CERTIFIED BUT NOT USEFUL: AN INFORMATION-THEORETIC CEILING ON LLM-BASED MATHEMATICAL DISCOVERY

Anonymous authors
Paper under double-blind review

ABSTRACT

We prove an information-theoretic ceiling for verifier-mediated LLM agents: in any hidden-world, no-side-channel verifier model with disjoint success sets, no admissible policy—at any query budget—has equal-prior success probability above $(1+\delta_\Gamma)/2$, where δ_Γ is the total-variation distance between the laws of the complete verifier signature (Theorem 4.1). The ceiling depends on the interface and environment pair, not the agent; exact maxima meet it in all 423 finite $\mathcal{G}_3$ worlds. It is vacuous precisely when $\delta_\Gamma = 1$, meaning that the complete signature distinguishes the environments almost surely; public input revealing the target is sufficient, but not necessary. A companion result gives a minimax $\Theta(d)$ checker-call separation between charged rebuilding and free frame restoration in an opaque marked-leaf model (Theorem 4.2).

A usefulness interpretation of the ceiling requires success sets that encode held-out utility. We define *statistical discoverability* as the probability that a research procedure returns an artifact that is both certified and useful on sealed theorems. Separately, we observe that installing a checker-certified library *lowered* held-out solve rate by 0.375 (95% CI $[-0.50, -0.24]$; 10 of 12 paired cells negative, none positive; b=16384), and every added failure was a completed proof the checker rejected, not truncation or timeout. The measured harm is a post-freeze utility contrast on four worlds with a fixed solver, not an estimate of discoverability or a test of the ceiling; δ_Γ is unmeasured in these runs. Together, the framework opens a principled path for analyzing what LLM-based agents can and cannot formally guarantee in automated mathematics.

1 INTRODUCTION

What can an LLM-based mathematical agent learn about usefulness through a proof-checker interface? Our object is a limit theorem for the *interface*, not for the agent's reasoning capacity. In a hidden-world, no-side-channel model with disjoint success sets, indistinguishability of complete verifier signatures limits every admissible policy, regardless of capability or query budget (Theorem 4.1). Consider $(a, b, c) \mapsto (a + 2,\ b + 2,\ c + 1)$ on $(\mathbb{Z}/5)^3$: a checker can confirm in one line that $a - b$ is invariant, but certification alone does not establish its usefulness for a solver on sealed tasks. This hand-written illustration from a pilot world, not an agent output, exemplifies a *structure*: a definition, a proved property, and a way to use it.

Automated provers now routinely produce intermediate results that pass a proof checker, but a result that passes locally need not be a structure that helps on theorems the system has never seen. We therefore keep three events apart: a construction is *locally correct* when its declared certificate passes the checker; it is *accepting* when it is installable and applied to the target family; and it is a *discovery* when the installed artifact clears a held-out utility threshold on sealed theorems. Figure 1(a) draws the protocol that keeps these events apart, Figure 1(b) audits the first oracle ceiling in a finite model, and Figure 1(c,d) report a frozen-library solve-rate ablation; none of the panels measures discovery. Only after this do we define the quantity, *statistical discoverability*, in §2. Tao describes the corresponding

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.

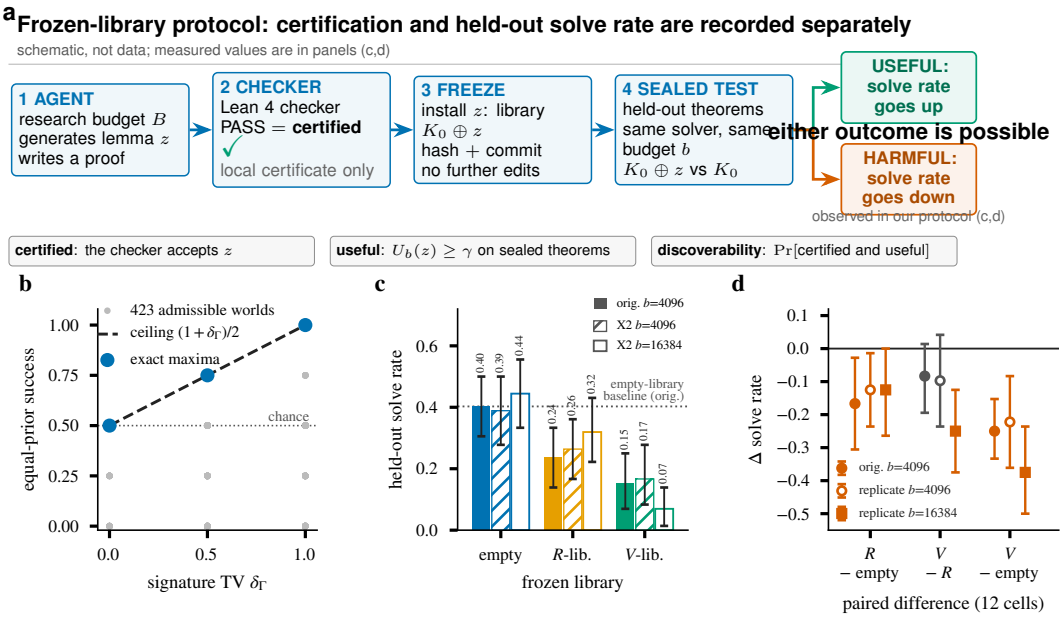


Figure 1: **In our frozen-library protocol (4 worlds×3 seeds), held-out solve rate was lower with each checker-certified library installed than with the empty library.** (a) *Protocol* (schematic, not data): a lemma that passes the checker is installed into a frozen library and evaluated only on sealed held-out theorems. (b) *Restricted model* (analytical, not an LLM measurement): in the 423 finite G-3 worlds satisfying the assumptions, exact maxima $1/2, 3/4, 1$ lie on the ceiling $(1+\delta_\Gamma)/2$ (zero violations). (c,d) *Measured* (E2-CorF-K0, consumer C_{base}, b=4096, 4 worlds×3 seeds; §5): held-out solve rate 0.403 (empty), 0.236 (R-library), 0.153 (V-library), and paired within-cell differences -0.17 $[-0.31, -0.03]$, -0.08 $[-0.19, +0.01]$, -0.25 $[-0.33, -0.15]$ (95% percentile bootstrap over 12 cells; descriptive, not an independent-world interval). In (c), hatched and open bars are an independent replicate of the same 12 cells at b=4096 (0.39, 0.26, 0.17) and the same cells at b=16384 (0.44, 0.32, 0.07); in (d), open circles and squares are the corresponding paired differences: $V-$empty -0.22 $[-0.36, -0.08]$ and -0.38 $[-0.50, -0.24]$. In these runs every failed test ended with finish reason `stop` and was rejected by the checker; no truncation and no Lean timeout occurred. Solve rate is not Theorem 4.1's $\Pr(\hat{a} = U)$; δ_Γ is unmeasured in (c,d), and no transfer claim beyond these four worlds is made.

problem-selection task as learning a field's "difficulty landscape"[1]; we study one coordinate of it under a fixed base library, checker, and sealed evaluation set.

The theory and measurement address distinct questions. Theorem 4.1 bounds identification through a verifier interface; Theorem 4.2 isolates the cost of rebuilding search context. Figure 1(b) and Appendix Figure 6 are respectively finite-model and local-accounting diagnostics, not deployed-LLM tests. The frozen-artifact target (§2) separates certification from usefulness; the exploratory utility contrast (§5, Figure 3) illustrates that distinction, not observed attainment of the ceiling. The repository records alone do not establish a preregistration timeline, and the 24-cell paired comparison does not establish general transfer. The measurements do not establish that a deployed LLM agent satisfies either oracle interface.

Contributions. A checker certifies local correctness; usefulness on unseen theorems is a separate target. We contribute an interface-conditional information ceiling, a definition of discoverability, and a controlled observation of reduced utility after installing certified libraries. The observation demonstrates the distinction between certification and usefulness; it does not test the ceiling. None of the three converts a restricted bound into a deployed-agent guarantee.

[1]T. Tao, Mathstodon thread, 8 September 2026, posts 2–3 of 4: `https://mathstodon.xyz/@tao/117237321472650215` and `https://mathstodon.xyz/@tao/117237322160500501`. This informal commentary is used only as motivation.

- **An information-theoretic ceiling on verifier-mediated identification.** In any hidden-world, no-side-channel verifier model with disjoint success sets, every admissible policy has equal-prior success probability at most $(1 + \delta_\Gamma)/2$, where δ_Γ is the TV distance between the complete-signature laws (Theorem 4.1; Figure 1(b)). Interpreting success as usefulness requires sets that encode held-out utility. The ceiling depends on the interface and environment pair, not the agent; exact maxima meet it in all 423 finite $\mathcal{G}_3$ worlds. A companion minimax separation (Theorem 4.2; Appendix Figure 6) gives a $\Theta(d)$ checker-call factor between charged rebuilding and free frame restoration. Neither result is tested by the experiment below, and neither describes a deployed LLM's provider-side state.
- **Evaluation target (definition).** We separate the checker event $\mathrm{Cert} = 1$ from the held-out utility event $U_b \geq \gamma$, and define *statistical discoverability* SD as the probability of both (Definition 2.4, §2). Figure 2 shows why a locally correct branch need not lead to an accepted artifact. The experiments estimate the post-freeze utility U_b of particular installed artifacts, not SD, whose denominator is every research trial.
- **Measured observation: installing certified libraries lowered held-out utility.** In four finite Lean worlds, installing a Lean-certified library *lowered* the held-out solve rate relative to an empty library (§5, Figure 3). Table 1 binds this result to its assumptions, code path, result columns, baseline, and provenance. We do not claim that transfer is negative in general.

2 SETUP: DISCOVERABILITY AS A STATISTICAL TARGET

This section defines the evaluation target; it reports no measured estimate of statistical discoverability. The experiments estimate the post-freeze held-out utility U_b of particular installed artifacts against a replacement library; SD, whose denominator is every research trial including failures, is not estimated.

A mathematical environment is $c = (K_0, S, V, Q_{\mathrm{test}})$, comprising a frozen base library, public research tasks, public development tasks, and sealed evaluation tasks. The four components, including their versions and splits, are part of the experimental unit. A research procedure may use only K_0, S, V while choosing an artifact; Q_{test} is opened only after that choice and the installed artifact have been frozen.

Definition 2.1 (Verified abstraction)**.** *A structural artifact is an executable package*

$$z = (\mathsf{definition}, \mathsf{properties}, \mathsf{proofs}, \mathsf{application}, \mathsf{dependencies}).$$

The package declares a representation, invariant, lemma, or transformation; the properties it needs; checker-verifiable proofs; an application to a theorem family; and its dependencies. Write $K_0 \oplus z$ only when the package and its dependencies are accepted under the same declared foundation, axiom policy, and library version as K_0. Natural-language suggestions, a name alone, a pasted gold proof, or a table of answers to evaluation tasks are not admissible installations.

Definition 2.2 (Held-out utility)**.** *Fix an environment c and a fixed solver T with per-theorem proof budget b. Here Q_{test} denotes the uniform distribution on the declared finite sealed test set. For an installed library K, define $p_b^T(K, q)$ as the probability that T returns a checker-accepted proof of q within budget b, including all solver randomness under the frozen evaluation protocol.*

For an admissible artifact z, define

$$U_b(z; c) = \mathbb{E}_{q \sim Q_{\mathrm{test}}} \left[p_b^T(K_0 \oplus z, q) - p_b^T(K_0, q) \right].$$

For an inadmissible artifact, define $U_b(z; c) = 0$ and its certification indicator to be zero. Both terms use the same solver, proof budget, checker, and frozen evaluation protocol. Thus utility measures a library intervention, not a change in model, prompt budget, test set, or verification policy.

Remark 2.3 (Accepting, certified, successful). Call an artifact *certified* if $\mathrm{Cert} = 1$ (locally correct, admissibly installable, with its designated application; a vertex with these properties is *accepting* in §3) and *successful* if it is certified and $U_b \geq \gamma$, the event counted by Definition 2.4. Successful implies certified, and the converse fails: an artifact can be installable and yet change the solve rate by less than γ. Hence the set of accepting vertices of a graph over-approximates the set of successful artifacts, and any count of certified terminals bounds the successful ones only from above. Experiment

tables keep the two columns apart: the certification rate for the first, the held-out $\widehat{U}_b \geq \gamma$ rate for the second.

Definition 2.4 (Statistical discoverability). *Let $\widehat{z}_B$ be the artifact output by a research procedure with research budget B, and let $\mathrm{Cert}(\widehat{z}_B) \in \{0,1\}$ be its declared admission check. For a prespecified threshold γ,*

$$\mathrm{SD}^{T}_{B,b}(\gamma; c) = \Pr[\mathrm{Cert}(\widehat{z}_B) = 1 \ \wedge \ U_b(\widehat{z}_B; c) \geq \gamma] \,.$$

Statistical discoverability is conditional on c; its probability is over the declared research procedure's randomness. An average across environments is a separate estimand requiring a declared environment distribution. Finite evaluation estimates U_b by $\widehat{U}_b$. Threshold crossing by $\widehat{U}_b$ is not identified with threshold crossing by U_b, and evaluation uncertainty is reported separately. Accepting-vertex reachability alone does not imply positive U_b. Reports retain proposal, certificate, and held-out-utility rates separately, so failure at one stage is not silently attributed to another.

Remark 2.5 (Evaluation boundaries). Discoverability is solver dependent: an artifact useful to one solver need not be usable by another. Research and evaluation tasks are disjoint, and artifacts, dependencies, proof objects, and retrieval settings are frozen before held-out evaluation. Finally, SD is not a definition of mathematical importance: generality, naturalness, explanatory value, and historical novelty lie outside this quantity.

The status of each exploratory comparison relative to this target is stated in Appendix A; the candidate-generation/selection identity used to read failures is in Appendix B.

3 Execution-state graphs

This section gives finite-model diagnostics for the evaluation target of §2. They are not the restricted hidden-world or marked-leaf analyses of Theorems 4.1 and 4.2, and held-out utility is not measured by them. Figure 2 is a schematic, not an experimental effect.

For a finite environment c, fix a typed artifact language, declared construction operators, and admissible size bounds. We distinguish the artifact graph from the state of an execution: the former records what one word depends on, whereas the latter records what has been retained, tried, and can still be done under the residual budget. This distinction matters even for a tree, since a failed checked prefix can be retained and extended or the executor can return to an earlier retained prefix.

The construction graph $\mathcal{G}_c$ used below records artifact dependencies. Its verification-gap statistic is pathwise: it does not account for all inputs to a multi-input operation or for retained memory and restart costs. The Family B instantiation (Appendix C) supplies the graph–execution connection for its single-input affine words only; a general finite transition system for every environment c, and its connection to $\mathcal{G}_c$'s verification-gap statistic beyond the declared Family B case, remain to be specified. The roots of $\mathcal{G}_c$ are the constructions supplied by K_0. A vertex is *locally correct* if its declared certificate passes the checker. It is *accepting* if it is locally correct, admissibly installable, and has the designated application to the target family; accepting is weaker than successful in the sense of Remark 2.3. The graph is an analysis object; it gives no solver an undeclared shortest-path oracle, and in particular an online agent is never told whether its current state can be extended to an accepting vertex.

Definition 3.1 (Correct but unhelpful branch). *A locally correct vertex $v \in \mathcal{V}_c$ is a* correct but unhelpful branch *for a target family if no directed continuation from v, using only the declared remaining operators and remaining construction budget, reaches an accepting vertex. Such a vertex certifies a local statement but not held-out utility or completion into a useful artifact. Evaluation traces record these branches rather than discarding them.*

The Family B transition system, the state-relative version of Definition 3.1, the branching–budget bound, and the verification-gap burden $w^{\star}(c)$ are in Appendix C.

4 Fundamental limits of verifier-mediated research agents

Theorem 4.1 bounds information: δ_Γ is the total-variation distance between the two laws of the complete verifier signature, including declared initial information and every possible checker reply.

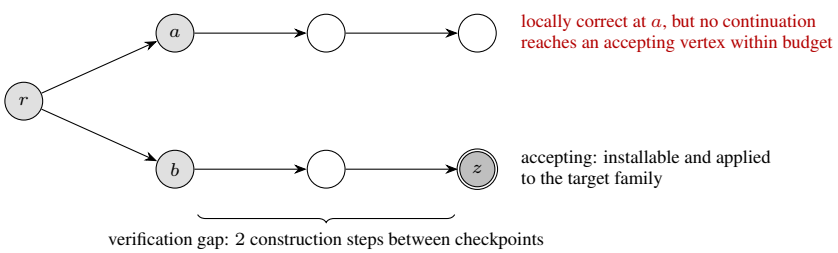


Figure 2: **Passing the checker does not locate the useful branch: a correct-but-unhelpful vertex certifies a local statement that no admissible continuation extends into an accepted artifact.** Shaded vertices are verification checkpoints (locally correct); the double vertex is accepting. Both branches from the root pass the checker at depth one; only one can be completed. The verification-gap burden of a path is the largest number of construction steps between consecutive checkpoints.

For a fixed environment pair, an admissible policy only processes this information, so equal-prior success is at most $(1 + \delta_\Gamma)/2$, regardless of solver or query budget; we verify the bound exactly on finite interfaces (Figure 1) and do not audit its premises for any deployed agent. Theorem 4.2 instead gives a minimax $\Theta(d)$ factor between charged rebuilding and free frame restoration in the opaque marked-leaf model. Bridge conditions for deployed agents are in Appendix D.1.

