 1112–1122, New Orleans, Louisiana, 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1101. URL https://aclanthology.org/N18-1101/.

Edwin B. Wilson. Probable Inference, the Law of Succession, and Statistical Inference. *Journal of the American Statistical Association*, 22(158):209–212, 1927. ISSN 0162-1459. doi: 10.1080/01621459.1927.10502953. URL https://www.tandfonline.com/doi/abs/10.1080/01621459.1927.10502953.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. Transformers: State-of-the-Art Natural Language Processing. In Qun Liu and David Schlangen (eds.), *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 38–45, Online, 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-demos.6. URL https://aclanthology.org/2020.emnlp-demos.6/.

Shitao Xiao, Zheng Liu, Peitian Zhang, Niklas Muennighoff, Defu Lian, and Jian-Yun Nie. C-Pack: Packed Resources For General Chinese Embeddings. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '24, pp. 641–649, New York, NY, USA, 2024. Association for Computing Machinery. ISBN 979-8-4007-0431-4. doi: 10.1145/3626772.3657878. URL https://dl.acm.org/doi/10.1145/3626772.3657878.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of Thoughts: Deliberate Problem Solving with Large Language Models. In *Advances in Neural Information Processing Systems*, volume 36, pp. 11809–11822. Curran Associates, Inc., 2023. doi: 10.52202/075280-0517. URL https://proceedings.neurips.cc/paper/2023/hash/271db9922b8d1f4dd7aaef84ed5ac703-Abstract.html.

Xueliang Zhao, Wei Wu, Jian Guan, Qintong Li, and Lingpeng Kong. DynaAct: Large Language Model Reasoning with Dynamic Action Spaces. In *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 35533–35561. Curran Associates, Inc., 2025. doi: 10.52202/085713-1194. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/330eeed77a5d8fd4fe5a5a5982868acd-Abstract-Conference.html.

APPENDICES

Apps. A, B, C, and D contain proofs and numerical examples for the coverage objective, sufficient representations, diagnostic threshold, and discovery bounds. Apps. E, F, G, and H give the experimental methods and additional results for coordinated generation, selection from shared previews, complete procedures, and executable tests. App. I provides extended related work.

A COVERAGE AND DECISION-RELEVANT PREDICTION

The six analyses in Sec. 2.1 make the allocation decision concrete: two completed analyses can supply more findings either by contributing more useful material or by repeating less. This appendix derives that accounting identity, then asks how accurately a predictor must compare batches. Pairwise overlap gives a useful lower score, but we show exactly what it omits and why even perfect pairwise information can leave the best batch unresolved. The final subsection connects coverage to shared correctness and the separate decision of which completed answers to retain.

A.1 FROM COMPLETED ARTIFACTS TO THE REFERENCE IDENTITY

Planning assumptions. We compare what different batches would add from the same point in a computation. Let $E = [M]$ index the available actions and let $\mathcal{B} \subseteq 2^E$ contain the batches satisfying resource and compatibility constraints. Each action fixes its prompt or prefix, continuation and stopping rules, and reserved resources, and may produce several outputs. Its potential output bundle Z_i is jointly defined with those of the other actions and does not change with the selected batch. This lets us compare, for example, completing analyses 1 and 2 with completing 1 and 4 while keeping analysis 1 itself fixed.

We condition throughout on the information permitted at this decision, such as previews. Allocation randomization is independent of the potential outputs conditional on this information; the outputs themselves may be dependent. Thus the decision can use the previews, while its remaining random choice does not reveal unseen outputs. In the finite formulation of Sec. 2.1, $\mathcal{U}$ is finite, $X_i(u) = \mathbf{1}\{\text{action } i \text{ supplies } u\}$, and $N_S(u) = \sum_{i \in S} X_i(u)$.

This is a comparison of batches using the same planning information. Sequential selection that updates beliefs after observing each outcome requires a corresponding observation model and separate guarantees (Golovin & Krause, 2011).

For any finding hit $N_S(u)$ times, one hit supplies coverage and the remaining $(N_S(u) - 1)_+$ hits are repeats. Weighting and averaging this count gives

$$\mathcal{D}(Q) = \mathbb{E}_{S \sim Q}\left[\sum_{u \in \mathcal{U}} w_u \, \mathbb{E}[(N_S(u) - 1)_+]\right], \qquad (t)_+ = \max\{t, 0\}.$$

The inner expectation averages over potential outputs for a fixed batch, and the outer expectation averages over the selected batch. The total expected value of all hits is $\sum_i v_i x_i(Q)$, where $v_i = G(\{i\})$ and $x_i(Q) = \Pr_Q(i \in S)$. Subtracting the repeats leaves coverage: $J_G(Q) = \sum_i v_i x_i(Q) - \mathcal{D}(Q)$; subtracting this equality for two rules gives Eq. (2).

Graded support. Useful support can be graded: raising support for a finding from 0.3 to 0.8 earns 0.5 times its weight, while repeating the same 0.8 support adds nothing. The construction below expresses each increase as coverage of the newly reached support levels.

Let (Θ, μ) be a finite measure space, with $\theta \in \Theta$ indexing findings or other useful contributions and μ assigning their weights. For each completed artifact y, let $\kappa(y, \theta) \in [0, 1]$ be its measurable degree of support for θ, and let $b(\theta) \in [0, 1]$ be baseline support. For a finite completed portfolio A, the best available support and its gain over baseline are

$$m_A(\theta) = \max\{b(\theta), \sup_{y \in A} \kappa(y, \theta)\}, \qquad F(A) = \int [m_A(\theta) - b(\theta)] \, d\mu(\theta), \qquad \text{(A1)}$$

with $\sup \varnothing = 0$. All support profiles and the baseline are fixed at the planning decision. An action i has a jointly defined, almost surely finite potential output bundle Z_i. To reduce this graded objective to coverage, treat each support level $r \in [0, 1]$ for each finding as a target $u = (\theta, r)$, with measure $d\nu = d\mu\, dr$. Action i hits this target exactly when its output reaches level r and the current portfolio does not:

$$X_i(\theta, r) = \mathbf{1}\{m_A(\theta) < r \leq \sup_{y \in Z_i} \kappa(y, \theta)\}.$$

For a finding whose support rises from 0.3 to 0.8, these new targets are the levels in $(0.3, 0.8]$. Taking the union of the levels reached by the selected actions and integrating their lengths gives

$$F\left(A \cup \bigcup_{i \in S} Z_i\right) - F(A) = \int \mathbf{1}\{N_S(u) \geq 1\}\, d\nu(u). \tag{A2}$$

Taking expectation and using Tonelli's theorem gives the integral extension of Eq. (1). This is a nonnegative mixture of coverage functions, so the usual normalization, monotonicity, and submodularity properties apply (Nemhauser et al., 1978; Lin & Bilmes, 2011).

Finite measure bounds every gain, so the following exchanges of integration and finite sums are valid:

$$\sum_{i \in S} v_i - G(S) = \int \mathbb{E}\big[N_S(u) - \mathbf{1}\{N_S(u) \geq 1\}\big] d\nu(u) = \int \mathbb{E}(N_S(u) - 1)_+ d\nu(u).$$

Averaging over Q and subtracting the corresponding equality for Q_0 proves Eq. (2) for graded support as well. Fixed inclusions cancel the singleton term, so all gain then comes from avoiding repeated support. Unrestricted selection can also gain by selecting actions of higher individual value.

The action specification matters. If two prerequisite operations produce a useful artifact only when both are selected, their values can be $0, 0, 0, 1$ on $\varnothing, \{1\}, \{2\}, \{1, 2\}$. That is not diminishing returns over the prerequisite operations. A bounded operation that completes the entire prerequisite bundle is a different action. Likewise, changing another selected action's prompt or budget changes the potential-output experiment rather than preserving (A2).

A.2 Comparison-relative prediction

To choose a good batch, a predictor needs to distinguish the available choices. It need not recover their absolute values: adding the same constant to every predicted batch value leaves the choice unchanged, whereas reversing two competing batches can cause a loss. This is the decision-relevant interpretation used in Sec. 3.1.

For any batch utility f and allocation law Q, write $J_f(Q) = \mathbb{E}_Q f(S)$. Let $\mathfrak{Q}$ be the feasible family of allocation laws; it may include constraints on selection probabilities. For an error function e, the largest error in comparing two feasible rules is

$$\|e\|_{\mathrm{dec}} = \sup_{Q, Q' \in \mathfrak{Q}} |J_e(Q) - J_e(Q')|.$$

Proposition A1(comparison-relative prediction). If $\widehat{Q}$ has predicted utility within $\tau \geq 0$ of the best value of $J_{\widehat{f}}$ over $\mathfrak{Q}$, then

$$\sup_{Q \in \mathfrak{Q}} J_f(Q) - J_f(\widehat{Q}) \leq \tau + \|\widehat{f} - f\|_{\mathrm{dec}}. \tag{A3}$$

Here τ is the loss from imperfect optimization of the predicted utility, while the second term measures how far prediction errors can move a comparison. Errors in a statistic with the same expectation under every $Q \in \mathfrak{Q}$ cancel. For example, if every rule preserves x, an individual-value error

$e(S) = \sum_{i \in S} a_i$ has expectation $\sum_i a_i x_i$ under all rules and cannot change their ranking. That cancellation no longer follows when the rules may change the inclusion probabilities.

Proof. Put $e = \widehat{f} - f$. For any Q,

$$J_f(Q) - J_f(\widehat{Q}) = J_{\widehat{f}}(Q) - J_{\widehat{f}}(\widehat{Q}) - [J_e(Q) - J_e(\widehat{Q})].$$

The predicted-utility difference is at most τ by the optimization assumption; the magnitude of the error difference is at most $\|e\|_{\mathrm{dec}}$. Taking the supremum proves Eq. (A3). The coefficient one cannot be reduced: two decisions with true values $(0, d)$, $d > 0$, and errors $(d, 0)$ have equal predicted value. If the estimated tie selects the first, the true loss and the comparison error are both d, with $\tau = 0$. This is the comparison-specific perturbation argument underlying the decision-focused interpretation (Elmachtoub & Grigas, 2022). □

A.3 Pairwise truncation and estimation error

A natural approximation starts with individual useful value and subtracts every pair's overlap. This uses information about one or two actions at a time. Let $\omega_{ij} = \int \mathbb{E}[X_i(u)X_j(u)]d\nu(u)$ be the true expected weighted overlap of actions i and j, and define the pairwise score

$$L_2(S) = \sum_{i \in S} v_i - \sum_{i<j \in S} \omega_{ij}.$$

When three or more actions hit the same target, this subtraction removes too much credit. The exact correction is

$$G(S) = L_2(S) + R_3(S), \qquad R_3(S) = \int \mathbb{E}\binom{(N_S(u)-1)_+}{2} d\nu(u) \geq 0. \tag{A4}$$

Three actions supplying one identical unit give $L_2 = 3 - 3 = 0$ although coverage is one. The nonnegative remainder restores that credit, so a lower-bound score need not rank batches correctly.

For every integer $n \geq 0$,

$$\mathbf{1}\{n \geq 1\} - n + \binom{n}{2} = \binom{(n-1)_+}{2}.$$

Substitute $N_S(u)$, integrate, and take expectation to obtain Eq. (A4). This also proves the second-order Bonferroni bound $G(S) \geq L_2(S)$.

To use an estimated pairwise score to certify an improvement over a reference, we must account for two sources of discrepancy: moment-estimation error and the higher-order overlap omitted by the pairwise score. Write $y_{ij}(Q) = \Pr_Q(i, j \in S)$ and use Δ for candidate Q minus reference Q_0, with $\Delta G = J_G(Q) - J_G(Q_0)$. Suppose simultaneous estimation bounds satisfy $|\widehat{v}_i - v_i| \leq e_i$ and $|\widehat{\omega}_{ij} - \omega_{ij}| \leq e_{ij}$. Let $\widehat{\Delta L_2}$ be the score difference obtained from these estimated moments. Then

$$\begin{aligned} \Delta G &\geq \widehat{\Delta L_2} - r_2 - \mathbb{E}_{Q_0} R_3(S), \\ r_2 &= \sum_i e_i |x_i(Q) - x_i(Q_0)| + \sum_{i<j} e_{ij} |y_{ij}(Q) - y_{ij}(Q_0)|. \end{aligned} \tag{A5}$$

Indeed, $\Delta G = \Delta L_2 + \Delta R_3$, and the candidate's remainder is nonnegative. The expected pairwise score is $\sum_i v_i x_i - \sum_{i<j} \omega_{ij} y_{ij}$; applying the triangle inequality to its difference proves the error term. Thus a moment estimate contributes to r_2 only when the corresponding single-action or pair-selection probability changes. The reference remainder is still subtracted because the pairwise score may have understated the reference's true coverage. The scalar inequality $\binom{(n-1)_+}{2} \leq \binom{n}{3}$ gives one way to bound that omitted credit using triple overlaps:

$$R_3(S) \leq \sum_{i<j<\ell \in S} \int \mathbb{E}[X_i X_j X_\ell] d\nu.$$

Multiplicity at most two makes the pairwise score exact. Small prediction error by itself does not bound the higher-multiplicity term. Since the overlaps are nonnegative, L_2 is submodular but need not be monotone; the monotone-coverage greedy guarantee does not automatically apply to its fixed-count maximization.

A.4 A SHARP LOW-ORDER INFORMATION OBSTRUCTION

The preceding remainder shows why one particular pairwise score can fail. We now ask the stronger question: could a different rule always choose well from the same exact singleton and pairwise information? Two outcome models with identical such information but different optimal batches show that the answer is no.

There is one useful target of weight one and six actions in two independent groups of three. A bit records whether an action supplies the target. In world one, the first group's hit bits are uniform over $001, 010, 100, 111$ (odd parity), and the second group's over $000, 011, 101, 110$ (even parity); world two swaps these laws. Every singleton hit probability is $1/2$ and every pairwise hit probability is $1/4$ in both worlds, including pairs crossing the independent groups.

Select three actions. The whole odd group always contains a hit, while the whole even group misses in its 000 outcome and therefore covers with probability $3/4$. A mixed triple takes two bits from one group and one from the other. The pair is jointly zero with probability $1/4$, and the remaining independent bit is zero with probability $1/2$, giving coverage $7/8$. Every fixed triple therefore averages $7/8$ across the two worlds, although each world's optimum is one. A rule observing only the common singleton and pairwise moments returns the same distribution in both worlds; randomizing over triples preserves this average, so its worst-world regret is at least $1/8$. A fixed mixed triple attains that regret in both, making the bound sharp.

Any fixed moment order. For order $q \geq 1$, use two independent groups of $r = q + 1$ bits and select r actions. Every assignment on $t < r$ coordinates has 2^{r-t-1} extensions of either parity, so all moments through order q agree between worlds. The odd group never misses, the even group misses with probability $2^{-(r-1)}$, and a mixed batch misses with probability 2^{-r}. The same averaging argument gives exact minimax regret $2^{-(q+1)}$. This obstruction applies when the rule sees only these moments; outcome labels or a justified joint model can provide additional information.

A.5 SHARED TRUTH AND FINAL-UTILITY ALIGNMENT

Repeated mentions of an answer can share one underlying truth: the entity is either correct for the question or it is not. A coverage predictor must represent that shared event before combining action-level emission probabilities. More generally, let Z be a shared latent state and suppose that, conditional on Z, action hit events for each target factorize with probabilities $r_{ij}(Z)$. If target weights $w_j(Z)$ are also conditional on that state, then

$$G(S) = \mathbb{E}_Z \sum_j w_j(Z) \left[1 - \prod_{i \in S} (1 - r_{ij}(Z)) \right]. \tag{A6}$$

The product is the conditional probability that every selected action misses a target; only after computing that probability do we average over shared truth. In particular, a single entity of correctness probability q and conditionally independent emission probabilities r_i contributes $q[1 - \prod_i (1 - r_i)]$. If two actions certainly emit the entity, its useful coverage is still q. Multiplying correctness into each action probability before taking the product would instead give $1 - (1 - q)^2$, as if the two mentions could be independently correct. This discrepancy can reverse a selection. Suppose actions A and B certainly emit the same entity, whose correctness probability is 0.5, while C emits a different entity with correctness probability 0.2. The product-union proxy scores $\{\mathrm{A}, \mathrm{B}\}$ at $1 - (1 - 0.5)^2 = 0.75$, above the $0.5 + 0.2 = 0.70$ of $\{\mathrm{A}, \mathrm{C}\}$. Their actual expected distinct-correct coverage is 0.50 and 0.70, respectively. After selecting A, greedy maximization also prefers B: its proxy increment is 0.25, versus C's 0.20. Repetition has been mistaken for another chance of being correct. Without conditional factorization, the joint hit law in Eq. (1) remains the appropriate object.

Coverage gains must also be checked against the actual output metric. Even improving the expected numbers of correct and incorrect answers need not improve expected F1: returning the sole gold answer with probability $1/2$, and two wrong answers otherwise, gives expected F1 $1/2$; returning the gold and one wrong answer with probability 0.6, and nothing otherwise, improves both counts but gives expected F1 0.4. F1 optimization is a separate decision problem (Waegeman et al., 2014); App. F.1 specifies the QAMPARI metric, which combines name-level precision with target-level recall.

This distinction gives repetition two different roles. A second occurrence of a correct entity adds no distinct-target coverage, but agreement among completed answers can inform a filter deciding which names to retain. App. E measures useful repeats avoided during generation; App. G evaluates final answers after aggregation. A filter can discard correct as well as incorrect names, so its effect must be assessed on the chosen final metric.

B LEARNING ENOUGH TO COORDINATE

Thm. 1 asks how much must be learned to choose complementary actions together while preserving how often each action is selected. Its answer is a common circular order of the contributor groups, the representation illustrated in Fig. 1D. We first show why a known order attains the best possible coverage. We then bound how often complete contributor observations can restrict the remaining orders, convert that bound to allocation regret, and construct problems on which every learner suffers a matching order of loss. The proof combines classical systematic sampling, circular-order constraints (Hsu & McConnell, 2003; Fink et al., 2023), and a statistical bound on the remaining uncertainty (Hanneke, 2016). The last two subsections explain how to assess an imperfect order and how the learning mechanism is illustrated numerically. App. E separately tests useful coordination through a language model's sampling interface.

B.1 SIMULTANEOUS COUNT BALANCE

For a contributor group B, fixed selection probabilities already bound its chance of supplying a finding: it cannot exceed either one or $x(B) = \sum_{i \in B} x_i$. The challenge is to attain these bounds for all groups with a single rule. A common circular order lets one rule spread selections evenly through every group.