4.1 Two bounds for LLM-based research agents

Definition 4.1 (Restricted research policy)**.** *A restricted research policy acts causally on declared initial information, observed checker replies, and private randomness independent of the world and target. A frozen solver specifies action selection; the environment separately specifies responses and state transitions. Every revealed diagnostic belongs to the observation interface. Flat and retained-frame policies are distinguished by the operations and charges of the opaque marked-leaf model, not by the presence of a scratchpad or retrieved text. Stored text alone does not establish access to a restorable frame.*

Setting for Theorem 4.1. The interaction model is that of §H.1 (appendix). A *pair of environments* (P_0, P_1) over hidden worlds W is given, $U \sim \mathrm{Bernoulli}(1/2)$ selects which one is used, $I(W)$ is the declared initial information, and the verifier answers each candidate-proof query q with $r_W(q)$, with no trace or side channel (A1)–(A6 there). The *complete verifier signature* is $\Gamma_V(W) = (I(W), (r_W(q))_q)$ and $\delta_\Gamma := \|\mathrm{Law}(\Gamma_V \mid U = 0) - \mathrm{Law}(\Gamma_V \mid U = 1)\|_{\mathrm{TV}}$. $\mathcal{P}_B^T$ denotes the policies of solver T that make at most B queries, H_{τ_π} their tagged history and $\widehat{a}_\pi$ their output. The *successful-artifact sets* $A^*(0), A^*(1)$ of the two environments are assumed disjoint (for binary U, $A^*(u) = \{u\}$); $A^*(u)$ is the set of artifacts the verifier of environment u certifies as proofs. The primary quantity of this section is the *budgeted verified-proof success probability*

$$\mathrm{SP}_B^T(\pi) := \mathbb{P}\big(T \text{ running } \pi \text{ outputs a verified proof, } \widehat{a}_\pi \in A^*(U), \text{ within } B \text{ queries}\big), \qquad \mathrm{SP}_B^T := \sup_{\pi \in \mathcal{P}_B^T} \mathrm{SP}_B^T(\pi), \tag{1}$$

with $U \sim \mathrm{Bernoulli}(1/2)$, so $\mathrm{SP}_B^T(\pi) = \frac{1}{2}\sum_u \mathbb{P}(\widehat{a}_\pi \in A^*(u) \mid U = u)$. On the finite search graph of Result C it coincides with SD_B^* (Appendix D.1).

Assumption 4.1 (Modelling assumptions for $\mathcal{M}_{\mathrm{LLM}}$)**.** *(B) The signature Γ_V is defined from* every *output the checker returns (including error strings and any other non-pass/fail text), not only from a pass/fail bit; if the checker returns more, δ_Γ is computed for that richer signature.*

Core proof idea. Every query response is a coordinate of the complete signature $\Gamma_V(W)$ and the policy acts causally, so its history is a measurable function of $\Gamma_V(W)$ and private randomness independent of (W, U): a U-independent garbling of Γ_V in the sense of Blackwell's comparison of experiments (Blackwell, 1953). Data processing bounds the TV distance between the conditional output laws by δ_Γ, and disjointness of $A^*(0), A^*(1)$ converts this into the success ceiling.

Theorem 4.1 (Verifier-interface ceiling (I))**.** *Fix an equal-prior environment pair, measurable disjoint successful-artifact sets A_0, A_1, and a stipulated policy class $\mathcal{P}_B^T$. Let Γ contain all declared initial*

information and responses, and write

$$\delta_\Gamma = \mathrm{TV}(\mathrm{law}(\Gamma \mid U = 0), \mathrm{law}(\Gamma \mid U = 1)).$$

For each policy in this class, assume that its history and output are jointly generated from Γ *by a policy-specific Markov kernel that is common to the two environments and independent of the world and label. Then every policy in precisely this class satisfies*

$$\mathrm{SP}_B^T(\pi) = \big(\Pr(\widehat{a}_\pi \in A_0 \mid U = 0) + \Pr(\widehat{a}_\pi \in A_1 \mid U = 1)\big)/2 \le (1 + \delta_\Gamma)/2,$$

and hence $\mathrm{SP}_B^T \le (1+\delta_\Gamma)/2$*. It concerns the equal-prior average, not either environment separately.*

Proof. The output is a garbling of Γ by the common kernel, so data processing gives $\mathrm{TV}(\nu_0, \nu_1) \le \delta_\Gamma$ for the conditional output laws ν_u. Disjointness gives

$$\nu_0(A_0) + \nu_1(A_1) \le 1 + \mathrm{TV}(\nu_0, \nu_1).$$

Divide by two, use (1), and take the supremum over the stipulated class. □

When the ceiling is vacuous. At $\delta_\Gamma = 1$, the complete-signature laws are mutually singular and the bound is vacuous, without guaranteeing success for a budgeted policy. Public-input revelation is sufficient: if $U = g(I)$ almost surely, then $\delta_\Gamma = 1$ (Proposition G.2).

Intuition. The separation concerns restorable state, not memory capacity: after a reset, a flat executor must pay for d extensions to reach another leaf, however much private bookkeeping it retains. A retained executor restores a stored ancestor at zero verifier-call cost; this difference yields the minimax $\Theta(d)$ separation, not a claim about token context or every search trajectory.

Theorem 4.2 (Context-reset lower bound (II): retained state is worth a depth factor)**.** *Let* $m \ge 2$*,* $d \ge 1$ *and let* $\mathcal{W}_{m,d} = \{W_t : t \in [m]^d\}$*, where* W_t *is the opaque marked-leaf world of the paragraph below with alphabet* $[m]$*, depth* d *and marked leaf* t*. Write* $T^\pi(W)$ *for the number of verifier calls (extends) until the current word is terminal, and* $n_{m,d} = \sum_{k=1}^{d} m^k < m^{d+1}/(m-1)$*.*

- *(a) For every deterministic policy* π *on* M^{flat}*, however much internal memory it has and with no budget cap, there is* t *with* $T^\pi(W_t) \ge d\, m^d$*.*
- *(b) For every randomized policy on* M^{flat}*,* $\mathbb{E}_{t\sim\mathrm{Unif}[m]^d}\, \mathbb{E}\, T^\pi(W_t) \ge d\,(m^d+1)/2$*.*
- *(c) The retained-state reference schedule* $\pi^{\mathrm{ret}}_{\mathrm{ref}}$ *(preorder DFS, Remark D.19 paragraph) on* M^{ret} *satisfies* $T(W_t) \le n_{m,d}$ *for every* t*.*
- *(d) (tightness of (a)) The flat policy that rebuilds the words of* $[m]^d$ *in lexicographic order has* $T(W_t) = d\,(\mathrm{rank}(t)+1) \le d\, m^d$ *for every* t*.*

Consequently $\min_{\pi\in\mathrm{flat}} \max_t T^\pi \big/ \max_t T^{\mathrm{ret}} \ge d\, m^d/n_{m,d} > d(m-1)/m \ge d/2$*, and the same ratio of averages is* $\ge d(m-1)/(2m) \ge d/4$*. These are minimax and uniform-leaf expected separations between interfaces, not a cost ratio for every marked leaf or every agent trajectory.*

Opaque marked-leaf model for Theorem 4.2. Fix a complete m-ary tree of depth d and one unknown marked leaf. Initial information is identical for every possible marked leaf. An extension costs one verifier call and reveals only whether the resulting word is marked; all other replies are identical. The marked leaf cannot be inspected or tested by an uncharged operation.

One policy is used across all possible marked leaves. A flat executor may extend its current word or reset to the root, but cannot restore an earlier frame. Private bookkeeping is unrestricted and does not restore a frame. A retained executor may restore a stored ancestor at zero verifier-call cost.

Remark 4.3 (Conditional bounds for an LLM agent with audited interfaces). Fix an LLM agent represented as a restricted research policy (Definition 4.1) whose relevant interface satisfies (A1)–(A3) of Appendix D.1.

(I) *Verifier-interface ceiling.* $\frac{1}{2}\big[\mathbb{P}(\hat{a}_\pi \in A^*(0) \mid U{=}0) + \mathbb{P}(\hat{a}_\pi \in A^*(1) \mid U{=}1)\big] \le \dfrac{1+\delta_\Gamma}{2}$ for every environment pair with disjoint successful-artifact sets (equal-prior average, Theorem 4.1), where δ_Γ is the TV-distinguishability of the checker's complete signature (Assumption 4.1(B), §G); it holds for every solver, budget, and search strategy.

Table 1: **The in-run installed-versus-K_0 comparison (n=24), one-to-one; a supporting check for the X2 main comparison.** Each row binds the measured claim of §5 to what it assumes, what code implements it, which recorded column it is computed from, what it is compared against, and where the raw record lives. Values are recomputed from the raw per-test arrays by an independent script.

Claim	Installing a Lean-certified library lowers the held-out solve rate relative to an empty library (installed 0.174 vs. empty 0.306; paired -0.132, cell-bootstrap 95% CI $[-0.236, -0.028]$; 4/13/7 cells +/−/tie; all 4 world means negative).
Assumptions	Definition 2.2: library, solver (C_{base}), proof budget, checker (Lean 4), and sealed test set fixed per cell; research and test theorems disjoint.
Implementation	Installed arm: in-run Lean-checked library of the same world/condition/seed. Baseline arm: forced-empty library K_0, same solver, tests, budget. Analysis: `e2p_k0_paired_utility_audit.py` (does not import the runner).
Result column	Per-test `solve[]` arrays → per-cell $\widehat{U}_b$; $n = 24$ cells (4 worlds × 2 conditions × 3 seeds); 5,000 paired-cell resamples, seed 20260925.
Baseline	Empty library K_0 (no artifact installed), paired per cell.
Robustness	Archived frozen libraries (E2-CoRF-K0, b=4096, n=12): empty 0.403, K_R 0.236, K_V 0.153; K_V-empty $= -0.250$, CI $[-0.333, -0.153]$ (`e2corf_k0_ablation.py`; rows sha256 `46f388a1`). Preliminary, stronger consumer (C_{strong}, 4 worlds × 1 seed, b=4096): empty 1.000, K_R 0.958, K_V 0.917; K_V-empty $= -0.083$ (1/4 cells negative, 3 ties); the empty arm is at ceiling, so this neither confirms nor refutes the main result.
Not claimed	General transfer; a world-level CI (cells cluster in 4 worlds); a test of either theorem; an artifact-addition effect for K_V vs. K_R (independently obtained libraries).

(II) *Context-reset cost.* In the opaque marked-leaf model of Thm 4.2, the flat *minimax* call cost is at least $d(m-1)/m \geq d/2$ times the retained reference's worst-case cost; under a uniform leaf prior the corresponding ratio of expected costs is at least $d(m-1)/(2m)$. Neither ratio applies to each fixed leaf, a particular LLM trajectory, or a fixed finite horizon.

Proof. (I). Condition (b) is exactly the causal no-side-channel condition of §G; Theorem 4.1 applies directly. *(II).* Condition (c) places the agent in M^{flat}; Theorem 4.2(a) and (c) give the ratio. □

Two structural obstacles. Theorem 4.1 limits information; Theorem 4.2 quantifies rebuilding cost under specified operations and charges. Their assumptions are separate, and no joint bound is claimed. The measured harm of §5 is consistent with, though not implied by, the ceiling; it measures post-freeze utility, not hidden-world identification.

The proof of Theorem 4.2, its interpretation, the general-prior and overlap remarks, and the supporting finite-model certificates (Results A–F, III, and P) are in Appendix D.

5 Empirical checks: exploratory evaluation

Attributing the harm (X2 replicate). A replicate of the frozen-library comparison (X2; 4 worlds × 3 seeds, n=12 pairs per budget, C_{base}) records the finish reason and token counts of every test. The harm reproduces and does not shrink from b=4096 to 16384 (diff-of-diff CI $[-0.33, +0.01]$ includes 0): K_V-empty $= -0.222$ $[-0.361, -0.083]$ and -0.375 $[-0.500, -0.236]$ (empty 0.444, K_R 0.319, K_V 0.069; 10 pairs negative, none positive). Installing a library roughly triples the mean prompt length (empty 172, K_R 304, K_V 519 tokens). All 313 failures end with finish reason `stop` and are rejected by Lean (309 with elaboration errors, 4 without); there are no length truncations and no Lean timeouts, and no test uses more than 368 output tokens. The harm is therefore incorrect proofs produced under the library-augmented prompt, not truncation. Theorem 4.1 shows that such a certified-but-not-useful gap is possible in principle whenever the verifier's signature does not determine usefulness; X2 is a measured instance of the gap, not a test of the theorem. Prompt growth co-varies with the arm, so we do not isolate it as the cause (`x2_k0_mech_replicate.py`; `results/e2_k0_mech_r120/x2/full/RESULT.json`).

The evaluation target is held-out utility as defined in §2. Theorem 4.1 and Theorem 4.2 concern separate restricted interfaces; none of the evaluations below tests both interface assumptions in a

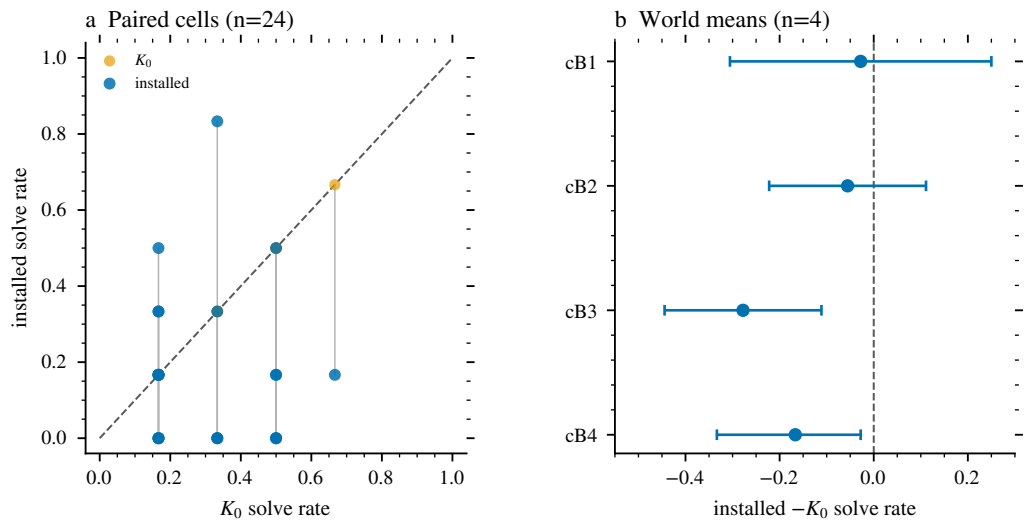


Figure 3: **Cell-level view of the certified-versus-useful comparison in Figure 1(c,d): paired installed-versus-K_0 outcomes, not an R–V contrast.** Figure 1(c,d) pools archived R/V libraries over 12 pairs; here each of 24 matched cells uses the library installed in that run. Left: each cell has its K_0 solve rate (orange) on the diagonal and installed solve rate (blue) at the same x coordinate; the dashed diagonal is equality. Right: four world-level mean differences with descriptive 95% within-world paired-cell bootstrap intervals (5,000 resamples each), not an independent-world confidence interval.

deployed LLM. Figure 3 is the exploratory paired-utility result, Figure 6 diagnoses local accounting, and Figure 1 is the exact finite-model check of the first theorem.

The repository records E2 and E3 protocols in `E2_PREREG.md` and `E3_PREREG.md`, but these records alone do not establish an independently verifiable preregistration timeline. The four-world Family B paired evaluation reported here is exploratory, not a confirmatory E2 or E3 result. Negative and null results are reported alongside positive ones.

Installed-versus-K_0 paired utility (n=24). On four Family B worlds (cB1–cB4), two conditions (R and V), and three seeds, we compare the Lean-checked in-run installed library with a forced-empty K_0 arm on the same world, condition, seed, solver, sealed tests, and proof budget, as in Definition 2.2. Recomputing each solve rate from the recorded per-test outcomes gives installed 0.174 versus K_0 0.306; the paired installed-minus-K_0 mean is -0.132 (24-cell bootstrap 95% CI $[-0.236, -0.028]$, 5,000 paired-cell resamples, fixed seed). Four cells are positive, seven tie, and thirteen are negative. All four world means are negative (exact two-sided world-level sign test $p = 0.125$). The cell bootstrap interval is descriptive because the 24 cells are clustered in only four worlds; it is not a world-level confidence interval. The recorded contrast is negative on average, although four cells have positive differences. This evaluation does not establish a positive population effect or the absence of beneficial artifacts. Figure 3 shows the cell pairs and within-world bootstrap intervals.

Frozen-artifact intervention outcomes: one completed run (negative), one invalidated. Two pre-specified E2 variants were run. (A) *Family-A frozen-bundle transfer to unseen worlds* (E2FU, $B = 600$ per world, 10 worlds, 5 seeds): *invalidated*. The post-compile `sorry`/`admit`/`axiom` filter was applied to the concatenated library-plus-proof source rather than to the generated proof alone; since the injected library contains two `sorry` placeholders, every compiling K_{frozen} cell was scored as a failure (30 of 300 error-free cells rejected, versus 6 of 300 under K_0), so the recorded reach $K_{\text{frozen}} = 0.00$ is a filter artifact and no estimate from this run is reported. (B) *Frozen-artifact utility with a stronger checker base* (E2-CoRF, $n = 12$ pairs, $b \in \{4096, 8192, 16384\}$; C_{base} = DeepSeek-V4-Pro; frozen solver libraries from E2-fix): $\overline{\Delta}_{sp} = -0.222$ (CI $[-0.389, -0.083]$) at $b = 4096$; approach rule not met at any budget ($b = 16384$: -0.111, CI $[-0.208, 0.000]$; $b = 8192$: -0.014, CI $[-0.153, 0.111]$).