For prescribed probabilities with integer sum $\sum_i x_i = k > 0$, the systematic sampler (Madow, 1949) places intervals of lengths $x_i \leq 1$ in that order on a circle of circumference k. It selects the actions whose intervals contain $V, V+1, \ldots, V+k-1$, for uniform $V \in [0, 1)$. These k points are one unit apart. Each action interval therefore receives at most one point almost surely, giving exactly k selections and inclusion probability x_i. Consecutive contributors occupy one arc of length $s = x(B)$. That arc contains either $\lfloor s \rfloor$ or $\lceil s \rceil$ points, with mean s: if $s \leq 1$, the count is zero or one with hit probability s; if $s \geq 1$, a hit is certain. Its hit probability is consequently $\min(1, s)$.

The union bound gives the same ceiling for every competing law with these inclusions. Attaining it simultaneously on every support arc and averaging over B proves the optimality claim in Thm. 1. This is why the order suffices even when support frequencies are unknown: it optimizes each support's contribution before those contributions are averaged. Empty/full supports, zero-length intervals, and $k = 0$ follow directly.

For the six analyses in Sec. 2.1, set $x_i = 1/3$ and $k = 2$. The three systematic batches $\{1, 4\}$, $\{2, 5\}$, and $\{3, 6\}$ occur equally often and each hit every finding's contributor group. A uniformly chosen two-subset misses a specified three-action support with probability $\binom{3}{2}/\binom{6}{2} = 1/5$, giving total expected coverage $3(1 - 1/5) = 2.4$, compared with three for the systematic rule. Both rules preserve the same x.

Choosing the inclusion probabilities. Choosing x is a further decision: it determines how often to invest in each action. For finitely many known supports B_j sharing the order and known nonnegative weights w_j, the best value is obtained by maximizing $\sum_j w_j \min\{1, x(B_j)\}$ over $x \in [0, 1]^M$, $\sum_i x_i = k$. This is a linear program: introduce a credited hit probability t_j for each support, maximize $\sum_j w_j t_j$, and impose $0 \leq t_j \leq 1$ and $t_j \leq x(B_j)$. At any chosen x, setting t_j to

its ceiling gives the coverage attained by the sampler. Thus the optimum is attainable. Unlike coordinating a prescribed x, this optimization uses support weights as well as the order; additional hard constraints require every sampled batch to be feasible.

B.2 A BOUND ON STRICT CIRCULAR-ORDER RESTRICTIONS

An observed contributor set rules out every order in which that set is split apart. Call the remaining family the *version space*; a restriction is *strict* if it removes at least one order. Most later observations can become redundant even before the true order is uniquely identified. We bound the total number of strict restrictions, because only these observations change what remains possible.

A PC-tree represents all circular orders consistent with a collection of consecutive-set constraints (Hsu & McConnell, 2003; Fink et al., 2023). Its M labelled leaves are the actions. A P-vertex permits any cyclic ordering of its branches; a C-vertex fixes their cyclic order up to reversal. These local choices are independent, and the leaf orders of the resulting embeddings form the represented family. An exact update intersects that family with the orders satisfying the new constraint. We suppress internal degree-two vertices; degree-three P- and C-vertices are interchangeable because they permit the same orders.

Classical PC-tree analyses use degree-based accounting to bound update time. Here the potential counts how often the represented family can strictly shrink. For $M \geq 3$, let $P(T)$ and $C(T)$ be its sets of P- and C-vertices, and define

$$\Phi(T) = \sum_{v \in P(T)} (2\deg(v) - 5) + |C(T)|. \tag{B1}$$

Lemma B1. A consistent strict arc restriction decreases Φ by at least one under an exact PC-tree update. Hence at most $2M - 6$ observations can strictly reduce a nonempty circular-order version space.

Proof. Initially a single P-vertex connects all M leaves, representing every circular order; this unrestricted P-star has potential $2M-5$. Every internal vertex contributes at least one, so a nontrivial representation has potential at least one. It is therefore enough to show that each strict restriction spends at least one unit of potential.

Colour leaves in the observed set full and those in its complement empty. For a consistent strict constraint, the edges with both colours on both sides form a path, and the incident branches can be arranged with full branches on one side and empty branches on the other. This is the *terminal path* of the cited update; when it has no edges, a single vertex separates the full and empty branches. Off-path branches are monochromatic, so their internal order choices remain unrestricted.

The update splits each path vertex into its full and empty pieces, deletes the path edges, and joins the pieces through a new central C-vertex in their order along the two sides of the path. The new constraint fixes the relative orientations of C-vertices on that path: the C-pieces created by splitting them are contracted into the central vertex. C-vertices inside off-path branches retain their independent reversals, including when they become adjacent to the central vertex after suppression. This is the exact-family update described in the cited PC-tree construction, rather than a choice of one convenient consistent embedding.

We count the change first at each P-vertex and then at the C-vertices. For a path P-vertex of degree d, let t be its number of path edges, and let f, e count the full and empty off-path branches. Set $u = \mathbf{1}\{f > 0\} + \mathbf{1}\{e > 0\}$ and $a_1 = \mathbf{1}\{f = 1\} + \mathbf{1}\{e = 1\}$, the numbers of nonempty and singleton colour groups. Then $f + e = d - t$. A group of $z \geq 2$ branches remains a P-vertex of degree $z + 1$ and contributes $2z - 3$; a singleton needs no internal P-vertex. The P-potential change is exactly

$$\begin{aligned}(2f-3)\mathbf{1}\{f \geq 2\} + (2e-3)\mathbf{1}\{e \geq 2\} - (2d-5)\\ = 5 - 2t - 3u + a_1.\end{aligned}$$

At an endpoint of a path with at least one edge, $t = 1$ and $f, e \geq 1$: the side of the terminal edge containing that endpoint must contain both colours. Thus $u = 2$, $a_1 \leq 2$, and the change is at most

-1. At an internal path vertex, $t = 2$ and at least one off-path branch exists because its degree is at least three. If just one colour occurs, the change is $-2+a_1 \le -1$; if both occur, it is $-5+a_1 \le -3$.

If the path contains P P-vertices and C C-vertices, the old path C-vertices disappear into the single new C-vertex, giving C-potential change $1 - C$. Both endpoints supply a full and an empty branch, so the central vertex has degree at least four after the contractions. Singleton P-pieces have already been suppressed in the displayed calculation; all other new P-pieces have degree at least three. Hence a path of at least two vertices changes total potential by at most $1 - P - C \le -1$. Off-path vertices do not change their degree or contribution.

For a one-vertex path, a C-vertex's fixed cyclic order either already keeps the two colours together or makes the constraint inconsistent; reversing it does not change this. At a P-vertex, a restriction is strict only if $f, e \ge 2$. Each group then becomes a P-vertex, of degree $f+1$ or $e+1$, while the central degree-two C-vertex is suppressed. The change is $(2f-3)+(2e-3)-(2f+2e-5) = -1$. If one colour occupies at most one branch, that colour or its complement is consecutive in every represented order, so there is no restriction. Empty and full sets are redundant for the same reason. Inconsistent constraints are excluded by realizability; redundant ones leave the tree unchanged. Degree-three P/C conversions preserve the potential because either type contributes one.

Every consistent strict update therefore decreases the potential by at least one. Its initial value is $2M - 5$, and its final value is at least one, giving at most $2M - 6$ such updates. For $M \le 3$, every subset is a circular arc and there are no strict restrictions. □

This argument uses the exact represented order family, not one arbitrarily maintained representative order. Rejecting a constraint merely because that representative violates it would not implement the version space used below.

B.3 From strict updates to statistical regret

A learner may return any order still consistent with its observations. To control all such choices at once, we bound the probability of seeing a contributor set on which even one of those orders fails. Let $\mathcal{V}_n$ be the nonempty set of orders that keep all n observed supports consecutive, and let $D_M = \max(0, 2M - 6)$. Its disagreement probability is

$$q_n = \Pr\{B \text{ is not an arc in at least one order in } \mathcal{V}_n\}.$$

The probability here is over a fresh support from the deployment distribution, conditional on the observed sample. If that support lies in the event defining q_n, its observation strictly shrinks the version space. Realizability ensures that the true order survives.

For the expected bound, view each order as a classifier that labels a support one exactly when it is an arc; every observed label is one. Process $n + 1$ observations and retain only those that strictly restrict the version space. Lem. B1 leaves at most D_M retained observations with exactly the same version space as the full sample. Every *essential* observation—one whose removal enlarges that version space—must be in this retained set. Now leave out one uniformly chosen observation. It rules out an order allowed by the other n exactly when it is essential. Since the observations are iid, this held-out observation has the same role as a fresh probe after n observations. Thus $\mathbb{E}q_n$ is the expected fraction of essential observations. This is the leave-one-out form of the version-space argument in Hanneke (2016, Definition 6 and the argument preceding Theorem 7), giving

$$\mathbb{E}q_n \le \frac{D_M}{n+1}. \tag{B2}$$

The confidence bound uses the same update count. If q_t stayed large for many observations, strict updates would be frequent, but at most D_M can occur. To make this argument quantitative with the stated constant, let $\mathcal{F}_t$ be the history of the first t observations and let U_{t+1} indicate a strict update. Then $\mathbb{E}[U_{t+1} \mid \mathcal{F}_t] = q_t$, $\sum_t U_{t+1} \le D_M$, and q_t decreases along every sample path because orders are only removed. For $\lambda > 0$, the conditional Bernoulli moment formula shows that

$$Z_n = \exp\left[(1-e^{-\lambda})\sum_{t<n} q_t - \lambda \sum_{t<n} U_{t+1}\right]$$

has expectation at most one, using $1 - z \le e^{-z}$. On $q_n > \epsilon$, monotonicity gives $Z_n \ge \exp[(1 - e^{-\lambda})n\epsilon - \lambda D_M]$. Markov's inequality with $\lambda = \log 2$ yields

$$\Pr(q_n > \epsilon) \le \exp(D_M \log 2 - n\epsilon/2). \tag{B3}$$

Outside the disagreement event defining q_n, every consistent order makes the fresh support an arc, so every such order attains its targetwise optimum for every admissible inclusion vector. On the event, the normalized targetwise shortfall is at most one. The regret of any consistent-order sampler is therefore at most Wq_n, simultaneously over all these orders and vectors. Eqs. (B2)–(B3) prove Eq. (4) and the simultaneous high-probability statement: for $0 < \delta < 1$, with probability at least $1 - \delta$,

$$\mathrm{OPT}_x - J_G(\widehat{Q}_x) \le W \min\left\{1, \frac{2[D_M \log 2 + \log(1/\delta)]}{n}\right\} \tag{B4}$$

for every consistent order and integer-sum inclusion vector. The same event controls all these vectors because it concerns only whether supports are arcs. A learned order can therefore be reused across batch sizes and selection probabilities without a separate confidence bound for each.

B.4 The matching lower construction

Could a different learner avoid the M/n dependence? The construction hides a separate allocation choice in each of many small blocks. Only a rare observation reveals that choice. Before it appears, any law with the prescribed selection probabilities must risk choosing two contributors to the same target together and wasting part of their combined hit probability.

Take $h \ge 5$ divisible by five, $M = 4h$, $k = 8h/5$, and $x_i = 2/5$ for every action. Partition actions into h four-action blocks, each with an independent hidden fair bit. A probe chooses a block uniformly. Within it, use local labels $1, 2, 3, 4$ and write ij for the support $\{i, j\}$. The support is 12 or 34 with probability $(1 - \epsilon)/2$ each, or a rare pair 13 or 14, chosen by the hidden bit, with probability $\epsilon \le 1/2$.

For either value of a block's hidden bit, its three support pairs form a path on four vertices. Concatenating the block path orders makes every supported pair an arc, so every hidden world satisfies the theorem's structural assumption. Each support has inclusion mass $4/5 < 1$; its coverage ceiling is therefore $4/5$, and co-selecting its two members reduces coverage below that ceiling.

An informed law selects $4h/5$ blocks uniformly and chooses one of the path's two colour classes equally in each selected block. These classes alternate along the path, so each contains two actions and neither contains both endpoints of a support pair. The law has exactly k actions, inclusion probabilities $(4/5)(1/2) = 2/5$, and never co-selects a support pair. Every support is hit with probability $4/5$, attaining its ceiling.

For any learner's returned law, write $z_{ij} = \Pr(i, j \in S)$ within a block. The hit probability of support ij is $x_i + x_j - z_{ij} = 4/5 - z_{ij}$, so its deficit is exactly z_{ij}. Some co-selection among actions $\{1, 3, 4\}$ is unavoidable: their selected count N has expectation $6/5$. Since $\binom{N}{2} \ge N - 1$,

$$z_{13} + z_{14} + z_{34} \ge 1/5.$$

Until that block's rare support has been observed, its hidden bit is still fair, even conditional on observations from all other blocks. The conditional average deficit from its oracle value is at least

$$\frac{1-\epsilon}{2}(z_{12} + z_{34}) + \frac{\epsilon}{2}(z_{13} + z_{14}) \ge \frac{\epsilon}{10}, \tag{B5}$$

where $1 - \epsilon \ge \epsilon$: every one of z_{13}, z_{14}, z_{34} has coefficient at least $\epsilon/2$, and their sum is at least $1/5$. The probability that the rare support of a given block has not appeared in n probes is $(1 - \epsilon/h)^n$.

Averaging over blocks and hidden worlds proves that some realizable world has expected regret at least $W\epsilon(1-\epsilon/h)^n/10$.

Choose $\epsilon = \min\{1/2, h/[2(n+1)]\}$, so the informative event remains rare enough for many blocks to be unresolved. Since $n\epsilon/h \le 1/2$, Bernoulli's inequality gives $(1-\epsilon/h)^n \ge 1/2$. For $n \ge 1$, the resulting bound is at least

$$\frac{W}{80}\min\{1, h/n\}. \tag{B6}$$

Since $M = 4h$, this is $\Omega(W\min\{1, M/n\})$. The learner was allowed any law with the prescribed inclusions, not merely a consistent-order sampler. This proves the lower statement of Thm. 1. The result is worst-case over admissible families; integral inclusions can leave no coordination choice at all.

B.5 When some supports violate the order

For $M \ge 4$, exact circular realizability excludes fully nondegenerate independent hits: independent Bernoulli events with probabilities strictly between zero and one give every contributor subset positive probability. Any fixed circle has a nonadjacent pair, so some of these subsets are not arcs. With four actions and hit probabilities $1/2$, the two nonadjacent pairs violate each order with total probability $2/16 = 1/8$. This example identifies a restriction on the support law; it does not imply that violations are frequent for every nonrealizable law.

A chosen order can still be useful when some contributor sets are split apart. Its loss is controlled by those exceptional sets: the sampler remains optimal for every support that is an arc. For any fixed order σ, let ϵ_σ be the probability that a support is not an arc, and $Q_{\sigma,x}$ its systematic law. Comparing each support with its universal ceiling gives

$$\mathrm{OPT}_x - J_G(Q_{\sigma,x}) \le W\mathbb{E}_B[\min\{1, x(B)\}\,\mathbf{1}\{B \text{ is not an arc in } \sigma\}] \le W\epsilon_\sigma.$$

Arc supports attain their ceiling; other supports lose at most $\min\{1, x(B)\}$. Thus violations are weighted by the coverage they could contribute at the chosen x.

This gives a way to assess an order learned or selected before evaluation. Let $\widehat{\epsilon}_\sigma$ be the non-arc frequency in $m \ge 1$ independent held-out complete contributor-set observations. Hoeffding's inequality (Hoeffding, 1963) gives the regret certificate $W\min\{1, \widehat{\epsilon}_\sigma + \sqrt{\log(1/\delta)/(2m)}\}$ with probability at least $1-\delta$, simultaneously over admissible x. The empirical violation frequency estimates the population frequency, and the square-root term allows for sampling uncertainty. The order must be fixed before this sample, and observations must follow the deployment support law. This certificate checks a chosen order; it does not supply a learning rule when noisy constraints leave no order consistent with all observations.

B.6 Numerical illustration: learning an order

The illustration measures the coverage lost when a sampler uses an order learned from finitely many complete contributor observations. Comparing it with the known-order and random-order rules keeps every action's selection probability fixed, so the difference comes from how the rules coordinate actions.

The construction has $W = 1$, $M \in \{6, 8, 10, 12\}$, and $k = \max\{2, \lfloor M/3 \rfloor\}$. For each of 200 instances per dimension, draw weights uniformly on $[0.2, 1]$, normalize them to sum k, and reject unless every inclusion lies in $[0.05, 0.95]$. A random action-label permutation hides the order. The support law is uniform over its circular arcs of lengths $1, \ldots, \lfloor M/2 \rfloor$.

A nested iid stream reveals complete contributor sets at observation counts $0, 1, 2, 3, 4, 6, 8, 12, 16, 24, 32, 48, 64$. The learner fixes one label as circular origin, complements sets containing it, and uses randomized backtracking to find a consistent order. At zero observations it uses a random order; the separate random-order reference averages five orders per instance. All systematic rules preserve the same inclusions and batch size.

For evaluation, partition the phase $V \in [0, 1)$ into intervals on which the selected batch is unchanged. We integrate these systematic phase cells exactly for each returned order and average over the full

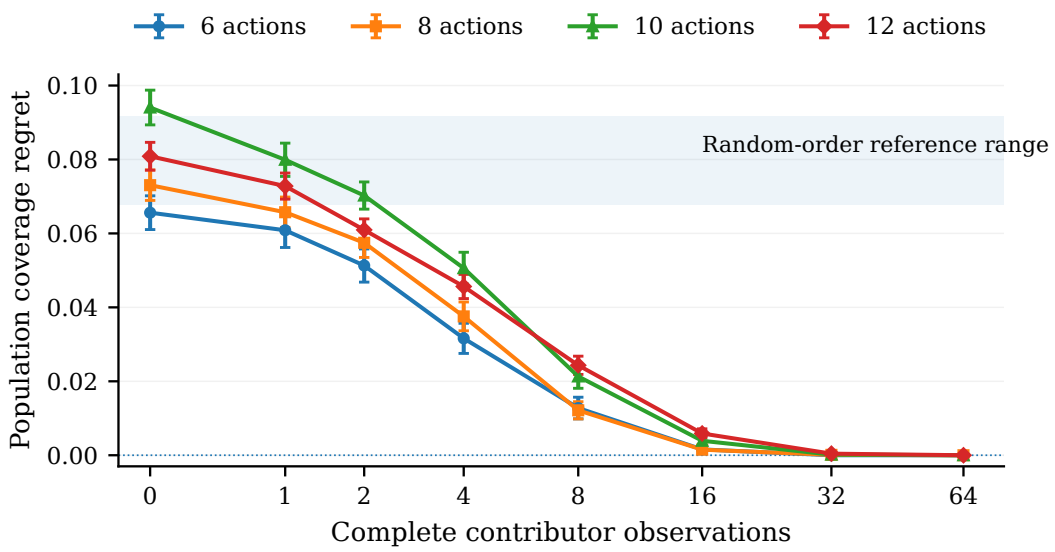


Figure 3: **A learned order can coordinate prescribed selections.** Population regret on 800 hidden-order instances (200 per action count), with full contributor observations and exact evaluation. Error bars show ± 1.96 standard errors across instances. Shading spans the four random-order reference means, not uncertainty. The known-order sampler has zero regret.

support law. Regret is therefore evaluated over the population support law. Fig. 3's error bars describe variation between 200 instances. The simulation uses random seed 20260916. Tab. B1 gives selected observation counts.