A strong-checker preflight (C_{strong} = GPT-6-Astra) reached $s_R = s_V = 1.0$ ceiling and was excluded from confirmatory analysis. E2-CorF compares two discovered libraries (R vs. V); it is directionally negative and does not support a claim that a frozen artifact transfers utility. The separate E2-CORF-K0 library ablation shown in Figure 1(c) compares an empty library, an archived R-condition library, and an archived V-condition library. Because these arms are not a base library and a proof-preserving extension of it, the V-minus-R contrast is a library-replacement comparison, not the artifact-addition intervention of Definition 2.2.

At $b = 4096$, the archived mean solve rates are 0.403, 0.236, and 0.153, respectively. The corresponding paired contrasts are -0.167 for R minus empty, -0.083 for V minus R, and -0.250 for V minus empty, with exploratory intervals $[-0.306, -0.028]$, $[-0.194, 0.014]$, and $[-0.333, -0.153]$ (1,000-resample percentile bootstrap over the 12 paired cells, not independent-world confidence intervals). These results establish neither equivalence nor non-increase, and do not exclude beneficial artifacts. The cell-level counterpart is Appendix Figure 5. Provenance: E2FU W&B `galois-e2fu` (invalidated); E2-CorF W&B `iclr2027-galois/wqkerxny`.

The E2/E3 protocols, GX-2/GX-3 accounting audits, theorem-faithful stack readout, reduced live stack run, consumer burden, and certification-versus-utility diagnostics are in Appendix E.

6 Related work

Statistical provability and machine-assisted proof. Sonoda et al. (2026b) model formal proof search as a finite-horizon MDP and bound finite-budget proof success by action-value error, coverage, and proof length. We keep their finite-budget perspective but move the object of evaluation one level up, from reaching a verified proof to producing a certified structure that helps on held-out theorems. Tao's account of machine-assisted proof (Tao, 2025) motivates the checker's restricted role as error filter and formalization aid.

Theorem-library growth and retrieval-augmented proving. A separate line of work grows and reuses a theorem or premise library: retrieval-augmented tactic prediction (Yang et al., 2023), premise and tactic co-training (Polu & Sutskever, 2020; Han et al., 2021), online self-play search (Lample et al., 2022), informal-to-formal proof sketching (Jiang et al., 2022b), and large-scale synthetic proof data (Xin et al., 2024a). The strong library baseline of §5 is built from this line; we ask what a search that keeps and returns to intermediate structure adds to it, and whether what it finds transfers when frozen. Appendix F tabulates the boundary against nearby systems.

Proof assistants, LLM provers, and discovery. Our checker abstracts a proof assistant (de Moura & Ullrich, 2021; The mathlib Community, 2020; Nipkow et al., 2002); hammers and learned premise selection (Blanchette et al., 2016; Kaliszyk & Urban, 2014; Irving et al., 2016; Bansal et al., 2019; Mikuła et al., 2024; Jiang et al., 2022a) change the oracle interface, to which Theorem 4.1 is agnostic. LLM provers on sealed Olympiad benchmarks (Zheng et al., 2022; Polu et al., 2023; First et al., 2023; Thakur et al., 2024; Song et al., 2024; Xin et al., 2024b; Ren et al., 2025; Lin et al., 2025; Trinh et al., 2024; AlphaProof and AlphaGeometry teams, 2024) report solve rates, the utility side of Definition 2.4; machine-aided discovery (Davies et al., 2021; Fawzi et al., 2022; Romera-Paredes et al., 2024) verifies value after the fact; discoverability asks how often a frozen procedure returns a useful certified artifact.

7 Limitations and conclusion

The two restricted theorems prove interface-conditional bounds; the exploratory evaluations do not measure protocol-valid discovery or transfer, and the identification ceiling is only a conditional necessary bound. Raising the ceiling for a fixed environment pair requires increasing δ_Γ; utility-revealing feedback is a candidate interface change, not a guarantee of discovery. The installed-versus-K_0 comparison (Figure 3) is negative on average but establishes no population transfer effect; confirmatory discovery and frozen-artifact transfer remain open under the stated protocol. By separating evaluation target from oracle interface and grounding each claim in a restricted model, this work charts a principled theoretical boundary for LLM-based automated mathematics—a boundary that must be understood before it can be crossed.

REFERENCES

Daattavya Aggarwal, Oisin Kim, Carl Henrik Ek, and Challenger Mishra. Discovering mathematical concepts through a multi-agent system. *arXiv preprint arXiv:2603.04528*, 2026. URL https://arxiv.org/abs/2603.04528.

AlphaProof and AlphaGeometry teams. AI achieves silver-medal standard solving international mathematical olympiad problems. Google DeepMind blog, 2024.

Kshitij Bansal, Sarah M. Loos, Markus N. Rabe, Christian Szegedy, and Stewart Wilcox. HOList: An environment for machine learning of higher-order theorem proving. In *International Conference on Machine Learning (ICML)*, 2019.

Jacob D. Bekenstein. Universal upper bound on the entropy-to-energy ratio for bounded systems. *Physical Review D*, 23(2):287–298, 1981.

Ian Berlot-Attwell, Frank Rudzicz, and Xujie Si. Library learning doesn't: The curious case of the single-use "library". *arXiv preprint arXiv:2410.20274*, 2024. URL https://arxiv.org/abs/2410.20274.

David Blackwell. Equivalent comparisons of experiments. *Annals of Mathematical Statistics*, 24(2):265–272, 1953.

Jasmin Christian Blanchette, Cezary Kaliszyk, Lawrence C. Paulson, and Josef Urban. Hammering towards QED. *Journal of Formalized Reasoning*, 9(1):101–148, 2016.

Edoardo Botta, Yuchen Li, Aashay Mehta, Jordan T. Ash, Cyril Zhang, and Andrej Risteski. On the query complexity of verifier-assisted language generation. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, 2025. URL https://arxiv.org/abs/2502.12123.

George Cybenko. Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems*, 2(4):303–314, 1989.

Alex Davies, Petar Veličković, Lars Buesing, Sam Blackwell, Daniel Zheng, Nenad Tomašev, Richard Tanburn, Peter Battaglia, Charles Blundell, András Juhász, Marc Lackenby, Geordie Williamson, Demis Hassabis, and Pushmeet Kohli. Advancing mathematics by guiding human intuition with AI. *Nature*, 600:70–74, 2021.

Leonardo de Moura and Sebastian Ullrich. The Lean 4 theorem prover and programming language. In *Automated Deduction – CADE 28*, volume 12699 of *LNCS*, pp. 625–635. Springer, 2021.

Kevin Ellis, Catherine Wong, Maxwell Nye, Mathias Sablé-Meyer, Luc Cary, Lucas Morales, Luke Hewitt, Armando Solar-Lezama, and Joshua B. Tenenbaum. Dreamcoder: Growing generalizable, interpretable knowledge with wake–sleep bayesian program learning. *arXiv preprint arXiv:2006.08381*, 2020. URL https://arxiv.org/abs/2006.08381.

Alhussein Fawzi, Matej Balog, Aja Huang, Thomas Hubert, Bernardino Romera-Paredes, Mohammadamin Barekatain, Alexander Novikov, Francisco J. R. Ruiz, Julian Schrittwieser, Grzegorz Swirszcz, David Silver, Demis Hassabis, and Pushmeet Kohli. Discovering faster matrix multiplication algorithms with reinforcement learning. *Nature*, 610:47–53, 2022.

Marc Finzi, Shikai Qiu, Yiding Jiang, Pavel Izmailov, J. Zico Kolter, and Andrew Gordon Wilson. From entropy to epiplexity: Rethinking information for computationally bounded intelligence. *arXiv preprint arXiv:2601.03220*, 2026.

Emily First, Markus N. Rabe, Talia Ringer, and Yuriy Brun. Baldur: Whole-proof generation and repair with large language models. In *Proceedings of the 31st ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering (ESEC/FSE)*, 2023.

Gabriel Grand, Lionel Wong, Matthew Bowers, Theo X. Olausson, Muxin Liu, Joshua B. Tenenbaum, and Jacob Andreas. Lilo: Learning interpretable libraries by compressing and documenting code. In *ICLR*, 2024. URL https://arxiv.org/abs/2310.19791.

Jesse Michael Han, Jason Rute, Yuhuai Wu, Edward W. Ayers, and Stanislas Polu. Proof artifact co-training for theorem proving with language models. *arXiv preprint arXiv:2102.06203*, 2021. URL https://arxiv.org/abs/2102.06203.

Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5):359–366, 1989.

Geoffrey Irving, Christian Szegedy, Alexander A. Alemi, Niklas Eén, François Chollet, and Josef Urban. DeepMath: Deep sequence models for premise selection. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2016.

Albert Q. Jiang, Wenda Li, Szymon Tworkowski, Konrad Czechowski, Tomasz Odrzygóźdź, Piotr Miłoś, Yuhuai Wu, and Mateja Jamnik. Thor: Wielding hammers to integrate language models and automated theorem provers. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022a.

Albert Q. Jiang, Sean Welleck, Jin Peng Zhou, Wenda Li, Jiacheng Liu, Mateja Jamnik, Timothée Lacroix, Yuhuai Wu, and Guillaume Lample. Draft, sketch, and prove: Guiding formal theorem provers with informal proofs. *arXiv preprint arXiv:2210.12283*, 2022b. URL https://arxiv.org/abs/2210.12283.

Cezary Kaliszyk and Josef Urban. Learning-assisted automated reasoning with Flyspeck. *Journal of Automated Reasoning*, 53(2):173–213, 2014.

Levente Kocsis and Csaba Szepesvári. Bandit based Monte-Carlo planning. In *European Conference on Machine Learning (ECML)*, volume 4212 of *LNCS*, pp. 282–293. Springer, 2006.

Guillaume Lample, Marie-Anne Lachaux, Thibaut Lavril, Xavier Martinet, Amaury Hayat, Gabriel Ebner, Aurélien Rodriguez, and Timothée Lacroix. Hypertree proof search for neural theorem proving. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. URL https://arxiv.org/abs/2205.11491.

Haohan Lin, Zhiqing Sun, Sean Welleck, and Yiming Yang. Lean-STaR: Learning to interleave thinking and proving. In *International Conference on Learning Representations (ICLR)*, 2025.

Maciej Mikuła, Szymon Tworkowski, Szymon Antoniak, Bartosz Piotrowski, Albert Q. Jiang, Jin Peng Zhou, Christian Szegedy, Łukasz Kuciński, Piotr Miłoś, and Yuhuai Wu. Magnushammer: A transformer-based approach to premise selection. In *International Conference on Learning Representations (ICLR)*, 2024.

Tobias Nipkow, Lawrence C. Paulson, and Markus Wenzel. *Isabelle/HOL: A Proof Assistant for Higher-Order Logic*, volume 2283 of *LNCS*. Springer, 2002.

Kazuki Ota, Takayuki Osa, and Tatsuya Harada. Self-supervised theorem discovery in a formal axiomatic system. *arXiv preprint arXiv:2606.28747*, 2026. URL https://arxiv.org/abs/2606.28747.

Stanislas Polu and Ilya Sutskever. Generative language modeling for automated theorem proving. *arXiv preprint arXiv:2009.03393*, 2020. URL https://arxiv.org/abs/2009.03393.

Stanislas Polu, Jesse Michael Han, Kunhao Zheng, Mantas Baksys, Igor Babuschkin, and Ilya Sutskever. Formal mathematics statement curriculum learning. In *International Conference on Learning Representations (ICLR)*, 2023.

Yuxiao Qu, Anikait Singh, Yoonho Lee, Amrith Setlur, Ruslan Salakhutdinov, Chelsea Finn, and Aviral Kumar. RLAD: Training LLMs to discover abstractions for solving reasoning problems. *arXiv preprint arXiv:2510.02263*, 2025. URL https://arxiv.org/abs/2510.02263.

Z. Z. Ren, Zhihong Shao, Junxiao Song, Huajian Xin, et al. DeepSeek-Prover-V2: Advancing formal mathematical reasoning via reinforcement learning for subgoal decomposition. *arXiv preprint arXiv:2504.21801*, 2025.

Bernardino Romera-Paredes, Mohammadamin Barekatain, Alexander Novikov, Matej Balog, M. Pawan Kumar, Emilien Dupont, Francisco J. R. Ruiz, Jordan S. Ellenberg, Pengming Wang, Omar Fawzi, Pushmeet Kohli, and Alhussein Fawzi. Mathematical discoveries from program search with large language models. *Nature*, 625:468–475, 2024.

David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, Timothy Lillicrap, Karen Simonyan, and Demis Hassabis. A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play. *Science*, 362(6419):1140–1144, 2018.

Ray J. Solomonoff. A formal theory of inductive inference. part I. *Information and Control*, 7(1):1–22, 1964.

Peiyang Song, Kaiyu Yang, and Anima Anandkumar. Towards large language models as copilots for theorem proving in Lean. *arXiv preprint arXiv:2404.12534*, 2024.

Sho Sonoda, Shunta Akiyama, and Yuya Uezato. Exponential sample complexity separation between flat and hierarchical agentic theorem provers. *arXiv preprint arXiv:2602.10512*, 2026a. URL https://arxiv.org/abs/2602.10512.

Sho Sonoda, Shunta Akiyama, and Yuya Uezato. Why agentic theorem prover works: A statistical provability theory of mathematical reasoning models. *arXiv preprint arXiv:2602.10538*, 2026b. URL https://arxiv.org/abs/2602.10538.

Terence Tao. Machine assisted proof. *Notices of the American Mathematical Society*, 72(1), 2025. URL https://www.ams.org/journals/notices/202501/noti3041/noti3041.html.

Alexander K Taylor, Junyi Zhang, Ethan Ji, Vigyan Sahai, Haikang Deng, Yuanzhou Chen, Yifan Yuan, Di Wu, Jia-Chen Gu, Kai-Wei Chang, Nanyun Peng, Amit Sahai, and Wei Wang. TaoBench: Do automated theorem prover LLMs generalize beyond MathLib? *arXiv preprint arXiv:2603.12744*, 2026. URL https://arxiv.org/abs/2603.12744.

Amitayush Thakur, George Tsoukalas, Yeming Wen, Jimmy Xin, and Swarat Chaudhuri. An in-context learning agent for formal theorem-proving. In *Conference on Language Modeling (COLM)*, 2024.

The mathlib Community. The Lean mathematical library. In *Proceedings of the 9th ACM SIGPLAN International Conference on Certified Programs and Proofs (CPP)*, pp. 367–381, 2020.

Trieu H. Trinh, Yuhuai Wu, Quoc V. Le, He He, and Thang Luong. Solving olympiad geometry without human demonstrations. *Nature*, 625:476–482, 2024.

Haiming Wang, Huajian Xin, Chuanyang Zheng, Lin Li, Zhengying Liu, Qingxing Cao, Yinya Huang, Jing Xiong, Han Shi, Enze Xie, Jian Yin, Zhenguo Li, Heng Liao, and Xiaodan Liang. Lego-prover: Neural theorem proving with growing libraries. In *ICLR*, 2024. URL https://arxiv.org/abs/2310.00656.

David H. Wolpert and William G. Macready. No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, 1(1):67–82, 1997.

Huajian Xin, Daya Guo, Zhihong Shao, Zhizhou Ren, Qihao Zhu, Bo Liu, Chong Ruan, Wenda Li, and Xiaodan Liang. DeepSeek-Prover: Advancing theorem proving in LLMs through large-scale synthetic data. *arXiv preprint arXiv:2405.14333*, 2024a. URL https://arxiv.org/abs/2405.14333.

Huajian Xin, Z. Z. Ren, Junxiao Song, Zhihong Shao, Wanjia Zhao, Haocheng Wang, Bo Liu, et al. DeepSeek-Prover-V1.5: Harnessing proof assistant feedback for reinforcement learning and Monte-Carlo tree search. *arXiv preprint arXiv:2408.08152*, 2024b.

Kaiyu Yang, Aidan M. Swope, Alex Gu, Rahul Chalamala, Peiyang Song, Shixing Yu, Saad Godil, Ryan Prenger, and Anima Anandkumar. LeanDojo: Theorem proving with retrieval-augmented language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. URL https://arxiv.org/abs/2306.15626.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of thoughts: Deliberate problem solving with large language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Youyuan Zhang, Jialiang Sun, Hangrui Bi, Chuqin Geng, Wenjie Ma, Zhaoyu Li, and Xujie Si. Dream-prover: Evolving transferable lemma libraries via a wake-sleep theorem-proving agent. *arXiv preprint arXiv:2604.26311*, 2026. URL https://arxiv.org/abs/2604.26311.

Kunhao Zheng, Jesse Michael Han, and Stanislas Polu. miniF2F: A cross-system benchmark for formal olympiad-level mathematics. In *International Conference on Learning Representations (ICLR)*, 2022.