Table B1: Mean population regret over 200 instances per action count, with fixed inclusion vectors and uniformly distributed circular-arc supports; each instance is evaluated exactly. The known-order value is optimal; smaller regret is better.

Actions	Batch	Random order	1 observation	8 observations	16 observations	32 observations
6	2	0.0678	0.0609	0.0128	0.0016	0.0000
8	2	0.0769	0.0657	0.0122	0.0015	0.0001
10	3	0.0917	0.0799	0.0213	0.0039	0.0002
12	4	0.0845	0.0728	0.0243	0.0059	0.0004

C When diagnostic information pays

A check is useful only if the completion work it saves pays for the information it buys. We prove Thm. 2 by first finding when one check per candidate beats blind completion, then showing that repeated checks cannot extend that profitable price range. The second step allows a procedure to set partly checked candidates aside and return to them later. We then consider a choice of diagnostics, reuse of preview work, and a buying rule based on independent calibration.

Throughout the proof, each independent candidate has persistent usefulness with known probability $0 < p < 1$. Its identifier gives no advance clue to usefulness. Completion costs $C > 0$ and reveals usefulness; a diagnostic costs $c > 0$, supplies information only, and leaves that completion charge unchanged. Reports take finitely many values and are independent draws from P_1 or P_0, conditional on the candidate being useful or a duplicate. We compare procedures that eventually obtain a useful completion with probability one.

C.1 THE COST OF CHECKING ONCE PER CANDIDATE

Start with the simplest screening procedure: check a fresh candidate once, use its report to decide whether to complete it, and move to another candidate after rejection or an unhelpful completion. Let $N = 1$ indicate usefulness and let $q(Y) = \Pr(N = 1 \mid Y)$ be its probability after report Y. An acceptance rule $a(Y) \in [0, 1]$ gives the probability of completing that candidate. Expectations below average over the report law $pP_1 + (1 - p)P_0$.

An attempt always pays c, pays C with probability $\mathbb{E}a$, and succeeds with probability $\mathbb{E}[aq]$. Since an unsuccessful attempt returns to the same fresh-candidate problem, repeating a fixed rule with positive success probability has expected cost

$$\frac{c + C\mathbb{E}a}{\mathbb{E}[aq]}. \tag{C1}$$

Thus the decision is governed by cost per successful attempt, not by report accuracy alone. Also allow blind completion, whose ratio is C/p. Choosing a different one-preview rule after each failure cannot improve the best ratio: if that ratio is t, every attempt has conditional expected cost at least t times its conditional success probability. Summing until the first success gives the same lower bound t, attained by repeating a minimizing rule.

To find the best rule, provisionally value a success at t. Completing after report Y then contributes cost minus expected value $C - tq(Y)$, so we accept precisely when this quantity is negative. Writing $(x)_+ = \max(x, 0)$, the smallest expected cost minus success value is

$$c + \inf_{0 \le a \le 1} \mathbb{E}[a(C - tq)] = c - \mathbb{E}(tq - C)_+. \tag{C2}$$

The zero of this expression identifies the best preview-only ratio: $\mathbb{E}(tq(Y) - C)_+ = c$. Its rule accepts when $q(Y) > C/t$, with arbitrary treatment of ties. With finitely many reports, no other randomization is needed. Comparing this ratio with C/p determines whether to check at all.

For a binary flag oriented toward usefulness, let $\alpha > 0$ be its sensitivity and $\beta \le \alpha$ its false-positive rate. The exact best blind-or-one-preview cost is

$$T_1^*(p) = \min\left\{\frac{C}{p}, \frac{c + C[p\alpha + (1 - p)\beta]}{p\alpha}\right\}. \tag{C3}$$

The second branch completes positive flags only. Its numerator includes the check and all accepted completions, including false positives; its denominator counts only useful accepted candidates. To see why no other binary rule improves the minimum, the positive flag has posterior at least p, while the negative flag has posterior at most p. Accepting negative flags alone therefore cannot beat blind completion. Accepting both pays for an unused check, and accepting neither cannot succeed. Positive-only acceptance has $\mathbb{E}a = p\alpha + (1 - p)\beta$ and $\mathbb{E}[aq] = p\alpha$, proving Eq. (C3).

For the main-text example, $p = 0.2$, $\alpha = 0.8$, and $\beta = 0.3$ give acceptance probability 0.40 and success probability 0.16 per attempt. At $c = 0.2C$, the ratio is $(0.2C + 0.40C)/0.16 = 3.75C$, compared with blind cost $5C$.

The reservation equation in Eq. (C2) specializes the partial-inspection reservation value of Aouad et al. (2026, Definition 1, Eq. (4)).[1] At the blind ratio $t = C/p$, Bayes' rule gives $\mathbb{E}(tq - C)_+ = C(1 - p)\,\mathrm{TV}(P_1, P_0)$. This is the check-price threshold below which one report permits a strict improvement over blind completion. The fresh-attempt argument above supplies our expected-cost-to-success objective; we next show why adaptive repetition cannot make a diagnostic profitable above the same boundary.

[1] Set their prize to RN, full-inspection cost to C, and $t = R - \sigma^P$, where σ^P is their partial-inspection reservation value, with $R > C/p$ and $\sigma^P \ge 0$.

C.2 AN INEQUALITY FOR THE VALUE OF ONE DIAGNOSTIC

The repeated-check proof needs to bound how much one additional report can improve a candidate's apparent value, even after earlier reports have changed our belief about it. The following inequality provides that bound. For laws P_1, P_0 on a finite alphabet and nonnegative weights A, B,

$$\sum_y (AP_1(y) - BP_0(y))_+ - (A - B)_+ \le \mathrm{TV}(P_1, P_0) \min(A, B). \tag{C4}$$

Before observing a report, the positive part $(A-B)_+$ records whether the useful weight exceeds the duplicate weight. After the report, the weights become $AP_1(y)$ and $BP_0(y)$; the sum records the total positive part across outcomes. The inequality limits the increase by the channel's discrimination times the smaller initial weight.

If $A \ge B$, the left side equals $\sum_y (BP_0(y) - AP_1(y))_+$ and is at most $B \sum_y (P_0(y) - P_1(y))_+$. If $B \ge A$, interchange the roles. Each positive-part sum of the difference of probability laws is total variation.

C.3 NECESSITY FOR REPEATED AND BUFFERED POLICIES

Now suppose a procedure may check repeatedly, switch candidates, or keep several partly checked candidates available. We show that an expensive diagnostic cannot create enough value to repay its cost. Set $T_0 = C/p$, the cost of obtaining one useful completion blindly, and assign that amount as a reward for success. A candidate currently useful with posterior probability z then has expected completion reward $T_0 z$. Assign it the nonnegative potential

$$V(z) = (T_0 z - C)_+.$$

This quantity measures the amount by which the expected reward exceeds the completion charge, allowing us to discard the candidate when it does not. A fresh candidate has $z = p$, so its potential is zero.

To bound a report's effect, represent the current posterior by unnormalized useful and duplicate weights w_1, w_0, so $z = w_1/(w_1 + w_0)$. The numerator of the potential is $(A - B)_+$, where $A = (T_0 - C)w_1$ and $B = Cw_0$. After a fresh diagnostic, Eq. (C4) bounds the expected potential increase by

$$\mathrm{TV}(P_1, P_0) \min\{C(1-p)z/p,\ C(1-z)\} \le C(1-p)\,\mathrm{TV}(P_1, P_0). \tag{C5}$$

The first term inside the minimum grows with z, while the second falls; they meet at $z = p$. This makes the rightmost bound valid at every posterior, including those reached after many favorable reports. If c is at least that bound, each diagnostic increases expected potential by no more than it costs. Introducing a fresh candidate adds zero potential, and discarding a candidate removes nonnegative potential. Completing one at posterior z produces expected reward $T_0 z$, costs C, and removes potential at least $T_0 z - C$. Each permitted operation therefore has nonpositive expected change in reward minus cost plus remaining potential.

For buffered independent candidates, sum the potentials of all active candidates. Observing one does not change the others' posteriors, so this same accounting applies when checks are interleaved. Formally, success reward minus accumulated physical cost plus remaining potential is a supermartingale: conditional on everything observed so far, its expected next value is no larger than its current value. It starts at zero. Stop after a fixed finite number of paid operations or the first success, whichever comes first. Dropping the nonnegative remaining potential gives expected reward at most expected cost. Letting that operation limit increase, eventual almost-sure success and monotone convergence give $T_0 \le \mathbb{E}[\mathrm{cost}]$. Infinite expected cost already satisfies the inequality. Thus neither randomized scheduling nor returning to a promising buffered candidate evades the price boundary.

C.4 SUFFICIENCY, MENUS, AND OPERATION COSTS

Below the boundary, a simple rule achieves a strict saving: accept after one diagnostic precisely on $A = \{y : P_1(y) > P_0(y)\}$, the reports that are more likely for a useful candidate than a duplicate. Its sensitivity and false-positive rate differ by $\mathrm{TV}(P_1, P_0)$. Subtracting its ratio (C1) from C/p shows strict improvement exactly when $c < C(1-p)\,\mathrm{TV}(P_1, P_0)$. Together with necessity this proves Thm. 2, including the equality boundary.

The same reasoning answers whether several diagnostic types can rescue individually overpriced checks. For a finite menu, apply Eq. (C5) separately to each technology. If every diagnostic costs at least its own profitability threshold, no adaptive combination can improve on blind completion; if one costs less than its threshold, that technology alone suffices. Here conditional freshness means that, given the candidate's usefulness and the full diagnostic history, the next report has the selected technology's specified law. It is this conditional law, rather than a marginal confusion matrix alone, that licenses the potential bound at each adaptive choice.

When previews change execution costs. If a preview performs some work that completion can reuse, its value includes that saved work as well as information. If direct execution costs D, while buying a preview and then executing costs $c+C$, the oriented binary one-preview optimum is instead

$$\min\left\{D/p,\,(c+C)/p,\,\frac{c+C[p\alpha+(1-p)\beta]}{p\alpha}\right\}. \tag{C6}$$

The middle branch always completes after the preview, so it captures any saving from reuse even if the report is ignored. The final branch additionally screens on a positive flag. Positive-only screening beats direct execution exactly when $c < \alpha(D-C)+C(1-p)(\alpha-\beta)$. Thm. 2's simpler expression assumes the same additional completion charge with or without a diagnostic.

If a preview also produces usable output, we must specify what the procedure can recognize and retain without completing it. Crediting output merely because it is useful in hindsight would give the procedure information it does not observe. An observable retention rule and a model of the value and cost of continuation are therefore needed to study the joint choice of information and productive work.

Why report freshness matters. Conditional freshness is also substantive. Repeating one noisy bit indefinitely preserves the marginal confusion matrix but provides no additional information after its first observation. Conversely, two individually uninformative bits can jointly reveal novelty through parity. Neither experiment is an iid repetition of the marginal diagnostic channel in Thm. 2.

C.5 A CONSERVATIVE GATE FROM INDEPENDENT CALIBRATION

In practice the report rates may need to be estimated before deciding whether to buy checks. The relevant quantity is their difference, because it determines the profitable price in Thm. 2. A conservative rule buys only when a lower confidence bound on that difference justifies the price; otherwise it completes blindly.

Fix a binary diagnostic independently of the calibration sample, then calibrate it on $n_+ > 0$ independent novel and $n_- > 0$ independent duplicate examples from the deployment conditional laws. Their positive-flag frequencies $\widehat{\alpha}, \widehat{\beta}$ estimate sensitivity α and false-positive rate β. For $0 < \delta < 1$, Hoeffding's inequality (Hoeffding, 1963) gives simultaneous radii

$$e_+ = \sqrt{\frac{\log(4/\delta)}{2n_+}}, \qquad e_- = \sqrt{\frac{\log(4/\delta)}{2n_-}}.$$

With probability at least $1-\delta$, both estimates differ from their true rates by at most their respective radii. Consequently, $\widehat{\alpha} - \widehat{\beta} - e_+ - e_-$ lower-bounds the true discrimination. At a known novelty probability $0 < p < 1$, buy one diagnostic and complete positive flags only when

$$C(1-p)(\widehat{\alpha} - \widehat{\beta} - e_+ - e_-) > c. \tag{C7}$$

On the same confidence event, every gate-selected one-preview policy beats blind completion in expected online cost to first success. The guarantee holds simultaneously for every known p, because only the two channel rates are estimated. Larger calibration samples shrink the uncertainty penalty in the buying rule.

This guarantee concerns online checking and completion under the theorem's separate charges. Fitting and calibration costs must also be charged or amortized when assessing a complete procedure. The fixed diagnostic must be calibrated on the deployment conditional laws; this is a different object from SCPS's fitted entity-correctness profile. These marginal rate estimates justify the one-preview decision, while repeated-report likelihood tests additionally need the channel law conditional on the full diagnostic history.

D DISCOVERY UNDER A RESOURCE BUDGET

To collect one completed finding from each of d classes, we must both encounter every class and pay to complete its output. Blind generation pays the completion price even for repeated encounters. This appendix proves Thm. 3: fresh checks can move much of that repeated-search cost to the diagnostic price, while the d necessary completions remain. We first bound expected cost, then construct a procedure with fixed operation caps, and finally show that every policy faces matching resource requirements. The numerical comparison illustrates why minimizing total spending and limiting expensive completions can favor different policies.

The model has independent candidates with persistent, uniformly distributed classes and identifiers that reveal no class information. A completion costs C, reveals the class, and supplies its output. A check costs c and reports a noisy novelty flag: its positive probabilities are a for a missing class and b for a collected class, with fixed $0 < b < a < 1$. Reports have fresh independent noise conditional on the candidate's class and current collection. With r classes missing, a fresh candidate is useful with probability $p_r = r/d$; checks cannot identify or target a particular missing class.

D.1 A SEQUENTIAL TEST THAT REJECTS CHEAPLY

Near the end of discovery, most candidates are duplicates. We therefore reject a candidate after modest evidence of duplication, but require stronger evidence of novelty before paying for completion. A positive flag adds u units of log-likelihood evidence for novelty, and a negative flag subtracts v, where

$$u = \log(a/b) > 0, \quad v = \log[(1-b)/(1-a)] > 0,$$
$$D_1 = au - (1-a)v > 0, \quad D_0 = (1-b)v - bu > 0.$$

The accumulated evidence has mean increment D_1 under novelty and $-D_0$ under duplication. Both drift magnitudes are positive, so a useful candidate tends toward acceptance and a duplicate toward rejection. Starting at zero, stop when the sum reaches $h > 0$ or falls below $-\log 2$, and accept for completion only at the upper exit. This is the classical sequential likelihood test (Wald & Wolfowitz, 1948), with a constant rejection boundary as in rare-population search (Malloy et al., 2013).

Let τ be the number of diagnostics in this test, and let α, β be the test's acceptance probabilities under novelty and duplication. These are probabilities for the whole sequence of reports, distinct from the single-report rates a, b.

The likelihood ratio is a nonnegative martingale of mean one under duplication, so its probability of ever reaching e^h is at most e^{-h}. Applying the reciprocal argument under novelty bounds rejection probability by $1/2$. Thus $\beta \le e^{-h}$ and $\alpha \ge 1/2$: raising the acceptance threshold suppresses false completions while retaining at least half the useful candidates. Bounded increments and nonzero drift ensure a finite expected stopping time. Wald's identity relates the expected final evidence to that stopping time; the final step overshoots the lower boundary by at most v and the upper boundary by at most u. Consequently,

$$\mathbb{E}_0\tau \le A_0 := \frac{\log 2 + v}{D_0}, \qquad \mathbb{E}_1\tau \le \frac{h+u}{D_1}. \tag{D1}$$

Here $\mathbb{E}_0$ and $\mathbb{E}_1$ condition on duplication and novelty. The first bound is independent of h: the many duplicates are cheap to reject even when acceptance requires strong evidence. The threshold-dependent term appears only in the useful-candidate bound.

At novelty probability p, test fresh candidates and complete accepted ones until a useful completion occurs. Each attempt succeeds with probability $p\alpha \geq p/2$. Set $h = \log(2/p)$, so $\beta \leq p/2$. This keeps the false-acceptance rate small enough relative to the rarity of useful candidates. The expected number of completions per discovery is

$$\frac{p\alpha + (1-p)\beta}{p\alpha} \leq 2,$$

because $p\alpha + (1-p)\beta$ is the completion probability per attempt, while $p\alpha$ is its success probability. Applying the same cost-per-success calculation to the diagnostic count gives at most

$$\frac{2A_0}{p} + \frac{2[\log(2/p) + u]}{D_1}. \tag{D2}$$

Sum over the discovery stages $p_r = r/d$. The identities $\sum_r 1/p_r = dH_d$ and $\sum_{r=1}^{d} \log(d/r) \leq d$ give an expected upper bound of order $Cd + cdH_d$, with constants depending only on a, b. The $1/p$ term charges the repeated search at the diagnostic price, whereas the at-most-two expected completions per discovery contribute only order Cd. Blind completion provides the alternative CdH_d, so we can use whichever procedure is cheaper. This expected-cost construction does not impose a pathwise cap of d completions.