A Deferred material from Sections 1–2

Status in this submission. Fix a declared research procedure throughout each definition. When procedures are compared, SD is indexed by the research procedure as well as the evaluation solver; artifact space, research output map, certification indicator, and utility-threshold event are assumed measurable.

The definitions distinguish certification, population held-out utility, and statistical discoverability. Accepting-vertex reachability concerns certification and does not establish the utility-threshold event. The primary E2 comparison is a paired difference in sealed-test solve rates between research procedures; it is not itself an estimator of SD. The manuscript reports distinct exploratory comparisons. The 24-cell audit (§5) compares recorded installed-library outcomes with forced-empty-library outcomes. The E2-CorF-K0 ablation (§5) compares an empty library, an archived R-condition library, and an archived V-condition library.

Interpreting the latter as the intervention of Definition 2.2 additionally requires demonstrating that the installed library preserves the declared base library and adds an admissible artifact under an unchanged evaluation protocol. Naming independently obtained libraries as a base and its extension does not establish this relationship. Neither comparison estimates statistical discoverability.

A finite-evaluation threshold-exceedance rate based on estimated utility is a separate statistic and must be reported with evaluation uncertainty. No procedure-specific estimate of statistical discoverability is reported. The completed exploratory library comparisons do not establish a protocol-valid artifact-addition or general transfer effect. The invalidated E2FU run is excluded from inference. The identification appendix applies to discoverability only under its explicit environment-design assumptions.

B A candidate-generation/selection identity

This section gives an accounting identity, not a measured intervention effect.

The experiments below report where a research procedure fails: in exposing a useful candidate, or in selecting it. The following identity fixes that vocabulary; it is an accounting device, not a result about any method.

At a decision state s, let $\mathcal{A}(s)$ be a finite action set, $Q^\star(s,a)$ the value of action a, and $C(s) \subseteq \mathcal{A}(s)$ the nonempty candidate set exposed to a selection rule that picks $\widehat{a} \in C(s)$.

Remark B.1 (Candidate-generation and selection decomposition). For every such state and selected action,

$$V^\star(s) - Q^\star(s,\widehat{a}) = \underbrace{V^\star(s) - \max_{a\in C(s)} Q^\star(s,a)}_{\text{candidate-generation loss}} + \underbrace{\max_{a\in C(s)} Q^\star(s,a) - Q^\star(s,\widehat{a})}_{\text{selection loss}},$$
$$V^\star(s) := \max_{a\in\mathcal{A}(s)} Q^\star(s,a).$$

The proof is in Appendix J.

The identity separates a failure to expose a high-value candidate from a failure to select one. It does not turn a locally verified candidate into a useful artifact, and it does not identify either loss from an observed success rate: estimating them separately requires the prespecified protocol of §5, in which candidate exposure and selection are logged as distinct events.

C Deferred material on execution-state graphs

Family B finite transition instance. The declared Family B instance is a finite word graph with a separate rollback executor and word-reachability oracle. Their state updates, costs, and independent DP/enumeration agreement are specified in Appendix J; the word oracle does not measure online discovery or rollback cost.

For a finite environment c, fix a typed artifact language, declared construction operators, and admissible size bounds. We propose recording execution by states $X = (K, F, M, b)$, comprising the installed library, open candidates, executor memory, and remaining budget. This is a state representation, not yet a fully specified finite transition system.

A complete instantiation must bound the memory domain and specify state updates, unsuccessful verification, candidate deletion, restart actions, and the cost of every action. An operator with multiple inputs may execute only when every input is available in the current state. This requirement concerns chronological execution and is not represented by the length of a single dependency path.

For the declared Family B instance above, this state representation is completely instantiated: the legal state space $\mathcal{X}$, initial state X_0, and the rules extend_i, $\mathsf{backtrack}$, and halt are a finite transition system in this sense, and Proposition J.1 closes the definition-to-enumeration connection for it by identifying the word-oracle recurrence $V(w, b)$ with independent breadth-first enumeration. That identity is independently audited, not only asserted, in Figure 13 (Table 12, Appendix J); the corresponding held-out utility comparison is reported separately, after the audit, in Figure 3 (§5), following the definition, proposition, audit, utility order of the S1 story.

Relativization to a state. The definition above is the special case of a state-relative one. For a state $X = (K, F, M, b)$ and a set ρ of permitted restart operations (which open candidates may be discarded or kept), a locally correct $a \in F$ is (K, F, b, ρ)-*correct but unhelpful* if no state reachable from X under ρ with residual budget b is accepting. Definition 3.1 is the case $F = \{v\}$ and $\rho = \varnothing$.

C.1 A one-stage branching–budget bound

We use a separate idealized attempt model. Its cost parameter is not identified with the verification-gap statistic. Suppose successive attempted completions have independent success indicators with common probability p. Every attempt, including a failed one, costs at least a, where $0 < a < \infty$.

Remark C.1 (Attempt-cost bound). For budget $b \geq 0$, let $n = \lfloor b/a \rfloor$. Then

$$\Pr(\text{success within budget } b) \leq 1 - (1-p)^n.$$

When $n = 0$, the right-hand side is interpreted as zero.

Proof. At most n attempts can be completed. Success among the completed attempts is contained in success among the first n independent trials. □

Remark C.2 (Idealized reusable-prefix comparison). Assume every attempt costs exactly a, including failures. Suppose a prefix of cost s, with $0 < s < a$, is shared by every attempt, can be built once, and reduces every subsequent attempt's cost to exactly $a - s$. Assume this intervention preserves independent success probability p.

Define $n_{\mathrm{keep}} = 0$ for $b < s$, and $n_{\mathrm{keep}} = \lfloor (b - s)/(a - s) \rfloor$ otherwise. A procedure completing all affordable attempts has success probability $1 - (1 - p)^{n_{\mathrm{keep}}}$. Rebuilding the prefix for every attempt gives $1 - (1 - p)^{\lfloor b/a \rfloor}$, which is no larger.

Proof. For $b < a$, neither procedure completes an attempt. For $b \geq a$, $(b - s)/(a - s) \geq b/a$, so retention admits at least as many attempts. Apply the independent-trial formula. □

These statements concern the stipulated attempt model. Applying them to Family B or an online solver requires separate evidence for the cost, independence, and success-probability assumptions.

No empirical retention result is used here. The available exploratory Family B cells do not identify a retention effect: their conditions alter retry opportunities as well as retained state, and they do not provide a paired installed-library versus K_0 held-out evaluation of the same pre-test frozen artifact. We therefore do not report their aggregates as evidence about discovery, held-out utility, or the effect of retention or rollback. The protocol in §5 states the additional frozen-bundle, paired-evaluation, and resource-accounting records required before such an empirical result can be interpreted.

The E1w discovery/usage-cost audit and forced-waypoint qualification are in Appendix J; neither establishes a held-out utility effect.

Whether a vertex is unhelpful depends on what the procedure can still do: a construction a that is insufficient on its own may become useful once a saved intermediate b is available, and a dead end can be abandoned only if the procedure can return to an earlier state. The interventions compared in §5 – keeping intermediate structure and returning from dead ends – change exactly these two capabilities.

The construction depth of a path is its number of construction edges. A reference depth d_{build} is the depth of a declared reference construction under this language. It is an operational upper bound and an experimental control variable, not a mathematical minimum. We call a depth or path "minimal" only when the corresponding finite graph has actually been completely enumerated; if enumeration reaches a bound, the relevant quantity is recorded as unknown.

C.2 Discovery burden on a finite search graph

Along a directed root-to-accepting path $\pi = (v_0, e_1, v_1, \ldots, e_m, v_m)$ of $\mathcal{G}_c$, call v_0 and every locally correct vertex a *verification checkpoint*. Write $0 = i_0 < i_1 < \cdots < i_r = m$ for the indices of the checkpoints on π; the accepting endpoint is a checkpoint. Its verification-gap burden is

$$g(\pi) := \max_{0 \leq \ell < r} (i_{\ell+1} - i_\ell), \qquad w^\star(c) := \min_{\pi \in \mathcal{P}_c} g(\pi),$$

where $\mathcal{P}_c$ is the set of root-to-accepting paths. Thus $w^\star(c)$ is the smallest, over useful paths, of the largest number of construction operations required between consecutive checkable states. Conventions: a path of length $m = 0$ has $g = 0$; if $\mathcal{P}_c = \emptyset$ then $w^\star(c)$ is undefined (not $+\infty$ and not 0). It is defined only when the finite graph and its accepting vertices have been completely enumerated; otherwise it is reported as undefined/unknown.

Remark C.3 (Finite-graph verification-gap bound). Assume the completely enumerated finite graph has a root-to-accepting path. Every such path π satisfies $g(\pi) \geq w^\star(c)$. For a positive-length path, at least one consecutive-checkpoint gap has length at least $w^\star(c)$. For a zero-length accepting path, $g(\pi) = w^\star(c) = 0$, and no gap is asserted. This is a dependency-path statement; application to an execution trace requires a specified dependency-record map and does not lower-bound total search work, token usage, or wall-clock verification intervals.

Proof in Appendix J.

What the bound does and does not say. Remark C.3 is a necessary condition on traces. It yields no discovery rate, no number of solver calls, no token or wall-time cost, and no effect of retaining intermediate structure. Equal $w^\star$ is compatible with very different difficulty: if every one of N candidates can be built in one step and passes the checker ($w^\star(c) = 1$) but only one is accepting, a procedure that generates candidates independently and uniformly produces the accepting one within B attempts with probability $1 - (1 - 1/N)^B$, which at $B = 8$ is about 0.996 for $N = 2$, 0.118 for $N = 64$, and 0.008 for $N = 1024$. Local verifiability therefore does not determine discoverability; the branching among locally correct candidates, and whether the procedure can keep and return to intermediate states, do. That gap is what the main experiments measure.

Remark C.4 (Interpretation of d_{build}). The reference value d_{build} specifies the depth of a declared reference construction in the fixed language. It may be used as an experimental stratum or as an upper bound on that reference path. It is not a mathematical minimum depth, nor does it determine $w^\star(c)$. A minimality claim requires the complete finite enumeration stated above.

D Deferred material on the restricted analyses

D.1 From interfaces to agents

SP and SD*. The budgeted verified-proof success probability SP_B^T of (1) is the success-probability definition of discoverability used throughout: on the finite deterministic search graph of Result C, $\mathrm{SP}_B^T(\pi) = \mathrm{SD}_B^\pi = \mathbb{P}^\pi(T_{\mathcal{G}_R} \le B)$ and $\mathrm{SP}_B^T = \mathrm{SD}_B^*$ (with the hitting time $T_{\mathcal{G}_R}$ of the verified-proof set), and $\mathrm{SP}_b^\pi(K)$ in §5 is the same quantity for a consumer with library K. History total variation is not a primary quantity; it enters only through the auxiliary Proposition H.1 (Appendix H).

Remark D.1 (What the main-text theorems answer). Part (I) is an information ceiling only for a hidden-world pair whose complete signature has $\delta_\Gamma < 1$; it is vacuous for a target determined by public input. Part (II) is a minimax or uniform-prior accounting separation in the opaque marked-leaf model, not a cost for every agent or world. The audits in Table 2 do not yet establish that one deployed agent meets all of the bridge assumptions.

The two oracle theorems above apply to an LLM research agent only after its *observed interface and charged operations* have been mapped to the corresponding mathematical experiment. A fixed model or a finite query budget does not establish that mapping: information available in the public problem, checker diagnostics, stored text, and provider-side state may each change the experiment. We therefore separate assumptions needed for an informative application from the statements that remain true but vacuous outside it. **Interface audit status.** GX-1′ tests the A1 feedback interface and is not a test of Theorem 4.2. GX-2/GX-3 test A2-style local checker-call accounting, while free restoration of provider-side LLM proof state remains unestablished.

(A1) **Pass/fail-only observation.** The checker reveals only the declared pass/fail bit, with no Lean error body, proof-state text, timing, or other target-dependent side channel. This can hold for an agent behind a verified bit-only wrapper; it fails when raw Lean diagnostics or other target-dependent outputs reach the policy. Theorem 4.1 still holds with a richer *complete* signature, but its distance must be recomputed and may equal one.

(A2) **Charged rebuilding, free restoration.** In the opaque marked-leaf comparison, each prefix extension/rebuild is one charged checker call, whereas a retained executor restores an earlier verified frame at zero checker-call cost; private notes alone cannot restore a frame. This can hold for an explicit proof-state stack with that accounting; a fresh prompt does not prove it, and a store of text or a provider session is not a free restoration unless its operational effect and cost are verified.

(A3) **Non-public hidden target.** The target label U is not a measurable function of the declared public input I, and the successful artifact sets for the compared worlds are disjoint. This can hold for sealed hidden-world tasks; it fails when a public theorem statement or generator already determines the target. Non-publicness alone is *not* sufficient for a nontrivial information bound: the complete-signature distance must also satisfy $\delta_\Gamma < 1$.

Under (A1) and (A3), and the no-side-channel Markov and disjointness hypotheses of Theorem 4.1, bound (I) limits the *equal-prior average* success across two hidden worlds to $(1 + \delta_\Gamma)/2$; it does not limit success on one public mathematics problem. Under (A2) and the common opaque, non-public

Assumption	Corresponding audit	Current inference
(A1)	GX-1′ feedback-slot wiring and exposed-error audit	Not yet verified: GX-1 has no exposed error slots and its raw cells are absent here; a successful GX-1′ audit is still needed.
(A2)	GX-2 local-frame paired comparison, 180 episodes	Checker-frame accounting is implemented locally; the ratios by depth are 1.292, 1.780, 1.897. Provider-retained LLM context is not established.
(A3)	G-3 exact counterexample worlds `g3w0005`/`g3w0022`	Finite counterexamples probe disjointness/garbling assumptions, not hidden-target status for deployed public-input tasks.

Table 2: Tests of the interface-to-agent bridge. None alone establishes that a deployed mathematical agent satisfies all three assumptions.

marked-leaf experiment of Theorem 4.2, bound (II) separates flat and retained interfaces in *worst-case* or *uniform-leaf expected* uncapped checker calls. Neither statement is a measured lower bound on the cost of a particular LLM agent or a fixed-horizon success ratio. A marked leaf encountered first costs d calls in the flat interface, so an individual trajectory need not show any depth-factor disadvantage.

Remark D.2 (General two-point prior). Under the hypotheses of Theorem 4.1, let $U \sim \mathrm{Bernoulli}(p)$ with $p \in (0,1)$. Then for every $\pi \in \mathcal{P}_B^T$ the Bayes success probability $S_p = (1-p)\,\mathbb{P}(\widehat{a}_\pi \in A^*(0) \mid U{=}0) + p\,\mathbb{P}(\widehat{a}_\pi \in A^*(1) \mid U{=}1)$ satisfies

$$S_p \le \max(p, 1-p) + \min(p, 1-p)\,\delta_\Gamma.$$

At $p = \frac{1}{2}$ this is $(1+\delta_\Gamma)/2$.

Proof. Write $\nu_u = \mathrm{Law}(\widehat{a}_\pi \mid U{=}u)$, $A_u = A^*(u)$, and $a = \min(p, 1-p)$, $b = \max(p, 1-p)$; let u° be the index carrying weight b. Then $S_p = a[\nu_0(A_0) + \nu_1(A_1)] + (b-a)\nu_{u^\circ}(A_{u^\circ})$. Disjointness and $\mathrm{TV}(\nu_0, \nu_1) \le \delta_\Gamma$ (data processing through $\Gamma_V \to H \to \widehat{a}$) give $\nu_0(A_0) + \nu_1(A_1) \le 1 + \delta_\Gamma$, and $\nu_{u^\circ}(A_{u^\circ}) \le 1$, so $S_p \le a(1+\delta_\Gamma) + b - a = b + a\delta_\Gamma$. □

Remark D.3 (Disjointness is necessary; overlap-aware bound). Let $C = A^*(0) \cap A^*(1)$ and suppose $C \neq \emptyset$. The constant policy π_c that outputs a fixed $c \in C$ without any query has $\widehat{a}_{\pi_c} \in A^*(u)$ for both u, hence equal-prior average success 1 even when $\delta_\Gamma = 0$. So the bound $(1+\delta_\Gamma)/2$ fails without disjointness. In general, with $\nu_u = \mathrm{Law}(\widehat{a}_\pi \mid U{=}u)$,

$$S_\pi \le \min\Big\{1,\ \frac{1 + \delta_\Gamma + \min\{\nu_0(C), \nu_1(C)\}}{2}\Big\}.$$

Indeed $\nu_0(A_0) + \nu_1(A_1) \le \nu_0(A_0) + \nu_0(A_1) + \delta_\Gamma = \nu_0(A_0 \cup A_1) + \nu_0(C) + \delta_\Gamma \le 1 + \nu_0(C) + \delta_\Gamma$, and symmetrically with $\nu_1(C)$. For $C = \emptyset$ this is Theorem 4.1; for π_c, $\nu_0(C) = \nu_1(C) = 1$ and the bound is 1. Disjointness is therefore not merely a convenience of the proof: the ceiling is about hidden targets that the artifact itself must distinguish.