For the matching lower bound, every successful full-coverage procedure needs d completions, costing at least Cd. It must also encounter candidates from every class. Call the first paid operation on a candidate its *first touch*. Because unopened identifiers are uninformative and candidates independent, the class of each newly touched candidate is still iid uniform, even if the policy chooses its identifier adaptively. Collecting every first-touch class is therefore the usual coupon-collection problem: it takes dH_d candidates in expectation. Each first touch costs at least $\min(c, C)$, so expected cost is also at least $\min(c, C)dH_d$. We need not add these potentially overlapping charges directly: their maximum is at least half their sum. This proves the expected-cost order in Eq. (6), including for policies that buffer candidates.

D.2 A DETERMINISTIC DIAGNOSTIC CAP WITH AT MOST d COMPLETIONS

A bound on average expenditure does not guarantee completion by a fixed budget. We now enforce at most d completions and a fixed number of diagnostics on every path, including failures. For $0 < \delta < 1/2$, allocate failure probability $\delta/3$ to each of three events: completing a duplicate, failing to encounter and accept every class soon enough, and spending too many diagnostics.

First make duplicate acceptance sufficiently rare. At novelty rate p, retain the lower boundary and use

$$h(p) = \log \frac{6d}{p\delta}. \tag{D3}$$

Since $\alpha \geq 1/2$ and $\beta \leq e^{-h(p)}$, the conditional probability that an accepted candidate is a duplicate is at most $\beta/(p\alpha) \leq 2e^{-h(p)}/p \leq \delta/(3d)$. Stop after d accepted completions. A union bound then gives probability at most $\delta/3$ that any of them is a duplicate. If none is, these d completions cover all classes.

Next bound how many fresh candidates must be tested. While a specified class remains missing, the next candidate belongs to it with probability $1/d$ and is then accepted with probability at least $1/2$. Consider the test process continued until full coverage, temporarily without the cap of d completions. A class remains unaccepted after N attempts with probability at most $e^{-N/(2d)}$. Taking

$$N = \left\lceil 2d \log \frac{3d}{\delta} \right\rceil \tag{D4}$$

and applying a union bound over classes gives probability at most $\delta/3$ that full coverage needs more attempts. Until a duplicate is accepted or full coverage is reached, this continued process and the completion-capped process agree. Thus, outside the two failure events already bounded, the capped process also finishes within N attempts.

The final step controls diagnostic work. Charging every attempt the largest possible acceptance threshold would lose an extra logarithmic factor. Instead, the following bound assigns the threshold-dependent cost only to accepted candidates, of which there are at most d. There are constants $\lambda, \kappa > 0$ and $K \geq 1$, depending only on a, b, such that for either candidate type and every $h \geq 0$,

$$\mathbb{E} \exp\{\lambda\tau - \kappa h A\} \leq K, \tag{D5}$$

where A indicates upper-boundary acceptance. The subtraction of $\kappa h A$ is the key: it permits long tests when they lead to acceptance, while demanding a uniform exponential bound for rejected tests.

Proof of Eq. (D5). Let S_n be the unrestricted sum of the first n log-likelihood increments, with $S_0 = 0$, and write $\ell = \log 2$. Define $\tau = \inf\{n \geq 0 : S_n \geq h \text{ or } S_n < -\ell\}$ and $A = \mathbf{1}\{S_\tau \geq h\}$. At $h = 0$ this means immediate acceptance with $\tau = 0$, so the bound is automatic. Assume henceforth that $h > 0$.

The increments lie in $[-v, u]$, have mean $-D_0$ under duplication and D_1 under novelty, and are independent under either law. To obtain a tail rate valid for both, set

$$R = u + v, \qquad \eta = \frac{\min\{D_0^2, D_1^2\}}{2R^2}, \qquad \lambda = \eta/2, \qquad \kappa = 2\lambda/D_1.$$

These constants are positive and depend only on the fixed channel. Under duplication, let $T_0 = \inf\{n \geq 0 : S_n < -\ell\}$, ignoring the upper boundary; then $\tau \leq T_0$. For $n \geq m_0 := \lceil 2\ell/D_0 \rceil$, survival implies $S_n \geq -\ell$, a deviation of at least $nD_0/2$ above its mean. Hoeffding's inequality (Hoeffding, 1963) therefore gives

$$\Pr_0(T_0 > n) \leq e^{-nD_0^2/(2R^2)} \leq e^{-\eta n}.$$

We use the following tail-sum identity to convert this into a moment bound. For a nonnegative integer-valued T and an integer $m \geq 0$, if $\Pr(T > n) \leq e^{-2\lambda n}$ for all $n \geq m$, then

$$\begin{aligned} \mathbb{E}e^{\lambda T} &= 1 + (e^\lambda - 1) \sum_{n \geq 0} e^{\lambda n} \Pr(T > n) \\ &\leq e^{\lambda m} + \frac{(e^\lambda - 1)e^{-\lambda m}}{1 - e^{-\lambda}} \leq e^{\lambda m} + e^\lambda. \end{aligned}$$

Consequently, under duplication the left side of Eq. (D5) is at most $e^{\lambda m_0} + e^\lambda$, uniformly in h.

Under novelty, separate rejected and accepted tests. Rejection at time $n \geq 1$ implies $S_n < -\ell$, although its mean is nD_1. Hoeffding gives $\Pr_1(\tau = n, A = 0) \leq e^{-2nD_1^2/R^2} \leq e^{-\eta n}$, and hence

$$\mathbb{E}_1[e^{\lambda\tau}\mathbf{1}\{A = 0\}] \leq \sum_{n \geq 1} e^{-\lambda n} = \frac{1}{e^\lambda - 1}.$$

For $n \geq m_h := \lceil 2h/D_1 \rceil$, survival instead implies $S_n < h \leq nD_1/2$. Thus $\Pr_1(\tau > n) \leq e^{-nD_1^2/(2R^2)} \leq e^{-\eta n}$. The same tail-sum bound yields $\mathbb{E}_1 e^{\lambda\tau} \leq e^{\lambda m_h} + e^\lambda$. Because $m_h \leq 2h/D_1 + 1$, the accepted contribution satisfies

$$e^{-\kappa h}\mathbb{E}_1[e^{\lambda\tau}\mathbf{1}\{A = 1\}] \leq e^{-\kappa h}(e^{\lambda m_h} + e^\lambda) \leq 2e^\lambda.$$

Combining the two contributions proves Eq. (D5), for example with the conservative choice

$$K = \max\left\{1,\ e^{\lambda m_0} + e^\lambda, 2e^\lambda + \frac{1}{e^\lambda - 1}\right\}. \qquad \square$$

The threshold-dependent factor has been absorbed entirely on accepted tests; the rejected-test bound is independent of h.

To combine these bounds, stop the procedure after full coverage, N attempts or d completions, before imposing a diagnostic cap. At the start of attempt i, condition on the preceding history. Its threshold h_i is then fixed, the collection stays unchanged during the test, and fresh reports follow the specified law conditional on the candidate's type. Since Eq. (D5) holds for either type with the same constants, it also holds for their conditional mixture. Pad stopped paths with $\tau_i = A_i = h_i = 0$ up to N attempts. Iterating the conditional bound gives

$$\mathbb{E}\exp\left\{\lambda\sum_{i=1}^{N}\tau_i - \kappa\sum_{i=1}^{N}h_iA_i\right\} \le K^N.$$

Before full coverage, at least one class remains missing, so the stage novelty rate is $p = r/d \ge 1/d$ and $h_i \le h_{\max} = \log(6d^2/\delta)$. At most d tests are accepted; hence $\sum_i h_iA_i \le dh_{\max}$ on every path. Markov's inequality now gives

$$\Pr\left\{\sum_i \tau_i > L\right\} \le K^N \exp\{-\lambda L + \kappa d h_{\max}\}.$$

Consequently, the deterministic cap

$$L = \left\lceil \frac{N\log K + \kappa d h_{\max} + \log(3/\delta)}{\lambda} \right\rceil = O_{a,b}(d\log(d/\delta)) \tag{D6}$$

is exceeded with probability at most $\delta/3$. We now impose all three caps. Reserve an operation's cost and check its applicable cap before executing it; stop when another attempt, diagnostic, or completion would exceed N, L, or d, respectively. Truncating a test at the diagnostic cap can cause failure, but cannot overspend. Combining the three failure events proves full coverage with probability at least $1-\delta$ and cost at most $dC + cL$ on every path. On success, the completions supply one output per class; a failed path may include duplicates. For $c \le C$, this is the desired high-confidence upper bound with at most d completions. When $c > C$, use blind generation instead, with the fixed completion count derived below. The constants depend on a, b and worsen as the two report laws become harder to distinguish.

D.3 The matching resource lower bound

Even perfect decisions about which encountered candidates to complete cannot supply a class never encountered. We use this fact to lower-bound the fixed budget for every policy, including those that repeat checks or buffer candidates. After n iid uniform first-touch classes, let Z be the number of classes not encountered and let $\mu = d(1-1/d)^n$. Then

$$\mathbb{E}Z^2 = \mu + d(d-1)(1-2/d)^n \le \mu + \mu^2, \qquad \Pr(Z>0) \ge \frac{\mu}{1+\mu}, \tag{D7}$$

where the first expression counts single missing classes and pairs of missing classes. The probability bound follows from Cauchy–Schwarz applied to $Z\mathbf{1}\{Z>0\}$: if the expected missing count μ remains substantial, there must be a substantial chance that at least one class is still missing.

A procedure with hard budget B first touches at most $\lfloor B/\min(c,C)\rfloor$ candidates. Extend its first-touch sequence by unused independent candidates if necessary. Covering all classes in its output requires encountering all classes within this sequence. Success probability at least $1-\delta$ therefore requires

$$\frac{B}{\min(c,C)} \ge \frac{\log[d(1-\delta)/\delta]}{-\log(1-1/d)}. \tag{D8}$$

It also requires $B \ge Cd$, because information alone produces no completed output. For $d \ge 2$ and $\delta < 1/2$, the right side of (D8) is of order $d\log(d/\delta)$. Taking the maximum of these two

requirements, which is at least half their sum, proves the lower high-confidence order. If only d completions are allowed, L diagnostics can first touch at most $d + L$ candidates; this gives the corresponding lower order on diagnostic work at the minimum completion count.

Without informative early observations, candidate choice before completion is independent of its class; the completed classes themselves are iid uniform. The same argument uses B/C in (D8). Conversely, $\lceil d \log(d/\delta) \rceil$ blind completions suffice by a union bound over unseen classes. The exact expected blind count is the sum of geometric waiting times with success probabilities r/d, namely dH_d. Thm. 3 follows.

D.4 The false-positive floor with one preview

Why use repeated checks when one check can already cross the profitability boundary? A single report with a nonzero false-positive rate still sends duplicates to expensive completion. Their contribution becomes particularly costly as useful candidates grow rare. For fresh-attempt one-preview screening with binary rates $\alpha > 0, \beta \le \alpha$, Eq. (C3) can be written

$$T_1^*(p) = \min\left\{\frac{C}{p}, C\left(1 - \frac{\beta}{\alpha}\right) + \frac{c_{\text{eff}}}{p}\right\}, \qquad c_{\text{eff}} = \frac{c + C\beta}{\alpha}. \tag{D9}$$

Here c_{eff} is the coefficient of the rare-candidate search cost $1/p$. It includes both the preview charge and the completion expense caused by false positives, divided by the chance of accepting a useful candidate.

Summing the optimal stage costs over $p_r = r/d$ is exact for the policy class that makes a blind-or-one-preview decision on a fresh candidate and discards it after that decision or completion. The sum is at least Cd and at least $dH_d \min(C, c_{\text{eff}})$, and at most either CdH_d or $Cd + c_{\text{eff}} dH_d$. This gives the order $Cd + dH_d \min(C, c_{\text{eff}})$ for that class. The false-positive floor $C\beta/\alpha$ remains even when previews themselves become very cheap. Repeated fresh reports address this floor by reducing the test's false-positive rate as useful candidates become rarer, as in App. D.1.

Thm. 2's profitability threshold leaves open the best finite-budget policy. Sharper repeated-testing costs depend on the report laws; unequal target probabilities and buffered candidates call for dynamic programs that track the remaining targets, budget, and candidate information.

D.5 Numerical illustration: price, freshness, and resource caps

The simulation asks how the theoretical choices behave at particular budgets: whether cheap fresh reports improve discovery, how much completion protection costs, and what happens when repeated reports contain no new information. It uses the stated encounter model, with $d \in \{4, 8, 16\}$ uniform persistent classes, channels $(a, b) \in \{(0.8, 0.3), (0.7, 0.5)\}$, prices $c/C \in \{0.05, 0.20\}$, and budgets $B/(dC) \in \{1, 1.1, 1.25, 1.5, 2, 3, 4, 6\}$. These give 96 settings, each evaluated with four policies and 2,000 independent paths per policy. Twelve blind controls make d completions at the minimum budget dC. Candidate class and novelty are latent to the policy; the simulator uses them to draw the specified reports. Completion alone reveals class and updates the missing-class count.

Policies and reservations. The completion-capped policy uses the test in App. D.2, with lower threshold $-\log 2$, upper threshold $\log(6d/(p\delta))$, $p = r/d$, and $\delta = 0.1$. Its caps are $\lceil 2d \log(3d/\delta) \rceil$ attempts, d completions, and $\lfloor (B - dC)/c \rfloor$ diagnostics. The budget-derived diagnostic cap may be smaller than the sufficient count L in Eq. (D6). The uncapped comparator uses upper threshold $\log(2/p)$, no completion cap, and the same total budget. Blind completion draws iid classes until full coverage or budget exhaustion. Copied-report screening reuses one noisy bit per candidate, preserving one-report marginals but violating conditional freshness.

We enforce the budget in integer resource units, with $C = 100$, $c \in \{5, 20\}$, and integer budgets. Each operation's cost is reserved before execution; diagnostic caps use integer division, and execution stops at the completion cap.

The base seed is 20260917, with independent PCG64 generators keyed by setting and policy.

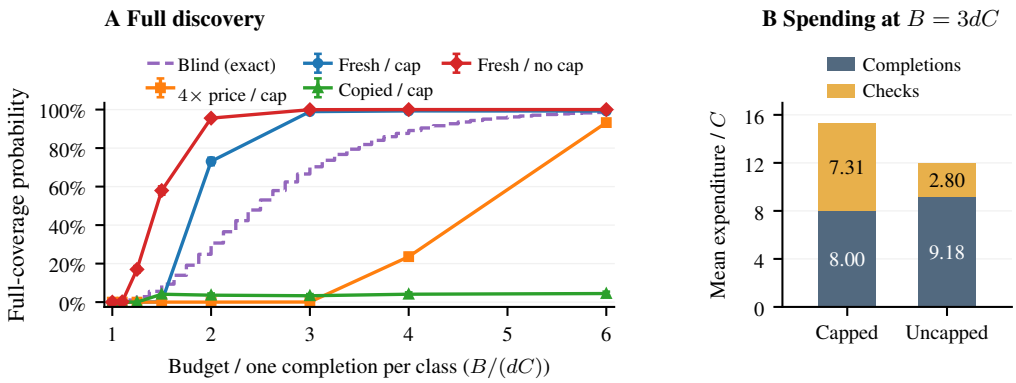


Figure 4: **More checks can protect completion spending.** The numerical model has eight persistent classes, report rates $(0.8, 0.3)$, and 2,000 paths per condition. **A:** Probability of collecting all classes as the budget increases. Checks cost $0.05C$, or $0.20C$ for the curves marked $4\times$. The capped rule allows at most eight completions; the uncapped rule also uses a lower acceptance threshold. Error bars show pointwise Wilson 95% intervals; the blind reference is exact. **B:** Mean expenditure across all paths at $B = 3dC$, with fresh $0.05C$ checks. The uncapped rule spends less here; the cap guarantees a completion limit on every path.

Uncertainty and exact reference. Pointwise Wilson intervals (Wilson, 1927) describe uncertainty in each simulated success probability. The blind reference in Fig. 4 is exact after $\lfloor B/C \rfloor$ draws: with $p_{0,0} = 1$, the occupancy recurrence

$$p_{n+1,j} = \frac{j}{d}\, p_{n,j} + \frac{d-j+1}{d}\, p_{n,j-1}$$

tracks $p_{n,j}$, the probability that n completed draws have revealed exactly j classes. The first term repeats one of those classes; the second discovers the jth class from a state with $j-1$. Thus $p_{n,d}$ is the full-coverage probability, including $d!/d^d$ at $n = d$. At the minimum budget dC, the capped policy cannot buy a diagnostic under its reservation; blind fallback is feasible. Tab. D1 reports simulated success for every class/channel/price condition at $3dC$.

Table D1: Full-coverage success (%) at $B = 3dC$ for every class/channel/price condition, with 2,000 independent paths per cell. The tested completion-capped policy has low discovery success when diagnostics are costly or weakly informative.

Classes	(a, b)	c/C	**Capped, fresh**	**Capped, copied**	**Uncapped**	**Blind**
4	(0.8,0.3)	0.05	99.35	32.15	99.95	87.00
4	(0.8,0.3)	0.20	9.55	30.50	91.25	88.10
4	(0.7,0.5)	0.05	0.20	14.05	79.15	86.80
4	(0.7,0.5)	0.20	0.00	0.00	6.25	86.65
8	(0.8,0.3)	0.05	98.95	3.30	99.95	72.05
8	(0.8,0.3)	0.20	0.05	3.95	78.25	70.25
8	(0.7,0.5)	0.05	0.00	0.65	57.15	68.90
8	(0.7,0.5)	0.20	0.00	0.00	0.00	71.55
16	(0.8,0.3)	0.05	99.10	0.10	100.00	43.40
16	(0.8,0.3)	0.20	0.00	0.00	60.20	45.95
16	(0.7,0.5)	0.05	0.00	0.00	30.05	46.20
16	(0.7,0.5)	0.20	0.00	0.00	0.00	46.10

For eight classes, channel $(0.8, 0.3)$, and price $0.05C$, capped screening succeeds on 98.95% of paths at $3dC$, averaging 146.14 diagnostics and 8.00 completions, with a pathwise cap of eight. The exact blind success probability is 70.28%; the 72.05% in Tab. D1 is its 2,000-path simulation estimate. The uncapped rule reaches 99.95% with 56.05 diagnostics and 9.18 completions on average; copied reports reach 3.30%. Fig. 4B separates the mean expenditure on checks from that on completions,

averaging all 2,000 paths per policy, including failures. The components sum to $15.30C$ for capped and $11.98C$ for uncapped screening. Thus the uncapped rule has higher success and lower mean expenditure at this budget, while the capped rule guarantees at most eight completions on every path.