Remark D.4 (Attribution). The inequalities of Theorem 4.1 and Remark D.2 follow from the Blackwell ordering of experiments (Blackwell, 1953) (TV data processing, a binary decision, a two-point prior, 0–1 payoff). Our contribution is the verifier-interface reduction that establishes the adaptive LLM transcript as a garbling of Γ_V, so that the classical bound applies to search agents of any budget. The bound uses the complete-signature distance δ_Γ; a budget-limited transcript distance $\delta_B^T \le \delta_\Gamma^B \le \delta_\Gamma$ is defined in §H.1 and is not used as the central quantity here.

Proof of Theorem 4.2.

Proof. Chains. In M^{flat} a macro-action either extends the current word w by a letter ($|w| < d$) or resets to the root and extends, so the word after a call is wi or the one-letter word i; from the initial root state it is a one-letter word. Cut the call sequence into maximal blocks (*chains*) that start at a call producing a one-letter word and continue while each call extends the previous word. Within a chain the produced words are $u_1, u_1u_2, \ldots$ of lengths $1, 2, \ldots, r$ with $r \le d$; chains are disjoint blocks of calls. A call that produces a word of length d is therefore the last call of a chain of length exactly d.

(a). Until the terminal is produced every world in $\mathcal{W}_{m,d}$ answers identically, so a deterministic π produces one fixed call sequence $(u_s)_{s \ge 1}$ (possibly infinite) independent of t. If some t never occurs,

$T^\pi(W_t) = \infty$. Otherwise the m^d words of $[m]^d$ have distinct first-occurrence times; let t be the last to occur first. Before and including that occurrence, the sequence contains m^d calls producing distinct words of length d (all of $[m]^d$), each the end of its own chain of d calls, and these chains are disjoint. Hence $T^\pi(W_t) \geq d\, m^d$.

(b). A randomized policy is a mixture of deterministic ones, so it suffices to fix π deterministic. Let $J(t) \in \mathbb{N} \cup \{\infty\}$ be the number of calls producing a length-d word up to and including the first occurrence of t. The finite $J(t)$ are distinct positive integers, so $\sum_t J(t) \geq \sum_{j=1}^{m^d} j = m^d(m^d+1)/2$. Each of these $J(t)$ calls ends its own chain of d calls, so $T^\pi(W_t) \geq d\, J(t)$; average over t.

(c). On M^{ret} the preorder DFS with free pops produces each non-root word once at cost 1 (proof of Remark D.19); the terminal t occurs at its preorder index, at most the number $n_{m,d}$ of non-root words.

(d). Visit $t_1 < t_2 < \dots$ in lexicographic order, each as a chain of d calls started by a reset; $t = t_r$ is produced at call dr. The ratios follow from (a)–(c): $dm^d/n_{m,d} > dm^d(m-1)/m^{d+1} = d(m-1)/m$, and $d(m^d+1)/(2n_{m,d}) > d(m-1)/(2m)$; for $m \geq 2$, $(m-1)/m \geq 1/2$. □

Remark D.5 (Interpretation of the counting separation). The bounds compare uncapped verifier-call costs in the opaque marked-leaf model, in the worst case or under its uniform leaf prior. They do not imply the same ratio at a fixed finite horizon: both capped trajectory lengths may equal the budget before either procedure reaches a leaf. In particular no statement about the certificate $2\bar{\varepsilon}\,\mathrm{Len}$ of Remark D.14 follows (the earlier "the reset agent's certificate is worse by the same factor" is withdrawn).

The result does not certify either reference schedule under Remark D.19. Its greedy-policy and action-value-error assumptions must be established separately. Solved-case ratios on the public affine instances (where generators and target are exposed, so the opaque model does not apply) are descriptive quantities from a different model, not an experimental test of this theorem or its depth scaling (`analysis/prop_e1_exact.csv`: $d = 2$ mean 1.51, $d = 3$ 1.95, $d = 4$ 2.87 on solved worlds within $B = 50$). The factor is counted in verifier calls; it is a statistical cost in the oracle model, not a computational-hardness claim, and says nothing about small d or large budgets.

D.2 Reduction and utility as provability shift

Remark D.6 (Reduction to a finite deterministic MDP (Result A)). Assume (F-R) (faithful research-layer abstraction, Sonoda's Prop. 2.2 applied to M_R). Then the optimal discoverability is the indicator of a reachability event:

$$\mathrm{SD}^*_B = \mathbf{1}\{\exists\, z \text{ committable within } B \text{ verifier calls} :\ \mathrm{Cert}(z) = 1,\ U_b(z) \geq \gamma\} \in \{0, 1\},$$

and $\mathrm{SD}^*_B = \max_\pi \mathbb{P}^\pi_{x_0}(T_{\mathcal{G}_R} \leq B)$ is computed by breadth-first search over M_R (finite, deterministic, absorbing commit states). For any policy π, $\mathrm{SD}^\pi_B \leq \mathrm{SD}^*_B$.

Proof. Throughout, $\mathcal{G}_R$ and the committed states are absorbing, and by (F-R) V^*_t of M_R is the optimal discoverability of Definition 2.4, i.e. $\mathrm{SD}^*_B := V^*_B(x_0)$ (this is the definition used in Remark D.8).

Step 1 ($\{0,1\}$-valuedness). Since M_R is deterministic, a stationary or history-dependent deterministic policy from x follows a single path, so $\mathbb{P}^\pi_x(T_{\mathcal{G}_R} \leq t) \in \{0, 1\}$; a randomized policy gives a convex combination of such values, hence is dominated by the best path. Thus $V^*_t(x) = 1$ iff some action sequence from x reaches $\mathcal{G}_R$ within t steps, and $V^*_t(x) \in \{0, 1\}$.

Step 2 (recursion). $V^*_0(x) = \mathbf{1}\{x \in \mathcal{G}_R\}$. For $t \geq 1$: if $x \in \mathcal{G}_R$ then $V^*_t(x) = 1$ ($\mathcal{G}_R$ is absorbing); if $x \notin \mathcal{G}_R$, reaching $\mathcal{G}_R$ within t steps requires a first action a followed by reaching $\mathcal{G}_R$ from $F_R(x, a)$ within $t-1$ steps, so $V^*_t(x) = \max_a V^*_{t-1}(F_R(x, a))$. (Both cases agree with $V^*_t(x) = \max\{\mathbf{1}\{x \in \mathcal{G}_R\}, \max_a V^*_{t-1}(F_R(x, a))\}$, and $Q^*_t(x, a) = V^*_{t-1}(F_R(x, a))$.)

Step 3 (unrolling). By induction on t, $V^*_t(x) = 1$ iff $\mathcal{G}_R$ is reachable from x in at most t steps. For $x = x_0$, $t = B$: $\mathrm{SD}^*_B = 1$ iff there is a path of $\leq B$ verifier-call units ending in a commit state $\langle z \rangle$ with $r_\gamma(z) = 1$, i.e. $\mathrm{Cert}(z) = 1$ and $U_b(z) \geq \gamma$. This is the displayed indicator.

Step 4 (BFS). M_R is finite, so breadth-first search to depth B from x_0 enumerates the states reachable in $\leq B$ steps and checks membership in $\mathcal{G}_R$; hence SD^*_B is decidable.

Step 5 (upper bound). For any policy π, $\mathrm{SD}_B^{\pi} = \mathbb{P}_{x_0}^{\pi}(T_{\mathcal{G}_R} \le B) \le \max_{\pi'} \mathbb{P}_{x_0}^{\pi'}(T_{\mathcal{G}_R} \le B) = \mathrm{SD}_B^{*}$ by the definition of the maximum. □

Remark D.7 (Utility as provability shift (Result B)). Let $L_q(K)$ be the length (in verifier calls) of a shortest checker-accepted proof of q from library K, and assume the consumption layer is a faithful finite deterministic MDP for K_0 and $K_0 \oplus z$ (F-C(K)).

(i) For the optimal consumer, $p_b^*(K, q) = \mathbf{1}\{L_q(K) \le b\}$ and

$$U_b^*(z) = \mathbb{E}_{q\sim Q_{\mathrm{test}}}\big[\mathbf{1}\{L_q(K_0 \oplus z) \le b\} - \mathbf{1}\{L_q(K_0) \le b\}\big],$$

i.e. utility is the (signed) mass of test theorems whose shortest proof crosses the budget b.

(ii) Let $\hat{\pi}_C$ be a learned consumer satisfying Sonoda's Assumption 3.1 on $M_C(K_0)$ and $M_C(K_0 \oplus z)$, with unsolved-mass curves and errors $\varepsilon_t^C(K)$, and set $\mathrm{gap}_C(K) = 2\sum_{s=1}^{b} \mathbb{P}_{Q_{\mathrm{test}}}^{\hat{\pi}_C}(T \ge s; K)\, \varepsilon_{b-s+1}^C(K)$. Then, on the intersection of the two Sonoda events (probability $\ge 1 - \sum_{t\le b}(\delta_t^{K_0} + \delta_t^{K_0\oplus z})$; z fixed),

$$\big|U_b^{\hat{\pi}_C}(z) - U_b^*(z)\big| \;\le\; \mathrm{gap}_C(K_0 \oplus z) + \mathrm{gap}_C(K_0).$$

Proof. (i) For $K \in \{K_0, K_0 \oplus z\}$ the consumption MDP $M_C(K)$ is finite and deterministic (F-C(K)) with initial state q, horizon b (one unit per verifier call) and absorbing goal set $\mathcal{G}_C(K) = \{$states holding a checker-accepted proof of $q\}$. Remark D.6, applied to $M_C(K)$ with $(\mathcal{G}_R, B)$ replaced by $(\mathcal{G}_C(K), b)$, gives $p_b^*(K, q) = V_b^*(q; K) = \mathbf{1}\{\mathcal{G}_C(K)$ reachable in $\le b$ steps$\} = \mathbf{1}\{L_q(K) \le b\}$ (with $L_q = \infty$ if no proof exists). Definition 2.2 with the optimal consumer in place of the fixed solver T and linearity of $\mathbb{E}_{q\sim Q_{\mathrm{test}}}$ (a finite average) yield the displayed formula.

(ii) Fix z (not data-dependent; if z is selected using the consumer's data, add a union bound over the candidate set). Apply Sonoda Thm 3.1 once to $M_C(K_0)$ and once to $M_C(K_0 \oplus z)$, each with initial distribution Q_{test} and horizon b. With $\mathrm{SP}_b^{\pi}(K) = \mathbb{E}_{q\sim Q_{\mathrm{test}}} p_b^{\pi}(K, q)$, this gives on events E_{K_0}, $E_{K_0\oplus z}$ of probabilities $\ge 1 - \sum_{t\le b}\delta_t^{K_0}$ and $\ge 1 - \sum_{t\le b}\delta_t^{K_0\oplus z}$ respectively,

$$0 \le \mathrm{SP}_b^*(K) - \mathrm{SP}_b^{\hat{\pi}_C}(K) \le \mathrm{gap}_C(K), \qquad K \in \{K_0, K_0 \oplus z\},$$

the left inequality because SP_b^* is a maximum. By the union bound, $E_{K_0} \cap E_{K_0\oplus z}$ has probability $\ge 1 - \sum_{t\le b}(\delta_t^{K_0} + \delta_t^{K_0\oplus z})$. On it, with $a = \mathrm{SP}_b^*(K_0 \oplus z) - \mathrm{SP}_b^{\hat{\pi}_C}(K_0 \oplus z) \in [0, \mathrm{gap}_C(K_0 \oplus z)]$ and $c = \mathrm{SP}_b^*(K_0) - \mathrm{SP}_b^{\hat{\pi}_C}(K_0) \in [0, \mathrm{gap}_C(K_0)]$,

$$U_b^*(z) - U_b^{\hat{\pi}_C}(z) = a - c \in [-\mathrm{gap}_C(K_0),\, \mathrm{gap}_C(K_0 \oplus z)],$$

so $|U_b^{\hat{\pi}_C} - U_b^*| \le \max\{\mathrm{gap}_C(K_0), \mathrm{gap}_C(K_0 \oplus z)\} \le \mathrm{gap}_C(K_0 \oplus z) + \mathrm{gap}_C(K_0)$. □

D.3 Bellman certificate for discovery

Research-layer MDP. Let M_R be the finite deterministic MDP $(\mathcal{X}_R, \mathcal{A}_R, F_R)$ of the research layer (§3): a state records library, open candidates, memory and residual budget; actions are construction operators, backtracks, and $\mathsf{commit}(z)$. Horizon B counts one unit per verifier call (free operations are folded into macro-actions, see §D.4). $\mathsf{commit}(z)$ enters an absorbing state $\langle z\rangle$ with terminal reward

$$r_\gamma(z) = \mathbf{1}\{\mathrm{Cert}(z) = 1 \;\wedge\; U_b(z) \ge \gamma\},$$

where $U_b(z) = \mathbb{E}_{q\sim Q_{\mathrm{test}}}[p_b^T(K_0 \oplus z, q) - p_b^T(K_0, q)] \in [-1, 1]$ is the held-out utility of Definition 2.2. Let $\mathcal{G}_R = \{\langle z\rangle : r_\gamma(z) = 1\}$ and let $\mathcal{G}_R^{\mathrm{abs}}$ be all absorbing (committed) states. Set $V_t^*(x) = \max_\pi \mathbb{P}_x^\pi(T_{\mathcal{G}_R} \le t) \in \{0, 1\}$ (deterministic, so $V_t^*(x) = 1$ iff a rewarded commit is reachable in $\le t$ steps), $Q_t^*(x, a) = V_{t-1}^*(F_R(x, a))$, so that $\mathrm{SD}_B^* = V_B^*(x_0)$ is the optimal discoverability of Definition 2.4. Faithfulness of the state abstraction ((F-R), §4) is assumed as in Sonoda's Prop. 2.2; it is an assumption about the declared abstraction, not about the LLM's internal state. Let $\hat{\pi}$ be greedy for estimates $\hat{Q}_t$, T the hitting time of $\mathcal{G}_R^{\mathrm{abs}}$, and

$$\mathrm{Len}_B^{\hat{\pi}} = \sum_{s=1}^{B} \mathbb{P}^{\hat{\pi}}(T \ge s) = \mathbb{E}[T \wedge B], \qquad \mathrm{SD}_B^{\hat{\pi}} = \mathbb{P}^{\hat{\pi}}(T_{\mathcal{G}_R} \le B).$$

Assumption C (utility-estimated labels). Clauses 1,2,3,5 of Sonoda's Assumption 3.1 hold on the research layer ($\hat{Q}_t \in \mathcal{H}_t \subseteq \mathrm{Lip}_{L_H}(\mathcal{C}_t, [0,1])$, Q_t^* is L_Q-Lipschitz on $\mathcal{C}_t$, i.i.d. query pairs from q_t with local mass $q_t(B(z,r)) \geq c_t r^{d_t}$, minimax regression). Clause 4 (rollout labels with $\mathbb{E}[Y|x,a] = Q_t^*(x,a)$) is replaced by:

(4′) For each query pair (x,a), m_t conditionally independent rollouts from (x,a) commit artifacts $Z_1, \ldots, Z_{m_t}$, with the *true* labels $Y_j = r_\gamma(Z_j)$ satisfying $\mathbb{E}[Y_j \mid x,a] = Q_t^*(x,a)$ (this is Sonoda's clause 4 verbatim, i.e. the labelling rollout continues optimally). The *observed* label is $\hat{Y}_j = \mathbf{1}\{\mathrm{Cert}(Z_j) = 1 \wedge \hat{U}_j \geq \gamma\}$, where $\hat{U}_j = \frac{1}{N_u}\sum_{l=1}^{N_u} D_{j,l}$ averages N_u paired differences $D_{j,l} \in [-1,1]$ that are, given Z_j, independent with mean $U_b(Z_j)$ (fresh evaluation seeds; $N_u = n_u|Q_{\mathrm{test}}|$ when every test theorem is run n_u times).

Define the *utility-threshold margin*

$$\kappa_t(\eta) = \sup_{(x,a)\in\mathcal{C}_t} \mathbb{P}\big(\mathrm{Cert}(Z_{x,a}) = 1,\ |U_b(Z_{x,a}) - \gamma| \leq \eta\big), \qquad \beta_t = \inf_{\eta>0}\Big\{2e^{-N_u\eta^2/2} + \kappa_t(\eta)\Big\},$$

with $Z_{x,a}$ the artifact committed by one labelling rollout from (x,a). Write $\varepsilon_t^{\mathrm{S}}$ for Sonoda's Eq. (3) (approximation + coverage + rollout noise) and

$$\varepsilon_t^R = \varepsilon_t^{\mathrm{S}} + 2\beta_t.$$

Remark D.8 (Discoverability-gap bound (Result C)). Under Assumption C, with probability at least $1 - \sum_{t=1}^B \delta_t$,

$$\mathrm{SD}_B^* - \mathrm{SD}_B^{\hat{\pi}} \ \leq\ 2\sum_{s=1}^{B} \mathbb{P}^{\hat{\pi}}(T \geq s)\, \varepsilon_{B-s+1}^R.$$

If moreover the conditional occupancies $O_s^{\hat{\pi}} = \mathcal{L}(X_{s-1} \mid T \geq s)$ satisfy $O_s^{\hat{\pi}}(\Delta_t \leq u) \leq C_\Delta u^\beta$ for $t = B - s + 1$, then on the same event $\mathrm{SD}_B^* - \mathrm{SD}_B^{\hat{\pi}} \leq C_\Delta \sum_s \mathbb{P}^{\hat{\pi}}(T \geq s)(2\varepsilon_{B-s+1}^R)^{\beta+1}$. In particular if $\sup_t \varepsilon_t^R \leq \bar{\varepsilon}$ then $\mathrm{SD}_B^* - \mathrm{SD}_B^{\hat{\pi}} \leq 2\bar{\varepsilon}\,\mathrm{Len}_B^{\hat{\pi}}$.

Proof. Step 1 (label bias). Fix $(x,a) \in \mathcal{C}_t$ and a rollout artifact Z. If $\mathrm{Cert}(Z) = 0$ then $\hat{Y} = Y = 0$. Otherwise $\hat{Y} \neq Y$ requires $\hat{U}$ and $U_b(Z)$ to lie on different sides of γ, hence $|\hat{U} - U_b(Z)| > \eta$ or $|U_b(Z) - \gamma| \leq \eta$ for every $\eta > 0$. The N_u paired differences are independent with range 2, so Hoeffding gives $\mathbb{P}(|\hat{U} - U_b(Z)| > \eta \mid Z) \leq 2\exp(-2N_u^2\eta^2/(4N_u)) = 2e^{-N_u\eta^2/2}$. (Equivalently: apply Hoeffding to the shifted variables $(D+1)/2 \in [0,1]$ at threshold $\eta/2$; the shift $[-1,1] \to [0,1]$ is needed only here, because $Y, \hat{Y}, Q_t^*, \hat{Q}_t$ are already $[0,1]$-valued and Sonoda's Lipschitz class clause 2 concerns only these.) Hence $|\mathbb{E}[\hat{Y} \mid x,a] - Q_t^*(x,a)| \leq \mathbb{P}(\hat{Y} \neq Y) \leq 2e^{-N_u\eta^2/2} + \kappa_t(\eta)$; take the infimum over η: bias $\leq \beta_t$ uniformly on $\mathcal{C}_t$.

Step 2 (uniform regression error; Sonoda Thm A.2 with biased labels). Let $\bar{Y}_{t,i} = m_t^{-1}\sum_j \hat{Y}_{t,i,j}$. Conditionally on the query pairs, $\hat{Y}_{t,i,j} \in \{0,1\}$ are independent with mean $\mu_{t,i}$, $|\mu_{t,i} - Q_t^*(X_{t,i}, A_{t,i})| \leq \beta_t$ (Step 1). Hoeffding with $\alpha_t = \sqrt{\log(4N_t/\delta_t)/(2m_t)}$ and a union bound over $i \leq N_t$ give, with probability $\geq 1 - \delta_t/2$, $\max_i |\bar{Y}_{t,i} - \mu_{t,i}| \leq \alpha_t$, so $\max_i |\bar{Y}_{t,i} - Q_t^*(X_{t,i}, A_{t,i})| \leq \alpha_t + \beta_t$. The remaining argument of Thm A.2 is unchanged: the coverage event (Lemma A.2, probability $\geq 1 - \delta_t/2$) gives an η_t-net with $\eta_t = C_{\mathrm{cov}} c_t^{-1/d_t}(\log(4N_t/\delta_t)/N_t)^{1/d_t}$; minimax regression yields $\max_i |\hat{Q}_t - \bar{Y}_{t,i}| \leq \varepsilon_{\mathrm{app,t}} + \xi + \alpha_t + \beta_t$; the Lipschitz clauses transfer the bound from sample points to all of $\mathcal{C}_t$. Result: $\sup_{\mathcal{C}_t} |\hat{Q}_t - Q_t^*| \leq \varepsilon_t^{\mathrm{S}} + 2\beta_t = \varepsilon_t^R$ (the noise term of $\varepsilon_t^{\mathrm{S}}$ is $2\alpha_t$, and the bias enters the same two places).

Step 3 (one-step regret). Sonoda Lemma A.1 (App. A.4, purely algebraic): if $\sup_{\mathcal{C}_t} |\hat{Q}_t - Q_t^*| \leq \varepsilon_t$ then $\sup_a Q_t^*(x,a) - Q_t^*(x, \hat{\pi}_t(x)) \leq 2\varepsilon_t$.

Step 4 (telescoping; Sonoda Thm A.1). M_R is finite and deterministic and every committed state is absorbing with $V_t^* = V_t^{\hat{\pi}} = r_\gamma$ there, so $D_t(x) = V_t^*(x) - V_t^{\hat{\pi}}(x) = 0$ on $\mathcal{G}_R^{\mathrm{abs}}$. The candidate-retention clause of Assumption 3.1-1 gives $V_t^*(x) = \sup_{a\in\mathcal{A}_t(x)} Q_t^*(x,a)$ (Sonoda Prop. A.2), hence $D_t(x) \leq 2\varepsilon_t \mathbf{1}[x \notin \mathcal{G}_R^{\mathrm{abs}}] + D_{t-1}(F_R(x, \hat{\pi}_t(x)))$. Running $\hat{\pi}$ from x_0 and summing, $\mathrm{SD}_B^* - \mathrm{SD}_B^{\hat{\pi}} = D_B(x_0) \leq 2\sum_s \mathbb{P}^{\hat{\pi}}(T \geq s)\varepsilon_{B-s+1}$. Intersecting the events of Step 2 over $t \leq B$ costs $\sum_t \delta_t$.

Step 5 (margin form; Sonoda Thm A.3). The per-step regret is $r_s(x) \le 2\varepsilon_t \mathbf{1}[\Delta_t(x) \le 2\varepsilon_t]$, so $\mathbb{E}[r_s \mathbf{1}[T \ge s]] \le \mathbb{P}(T \ge s)\, 2\varepsilon_t\, C_\Delta (2\varepsilon_t)^\beta$. (Sonoda's Thm A.3 states the constant as $2C_\Delta$; the line above already gives C_Δ, and we use the tighter constant.) □

Remark D.9 (What is new, and what is not). Steps 3–4 are Sonoda's argument unchanged; the only new ingredient is the bias term $2\beta_t$ of Step 1–2. Two consequences. (i) $\beta_t \ge \inf_\eta \kappa_t(\eta) = \kappa_t(0^+)$ does not decrease with N_u: if the utility of committed artifacts has mass at γ, no amount of evaluation removes it. (ii) Rates: if $U_b(Z_{x,a})$ has density $\le \rho_t$ near γ then $\kappa_t(\eta) \le 2\rho_t\eta$ and, with $\eta = \sqrt{2\ln N_u/N_u}$, $\beta_t \le 2/N_u + 2\rho_t\sqrt{2\ln N_u/N_u}$, i.e. polynomial in N_u; only if U_b has a gap η_0 around γ ($\kappa_t(\eta_0) = 0$) do we get the exponential $\beta_t \le 2e^{-N_u\eta_0^2/2}$. Statistical difficulty of discovery is therefore governed by the density of artifact utilities *near the threshold* γ, which is a property of the environment and of where γ is placed, not of the learner.

Remark D.10 (Scope of Thms A–C and retractions). (i) *Oracles.* Result A's "BFS computes SD^*_B" presupposes oracle access to $\mathrm{Cert}(z)$ and to the population utility $U_b(z)$; it is a statement about the definition of optimal discoverability on the declared abstraction, not an executable procedure for a real research run. (ii) *Optimal rollout.* Clause (4′) of Assumption C takes the labelling rollouts to continue *optimally* ($\mathbb{E}[Y \mid x, a] = Q^*_t$); no implementable learner is claimed to satisfy this, so Result C is a conditional statement about a hypothetical fitted-Q procedure with optimal-continuation labels. (iii) *Sealed set.* The N_u paired differences defining $\hat{Y}$ are drawn from the evaluation split; if the same split is later used to report the utility of the committed artifact, the reported value is selected on its own noise and the guarantee of Result C does not transfer to it. The certificate applies to the label used for training $\hat{Q}$, and any final report requires a disjoint, previously untouched split. (iv) *Retractions.* β_t is a property of the environment and of γ (mass of artifact utilities near the threshold). It is not estimable from a single run's traces without a margin assumption, so we *withdraw* any reading of the bias-floor $\beta_t \ge \kappa_t(0^+)$ or of an empirically estimated margin as a measurable, trace-derived guarantee. Result C bounds the gap by unobservable quantities; it is not an executable discovery certificate.

Remark D.11 (Two-point bound for terminal-label means). Fix a rollout distribution at one query pair. Let θ_+ and θ_η be its true terminal-label means in two evaluation environments, with $\theta_+ - \theta_\eta = \kappa > 0$. For fixed finite evaluation and rollout counts, let P_+ and P_η be the laws of all observations available to an estimator. Assume

$$\mathrm{TV}(P_+, P_\eta) \longrightarrow 0 \quad \text{as } \eta \downarrow 0.$$

For every estimator A taking values in $[0, 1]$, define $e_+ = |\mathbb{E}_+ A - \theta_+|$ and $e_\eta = |\mathbb{E}_\eta A - \theta_\eta|$. Then

$$\liminf_{\eta \downarrow 0} \max(e_+, e_\eta) \ge \kappa/2.$$

The same conclusion holds for expected absolute estimation error.

Proof. Boundedness of A gives

$$|\mathbb{E}_+ A - \mathbb{E}_\eta A| \le \mathrm{TV}(P_+, P_\eta).$$

The triangle inequality yields

$$e_+ + e_\eta \ge \kappa - \mathrm{TV}(P_+, P_\eta).$$

Take the larger error and then the lower limit. Expected absolute error is at least absolute bias. □

Remark D.12 (Not an optimal-action-value lower bound). These parameters are means under a fixed rollout distribution, not optimal reachability values of the deterministic research MDP, which are $\{0, 1\}$-valued (Result A). Moving a threshold atom need not preserve the optimal-continuation condition of Assumption C, and no bias bound for a fitted Q follows. The proposition is a non-uniform estimation statement over nearby environments, not an irreducible-error statement for one fixed environment or a lower bound on the Bellman discovery gap. The observation-law continuity hypothesis holds, for example, when a label thresholds the mean of N_u fresh paired differences whose installed mean is $p_0 + u$ against a common baseline mean p_0: the label law is a polynomial in u, so the two laws merge as $\eta \downarrow 0$ (in the earlier version this was stated for $\{\pm 1\}$ differences, which is incompatible with a common baseline and is withdrawn).

Remark D.13 (Limits). (a) Clause (4′) assumes the labelling rollouts continue optimally; for an LLM this is an idealization, as is (F-R). (b) κ_t is unobservable in general; it can be estimated from the empirical distribution of $\hat{U}_b$ over committed artifacts (this needs the E2 trace fields listed in the protocol). (c) We use Result C only as a certificate: it bounds a gap in terms of quantities of a prover's trace and does not say that any prover attains it.

Remark D.14 (Trajectory-length form of Result C (Result D)). Under Assumption C, on the event of Remark D.8, if $\sup_t \varepsilon_t^R \le \bar{\varepsilon}$ then

$$\mathrm{SD}_B^* - \mathrm{SD}_B^{\hat{\pi}} \;\le\; 2\bar{\varepsilon}\,\mathrm{Len}_B^{\hat{\pi}}, \qquad \mathrm{Len}_B^{\hat{\pi}} = \mathbb{E}[T \wedge B] \le B.$$

Hence for two research policies with the same valid $\bar{\varepsilon}$ the certificate is smaller for the one with the shorter expected trajectory; trivially the certificate is $\le 2\bar{\varepsilon}B$.

Proof. By Remark D.8, on its event, $\mathrm{SD}_B^* - \mathrm{SD}_B^{\hat{\pi}} \le 2\sum_{s=1}^B \mathbb{P}^{\hat{\pi}}(T \ge s)\,\varepsilon_{B-s+1}^R$. Substituting $\varepsilon_{B-s+1}^R \le \sup_t \varepsilon_t^R \le \bar{\varepsilon}$ (all terms are nonnegative) gives $\le 2\bar{\varepsilon}\sum_{s=1}^B \mathbb{P}^{\hat{\pi}}(T \ge s)$. Since $T \wedge B = \sum_{s=1}^B \mathbf{1}\{T \ge s\}$ for integer-valued $T \ge 0$, taking expectations gives $\sum_{s=1}^B \mathbb{P}(T \ge s) = \mathbb{E}[T \wedge B] = \mathrm{Len}_B^{\hat{\pi}} \le B$. For the comparison: if $\bar{\varepsilon}$ is a valid uniform bound for both policies, the two certificates are $2\bar{\varepsilon}\mathrm{Len}_B^{\hat{\pi}_1}$ and $2\bar{\varepsilon}\mathrm{Len}_B^{\hat{\pi}_2}$, so their order is the order of Len; this is the exact-α statement of Remark D.21(a) with $\bar{\varepsilon} = 2\alpha$. Since these are upper bounds, a smaller certificate does not by itself imply a smaller realized gap. □

Remark D.15 (Two-action evaluation-oracle lower bound (Result C$^{\downarrow}$)). Consider an auxiliary one-step experiment with two certified artifacts and an unknown environment sign. All public information is identical under the two signs. Selecting an action commits its artifact within that step.

Fix $0 < \gamma \le 1/4$ and $0 < \eta_0 \le 1/4$. The baseline solve indicator has mean $p_0 = 1/2 - \gamma$ for both artifacts. Under the positive sign, the installed indicators for actions one and two have means $1/2 + \eta_0$ and $1/2 - \eta_0$, respectively; under the negative sign these means are exchanged. Their utilities (installed minus the common baseline mean) are therefore $\gamma + \eta_0$ and $\gamma - \eta_0$.

Each paired evaluation draws independent baseline and installed indicators and returns their difference. All evaluation pairs are fresh. A rollout label thresholds the mean of N_u differences at γ. After n labels, allowing adaptive action queries, a learner selects one artifact using only its queries, observed labels, and private randomness. If R_+ and R_- are its expected discovery gaps under the two signs, then

$$\max(R_+, R_-) \ge \frac{1}{2}\left(1 - 2\eta_0\sqrt{3nN_u}\right).$$

This is an evaluation-oracle lower bound, not an assertion that the experimental solver satisfies Assumption C.

Proof. The baseline distribution is identical under the two signs. The installed Bernoulli means differ by $2\eta_0$ and lie in $[1/4, 3/4]$, so their relative entropy is at most $24\eta_0^2$. Data processing bounds the information in one threshold label by $24N_u\eta_0^2$. The chain rule gives the bound $24nN_u\eta_0^2$ for the complete observation law, including adaptive queries generated by the same conditional query rule under both signs. Pinsker's inequality bounds total variation by $2\eta_0\sqrt{3nN_u}$. Selecting the unsuccessful artifact is a testing error and incurs gap one. The two-point testing inequality proves the result. □

Remark D.16 (Scope). The construction shares one baseline distribution across artifacts, as required by the utility definition: each artifact's utility is the mean of its own installed-minus-baseline difference, so the two artifacts have different utilities although the baseline is common (the earlier $\{\pm 1\}$-difference instance, which forced equal utilities, is withdrawn). The bound concerns total evaluation information. It does not establish sharp dependence on N_u alone, a matching coverage rate, or non-vacuity of the complete Bellman certificate of Remark D.8.

D.4 Exact certificate quantities on Family B

Sonoda et al. name two extensions: estimating the bound's quantities from traces, and analysing policies that backtrack (Sonoda et al., 2026b). On the finite Family B instance both can be done exactly.

Two finite MDPs with the horizon in verifier calls. Fix a Family B world (alphabet $[m]$, depth cap d, terminal set $\mathcal{T} \subseteq [m]^{\le d}$) and budget B. Rules (E),(B),(H) of Appendix J charge one unit only to (E), whereas Sonoda's identity $Q_t^*(x,a) = V_{t-1}^*(F(x,a))$ needs one horizon unit per step. We therefore fold free operations into macro-actions: M^{ret} has state (R, b) (stack of frames (w_k, J_k), residual budget b) and macro-action (k, i) = pop to frame k, then extend$_i$ with $i \notin J_k$, $|w_k| < d$; M^{flat} has state (w, b) (no retained frames) and macro-action (r, i) = [reset to the root if $r = 1$], then extend by i. Both are finite and deterministic; halt is dropped (it never raises V^*), and a state is absorbing-failure when $b = 0$ or no extend is legal. For a word u let $\mathrm{dist}(u) = \min\{|t| - |u| : t \in \mathcal{T},\ u \preceq t\}$ (∞ if none) and $L^\star = \mathrm{dist}(\epsilon)$.

Remark D.17 (Closed-form V^* and Q^*). $V_b^{*,\mathrm{ret}}(R) = 1$ iff $\min_{k,\, i \notin J_k,\, |w_k| < d} \big(1 + \mathrm{dist}(w_k i)\big) \le b$; $V_b^{*,\mathrm{flat}}(w) = 1$ iff $\min(\mathrm{dist}(w), L^\star) \le b$. $Q_b^*(x,a) = 1$ iff the child word is terminal or $V_{b-1}^*(F(x,a)) = 1$.

Proof. A terminal is reachable from a stack state exactly by choosing a frame k and an untried i and then descending along a shortest terminal continuation of $w_k i$; every extend costs one unit and pops are free. Frames above k are lost, but they are never needed since $w_k i$ is a new frame whose descendants have fresh tried-sets. A tried $i \in J_k$ cannot be re-extended (rule (E)), hence the restriction. In M^{flat} the only alternatives to continuing from w are reset-and-extend, at cost $L^\star$. The last statement is the definition $Q_b^* = V_{b-1}^* \circ F$ together with terminal absorption. □

Trace-local coverage and the finite-support coverage lemma. For a deterministic reference executor $X \in \{\mathrm{ret}, \mathrm{flat}\}$ with trajectory $x_1, x_2, \ldots$ set $\mathcal{C}_t^X = \{(x_s, a) : a \text{ legal at } x_s\}$, $s = B - t + 1$, and $n_s^X = |\mathcal{C}_t^X|$. This is the trace-local reading of Assumption 3.1-1 in Sonoda's discussion (the region occupied by available traces); we make no claim outside it.