Increasing the check price to $0.20C$ at the same $3dC$ budget exposes the cost of completion protection. Capped screening succeeds on only 0.05% of paths, compared with 70.25% for blind completion and 78.25% for uncapped screening (Tab. D1); all three are simulation estimates, and the exact blind probability remains 70.28%. The capped rule reserves eight completions and spends its diagnostic allowance seeking stronger evidence before accepting candidates. Its budget-derived check cap need not meet the sufficient bound in Eq. (D6). A pathwise limit on completions can therefore be expensive at a particular deadline, even when a less conservative screening rule succeeds more often than blind completion.

At budget exactly dC, buying any check displaces a necessary completion, so full-coverage probability cannot exceed the blind value $d!/d^d$. The budget must therefore cover both information and useful work; an expected-cost improvement alone does not establish a gain at every deadline.

E Output format and coordinated sampling

Can a batch cover more correct answers by changing how its completions are sampled, without purchasing previews? We use QuasiMoTTo's existing sampler to test when coordination reduces repetition of useful content. The key comparison asks whether naming an entity before explaining it makes coordination more useful than explaining first. This connects the repeated answers in Fig. 1B to a choice available before generation begins. We first compare the two formats under matched sampling conditions, then test coordination at a later answer delimiter and on questions not used in development.

E.1 Comparing independent and coordinated generation

A 2×2 QAMPARI design crosses two sampling rules with two output formats. Independent sampling draws randomness separately for each output; QuasiMoTTo lattice sampling coordinates those draws across the batch (Li et al., 2026). Both use the same vLLM implementation (Kwon et al., 2023), with independent random coordinates or a shared randomly shifted rank-one lattice. Entity first requests `ANSWER: <entity>` followed by a brief justification; rationale first requests the reverse. Only that output-contract sentence changes: question, evidence, and diversity instruction are shared. Comparing samplers within a format isolates the change in dependence. Comparing formats also changes the instruction, so their individual-output distributions may differ.

The same 164 questions receive retrieved evidence (ALCE's top 20 GTR passages (Gao et al., 2023; Ni et al., 2022)) or oracle proof snippets on Qwen3.5-9B (Qwen Team, 2026a), Gemma-4-12B-it (Gemma Team et al., 2026), and Qwen3.5-27B. Both evidence conditions use the QAMPARI eligibility rule in App. F.1. A disjoint 160-question development set selects the prompt and 384-token cap by compliance and truncation, without comparing sampler effects. Every factorial cell uses that cap, temperature 1.0, and top-$p = 0.95$.

The parser takes the first answer line under entity first and the last under rationale first, then applies shared normalization and alias-aware gold matching. Invalid outputs and order violations remain in the denominator, without regeneration. To see how coordination changes coverage, separate correct outputs from repeated correct answers. For a batch of k outputs, let p be the fraction matching a gold entity and D the number of useful occurrences beyond the first per entity. Distinct coverage is exactly $C = kp - D$, hence

$$\Delta C = k\Delta p - \Delta D$$

for each within-format coordinated-minus-independent contrast. The first term measures the change in correct outputs, counting repetitions; the second removes the change in repeats. Thus coverage can improve even without more individually correct outputs. Fig. 2 displays these two contributions, with repeats avoided shown as $-\Delta D$. Each interval is computed for its own paired contrast rather than by adding interval endpoints.

E.2 PRIMARY INTERACTION AND UNCERTAINTY

The primary question is whether the same sampler helps more under one format. For each question and seeded repetition, or *stream*, we subtract independent coverage from coordinated coverage within each format, then subtract the rationale-first gain from the entity-first gain. This difference of gains is the *interaction*; a positive value means that naming the entity first increases the benefit of coordination, regardless of the formats' raw coverage levels.

We average eight streams within each question, then resample questions with 10,000 paired bootstrap draws (Efron & Tibshirani, 1994) (seed zero). Intervals are percentile 95%. The six $k = 4$ interactions form the declared Holm family (Holm, 1979) at $\alpha = 0.05$: the adjustment controls error across these six primary tests. Within-format and decomposition contrasts are secondary and unadjusted.

Table E1: Additional distinct correct entities for $k = 4$, with paired 95% intervals. Models 9B/12B/27B denote Qwen3.5-9B, Gemma-4-12B-it, and Qwen3.5-27B; R/O denote retrieved/oracle evidence. The last column gives interaction Holm p values for the primary six-member family and the four-member sensitivity excluding 27B. Bootstrap tail probabilities are floored at $1/10001$ before adjustment; the displayed intervals are not multiplicity-adjusted.

Model / evidence	Entity-first gain	Rationale-first gain	Interaction	Holm p: 6 / 4
9B / R	$0.112\ [0.075, 0.150]$	$-0.014\ [-0.052, 0.023]$	$0.127\ [0.076, 0.178]$	0.0006 / 0.0004
12B / R	$0.038\ [0.018, 0.062]$	$-0.014\ [-0.041, 0.014]$	$0.052\ [0.014, 0.094]$	0.0066 / 0.0066
27B / R	$0.094\ [0.061, 0.129]$	$0.010\ [-0.027, 0.047]$	$0.084\ [0.035, 0.133]$	0.0012 / —
9B / O	$0.202\ [0.152, 0.252]$	$0.002\ [-0.048, 0.051]$	$0.200\ [0.131, 0.270]$	0.0006 / 0.0004
12B / O	$0.085\ [0.053, 0.119]$	$0.009\ [-0.029, 0.048]$	$0.076\ [0.026, 0.126]$	0.0060 / 0.0060
27B / O	$0.186\ [0.135, 0.238]$	$-0.008\ [-0.050, 0.034]$	$0.194\ [0.127, 0.262]$	0.0006 / —

All six interactions survive adjustment: naming the entity before explaining it increases the coverage benefit of coordination. Under entity first, all six duplicate-count contrasts are negative with intervals excluding zero, so fewer repeated correct answers support the gain. The individual-usefulness intervals include zero. All rationale-first coverage intervals include zero. The comparison therefore identifies an output-format choice that makes this sampler more useful for distinct-answer coverage; diversity in early tokens alone is insufficient evidence of that benefit.

E.3 ACCOUNTING SENSITIVITIES AND SECONDARY BATCH SIZE

The decision to include Qwen3.5-27B under the shared-prefix cost-accounting rule was made after its results were available. Generated outputs were unchanged, and all reservations and total allowances were respected. Excluding this backbone leaves all four remaining format interactions significant after Holm correction (Tab. E1).

For Gemma, a sensitivity analysis retains questions for which measured prefix reuse was at least the amount assumed when reserving the call's cost, leaving 37 retrieved-evidence and 45 oracle-evidence questions from the respective 164-question populations. The groups differ in prompt length. Both restricted $k = 4$ intervals include zero, as do the differences between the two groups (Tab. E2).

Table E2: Gemma interactions for all questions and for the subset whose measured prefix reuse was at least the amount assumed when reserving the call's cost. Full-population and restricted intervals resample questions; the last column independently resamples the two disjoint question strata. All entries are distinct correct entities with unadjusted 95% intervals.

Evidence	k	Full ($n = 164$)	Reuse threshold met	Met minus below
Retrieved	4	$0.052\ [0.014, 0.094]$	$0.061\ [-0.003, 0.135]$	$0.012\ [-0.071, 0.099]$
Oracle	4	$0.076\ [0.026, 0.126]$	$0.086\ [-0.008, 0.186]$	$0.014\ [-0.098, 0.128]$

With sixteen outputs and three streams on the same 164 questions, the format interactions are $0.226\ [0.132, 0.325]$ and $0.368\ [0.244, 0.492]$ on Qwen9B under retrieved and oracle evidence, and

0.041 $[-0.008, 0.092]$ and 0.108 $[0.035, 0.181]$ on Gemma. These secondary 95% intervals are unadjusted; batches are randomized separately at each size.

In timing checks with four outputs and a 256-token completion cap, lattice sampling took 23.1% longer than stock sampling on Qwen9B and Gemma12B, and 10.2% longer on Qwen27B. The factorial comparison holds output count and sampler implementation fixed, with both arms generating the full requested rationale.

E.4 Coordinating at the answer delimiter

We also test coordination after a rationale has been generated. On the same 164 QAMPARI questions with retrieved evidence, four outputs sample independently until an online detector recognizes `ANSWER:`; a new lattice draw then coordinates their remaining tokens. The matched control resumes independently from the same stopped token prefixes. Both use a 384-token cap including the prefix, and outputs that never reach the delimiter remain included. Averaging twelve streams within each question, the coverage differences are $+0.005$ $[-0.003, 0.013]$ on Qwen3.5-9B and -0.004 $[-0.010, 0.002]$ on Gemma-4-12B, with pointwise 95% intervals from 10,000 paired question-bootstrap resamples. Among outputs that reach the delimiter and yield a parsed entity, 88–95% already name that entity in the preceding rationale across the retrieved and oracle conditions. The visible delimiter therefore often follows the first textual statement of the answer.

E.5 Entity-first coordination on AmbigQA

We compare independent and lattice sampling with uninterrupted entity-first outputs on 256 questions from AmbigQA's official training split (Min et al., 2020), held out from prompt and procedure development. These questions satisfy the AmbigQA eligibility rule in App. G.5. Both backbones receive gold-supporting snippets, with four streams of four outputs and a 384-token cap. Coverage counts distinct targets within one reference annotation per question. Lattice sampling adds 0.111 $[0.073, 0.149]$ correct entities per batch on Qwen3.5-9B and 0.016 $[0.000, 0.031]$ on Gemma-4-12B. These secondary, unadjusted 95% intervals resample questions after averaging streams; Gemma's unadjusted p-value is 0.0526.

F Selecting from shared early information

A promising attempt may add little if its likely answer is already represented in the batch (Fig. 1A–B). We compare ranking, deduplication, and overlap-aware selection using shared candidate previews, first for entity answers and then for cited claims. Fixed-order selection provides a reference that ignores the previews and, deployed on its own, could skip them altogether. The controlled eight-prefix and SummHay comparisons hold early information and the selected count fixed; App. G compares complete procedures under the same online allowance, including the cost of obtaining that information.

F.1 QAMPARI: the controlled eight-prefix study

Decision and population. The QAMPARI comparison (Amouyal et al., 2023) in Tab. F1 chooses four of eight unfinished attempts before seeing their final answers. Each chosen attempt supplies four continuations, giving sixteen outputs in total. The study uses 256 official-test questions, selected by a fixed SHA-256 ordering from 828 eligible questions in the 1,000-question file. Eligibility requires 5–20 gold targets and nonempty proof snippets totaling at most 1,000 words, after a 24-snippet cap. Giving every selector the same evidence and candidate pool isolates allocation after that information is supplied.

Generation and early information. Qwen3.5-9B (Qwen Team, 2026a) constructs eight prefixes, or unfinished attempts, using diversity instructions without gold answer labels and a 64-token cap. Each has four continuations: at most 200 reasoning tokens, then an extraction capped at 24 tokens returning one entity or `NONE`. Temperature is 1.0, top-$p = 0.95$, and seeds are 13, 29, and 47. The $24{,}576$ terminal outcomes have 99.97% validity. Before selection, we obtain two stochastic 24-token entity previews, a greedy answer prediction called the *intent* with its confidence, and scores from a verifier checking evidence support. The verifier uses the same Qwen backbone to judge each

distinct preview entity against the question and evidence. Its score is the next-token logit difference between `YES` and `NO` after an `ANSWER:` primer. Generation uses Hugging Face Transformers (Wolf et al., 2020).

Estimator and policies. The logistic correctness estimator is fitted on 2,998 prefix–entity pairs from 96 disjoint development questions, using reference-entity matches as binary labels. Its four standardized inputs are the verifier score, the greedy intent's first-token probability, an indicator that all valid entity previews agree, and a constant 224-token completion-cost proxy. A single valid entity preview counts as agreement; preview frequencies still divide by both previews. Fitting minimizes summed logistic loss with unit-strength L2 regularization on the weights and an unpenalized intercept. The fit and standardization remain fixed during evaluation. Let $\widehat{q}_i(e)$ be the predicted correctness of entity e from prefix i, and let $\widehat{\mathrm{freq}}_i(e)$ be the fraction of its previews naming e. Multiplying these quantities gives a contribution profile; product union combines the profiles into the batch score

$$\widehat{p}_i(e) = \widehat{\mathrm{freq}}_i(e)\,\widehat{q}_i(e), \qquad \widehat{U}(S) = \sum_e \left[1 - \prod_{i \in S}(1 - \widehat{p}_i(e))\right]. \tag{F1}$$

Any remaining probability mass stays on the null outcome rather than being redistributed among the predicted entities. A profile summarizes predicted contributions for selection; it does not model the four continuations' joint outcome or shared uncertainty about whether an entity is correct. The batch score can consequently prefer a redundant batch to a complementary one; App. A.5 gives a concrete example and the dependence assumptions for a probabilistic interpretation.

Every profile policy sees the same candidates and profiles. Additive ranks each prefix by its individual score; product union repeatedly adds the prefix with the largest predicted contribution beyond those already selected (Nemhauser et al., 1978; Lin & Bilmes, 2011), breaking ties by action identifier. Confidence ranks greedy-intent confidence. Intent deduplication takes the highest-confidence prefix for each distinct normalized intent, then fills any remaining places by confidence. All five arms select four prefixes at the primary operating point. Fixed order takes the first four candidates by sorted identifier. The primary product-union comparison imposes no false-positive constraint.

Metrics. Let $\mathcal{E}(S)$ contain the distinct valid normalized names from selected continuations, $\mathcal{G}$ the gold targets, and $\mathcal{M}(e)$ the empty set or singleton gold identifier assigned by the fixed alias matcher. Then

$$C(S) = \left|\bigcup_{e \in \mathcal{E}(S)} \mathcal{M}(e)\right|, \qquad R(S) = C(S)/|\mathcal{G}|,$$
$$P_{\mathrm{name}}(S) = \frac{|\{e \in \mathcal{E}(S) : \mathcal{M}(e) \neq \varnothing\}|}{|\mathcal{E}(S)|}, \quad F_{\mathrm{reported}}(S) = \frac{2P_{\mathrm{name}}(S)R(S)}{P_{\mathrm{name}}(S) + R(S)}. \tag{F2}$$

Coverage C counts distinct correct targets, whereas name precision P_{name} asks what fraction of emitted names match any target. Empty denominators give zero. Unmatched valid names count against precision; invalid outputs and `NONE` contribute no entity. Two different names can match one target: both receive precision credit, but together contribute only one covered target. Reported F1 therefore combines name-level precision with target-level recall and differs from the target F1 in App. G. Selection never uses gold alias identifiers.

Mean outcomes and interpretation. We average each policy's coverage, F1, and precision over the 256 questions and three generation seeds. All five policies use the same question–seed pairs and select four prefixes.

Joint overlap scoring and simple intent deduplication both yield higher mean coverage than individual ranking. Union gains 0.279 correct entities over additive; its mean is 0.040 above deduplication and 0.096 above fixed order, which itself exceeds additive by 0.182. Overlap correction improves mean coverage over additive ranking on this shared candidate pool, while the nearby deduplication mean identifies a simple alternative for complete-procedure evaluation. Union-minus-additive mean name precision is $+0.10$ percentage points, while mean per-continuation precision, which credits repeated correct outputs separately, falls by 1.91 points. More distinct correct answers can coexist with weaker individual correctness.

Cross-backbone confirmation. A separate study evaluates pairwise overlap correction on two other backbones. It uses a separate 78-question confirmation split and three seeds on Gemma-4-12B-it

Table F1: **Joint selection improves mean coverage over individual ranking from the same predictions.** QAMPARI uses Qwen3.5-9B on 256 official-test questions and three seeds (768 paired records). All rules select four prefixes from the same candidate pool with shared precomputed previews; fixed order takes the first four identifiers. Coverage counts distinct gold targets. Reported F1 combines name-level precision with target-level recall, as defined in App. F.1.

Allocation policy	Distinct correct	Recall (%)	Name precision (%)	F1 (%)
Confidence top-k	1.928	28.22	45.72	33.12
Additive ranking	2.139	31.02	47.38	35.57
Fixed order	2.322	33.65	45.96	37.07
Intent deduplication	2.378	34.53	47.32	38.16
Product-union selection	2.418	34.99	47.47	38.69

(Gemma Team et al., 2026) and Qwen3.6-35B-A3B (Qwen Team, 2026b). It retains eight prefixes, two previews, four selected prefixes and four continuations each, oracle evidence, the same token caps, temperature and top-p. Each backbone's estimator is fixed after fitting on 48 disjoint questions. The primary score subtracts predicted pairwise overlap from summed individual contributions, using the pairwise Bonferroni score in App. A.3 with penalty weight one. Against identical-information additive ranking, coverage gains are $+0.248$ $[0.158, 0.342]$ and $+0.299$ $[0.171, 0.436]$, respectively. These pointwise 95% percentile intervals use 10,000 paired question-bootstrap draws (seed 0), averaging the three observed seeds within each question. Inference therefore conditions on those generation streams. Two-sided null-centered bootstrap tests remain significant after Holm adjustment across the two primary backbone comparisons ($p = 0.0002$ each). Product union's secondary mean gains are $+0.239$ and $+0.316$. Selected-terminal costs match within 2% for the primary comparison; this excludes full acquisition costs.

Simple deduplication, product union, and pairwise correction have similar mean coverage in this study. Coverage means for intent deduplication, union and pairwise scoring are $1.581/1.624/1.632$ on Gemma and $2.444/2.500/2.483$ on Qwen; Qwen deduplication's reported F1 is 0.4300, versus union's 0.4288. The empirical gain over individual ranking can therefore be obtained with pairwise overlap correction. The need for higher-order information in App. A.4 concerns worst-case instances. These confirmation populations are analyzed separately from Tab. F1.

F.2 Shared coordinates and independent terminal outcomes

F.2.1 Shared answers across candidates

Overlap correction relies on recognizing when different candidates predict the same entity. In a separate study with sixteen candidates, two previews and one terminal outcome per candidate, a raw profile assigns each valid normalized preview its observed frequency; invalid previews retain null mass. Here an entity *coordinate* is simply the shared label used to match that answer across profiles. Renaming all labels consistently leaves choices unchanged. Giving each candidate its own labels instead preserves its individual score and preview agreement but removes all overlap between candidates: union and additive then agree algebraically, including ties. This control identifies what the score uses; its zero difference is a mathematical identity rather than an estimated effect. This study uses a separate population and estimator from the eight-prefix comparison, with one continuation per candidate.