Remark D.18 (Finite-support coverage). Let $\mathcal{C}_t$ be finite with $|\mathcal{C}_t| = n_t$, q_t uniform on $\mathcal{C}_t$, $\mathcal{H}_t$ all functions $\mathcal{C}_t \to [0,1]$, discrete metric. If $N_t \ge n_t \ln(2n_t/\delta_t)$, then with probability $\ge 1 - \delta_t$ (label noise included), $\sup_{\mathcal{C}_t} |\hat{Q}_t - Q_t^*| \le 2\alpha_t + 2\beta_t$, i.e. Sonoda's approximation and coverage terms vanish: $\varepsilon_t = 2(\alpha_t + \beta_t)$, $\alpha_t = \sqrt{\ln(4N_t/\delta_t)/(2m_t)}$.

Proof. Every function into $[0,1]$ is 1-Lipschitz for the discrete metric, and $Q_t^* \in \mathcal{H}_t$ so $\varepsilon_{\mathrm{app}} = 0$. A fixed point is missed by all N_t i.i.d. samples with probability $(1 - 1/n_t)^{N_t} \le e^{-N_t/n_t}$, so by a union bound all points are sampled with probability $\ge 1 - n_t e^{-N_t/n_t} \ge 1 - \delta_t/2$. Then $\sup_{\mathcal{C}_t}$ equals a max over sample points and Steps 1–2 of Result C give the claim (no net argument, η_t-term $= 0$); the noise event costs $\delta_t/2$. □

Remark D.19 (Conditional certificate for the greedy retained-state policy (Result E)). For $X \in \{\mathrm{ret}, \mathrm{flat}\}$ let $\mathcal{C}_t^X = \{(x,a) : x \text{ reachable in exactly } B - t \text{ macro-steps from the root of } M^X,\ a \text{ legal at } x\}$ be the *full reachable coverage set* (finite; it contains every state-action pair that *any* policy, in particular the learned one, can visit with t steps left), $n_t^{X,\mathrm{reach}} = |\mathcal{C}_t^X|$. Let $\hat{\pi}^X$ be greedy for the learned $\hat{Q}_t^X$ fitted on $\mathcal{C}_t^X$ with q_t uniform on $\mathcal{C}_t^X$ (Remark D.18; labels are certificate-only, so $\beta_t = 0$), and assume $N_t \ge n_t^{X,\mathrm{reach}} \ln(2n_t^{X,\mathrm{reach}}/\delta_t)$. Then on an event of probability $\ge 1 - \sum_t \delta_t$, for the *same* policy $\hat{\pi}^X$, whose occupancy $\mathbb{P}^{\hat{\pi}^X}(T \ge s)$ is the one in the bound and the one whose coverage was paid for,

$$\mathrm{SP}_B^* - \mathrm{SP}_B^{\hat{\pi}^X} \ \le\ 4 \sum_{s=1}^{B} \mathbb{P}^{\hat{\pi}^X}(T \ge s)\, \alpha_{B-s+1}^X \ \le\ 4\,\mathrm{Len}_B^{\hat{\pi}^X} \max_t \alpha_t^X \ \le\ 4B \max_t \alpha_t^X.$$

$\mathrm{Len}_B^{\hat{\pi}^X}$ is a random (data-dependent) quantity, computed exactly *after* fitting since M^X is finite; a priori only the last bound ($\mathrm{Len} \le B$) is available. The theorem is silent about any fixed executor (e.g. the preorder DFS of `family_b_executor.py`) unless that executor coincides with $\hat{\pi}^X$.

Proof. Both MDPs are finite and deterministic with absorbing success/failure, and $\mathcal{C}_t^X$ contains every action available at any state the policy can occupy with t steps left, so clause 1 (candidate retention) and trajectory containment hold for *every* greedy policy, including $\hat{\pi}^X$, without conditioning on which states it visits. Remark D.18 on $\mathcal{C}_t^X$ gives $\varepsilon_t = 2\alpha_t$; Result C (Steps 3–4, $\beta_t = 0$) applied to $\hat{\pi}^X$ yields the first inequality; the others use $\mathbb{P}(T \ge s) \le 1$ and $\sum_s \mathbb{P}(T \ge s) = \mathrm{Len} \le B$. □

Remark D.20 (What changed and why). An earlier version evaluated the bound at the occupancy and trace-local coverage set $\{(x_s, a)\}$ of a *fixed* preorder schedule, which is not $\hat{Q}$-greedy; the guarantee then concerned a policy other than the one whose occupancy entered the bound. Trace-local coverage is not a valid input for a learned greedy policy, because the states it visits depend on the fitted $\hat{Q}$. We now pay for the full reachable set. Upper bounds: $n_t^{\text{flat,reach}} \le 2m \sum_{k \le d} m^k$ (states are words, macro-actions (r, i)), and $n_t^{\text{ret,reach}} \le m(d{+}1) \cdot |\{\text{stack states}\}|$, exponentially larger since a state carries the tried-sets J_k of all frames. The comparison of Remark D.21 is therefore about certificate cost, and retention is *not* cheaper to certify in general.

Reference schedules (descriptive functionals, not certified policies). $\pi_{\text{ref}}^{\text{ret}}$: least-untried-generator preorder DFS with free pops (the policy of `family_b_executor.py` without duplicate suppression); $\pi_{\text{ref}}^{\text{flat}}$: the same preorder with retention ablated. Their trajectories are deterministic, so $\text{Len}_{\text{ref}}^X = \min(T_{\text{ref}}^X, B)$ with $\text{Len}_{\text{ref}}^{\text{ret}} \le \text{Len}_{\text{ref}}^{\text{flat}}$ pointwise: the ret schedule visits each non-root word of the preorder once at cost 1, the flat one visits the same words but pays 1 only for a first child and $|w'| \ge 1$ otherwise (reset and rebuild). Their trace-local counts n_s^X ($\le 2m$ for flat; stack-dependent for ret) give the descriptive quantity $\Phi^X = \sum_s \mathbf{1}[s \le T_{\text{ref}}^X] \alpha_{B-s+1}^X$. Φ^X **is not a certificate for any learned policy**; it measures the executor cost/coverage functional of the reference schedule (as a functional of occupancy, cf. Sonoda's Cor. 3.1).

Remark D.21 (Certificate cost of retention vs. flat execution (Result E.1)). Work under Remark D.19 with $\varepsilon_t^X = 2\alpha_t^X$ and the full reachable coverage sets, and write $\text{Len}^X := \text{Len}_B^{\hat{\pi}^X}$ (random; fixed once $\hat{Q}^X$ is fitted). On the event of Remark D.19:

(a) *Common* (N, m). If $N \ge \max_{X,t} n_t^{X,\text{reach}} \ln(2n_t^{X,\text{reach}} B/\delta)$ then $\alpha^{\text{ret}} = \alpha^{\text{flat}}$, and the ret bound is smaller than the flat one iff $\text{Len}^{\text{ret}} < \text{Len}^{\text{flat}}$ (occupancies of the two *greedy* policies, not of reference schedules).

(b) *A priori certificate sample cost* for slack ρ ($\text{SP}_B^* - \text{SP}_B^{\hat{\pi}^X} \le \rho$). Since $\text{Len}^X \le B$ is the only a priori bound: $N_t^X = \lceil n_t^{X,\text{reach}} \ln(2n_t^{X,\text{reach}} B/\delta) \rceil$, $m^X = \lceil 8B^2 \max_t \ln(4N_t^X B/\delta)/\rho^2 \rceil$, $\Xi^X = m^X \sum_{t \le B} N_t^X$ (labelled rollouts, $\delta_t = \delta/B$); $4B\alpha \le \rho$ iff $m \ge m^X$. The length factor cancels, so retention is cheaper to certify a priori iff $\Xi^{\text{ret}} < \Xi^{\text{flat}}$, which is governed by the reachable-set sizes $n^{X,\text{reach}}$; by the bounds in the Remark above retention is in general the *more* expensive one.

(c) *A posteriori (data-dependent) cost.* Once $\hat{\pi}^X$ is fitted and Len^X computed, the certificate $4\text{Len}^X \alpha \le \rho$ needs $m \ge 8(\text{Len}^X)^2 \ln(4N/\delta_t)/\rho^2$; using it requires Len^X chosen independently of the labels used for the bound (e.g. a fresh split), otherwise use (b).

The success event is that of Remark D.19 (finite coverage and label noise), the ceiling is $\text{Len}^X \le B$, and the statement concerns certificates, not which executor performs better.

Proof. (a) Equal (N, m) make α_t^X independent of X, so the bounds are $4\alpha\text{Len}^X$. (b) $4B\alpha \le \rho$ with $\alpha = \sqrt{\ln(4N/\delta_t)/(2m)}$ is equivalent to $m \ge 8B^2 \ln(4N/\delta_t)/\rho^2$; take the worst t. (c) is the same algebra with Len^X in place of B; the independence caveat is needed because Len^X depends on $\hat{Q}^X$. □

Remark D.22 (Descriptive functional of the reference schedules). For the deterministic reference schedules, $\Phi^X = \sum_{s \le \text{Len}_{\text{ref}}^X} \alpha_{B-s+1}^X$ with trace-local counts $N_s^X = \lceil n_s^X \ln(2n_s^X B/\delta) \rceil$, $m^X = \lceil 8(\text{Len}_{\text{ref}}^X)^2 \max_s \ln(4N_s^X B/\delta)/\rho^2 \rceil$, $\Xi_{\text{ref}}^X = m^X \sum_{s \le \text{Len}_{\text{ref}}^X} N_s^X$ is the quantity tabulated in Table 3. It compares the executors' *cost/coverage profiles* along their fixed traces and **does not certify the sample size for any learned policy**.

Exact computation on the 40 Family B worlds. `analysis/prop_e1_exact.py` (CPU-only) computes dist, $\mathcal{T}$, both reference trajectories, n_s^X, the optimal-action counts (hence Δ_t-ties) and Ξ^X for the 10 worlds × depth $d \in \{1, .., 4\}$ at $B = 50$, $\delta = 0.05$, $\rho = 0.5$ from the enumeration in `experiments/e1_enum/`; per-world output is `analysis/prop_e1_exact.csv`. (Our

d	solv.	$L^\star$	$\mathrm{Len}^{\mathrm{ret}}$	$\mathrm{Len}^{\mathrm{flat}}$	succ. ret/flat	$\bar{n}^{\mathrm{ret}}$	$\bar{n}^{\mathrm{flat}}$	$\log_{10}\Xi^{\mathrm{ret}}$	$\log_{10}\Xi^{\mathrm{flat}}$	ret<flat (cost)
1	0/10	–	4.2	4.2	0/0	2.6	8.4	5.77	6.35	–
2	10/10	2	13.2	20.8	10/10	4.9	8.4	7.33	8.13	10/10
3	10/10	2	29.4	37.2	7/5	7.0	8.4	8.51	9.09	10/10
4	10/10	2	43.5	49.3	5/1	9.6	8.4	9.43	9.59	5/10

Table 3: **Retention shortens the unsolved-mass curve but pays for it in coverage; the certificate cost is lower for retention only when it actually solves within B.** Means over 10 worlds per depth; $d = 1$ worlds have no terminal ($\mathrm{SP}^* = 0$, comparison void). $\bar{n}$: mean legal macro-actions along the trace; Ξ: total labelled rollouts Ξ_{ref} of the reference schedules at slack $\rho = 0.5$ (descriptive; Remark after Remark D.21); last column: worlds with $\Xi^{\mathrm{ret}} < \Xi^{\mathrm{flat}}$. Source: `analysis/prop_e1_exact.csv`.

$\mathrm{Len}^{\mathrm{ret}}$ equals the CSV `rollback_cost` minus one on all 22 solved cases where $\mathrm{Len}^{\mathrm{ret}} < B$: the latter also counts the root check.)

Reading of Table 3. Of the 30 solvable worlds, retention has the shorter curve in 22 ($\mathrm{Len}^{\mathrm{ret}} < \mathrm{Len}^{\mathrm{flat}}$; the same 22 where it solves within $B = 50$) and the lower certificate cost in 25. At $d = 4$ the cost is lower in only 5 of 10 worlds: in the other 5 both executors exhaust B, $\mathrm{Len}^X = B$, the length factor cancels and retention's larger n makes its coverage requirement dearer. In 99.0% (ret) and 99.8% (flat) of trace steps the optimum is not unique ($\Delta_t = 0$), because $B = 50 \gg L^\star = 2$ keeps $V^* = 1$ from almost every state; hence the margin condition of Result C holds only with $\beta = 0, C_\Delta = 1$ and gives no fast-rate improvement here. The certificate is far from tight on E1 (any $\hat{Q}$-greedy prover would incur regret ≈ 0 at this budget); we report exact descriptive functionals of the reference schedules (Table 3 is *not* a certificate for any learned policy), not performance.

Remark D.23 (Null-utility interpretation (Result F)). Let $\hat{U} = U_b^{\hat{\pi}_C}(z)$ be the utility measured with a learned consumer and $g = \mathrm{gap}_C(K_0 \oplus z) + \mathrm{gap}_C(K_0)$ as in Remark D.7(ii). On the event of that theorem:

(a) $U_b^*(z) \in [\hat{U} - g,\ \hat{U} + g]$. In particular an observed null or negative $\hat{U} \le 0$ is compatible with $U_b^*(z) \ge \gamma$ whenever $g \ge \gamma - \hat{U}$; conversely $\hat{U} < \gamma - g$ certifies $U_b^*(z) < \gamma$.

(b) For consumers $j = 1, \dots, J$ (different provers, same budget b) with gaps g_j, $U^* \in \bigcap_j [\hat{U}_j - g_j, \hat{U}_j + g_j]$; so $\max_j(\hat{U}_j - g_j) \le U^* \le \min_j(\hat{U}_j + g_j)$.

Proof. (a) By Remark D.7(ii), on its event $|\hat{U} - U_b^*(z)| \le g$, i.e. $\hat{U} - g \le U_b^*(z) \le \hat{U} + g$. If $\hat{U} \le 0$ and $g \ge \gamma - \hat{U}$ then $\hat{U} + g \ge \gamma$, so the interval does not exclude $U_b^*(z) \ge \gamma$. If $\hat{U} < \gamma - g$ then $U_b^*(z) \le \hat{U} + g < \gamma$. (b) For each j the interval statement of (a) holds on an event E_j (Remark D.7(ii) with consumer j), so on $\bigcap_j E_j$, whose probability is $\ge 1 - \sum_j \mathbb{P}(E_j^c)$ by the union bound, $U_b^*(z)$ lies in every interval, hence in their intersection $[\max_j(\hat{U}_j - g_j), \min_j(\hat{U}_j + g_j)]$. ($U_b^*$ depends on b, so the intersection is over consumers evaluated at the same b.) □

Remark D.24 (Finite evaluation error and common optimal consumer (Result F′)). Let $\hat{U}^{\mathrm{emp}}$ be the empirical mean of n paired differences $D_l \in [-1, 1]$ that, given z and the consumer, are independent with mean $U_b^{\hat{\pi}_C}(z)$ (same seeds for K_0 and $K_0 \oplus z$, fresh split disjoint from any split used to select z). With $\epsilon_{\mathrm{ev}}(\delta') = \sqrt{2\ln(2/\delta')/n}$, on the event of Remark D.7(ii) intersected with the Hoeffding event of probability $\ge 1 - \delta'$ (total failure $\le \sum_{t \le b}(\delta_t^{K_0} + \delta_t^{K_0 \oplus z}) + \delta'$):

$$\left|\hat{U}^{\mathrm{emp}} - U_b^*(z)\right| \le g + \epsilon_{\mathrm{ev}}(\delta'), \qquad \text{so Remark D.23(a),(b) hold with } \hat{U} \to \hat{U}^{\mathrm{emp}},\ g \to g + \epsilon_{\mathrm{ev}}.$$

If a single consumer π_c is optimal for both $M_C(K_0)$ and $M_C(K_0 \oplus z)$ (a *common optimal consumer*), then $\mathrm{gap}_C(K) = 0$ for $\hat{\pi}_C = \pi_c$, hence $U_b^{\pi_c}(z) = U_b^*(z)$ and only ϵ_{ev} remains. Otherwise U_b^* is defined with a library-specific optimum (Remark D.7(i)) and g cannot be removed by evaluation alone.