A stronger intervention keeps overlap in the profiles but reassigns which candidates share an answer. The intervention selects four of sixteen candidates on 160 development questions, three backbones, two evidence conditions, and three seeds (2,880 records), with two previews and one continuation per candidate. Its logistic model is fitted on disjoint calibration questions using preview agreement, presence in evidence, novelty relative to earlier previews, and prefix length; it has no verifier-logit feature. Reference rules include identical-profile additive ranking, confidence, backfilled intent deduplication, fixed order, and random subsets (20 seeded draws per record).

Represent each predicted contribution as a link between a candidate and an entity, then exchange links while preserving their counts at both ends. These *degree-preserving edge swaps* preserve

each candidate's weight multiset, singleton mass, and entity count, and each entity's number of contributing candidates. The resulting profiles assign different candidates to the same predicted answers while retaining these individual and entity-level summaries. Tab. F2 reports both the contrasts against additive ranking and the direct loss from reassignment. Under oracle evidence, the intervals for the loss from reassignment are wholly positive on Qwen3.5-9B and Gemma-4-12B; all three retrieved-evidence intervals include zero. This direct comparison tests whether the actual assignment of answers contributes to coverage. The exploratory, pointwise intervals use 10,000 paired question-bootstrap draws (seed 0), keeping each question's three generation seeds together.

Table F2: Reassigning shared answers across sixteen candidate profiles. The first three numerical columns give union minus identical-profile additive coverage at $k = 4$; the final column directly subtracts reassigned from original calibrated-union coverage. Pointwise 95% paired question-bootstrap intervals use 160 development questions and three seeds per cell. Reassignment preserves each candidate's singleton score and each entity's sharing degree. Each contrast is exploratory and analyzed within its own backbone and evidence condition.

Backbone / evidence	Raw profiles	Calibrated profiles	Calibrated, reassigned	Original minus reassigned
Qwen3.5-9B / Oracle	$+0.092$ $[0.042, 0.142]$	$+0.154$ $[0.094, 0.217]$	$+0.017$ $[-0.033, 0.067]$	$+0.138$ $[0.079, 0.198]$
Qwen3.5-9B / Retrieved	-0.013 $[-0.052, 0.029]$	$+0.031$ $[0.000, 0.062]$	$+0.000$ $[-0.029, 0.027]$	$+0.031$ $[-0.002, 0.067]$
Gemma-4-12B / Oracle	$+0.008$ $[-0.029, 0.046]$	$+0.056$ $[0.008, 0.104]$	-0.015 $[-0.056, 0.027]$	$+0.071$ $[0.027, 0.115]$
Gemma-4-12B / Retrieved	-0.008 $[-0.040, 0.023]$	$+0.015$ $[-0.015, 0.044]$	$+0.008$ $[-0.023, 0.040]$	$+0.006$ $[-0.021, 0.033]$
Gemma-4-26B / Oracle	-0.008 $[-0.040, 0.021]$	$+0.021$ $[-0.010, 0.052]$	$+0.004$ $[-0.019, 0.027]$	$+0.017$ $[-0.008, 0.042]$
Gemma-4-26B / Retrieved	-0.013 $[-0.031, 0.006]$	-0.010 $[-0.033, 0.013]$	-0.006 $[-0.027, 0.013]$	-0.004 $[-0.025, 0.017]$

Calibration also changes individual ranking. Under Qwen oracle evidence, union-minus-additive rises from 0.092 to 0.154, but the direct calibrated-minus-raw union gain is 0.058 $[-0.004, 0.121]$. The two aggregation contrasts alone do not establish a calibration benefit.

F.2.2 Predicted scores and realized coverage

On the same 160-question Qwen3.5-9B development bank, greedy product-union selection chooses four of sixteen prefixes using a correctness fit fixed on 96 separate questions. We compare it with modal-name deduplication, fixed order, and uniform selection. Deduplication ranks candidates by preview agreement, selects distinct most frequent preview names, and then fills any remaining places from that ranking. The score difference evaluates union and each reference with the same product-union objective; uniform averages all 1,820 four-prefix subsets. Realized coverage counts distinct correct targets across four terminal continuations per selected prefix. Mean score differences favor union in both evidence conditions, while all six coverage intervals include zero (Tab. F3).

For one continuation per selected prefix, averaged over two independent terminal draws, exact rather than greedy maximization changes coverage by $+0.001$ $[-0.011, 0.014]$ under oracle evidence and -0.001 $[-0.006, 0.004]$ under retrieved evidence. These pointwise 95% intervals use the same question-bootstrap procedure.

Rescoring the output sets with four continuations per selected prefix under name precision and target precision (Apps. F.1 and G.2) changes every mean union–reference F_1 contrast on this bank by less than 0.0006, with each output set scored before averaging.

Table F3: Product-union selection minus each reference on 160 QAMPARI development questions. Coverage combines four continuations from each of four selected prefixes; the predicted-score column uses the fixed preview-based objective. The exploratory pointwise 95% coverage intervals use 10,000 paired question-bootstrap resamples, averaging three streams within each question (seed 0).

Evidence	Union minus	Predicted score difference	Coverage difference (95% interval)
Oracle	Backfilled modal-name deduplication	+0.390	+0.073 [-0.008, +0.156]
Oracle	Fixed order	+0.885	+0.083 [-0.013, +0.181]
Oracle	Uniform	+0.798	+0.047 [-0.020, +0.116]
Retrieved	Backfilled modal-name deduplication	+0.278	+0.033 [-0.029, +0.098]
Retrieved	Fixed order	+0.435	-0.002 [-0.069, +0.067]
Retrieved	Uniform	+0.336	+0.021 [-0.022, +0.064]

F.2.3 Prediction across independent terminal outcomes

A favorable completed subset is useful only if it can be selected before those outcomes are known. The independent-terminal study holds sixteen prefixes and their two previews fixed, then draws two independent terminal outcomes per candidate on 96 hash-selected questions, two backbones, and two evidence conditions. Selection reads only the shared previews. The reference averages coverage over all $\binom{16}{4}$ subsets of size four. It is therefore the exact expected coverage of uniform random selection, rather than fixed identifier order.

The five rules chosen before terminal outcomes are observed are calibrated-profile union and additive, raw deduplication, fixed order, and an ordering by preview agreement. Across two backbones, two conditions, and two terminal draws, none of their 40 rule-minus-uniform contrasts has a 95% interval wholly above zero. Three intervals favor uniform selection: Gemma additive under retrieved evidence on draw 13, and Gemma raw deduplication under both evidence conditions on draw 29. These are exploratory question-bootstrap intervals without multiplicity adjustment.

Table F4: Independent terminal outcomes with the prefix and previews fixed. Differences in distinct correct entities against exact uniform size-four expectation, with pointwise 95% paired question-bootstrap intervals; 96 questions per cell. Seeds identify the two terminal draws. These comparisons are exploratory.

Backbone / evidence	**Draw**	**Union − uniform**	**Additive − uniform**
Gemma-4-12B / Retrieved	13	−0.042 [-0.101, +0.013]	−0.042 [-0.081, -0.003]
Gemma-4-12B / Retrieved	29	−0.027 [-0.093, +0.042]	−0.017 [-0.082, +0.047]
Gemma-4-12B / Oracle	13	+0.070 [-0.007, +0.150]	−0.045 [-0.118, +0.028]
Gemma-4-12B / Oracle	29	+0.052 [-0.037, +0.141]	−0.053 [-0.138, +0.029]
Qwen3.5-27B / Retrieved	13	+0.020 [-0.064, +0.108]	+0.010 [-0.069, +0.087]
Qwen3.5-27B / Retrieved	29	−0.009 [-0.072, +0.055]	−0.029 [-0.092, +0.035]
Qwen3.5-27B / Oracle	13	+0.061 [-0.045, +0.164]	−0.002 [-0.106, +0.102]
Qwen3.5-27B / Oracle	29	+0.094 [-0.018, +0.206]	−0.010 [-0.130, +0.107]

Raw deduplication does not backfill here: it selects 2.44–2.59 candidates on Gemma and 3.75–3.78 on Qwen. Its execution count therefore differs from fixed-count selection. A second preview changes choices without a stable coverage gain. To distinguish a lucky completed subset from repeatable candidate quality, select the best subset on one terminal draw and evaluate it on the other. Across the eight backbone/evidence/draw cells, uniform selection has expected coverage 0.849 distinct entities. Choosing a subset on one terminal draw and scoring it on the independent draw gives 0.905; choosing with knowledge of the evaluated draw gives 1.363. The other-draw selector therefore recovers 10.9% of the mean hindsight-minus-uniform gap, a descriptive ratio. This comparison uses one terminal draw to select for the other, so it measures transfer across outcomes rather than preview-based prediction. The control explains why hindsight headroom alone cannot establish usable early information.

F.3 SUMMHAY: PREDICTED CLAIMS AND CITED INSIGHTS

Can the same selection idea help when the desired contributions are supported claims rather than entity names? The SummHay extension (Laban et al., 2024) chooses among *document views*, each a subset of a larger collection called a haystack. The split keeps haystacks separate: two fitting haystacks containing 17 tasks in total, two development haystacks containing 19 tasks in total, and six confirmation haystacks containing 56 tasks in total. Eight fixed views combine BM25 (Robertson & Zaragoza, 2009) and dense-retrieval ranks 1–10 and 11–20, diversified (Carbonell & Goldstein, 1998) and reciprocal-rank-fused retrieval (Cormack et al., 2009), and the first and last ten documents. Views use only the query and documents.

Qwen3.5-9B produces two 24-token previews per view. Each preview is represented by a *claim signature*: its entities, predicate, polarity, and qualifiers. To recognize paraphrases as the same predicted contribution, BGE embeddings (Xiao et al., 2024) and bidirectional DeBERTa-MNLI entailment (He et al., 2021; Williams et al., 2018) merge signatures using thresholds selected on fitting haystacks, adapting semantic-equivalence clustering (Kuhn et al., 2023). Profiles use preview frequency, source-entailment scores, confidence in the most common signature cluster, and prompt length plus generation cap as a cost proxy. Missing parses retain null mass. Gold insights are used for fitting and evaluation, never confirmation-time selection.

Each rule selects four of the eight views. Fixed order takes the first four views by identifier; signature deduplication prioritizes views with distinct most likely claim signatures and fills remaining places by confidence. Each selected view emits one supported insight with citations (128-token cap, temperature 0.7, top-$p = 0.95$). Citation identifiers index the full haystack; invented citations receive no credit. Coverage measures recovered gold insights, citation score assesses their source attribution, and Joint Score combines the two. We use the official metric formulas and judgment prompt, with `gpt-5.6-luna-2026-07-09` as the coverage evaluator in place of GPT-4o. Coverage takes the majority of three judgments, shared across selection rules. Joint Score uses 168 question–seed pairs; the citation contrast uses 165 paired records from the 55 tasks with a defined citation score for both policies at every seed.

Table F5: Selection of four document views on the held-out SummHay set: six haystacks and 56 tasks. All rules select from the same candidate views; preview-based rules receive the same previews. Each policy's citation mean averages its defined task–seed scores; paired citation contrasts retain only tasks with scores defined for both policies at all three seeds. Larger values indicate better performance.

Allocation policy	**Joint Score**	**Coverage**	**Citation**
Fixed order	0.16037	0.4980	0.2997
Confidence top-k	0.15496	0.4876	0.2962
Signature deduplication	0.15163	0.4881	0.2914
Additive ranking	0.13697	0.4835	0.2656
Product-union selection	0.15574	0.4891	0.2954
Pairwise scoring ($\beta = 1$)	0.13785	0.4823	0.2685

Fixed order has the highest mean Joint Score in Tab. F5, exceeding product union by 0.00463. Compared with additive ranking, product union has a mean Joint Score gain of 0.01877, a paired citation-score gain of 0.03073 and a coverage gain of 0.0056. The Joint Score difference is positive in each of the six haystacks. The two-sided cluster sign-symmetry sensitivity gives $2/2^6 = 0.03125$, above the prespecified threshold of 0.025. The higher mean Joint Score accompanies a larger difference in citation score than in coverage.

G COMPLETE PROCEDURES AND THE COST OF INFORMATION

We compare direct generation, agreement filtering, memory-conditioned generation, and preview-based selection under a common online token allowance. Each procedure divides computation differently between obtaining information and completing answers. Coverage and target F_1 measure

the resulting answers, while charged tokens measure the resources used. The evaluations include QAMPARI and AmbigQA, with a separate held-out QAMPARI set.

G.1 Alternative uses of the allowance

The direct-generation references spend the allowance on completed answers. Single-list generation greedily produces one answer list (temperature 0, top-$p = 1$, 320-token cap). Agreement-filtered generation draws sixteen independent lists and retains names appearing in at least two, using no gold information or extra generation for filtering. Each list has a 320-token cap, temperature 1, and top-$p = 0.95$. Its threshold was chosen by mean per-question target F_1 on Qwen3.5-9B's development set (96 retrieved and 48 oracle questions), and the same threshold is used across assessment conditions and backbones. Memory-conditioned generation repeats sixteen-list batches while they fit, returns the unfiltered union of their names, conditions on previous answers, and stops after two rounds without a new answer. Repeated generation pursues additional coverage; agreement filtering uses recurrence to decide which completed answers to retain, following the broader idea of consistency-based aggregation (Wang et al., 2023).

Each admitted SCPS round purchases a fixed bundle of early information before selecting completions. At allowance 65,536, SCPS samples sixteen prefixes per round (32-token caps), obtains two previews per prefix (24-token caps), selects four, and generates one continuation each (224-token cap). Prefixes, previews, continuations, and any 24-token answer-extraction calls use temperature 1 and top-$p = 0.95$; previews and extractions stop at a newline. Completed terminal answers enter memory; previews do not. Before admitting a round, the procedure reserves enough budget for its planned operations; admission tests affordability. Execution stops when that reservation no longer fits or after 40 rounds, with a single-list fallback if the initial round cannot fit. This is a different procedure from the eight-prefix, four-continuation study in App. F. A prefix-text-diversity reference omits the entity previews and selects four of the sixteen prefixes by max-min Jaccard distance between their lowercased whitespace-token sets. It starts with the first prefix and breaks ties by prefix order. Prefix generation, continuations, memory updates, and whole-round admission follow the same iterative framework; prefix generation remains charged.

The support estimator is fitted separately by backbone and evidence condition on disjoint development data. Its features are preview agreement, evidence occurrence, novelty relative to prior previews, and prefix length; selection uses neither a verifier nor gold answers. SCPS applies product-union selection to its predicted profiles, whereas additive ranks those same profiles by individual contribution. Preview deduplication instead ranks raw agreement and uses each candidate's most frequent previewed entity, keeping distinct entities absent from memory. It may select fewer than four or none. These alternatives begin with the same selection problem, but their completed answers then update memory. Later rounds can consequently contain different prefixes and predictions; the full procedure includes this adaptive divergence.

A separate reservation-threshold screen fits a logistic estimate $\widehat{q}$ of whether completing a candidate will add a new correct name. Its inputs are preview validity and agreement, novelty relative to memory, occurrence in the evidence, and memory size. Using estimated preview-call and completion costs c and C, it solves $\mathbb{E}[(t\widehat{q} - C)_+] = c$ on calibration data and fixes the completion threshold at C/t; c excludes prefix generation. It buys previews before applying this threshold, then considers candidates in descending predicted usefulness. If all valid previewed names are already in memory or among the previews of accepted candidates, their candidate's score is capped at 0.05 before thresholding. This procedure decides which previewed candidates to complete; the purchasing rule in App. C.5 acts before observing the diagnostic.

A charged token is a measured uncached input or output token. Charged cost includes all candidate prefixes and previews, selected continuations, extraction, invalid outputs, and unsuccessful rounds. On retrieved-evidence questions, SCPS's information acquisition alone costs 45,771, 34,942, and 41,759 tokens for Qwen3.5-9B, Gemma-4-12B, and Qwen3.5-27B, already included in the totals. Dividing these sample means by total SCPS spend gives 85.3%, 79.5%, and 84.5%. Acquisition includes prefix generation as well as previews. These shares describe expenditure within the tested procedure; changing the previews would also change the information available to selection. Offline calibration and evidence preparation are outside this online allowance; CPU selection is timed separately. Before each procedure processes a question batch, its prefix cache is cleared. Calls at the

same generation stage are batched, and prompt prefixes may be reused within the batch. Admission credits only cached work attributable to the same episode; actual expenditure subtracts cache hits reported by the inference engine. These conventions are shared within each backbone. Token costs measure uncached input and output rather than elapsed time.

G.2 Populations, metrics, and comparison families

The common allowance makes the same resource ceiling available to each procedure; it does not force equal realized spending. Single-list and agreement-filtered procedures finish their specified calls even when more budget remains. Memory-conditioned generation is the separate list-based reference that can use additional rounds.

The retrieved assessment uses 164 questions from QAMPARI's official development split, selected separately from the fitting and procedure-tuning pools, with GTR-20 evidence (Gao et al., 2023; Ni et al., 2022); the oracle assessment uses a separate 256-question official-test population, disjoint from the eight-prefix study in App. F.1. All QAMPARI populations in this appendix, under either evidence condition, use the eligibility rule in App. F.1. The 164-question set is shared with the coordination study (App. E). Three seeded repetitions, or streams, are averaged inside each question. Coverage counts distinct matched gold targets; target precision divides that count by emitted-name count, and recall divides it by gold-target count. Target F_1 is their harmonic mean per episode before averaging. Thus additional aliases for an already covered target do not earn additional precision credit, unlike the name-precision metric in App. F.1. Empty outputs receive zero precision and F_1 and remain included.

Paired differences use 10,000 question-cluster bootstrap draws (seed 0) and percentile 95% intervals. The Qwen3.5-9B comparisons are post hoc and form two families. Both compare SCPS with a reference on four metrics: coverage, target F_1, name F_1, and unmatched emitted names. They cross these metrics with three settings: the same 164 assessment questions under retrieved and oracle evidence, and the 256-question oracle-test set. The twelve-test family uses agreement-filtered lists as the reference. The broader 132-test family uses eleven references: a single greedy list; the union of sixteen lists; a three-of-sixteen vote; repeated memory-conditioned batches with union, two-list, or three-list agreement; preview deduplication; additive selection; prefix-text diversity; one-preview product-union selection; and a calibrated reservation-threshold controller. Eight unobserved comparisons—the prefix-diversity and one-preview oracle-test cells, across four metrics—receive $p = 1$ for adjustment, retaining the full family without treating them as observed null results.

Gemma and Qwen3.5-27B use the same question populations, with four comparisons prespecified for each backbone: SCPS minus preview deduplication on coverage and minus agreement-filtered lists on target F_1, each under retrieved and oracle-test evidence. Holm adjustment is within these families. Tab. G1 supplies descriptive means; paired comparisons follow it.

G.3 Coverage, answer quality, and spending

G.3.1 Complete-procedure comparisons

The procedures trade coverage against answer quality and cost. In Tab. G1, memory-conditioned lists have the highest mean coverage on both Qwen backbones; prefix-text diversity has the highest on Gemma. Single-list generation has the lowest token cost in every cell. Under retrieved evidence, filtered direct generation has higher mean target F_1 and lower mean token cost than the SCPS procedure on all three backbones.

The reservation-threshold screen has approximately the same mean F_1 as filtered lists under retrieved evidence and a slightly higher mean under oracle evidence, while using 4.56 and 5.94 times their mean tokens. These results concern a completion rule after previews have been purchased.

For the following comparisons with filtered lists and preview deduplication, differences subtract the reference from SCPS. Against filtered lists, retrieved-evidence target-F_1 differences are -0.019 $[-0.036, -0.002]$, -0.026 $[-0.045, -0.007]$, and -0.039 $[-0.058, -0.021]$ for 9B, Gemma, and 27B; Holm $p = 0.1144, 0.0368, 0.0004$. The negative F_1 differences therefore survive their declared adjustments on Gemma and 27B. Coverage differences are -0.360 $[-0.553, -0.171]$, $+0.238$ $[0.045, 0.449]$, and -0.911 $[-1.177, -0.661]$; the latter two are secondary, unadjusted compar-

Table G1: Complete-procedure sample means at allowance 65,536. Coverage counts distinct gold targets; target F_1 is computed per episode, then averaged over streams and questions. Tokens are measured uncached input plus output, rounded to the nearest token. Prefix-text diversity is evaluated in five cells. Arm means and token costs are descriptive; the following text reports paired quality comparisons, with Holm adjustment for the specified families.

	Retrieved evidence, 164 questions			Oracle evidence, 256 questions		
Procedure	Coverage	F_1	Tokens	Coverage	F_1	Tokens
Qwen3.5-9B						
Single list	1.555	0.204	3,210	3.695	0.537	665
Filtered 16 lists	1.955	0.207	8,081	4.495	0.579	5,691
Memory-conditioned lists	2.764	0.081	47,246	5.240	0.352	34,755
Prefix-text diversity	2.169	0.174	59,743	—	—	—
Reservation-threshold screen	1.650	0.207	36,861	4.620	0.583	33,791
Preview deduplication	1.622	0.192	53,345	4.094	0.531	53,354
SCPS	1.596	0.189	53,685	4.215	0.539	53,462
Gemma-4-12B						
Single list	1.518	0.212	3,123	3.773	0.549	647
Filtered 16 lists	1.638	0.220	4,303	3.897	0.555	1,704
Memory-conditioned lists	2.134	0.194	11,111	4.423	0.550	5,535
Prefix-text diversity	2.167	0.185	45,653	4.710	0.561	59,165
Preview deduplication	1.870	0.208	45,618	4.495	0.577	59,967
SCPS	1.876	0.194	43,947	4.611	0.572	60,210
Qwen3.5-27B						
Single list	2.045	0.246	3,222	3.952	0.569	667
Filtered 16 lists	2.467	0.239	10,614	4.573	0.600	8,268
Memory-conditioned lists	3.793	0.071	55,041	5.421	0.465	38,185
Prefix-text diversity	2.400	0.217	57,868	4.820	0.606	56,280
Preview deduplication	1.563	0.200	49,329	2.848	0.463	45,403
SCPS	1.557	0.199	49,441	2.914	0.467	45,685

isons. On Gemma, the additional correct targets from preview selection coexist with lower final F_1.

Under oracle-test evidence, the same F_1 contrasts are -0.041 $[-0.063, -0.018]$, $+0.016$ $[-0.004, 0.039]$, and -0.133 $[-0.158, -0.107]$, with Holm $p = 0.0028, 0.2604, 0.0004$. Gemma's secondary coverage gain is $+0.714$ $[0.482, 0.983]$. These results make the objective consequential: a procedure can recover more correct targets without improving the balance of correct and incorrect answers.

Within the preview-based alternatives, SCPS's oracle-test coverage gains over preview deduplication are $+0.121$ $[0.056, 0.186]$, $+0.116$ $[0.023, 0.241]$, and $+0.066$ $[0.003, 0.130]$. The exploratory 9B comparison passes the 132-test Holm adjustment ($p = 0.0132$), as does Gemma's comparison in its prespecified family; 27B's does not. All corresponding F_1 intervals contain zero. Gemma deduplication selects nothing in 8,751 of 17,291 rounds (50.6%), which remain in the outcome. This contrast asks how the purchased previews are used, whereas the direct-generation comparison also changes how much work is spent acquiring them.

To measure how evidence affects selection, a direct test compares the coverage advantage over deduplication under both evidence conditions on the same 164 questions. The paired oracle-minus-retrieved difference is $+0.081$ $[-0.049, 0.228]$ for Gemma and $+0.047$ $[-0.059, 0.161]$ for 27B; both paired intervals include zero. This comparison uses the same questions under both evidence conditions, separately from the 256-question oracle-test assessment.

Prefix-text diversity provides a comparison within the iterative framework without purchasing separate entity previews (Tab. G2). Its mean coverage is higher than SCPS in all five evaluated cells. Target F_1 is lower on 9B and Gemma and higher on 27B with oracle evidence; the 27B retrieved-evidence interval includes zero. These exploratory, pointwise intervals use the same 10,000 question-bootstrap draws, averaging three streams within question, and form no additional Holm family. Omitting previews can admit more rounds; mean total spend is higher in four of the five cells. Selections and later memory also differ, so this comparison measures an alternative allocation of work rather than isolating the price of information.

Table G2: Selecting diverse prefix texts without entity previews, compared with SCPS at allowance 65,536. Differences subtract SCPS from prefix-text diversity. The 95% intervals are exploratory, pointwise paired question-bootstrap intervals, conditional on the three observed streams. Costs are ratios of mean measured uncached input-plus-output tokens. Retrieved cells have 164 questions and oracle cells 256; absolute means appear in Tab. G1.

Backbone	Evidence	Δ coverage [95% CI]	ΔF_1 [95% CI]	Cost / SCPS
Qwen3.5-9B	Retrieved	$+0.573$ $[0.423, 0.742]$	-0.0146 $[-0.0274, -0.0022]$	1.113
Gemma-4-12B	Retrieved	$+0.291$ $[0.171, 0.429]$	-0.0096 $[-0.0194, -0.0006]$	1.039
	Oracle	$+0.099$ $[0.033, 0.174]$	-0.0102 $[-0.0194, -0.0003]$	0.983
Qwen3.5-27B	Retrieved	$+0.843$ $[0.659, 1.043]$	$+0.0176$ $[-0.0005, 0.0361]$	1.170
	Oracle	$+1.906$ $[1.690, 2.128]$	$+0.1386$ $[0.1169, 0.1605]$	1.232

G.3.2 Agreement filtering and preview retention

Filtering the same generated lists. To isolate the effect of filtering, we compare the union of all names from the same sixteen direct-generation lists with the names occurring in at least two lists. Filtering retains 88.0–98.6% of mean correct-target coverage while removing 20.2–67.5% of unmatched names (Tab. G3). All six mean target-F_1 differences are positive; the four Qwen gains survive Holm adjustment, while the two Gemma gains do not. Recurrence therefore helps select which uncertain names to return, although it can retain repeated errors and discard correct singletons. The comparison changes only the output filter and requires no additional model calls.

For this comparison and the direct retention-versus-filtering comparison below, we average the three paired stream differences within each question and resample questions 10,000 times (seed 0). Inference is conditional on these observed streams. Target-F_1 tests use two-sided null-centered bootstrap tails with a plus-one correction; Holm adjustment covers the six filtering comparisons and, separately, the 24 retention comparisons. Their displayed intervals are pointwise 95% intervals. These exploratory families are separate from the complete-procedure families above.

Table G3: Filtering the same sixteen answer lists removes many unmatched names while preserving most correct-target coverage. A name is retained when it occurs in at least two lists. The two percentage columns are descriptive ratios of means; unmatched names match no gold target, so aliases matching a target are excluded from this count. Target-F_1 differences subtract the unfiltered union from the filtered output and have pointwise 95% paired question-bootstrap intervals. Holm adjustment covers these six exploratory F_1 tests. Retrieved and oracle evidence use 164 and 256 separate questions, respectively, each with three streams. Generation and charged tokens are identical within each pair.

Evidence	Coverage retained	Unmatched names removed	Δ target F_1 [95% CI]	Holm p
Qwen3.5-9B				
Retrieved	90.4%	53.8%	$+0.028$ $[+0.020, +0.037]$	0.0006
Oracle	94.4%	67.5%	$+0.033$ $[+0.022, +0.044]$	0.0006
Gemma-4-12B				
Retrieved	97.6%	20.2%	$+0.0028$ $[+0.0001, +0.0057]$	0.1006
Oracle	98.6%	31.8%	$+0.003$ $[-0.001, +0.008]$	0.1635
Qwen3.5-27B				
Retrieved	88.0%	63.3%	$+0.040$ $[+0.028, +0.053]$	0.0006
Oracle	95.0%	55.9%	$+0.018$ $[+0.010, +0.027]$	0.0006

Retaining previewed names. A preview has already consumed tokens and may name a correct entity even when its candidate is not selected. We therefore augment each SCPS episode's final answers with all valid previewed names from its existing trajectory. An uncapped rule keeps the entire union. Three capped rules keep at most 4, 8, or 16 names, ranked by their total number of preview and selected-completion mentions; ties use first emission, then alphabetical order. This use of agreement lets a previewed name displace a terminal answer: the caps apply to the whole augmented answer set. These output rules leave generation, memory updates, stopping, and every token charge unchanged.

Tab. G4 compares every retention rule with equally capped terminal-only output and with agreement-filtered lists. At a matched output cap, previewed names improve Qwen27B's mean F_1 at caps four and eight under both evidence conditions, and at cap sixteen under oracle evidence; their pointwise intervals exclude zero. On 9B and Gemma, no matched-cap F_1 interval lies wholly above zero. Thus previewed answers can contribute useful information beyond shortening the final list.

Every retention rule has lower mean F_1 than agreement-filtered lists, with 20 of the 24 differences surviving Holm adjustment. Retention leaves generation cost unchanged: mean charges are 4.66–35.34 times those of filtered lists. Retained previews nevertheless have higher mean coverage in seven of these 24 comparisons.

Table G4: Previewed answers can improve output at a fixed cap, while filtered generation has higher mean F_1 at lower cost. The first two outcome columns compare preview-augmented and terminal-only SCPS output with the same mention-based ranking rule and name cap; "All" applies no name cap. The final column compares retained-preview SCPS with agreement-filtered lists. Intervals are exploratory, pointwise 95% paired question-bootstrap intervals; a dagger in the final column denotes Holm-adjusted $p < 0.05$ across its 24 comparisons. Retrieved and oracle evidence use 164 and 256 separate questions, respectively, with three streams and a 65,536-token allowance. Block headings give the ratio of mean charged tokens for SCPS to filtered lists, unchanged across retention rules.

	Versus equally capped terminal output		Versus filtered lists
Retained names	Δ coverage [95% CI]	Δ target F_1 [95% CI]	Δ target F_1 [95% CI]
Qwen3.5-9B / Retrieved evidence (6.64× tokens)			
All	+0.486 [+0.382, +0.598]	−0.048 [-0.063, -0.034]	−0.067† [-0.090, -0.046]
At most 4	−0.016 [-0.047, +0.014]	−0.003 [-0.008, +0.003]	−0.035† [-0.054, -0.017]
At most 8	+0.106 [+0.051, +0.167]	−0.002 [-0.011, +0.006]	−0.015 [-0.031, +0.001]
At most 16	+0.295 [+0.207, +0.388]	−0.021 [-0.033, -0.010]	−0.039† [-0.058, -0.021]
Qwen3.5-9B / Oracle evidence (9.39× tokens)			
All	+0.563 [+0.465, +0.668]	−0.153 [-0.171, -0.135]	−0.193† [-0.222, -0.165]
At most 4	−0.010 [-0.033, +0.012]	−0.004 [-0.008, +0.0004]	−0.108† [-0.132, -0.082]
At most 8	+0.163 [+0.104, +0.224]	−0.023 [-0.032, -0.014]	−0.053† [-0.074, -0.031]
At most 16	+0.432 [+0.345, +0.526]	−0.089 [-0.103, -0.076]	−0.128† [-0.152, -0.103]
Gemma-4-12B / Retrieved evidence (10.21× tokens)			
All	+0.220 [+0.154, +0.293]	−0.033 [-0.045, -0.021]	−0.058† [-0.083, -0.035]
At most 4	+0.000 [-0.039, +0.043]	−0.001 [-0.008, +0.008]	−0.066† [-0.092, -0.042]
At most 8	+0.016 [-0.033, +0.071]	−0.006 [-0.013, +0.002]	−0.033† [-0.051, -0.015]
At most 16	+0.081 [+0.024, +0.144]	−0.013 [-0.022, -0.005]	−0.034† [-0.054, -0.014]
Gemma-4-12B / Oracle evidence (35.34× tokens)			
All	+0.173 [+0.124, +0.227]	−0.081 [-0.098, -0.064]	−0.065† [-0.091, -0.037]
At most 4	+0.010 [-0.025, +0.049]	−0.0001 [-0.006, +0.007]	−0.117† [-0.144, -0.090]
At most 8	+0.039 [-0.004, +0.085]	−0.014 [-0.022, -0.006]	−0.011 [-0.032, +0.011]
At most 16	+0.078 [+0.030, +0.128]	−0.043 [-0.055, -0.032]	−0.024 [-0.047, -0.0001]
Qwen3.5-27B / Retrieved evidence (4.66× tokens)			
All	+0.589 [+0.455, +0.734]	−0.015 [-0.032, +0.001]	−0.055† [-0.074, -0.036]
At most 4	+0.118 [+0.065, +0.175]	+0.015 [+0.006, +0.026]	−0.040† [-0.061, -0.019]
At most 8	+0.299 [+0.201, +0.407]	+0.018 [+0.006, +0.031]	−0.019 [-0.036, -0.003]
At most 16	+0.463 [+0.341, +0.600]	+0.005 [-0.010, +0.020]	−0.033† [-0.051, -0.016]
Qwen3.5-27B / Oracle evidence (5.53× tokens)			
All	+1.466 [+1.302, +1.641]	+0.081 [+0.062, +0.101]	−0.051† [-0.071, -0.030]
At most 4	+0.479 [+0.404, +0.559]	+0.070 [+0.058, +0.082]	−0.104† [-0.127, -0.081]
At most 8	+1.130 [+0.995, +1.268]	+0.101 [+0.085, +0.117]	−0.034† [-0.053, -0.014]
At most 16	+1.387 [+1.232, +1.549]	+0.093 [+0.075, +0.111]	−0.039† [-0.058, -0.019]

G.4 Evaluation on held-out QAMPARI questions

A held-out set of 60 oracle-evidence questions tests the comparison on questions disjoint from the other evaluation sets and procedure-development data. Qwen3.5-9B is evaluated with three streams at allowance 65,536. SCPS returns fewer correct targets but higher target F_1 than memory-conditioned lists: both contrasts are significant after Holm adjustment in the prespecified two-test family ($p = 0.0002$). Tab. G5 shows this tradeoff and the secondary agreement-filtered comparison. Intervals use the same paired question-cluster percentile bootstrap; a coverage bootstrap-t sensitivity widens the primary interval to $[-1.655, -0.659]$ without changing its direction. Both secondary contrasts against agreement-filtered lists have intervals that include zero. The held-out comparison confirms the coverage–quality tradeoff against continued memory-conditioned generation.

Table G5: SCPS minus each reference on the held-out set of 60 QAMPARI questions, with paired 95% percentile intervals. The two contrasts with memory-conditioned lists form the complete primary Holm family; contrasts with filtered lists are secondary and unadjusted. Coverage is distinct correct targets, and target F_1 is computed within episodes.

Reference	Coverage difference	Target F_1 difference
Memory-conditioned lists	-1.006 $[-1.461, -0.622]$	$+0.209$ $[0.137, 0.284]$
Filtered 16 lists	-0.239 $[-0.628, 0.122]$	-0.032 $[-0.068, 0.006]$

G.5 Transfer to AmbigQA

The transfer study asks whether the same allocation of generation and preview work is useful on a different QA benchmark. It uses 116 eligible questions from AmbigQA's official development split, three streams, the same three backbones, and allowances 16,384 and 65,536. Eligibility requires 3–20 gold targets and nonempty proof snippets totaling at most 1,000 words after a 24-snippet cap, with at least as many retained snippets as targets. Procedures use the complete QA interface above with supplied answer-bearing evidence. The outcome is AmbigQA's official `F1 answer`, which scores against the example's annotations and takes the best match (Min et al., 2020), rather than QAMPARI's target F_1. Compared with target F_1 on the same outputs, official answer F_1 changes procedure ordering in all six backbone–allowance conditions. The scorer receives the names emitted by each procedure after Unicode decomposition, removal of diacritics, lowercasing, and removal of most punctuation and articles, and whitespace normalization. Empty predictions score zero and remain included. Applying the same normalization to gold aliases changes all twelve SCPS-minus-reference mean differences by less than 0.004 F_1. Their signs and conclusions from paired 95% question-bootstrap intervals are unchanged (20,000 resamples, seed 20260923).

All displayed contrasts are secondary and unadjusted, with 10,000 paired question-cluster bootstrap draws and percentile 95% intervals. Tab. G6 reports both allowances and both native references. At 65,536, all six official-F_1 contrasts favor the native references, and SCPS spends 64–84 times the single-list charged tokens. At 16,384, the initial round's reservation does not fit the allowance, triggering single-list fallback in 64.7%, 0.9%, and 88.8% of SCPS episodes on 9B, Gemma, and 27B. The agreement-filtered reference also falls back at this allowance on 0%, 1.7%, and 10.3% of episodes, respectively. The full-procedure scores include these fallback outcomes. The interval for the positive 27B contrast against agreement-filtered lists at this allowance includes zero.

Table G6: Secondary AmbigQA comparisons: SCPS minus single-list and filtered-sixteen-list native references on official answer F_1, with unadjusted paired 95% intervals. Cost is the descriptive ratio of mean charged tokens, SCPS/single-list. Fallback is the percentage of SCPS episodes using single-list fallback. Each cell contains 116 questions and three streams.

		Official F_1 difference			Fallback
Backbone	Allowance	vs. single list	vs. filtered 16	Cost	(%)
Qwen9B	16,384	-0.010 $[-0.034, 0.014]$	-0.023 $[-0.051, 0.006]$	$5.7\times$	64.7
Qwen9B	65,536	-0.112 $[-0.153, -0.072]$	-0.130 $[-0.163, -0.098]$	$75.3\times$	0.0
Gemma12B	16,384	-0.047 $[-0.082, -0.014]$	-0.043 $[-0.079, -0.008]$	$14.7\times$	0.9
Gemma12B	65,536	-0.129 $[-0.176, -0.083]$	-0.139 $[-0.183, -0.097]$	$83.6\times$	0.0
Qwen27B	16,384	-0.016 $[-0.029, -0.003]$	$+0.032$ $[-0.016, 0.083]$	$2.5\times$	88.8
Qwen27B	65,536	-0.071 $[-0.112, -0.032]$	-0.085 $[-0.118, -0.055]$	$63.7\times$	0.0

G.6 Reference validation and displaced work

The empirical comparisons leave a practical decision: when does validation justify replacing a reference procedure? A prespecified family can include direct generation, a simple filter, and information-using alternatives. Comparing their final utility under the available budget allows the reference itself to remain the selected choice when improvement is uncertain.

For a finite prespecified family containing reference Q_0, let $\Delta(Q)$ be procedure Q's true utility gain over Q_0, $\widehat{\Delta}(Q)$ its estimate, and $r(Q)$ its uncertainty radius. Suppose simultaneous intervals cover every gain: $|\widehat{\Delta}(Q) - \Delta(Q)| \le r(Q)$, with the reference's estimate and radius zero. Select $\widehat{Q}$ to maximize the lower bound $\widehat{\Delta}(Q) - r(Q)$. On that simultaneous coverage event,

$$\Delta(\widehat{Q}) \ge 0, \qquad \Delta(\widehat{Q}) \ge \max_{Q} \{\Delta(Q) - 2r(Q)\}. \tag{G1}$$

The selected gain is at least its lower bound, which is at least the reference's zero and every competitor's lower bound. Each competitor's lower bound is in turn at least its true gain minus twice its radius. This proves both statements: selection preserves the reference's utility and loses at most the indicated uncertainty penalty relative to another member. This is a direct high-confidence-improvement argument (Thomas et al., 2015); the family must be independent of validation outcomes or covered by the simultaneous event. Applying this guarantee requires simultaneous, rather than pointwise, intervals across the candidate family.

The same logic must charge information for the work it displaces. If acquisition costs a deterministic c_P, the information-using procedure has only $B - c_P$ left for continuation. In Eq. (3), Δ_P compares its value with the reference at that reduced allowance. If $\ell \le \Delta_P$ certifies this remaining-budget improvement, adding and subtracting $J_0(B - c_P)$ gives

$$J_P(B) - J_0(B) \ge \ell - [J_0(B) - J_0(B - c_P)]. \tag{G2}$$

Thus the certified selection improvement must cover the reference value lost by reducing its allowance. The inequality preserves the certificate's status as a lower bound. The bracketed loss is signed: it is nonnegative for a budget-optimal reference that can stop or ignore extra resources, but a fixed procedure can lose utility at larger allowances, for example by adding incorrect answers that lower F1. The term can then be negative. Random acquisition costs require the remaining-budget and continuation-value terms inside their joint expectation. A mean cost cannot enforce a hard cap; App. H examines actual operation reservations.

H Executable tests and displaced work

Generating fresh tests, revising existing tests, and gathering execution feedback are alternative uses of the same allowance. Coverage and mutation feedback also guide CodaMosa and MuTAP, respectively (Lemieux et al., 2023; Dakhel et al., 2024). Our studies ask when information improves later work enough to repay what it displaces. Final utility is mutation score: how well the suite detects deliberately introduced code faults. Explicit reservations reveal a second constraint: acquisition can prevent a later call from fitting even when the reference procedure eventually leaves tokens unused.

H.1 When acquisition prevents later work from fitting

Suppose we want to buy information upfront while preserving every call made by a reference procedure. For its recorded path, let s_{t-1} be charged expenditure before call t and r_t the amount reserved to admit that call. Replaying the same calls with the same caps requires

$$B_{\text{required}} = \max_t (s_{t-1} + r_t).$$

At each step, already spent tokens and the next reservation must fit together; their largest sum determines the required budget. An upfront acquisition charge c_a preserves that path only if $B - c_a \geq B_{\text{required}}$. For example, two calls that each reserve 80 but consume 20 fit sequentially within a budget of 100, leaving 60 unused; buying information for 30 upfront prevents even the first call from fitting. Final slack therefore cannot price the information that could have been bought without displacing reference work. This is the displaced-completion mechanism in Fig. 1C, with admission determined before the actual bill is known.

The TestGenEval study (Jain et al., 2025) uses gpt-oss-20b (OpenAI et al., 2025) with medium reasoning and a 16,384-token incremental allowance after a supplied test suite and initial execution feedback. Constructing that starting state is outside the allowance. Mutation scoring uses Cosmic Ray 8.4.1[2]. The full rendered prompt plus the maximum allowed output length determines a call's reservation; actual uncached input and output determine expenditure. Output caps shrink with remaining allowance, rounded down to multiples of 64, until their minimum is infeasible. A smaller allowance can thus shorten a call before removing it. Across 141 reference paths, median final slack is 4,498 tokens. Among the 135 reference paths containing at least one call, the minimum spare budget after admission has median 41 tokens. A run can therefore have little room for an upfront purchase despite substantial final slack. The following comparison tests what these constraints mean for final utility under matched runtime conditions.

H.2 Does information repay its opportunity cost?

The experiment uses 47 suite-state units from 25 benchmark instances in 24 source-file lineages, with three randomized blocks. Fresh continuation tests the value of spending directly on more work; matched, exchanged-information, and blind patches test the value of revising with different information; a matched rewrite changes the size of the revision. A patch edits a test; a rewrite generates a revised bundle. Patching and rewriting produce executable tests as well as information; their measured value includes both. Patch selection uses each test's executed lines and the suite's assertion and input-signature coverage. Matched selection scores a test's additional coverage in the current suite; exchanged selection uses the other paired suite state, and blind selection ignores suite coverage. The patch writer receives the selected test, its intended behavior, inputs and expected result, and the relevant source code. Matched versus fresh tests the final value of patching; matched versus exchanged tests the value of using the current suite to choose which test to edit.

Table H1: TestGenEval comparisons at 16,384 incremental charged tokens after a supplied suite and initial feedback: mutation-score points on a 0–100 scale. The reported 95% intervals are the wider of percentile and bootstrap-t intervals (Efron & Tibshirani, 1994), giving bootstrap-t in all five comparisons. Blocks are averaged within units; units receive equal weight and source-file lineages are resampled.

Final-score contrast	Difference [95% CI]	Raw p	Holm p ($m = 5$)
Matched − fresh	-1.274 $[-3.270, 0.392]$	0.125	0.502
Matched − exchanged information	$+0.537$ $[-0.527, 2.294]$	0.351	1.000
Matched − blind patch	$+0.097$ $[-0.824, 1.217]$	0.864	1.000
Rewrite − fresh	$+0.241$ $[-1.396, 1.336]$	0.771	1.000
Rewrite − matched patch	$+1.515$ $[0.047, 3.001]$	0.042	0.210

The writer, action-selection rules, rules for accepting or reverting a proposed change, and subsequent reference policy are fixed. Branches share the runtime conditions within each block and use separate caches. Each final-score contrast has 141 paired unit–block differences, averaged to 47 unit

[2] https://github.com/sixty-north/cosmic-ray

means; inference resamples the 24 source-file lineages. All attempted operations are charged, and unsuccessful changes leave the prior suite in place. The final-score comparisons in Tab. H1 therefore include both the operation and the continuation that remains affordable afterward.

Neither the primary comparison of matched patching with fresh generation nor the comparison of matched with exchanged information establishes a gain in final mutation score. None of the five comparisons is significant after Holm correction.

H.3 CoverUp under a shared incremental budget

CoverUp uses coverage feedback and code-context queries to guide test generation (Altmayer Pizzorno & Berger, 2025). The adaptation answers cross-file context queries and allows at most 50 information-query interactions per attempt. Every arm uses gpt-oss-20b and 32,768 incremental charged tokens after a supplied seed suite, with full-prompt reservations and recorded uncached-input/output charges. Native feedback observes the suite and execution results; the numerical path allocator ranks source-region work by predicted mutation gain per cost, using a predictor fitted on disjoint source-file lineages without deployment-time mutant outcomes. CoverUp uses temperature zero and the references 0.6, so this compares complete procedures. Seed construction and offline training/evaluation are outside the incremental allowance.

Table H2: CoverUp minus each reference, in mutation-score points. Three streams per arm; 95% bootstrap-t intervals resample source-file lineages. The four-member Holm family includes both the development and evaluation populations.

Population (instances/lineages)	Reference	Difference [95% CI]	Holm p
Development (36/27)	Native feedback	-6.84 $[-16.78, 1.44]$	0.099
Development (36/27)	Numerical path	-11.11 $[-19.78, -2.54]$	0.012
Evaluation (99/62)	Native feedback	-8.07 $[-13.13, -3.40]$	< 0.01
Evaluation (99/62)	Numerical path	-9.13 $[-13.07, -5.64]$	< 0.01

CoverUp has lower mutation scores than both references on the 99-instance population. Empty or unusable suites receive score zero, with all incurred costs charged.

I Extended related work: what early information enables

Complementary selection and early stopping raise three information questions: what a predictor must preserve, when its information repays its cost, and which completions it can replace. Tab. I1 connects our findings to language-model methods; Tab. I2 distinguishes the mathematical tools from the guarantees added here.

I.1 Coordination: information and useful outputs

What the predictor must preserve. Complementary choices matter in coverage, relevance–novelty selection, and solution portfolios (Nemhauser et al., 1978; Carbonell & Goldstein, 1998; Lin & Bilmes, 2011; Drygala et al., 2025). Retriever Portfolios, for example, values a set through its best member's score. This can also be viewed as coverage: for each score threshold, the set receives credit if any member reaches it. The additional question is how much of the outcome model a predictor must preserve to choose well.

Prop. A1 specializes decision-focused prediction (Elmachtoub & Grigas, 2022) to comparisons between feasible allocation rules. The obstruction in App. A.4 concerns rules given only individual and pairwise useful-hit probabilities. This distinguishes it from answer-equivalence prediction: exact answer equality identifies its equivalence classes, while unrestricted embeddings may contain richer information.

A reusable representation. Given a circular action order in which every contributor group is consecutive, systematic sampling coordinates selections optimally at prescribed selection probabilities (Madow, 1949). Our learning question is how many complete contributor observations suffice to

Table I1: What our information and resource questions add to existing language-model methods.

Related work: focus and evidence	Result in this paper
Coordination: what information supports useful joint selection?	
Retriever Portfolios (Stouras et al., 2026); ICML 2026. Learns retriever libraries from query scores; greedy best-member guarantee, individual-ranking, routing, and cost controls.	**Reusable information (Thm. 1).** A known order of overlapping contributor groups gives optimal coordination at prescribed selection probabilities. Complete contributor observations learn an order reusable across batch sizes and selection probabilities, with matching worst-case coverage-loss rates.
DynaAct (Zhao et al., 2025); NeurIPS 2025. Selects actions by learned utility and embedding diversity; tests critical-step coverage, utility-only selection, and runtime.	**Coverage–correctness tradeoff (Eq. (2); Sec. 4.2).** On shared QAMPARI predictions, joint selection improves distinct-correct-answer coverage over individual ranking even though individual outputs are less often correct. Avoiding repeats offsets that loss.
DeepPrune (Tu et al., 2026); Findings of ACL 2026. Predicts answer equivalence from prefixes and selects traces for voting; accuracy–token comparisons and judge ablations; acknowledges judge overhead.	**Information limitation (App. A.4).** Even exact individual usefulness and pairwise overlap probabilities can leave the best batch unidentified. The construction concerns overlapping contributions; known answer-equivalence classes give different information.
Coordination: when does coordinated generation improve useful coverage?	
Arithmetic Sampling (Vilnis et al., 2023), ICML 2023; **QuasiMoTTo** (Li et al., 2026). Coordinate outputs with unchanged individual distributions; sampling guarantees and translation/reasoning gains.	**Format-dependent useful coverage (Sec. 4.1).** Entity-first output increases QuasiMoTTo's correct-entity coverage gain in all six primary conditions, with fewer repeated correct answers: output format changes the same sampler's useful return.
Acquisition: when is early information worth buying?	
CIPHER (Heuillet & Peddiraju, 2026); arXiv preprint, July 2026. Selects plans before execution and studies how selection and aggregation affect accuracy; reports token use.	**Selection versus acquisition value (Eq. (3)).** Separate fixed-information and full-budget studies distinguish selection gains from the value of buying information at the expense of execution (Secs. 4.2–4.3).
Snell et al. (2025); ICLR 2025. Shows difficulty-dependent gains from adaptive search and revision; charges verification but excludes difficulty-estimation cost.	**Exact profitability (Thm. 2).** In a separate diagnostic model, price, redundancy, and discrimination determine whether repeated fresh checks can reduce expected total cost to a first useful completion.

recover such a representation. PC trees represent the remaining permissible orders; disagreement bounds and contributor-set sampling supply further tools (Hsu & McConnell, 2003; Fink et al., 2023; Hanneke, 2016; Tang et al., 2014). App. B combines them with a bound on strict order restrictions to prove matching coverage-loss rates. Under the same observation distribution, the learned representation serves different batch sizes and selection probabilities.

Useful overlap and output format. The controlled selection study asks how much overlap information is useful when candidate predictions are fixed (Sec. 4.2). Coordinated decoders act at a different point: they preserve the distribution of each generated output, whereas Thm. 1 fixes each action's probability of being selected. Our output-format comparison asks when the same QuasiMoTTo sampler yields more useful coverage, with Eq. (2) separating individual usefulness from repeated support avoided (Sec. 4.1).

Final aggregation determines which repetitions matter. Self-consistency votes across answers (Wang et al., 2023); DeepPrune retains multiple traces for voting, whereas its one-per-cluster study concerns pass@k. CIPHER varies aggregation on fixed execution logs and finds that maximum embedding diversity need not maximize accuracy. These comparisons motivate holding aggregation fixed when isolating a generation or selection change. Agreement can inform which uncertain names to retain even though repeated correct answers add no distinct-target coverage; App. G measures both the correct and incorrect names removed by filtering.

Table I2: Information and resource guarantees.

Related work: focus and evidence	Result in this paper
Coordination: what information supports useful joint selection?	
Data-Driven Solution Portfolios (Drygala et al., 2025); ITCS 2025. Approximates expected best-solution value under independent element activations and matroid constraints.	**Learnable sufficient structure (Thm. 1).** A known circular order gives optimal coordination at prescribed selection probabilities. Learning from complete contributor sets bounds coverage loss for all such probability vectors, with a matching worst-case lower bound even for unrestricted randomized selection rules.
Acquisition: when is early information worth buying?	
Sequential-inspection Pandora (Aouad et al., 2026); *Operations Research*, online May 2026. One partial report per box and separately charged full opening; reservation and buffered-dominance results yield our one-preview boundary.	**Exact economic boundary (Thm. 2).** The one-preview threshold characterizes profitability even with unbounded fresh reports, adaptive finite diagnostic menus, and interleaved independent candidates: repetition and switching cannot rescue overpriced checks for first useful completion.
Discovery: which expensive completions can checks replace?	
Search over multiple populations (Malloy et al., 2013); *IEEE Trans. Information Theory*, 2013. Finds one rare target population through repeated observations, with asymmetric tests and sample-complexity bounds.	**Joint resource control (Thm. 3).** Matching total-cost bounds for all d classes. When a check costs no more than a completion ($c \leq C$), the procedure makes at most d completions and $O_{a,b}(d \log(d/\delta))$ checks on every path. It covers all classes with probability at least $1 - \delta$.

Here c and C price a check and a completion; δ is the allowed failure probability. Discovery constants depend on the fixed diagnostic channel (a, b).

I.2 Acquisition: when is information worth buying?

DeepConf counts tokens from stopped partial traces, and difficulty-adaptive self-consistency charges its ranking calls (Fu et al., 2026; Wang et al., 2025). Bayesian stopping uses compressed answer counts to decide when to stop sampling for a modal answer (Huang et al., 2026). Our comparison asks what that spending displaces: the same budget could complete more work. This motivates evaluating complete procedures with information acquisition charged (Eq. (3) and App. G). When calls require an upfront reservation, final unused tokens need not measure that earlier opportunity (App. H).

Pricing computation and interleaving paid stages are established in metareasoning and information-acquisition models (Hay et al., 2012; Gupta et al., 2019; Ma & Tzamos, 2023; Aouad et al., 2026). Thm. 2 extends the one-preview profitability boundary to policies that repeat reports, choose adaptively from finite diagnostic menus, and interleave independent candidates, while obtaining a useful completion almost surely. Fixed candidate status, conditionally fresh reports, and separately charged completion make the price comparison possible (App. C); App. C.4 gives the accounting when a preview also advances generation.

I.3 Discovery: which completions can checks replace?

Classical sequential tests and rare-population search supply quick rejection and stagewise expected costs (Wald & Wolfowitz, 1948; Malloy et al., 2013; Schmit et al., 2019). Collecting all classes adds a resource-control question: how many checks and completions suffice on every path, including failures? App. D proves matching total-cost bounds and a construction with both operation caps. Repeated fresh checks suppress the false positives that would otherwise send duplicate candidates to completion. They reduce the price of repeated search but cannot remove the need to encounter missing classes: at fixed positive c/C, total search cost still scales as $d \log d$.