Proof. $|D_l| \le 1$ gives, by Hoeffding for variables of range 2, $\mathbb{P}(|\hat{U}^{\mathrm{emp}} - U_b^{\hat{\pi}_C}| > t) \le 2e^{-nt^2/2}$; set $t = \epsilon_{\mathrm{ev}}(\delta')$. Add $|U_b^{\hat{\pi}_C} - U_b^*| \le g$ (Remark D.7(ii)) and the union bound. For a common optimal consumer $p_b^{\pi_c} = p_b^*$ on both libraries, so $U_b^{\pi_c} = U_b^*$. □

Remark D.25 (Unified notation). Population vs. estimate: $\mathrm{SD}_B^* = V_B^*(x_0)$ and $\mathrm{SD}_B^{\hat{\pi}} = \mathbb{P}^{\hat{\pi}}(T_{\mathcal{G}_R} \le B)$ are population, *path*-level quantities (a path is one committed trajectory); $\hat{U}^{\mathrm{emp}}$, $\hat{Y}$, $\hat{Q}$ are estimates computed from traces. $\mathrm{gap}_C(K)$ and ε_t are *bounds on population differences* (from Assumption 3.1 and Lemma A.1), never empirical differences; TV and any occupancy-based quantity refer to the *information* carried by the consumer's traces, whereas $g, \epsilon_{\mathrm{ev}}$ concern the consumer's computational error and finite evaluation. Statements about the verifier's information (e.g. ceilings) do not constrain agent-side learning, and vice versa.

Remark D.26 (Falsifiable prediction (empirical; not a consequence of Remark D.23)). if the consumer is strengthened ($g \downarrow$), $\hat{U}$ moves toward $U_b^*(z)$; thus $U_b^*(z) \ge \gamma$ predicts $\hat{U}$ approaching γ from below, whereas $U_b^*(z) < \gamma$ predicts $\hat{U}$ never reaching γ under any stronger consumer. The bound gives an interval of width $2g$ around $\hat{U}$ but does not imply monotonicity of $\hat{U}$ in the consumer's strength; the frozen E2 bundle (§E.1) tests it by re-evaluation at $b \in \{4096, 8192, 16384\}$ with two consumers.

D.5 A CONDITIONAL CEILING ON CONSUMER UTILIZATION

For a fixed consumer T and held-out query law Q_{test}, write $S_T(K, h) := \mathbb{E}_{q \sim Q_{\mathrm{test}}} p_h^T(K, q)$ for success by h verifier calls. This is a population quantity, not an empirical estimate. The following assumptions make the relation between library contents, prompt length, and calls explicit.

- (U1) Installing K in the initial prompt consumes exactly $a(K) := c\ \mathrm{tok}(K)$ call-equivalent units, with $0 \le a(K) \le b$. The prompt fits the finite context window without truncation, and prompting sets only the initial consumer state; all other effects of its contents are represented by $S_T(K, h)$. The empty library K_0 has $a(K_0) = 0$.
- (U2) Success is recorded upon the first checker-accepted proof and is absorbing. Thus $S_T(K_0, h)$ is nondecreasing in h, and its limit as $h \to \infty$ exists. For a fixed policy this requires horizon-consistent execution; for the optimal consumer it follows from the reachability characterization in Remark D.7(i).
- (U3) Where the infinite-horizon version below is used, the *matched-budget content gain* is bounded by its limiting gain: $S_T(K, h) - S_T(K_0, h) \le S_T(K, \infty) - S_T(K_0, \infty)$ at $h = b - a(K)$. This is an additional assumption, not a consequence of Remark D.7 or D.8.

Remark D.27 (Consumer-utilization ceiling (Result III)). Under (U1)–(U2), set $h = b - a(K)$, $U_T(K, b) := S_T(K, h)$ and $U_T(K_0, b) := S_T(K_0, b)$. Define

$$\eta_T(K; h) := S_T(K, h) - S_T(K_0, h), \qquad \beta_T(t) := S_T(K_0, b) - S_T(K_0, b - ct) \ge 0.$$

Then the exact content–burden decomposition and its unconditional ceiling are

$$U_T(K, b) - U_T(K_0, b) = \eta_T(K; h) - \beta_T(\mathrm{tok}(K)) \le \sup_{0 \le j \le b} \eta_T(K; j) - \beta_T(\mathrm{tok}(K)). \quad (2)$$

If (U3) also holds, the requested infinite-horizon form is valid:

$$U_T(K, b) - U_T(K_0, b) \le \eta_T(K) - \beta_T(\mathrm{tok}(K)), \qquad \eta_T(K) := U_T(K, \infty) - U_T(K_0, \infty) = S_T(K, \infty) - S_T(K_0, \infty). \quad (3)$$

Proof. Add and subtract $S_T(K_0, h)$ from $S_T(K, h) - S_T(K_0, b)$ to obtain the equality in (2); the supremum bounds its first term. By (U2), $S_T(K_0, h) \le S_T(K_0, b)$, so $\beta_T \ge 0$. Under (U3), replace $\eta_T(K; h)$ by $\eta_T(K)$ to get (3). Remark D.7(i) supplies the absorbing reachability interpretation for an optimal consumer; Remark D.8 instead bounds a discovery-policy gap and does not supply (U3) or budget monotonicity for an arbitrary fixed T. □

Remark D.28 (Why the extra assumption and what the measurements say). The infinite-horizon inequality is false without (U3): for one query with $L_q(K_0) = 3$, $L_q(K) = 1$, $b = 2$, and $a(K) = 1$, Remark D.7(i) gives finite-budget gain 1 but $\eta_T(K) = 0$ because both libraries eventually prove q. Thus a useful lemma that merely shortens a proof is precisely a counterexample to deriving (U3) from Remark D.7. The paired E2p audit reports installed-minus-K_0 mean -0.131944 over 24 cells

Stage	Factor or loss	Source and applicability
Discover	$P_{\text{disc}} \le (1+\delta_\Gamma)/2$	Thm 4.1; binary hidden world
Retain	$P_{\text{ret}} \le \min\{1, \lfloor B_r/d \rfloor / m^d\}$	Thm 4.2 chain count; uniform marked leaf
Use	$\bar{\eta}_T - \beta_T(t)$	Remark D.27; matched budget, fixed charge

Table 4: Conditions needed to compose the three ceilings. The uncapped call-cost ratio in Thm 4.2 alone is not a finite-budget bound on P_{ret}, and the three factors are not universal properties of a deployed LLM.

(`experiments/e2p_k0_paired_utility_audit/results-20260923/summary.json`). G-4 reports gold-minus-K_0 mean -0.0093, world-bootstrap 95% interval $[-0.0648, 0.0556]$ over six worlds (`results/g4_finite_consumer/g4_summary.json`). Both observations are compatible with a positive prompt burden, but neither identifies β_T: matched-budget content gain and burden were not separately intervened on. G-4's gold library has six entries in every cell (`g4_dryrun_K_sizes.json`); in entry count it is not the smallest nonempty arm. These finite-budget measurements also do not estimate $\eta_T(K)$ at infinite horizon.

Remark D.29 (Gold-content ceiling (Result III′)). Suppose (U1)–(U3) hold for every candidate K under consideration and, in addition, every K whose verified contents lie in $\mathrm{Th}(\mathrm{gold})$ obeys the *consumer-content domination* assumption $\eta_T(K) \le \eta_T(\mathrm{gold}) \le \varepsilon$. Then every such K satisfies

$$U_T(K, b) - U_T(K_0, b) \le \varepsilon - \beta_T(\mathrm{tok}(K)).$$

Proof. Apply (3) and the stated content-domination assumption. □

For a fixed LLM consumer, logical inclusion of verified theorems alone does not imply content domination: prompt phrasing, retrieval and policy behavior can change success even when the theorems are equivalent. Nor does G-4's finite-budget confidence interval establish the population premise $\eta_T(\mathrm{gold}) \le \varepsilon$.

Remark D.30 (Conditional discover–retain–use pipeline ceiling (Result P)). Consider a two-stage installation protocol. Let D mean that the discovery stage returns the correct hidden-world artifact, and R mean that this artifact is available to the held-out consumer after context reset. Set $P_{\text{disc}} := \mathbb{P}(D)$ and $P_{\text{ret}} := \mathbb{P}(R \mid D)$ (zero if $P_{\text{disc}} = 0$). Assume (i) a fixed prompt charge $a = ct \le b$ is paid on every trial, including failure; (ii) on $D \cap R$ the matched-budget content gain at $h = b - a$ is at most a constant $\bar{\eta}_T \ge 0$, while on its complement the consumer receives K_0 at horizon h; and (iii) $S_T(K_0, h)$ is nondecreasing. Then

$$\mathbb{E}[U_T(\mathrm{installed}, b)] - U_T(K_0, b) \le P_{\text{disc}} P_{\text{ret}}\, \bar{\eta}_T - \beta_T(t). \tag{4}$$

In the equal-prior, disjoint-success hidden-world model, if D entails correctly distinguishing the world through the no-side-channel verifier, $P_{\text{disc}} \le (1+\delta_\Gamma)/2$ by Thm 4.1. Separately, if conditional on D the retention stage receives no information identifying the leaf and is an M^{flat} search for a uniformly random marked leaf in $\mathcal{W}_{m,d}$ with at most B_r verifier calls, then $P_{\text{ret}} \le \min\{1, \lfloor B_r/d \rfloor / m^d\}$; hence both factors may be substituted in (4) when these additional model conditions hold. If (U3) also holds for every successfully installed library, one may set $\bar{\eta}_T$ to a common nonnegative upper bound on their $\eta_T(K)$; in the special case of one fixed K with $\eta_T(K) \ge 0$, this yields the requested $P_{\text{disc}} P_{\text{ret}} \eta_T(K) - \beta_T(t)$ form.

Proof. At horizon h, the content gain is zero outside $D \cap R$ and at most $\bar{\eta}_T$ on it. Its expectation is therefore at most $\mathbb{P}(D \cap R)\bar{\eta}_T = P_{\text{disc}} P_{\text{ret}} \bar{\eta}_T$; subtract the fixed baseline loss $S_T(K_0, b) - S_T(K_0, h) = \beta_T(t)$. The discovery bound is Thm 4.1 under its stated conditions. For the retention bound, the chain argument in the proof of Thm 4.2(a)–(b) shows each distinct depth-d leaf first visited by a flat policy needs a disjoint chain of d calls. Thus by B_r calls it can visit at most $\lfloor B_r/d \rfloor$ of the m^d equally likely leaves. Randomization averages the same bound. □

E Deferred experimental material

Omitted main-result panel. The planned panel showing predicted lines for $B \in \{1, 2, 4\}$ with measured SD points, 95% CIs, and a strong baseline is omitted: no run exists with predictions frozen in advance for $B = 1, 2, 4$, measured SD, a strong baseline, and 95% CIs. We do not approximate the panel from other runs.

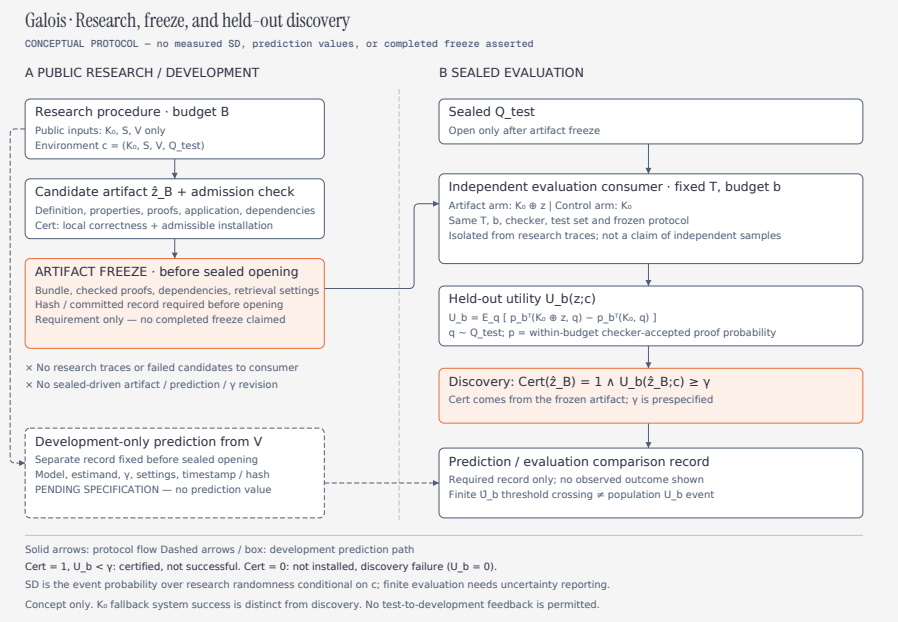


Figure 4: **Discovery-then-sealed-evaluation protocol (schematic, not an empirical SD result).** Research budget B produces an artifact whose proofs, dependencies, and retrieval settings are frozen before sealed evaluation. An independent consumer compares $K_0 \oplus z$ with K_0 using the same solver, checker, and per-theorem budget b. The population success event is $\mathrm{Cert}(\hat{z}_B) = 1$ and $U_b(\hat{z}_B; c) \geq \gamma$; development-based prediction is separate from sealed evaluation. This schematic asserts neither fitted prediction values nor a completed prediction-freeze receipt. The lemma-pair companion panel is Figure 5. The remaining companion panel, $B \in \{1, 2, 4\}$ predictions against measured SD with 95% CIs and a strong baseline, is omitted rather than shown as a placeholder because it requires a frozen prediction and SD measurements that have not been run.

E.1 Intervention comparison against a strong library baseline

Question. How do the ability to keep verified-or-partial intermediate structure and the ability to return from a dead end change, separately and jointly, the gap between discovering a structure and using it on unseen theorems?

Conditions. Five conditions run on the same model, tools, public information, and final library cap; they differ only in the two capabilities and in a diagnostic oracle: C1, a strong library baseline that stores and retrieves only checker-verified lemmas; C2, C1 plus retention of unfinished intermediate states as checkpoints, without the ability to resume from an earlier one; C3, C1 plus rollback to a single best-so-far state, without multiple checkpoints; C4, both (the proposed search); and C5, gold conditions that supply the target structure either as a specification only or with its proofs, used to locate difficulty (ideation, formalization, or use) and not counted as a competitor.

Matched resources and reported costs. Caps on research-phase output tokens (with hidden reasoning tokens counted inside the cap), verifier calls, per-theorem proof tokens, and final artifact size are identical across conditions, and all conditions run interleaved on the same day and provider. Matching generation tokens is not called "matched total compute": input tokens, verifier CPU time, verifier and total wall time, peak memory, and cost are recorded per run and reported alongside the primary result as distributions per condition.

Primary endpoint and analysis unit. The primary quantity is the sealed-test verified success rate; the primary estimate is the base-world-paired difference between C4 and C1 at the main research budget, tested two-sided with a prespecified minimal effect of 0.10. The independent unit is the base world: seed replicates and notational variants of one world are clustered, never counted as separate samples. Worlds on which discovery fails are not dropped: the follow-up runs fall back

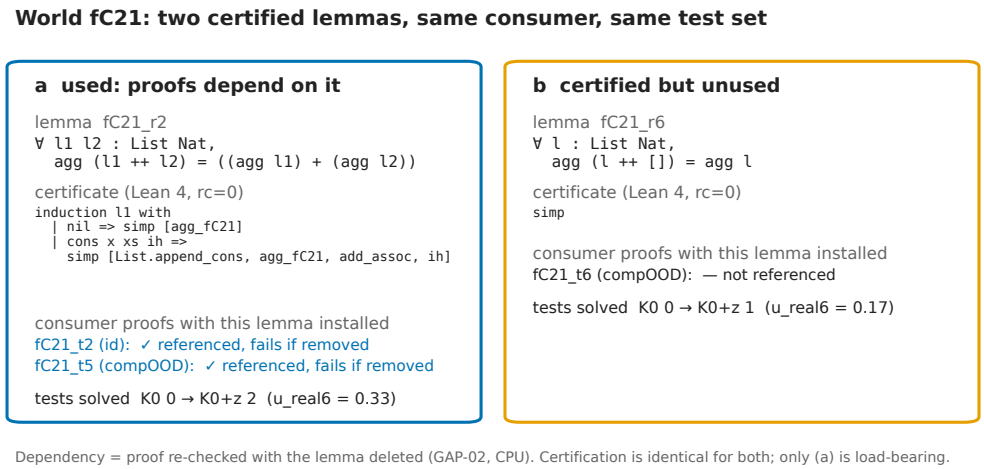


Figure 5: **Same world, two certified lemmas: one load-bearing, one not (world fC21, exploratory).** Both lemmas were certified by the same Lean 4 checker (return code 0) from the same research run. (a) `fC21_r2` is referenced by two sealed consumer proofs (`fC21_t2`, `fC21_t5`), and re-checking those proofs with the lemma deleted fails, so the proofs depend on it; the consumer solved 2 tests with $K_0 \oplus z$ against 0 with K_0. (b) `fC21_r6` is referenced by no consumer proof; the one $K_0 \oplus z$ success in that cell (`fC21_t6`) does not use it. Dependency is the GAP-02 proof re-check (CPU, one consumer model, one sample per cell); it is evidence of reuse, not a utility estimate, and panel (b) shows a lemma the proofs do not depend on rather than one shown to add no solves. Data: `results/gap02_lemma_dependency/` and `results/g5_real_consumer/summary.json`.

to K_0, discovery success and system success are recorded as separate columns, and the cost of the failed attempt is retained. Secondary contrasts (the two main effects, their interaction, and a compute-matched direct baseline) are Holm-corrected as a separate family.

Scale. A separate twelve-world pilot, excluded from the confirmatory analysis, fixes the confirmatory scale (three families of twenty base worlds) by a prespecified rule; pilot estimates are not reported as results.

Family B exploratory pilot: frozen-artifact utility ($n = 4$ worlds). The exploratory four-world pilot gives sealed-test success rates 0.3333 with the frozen bundle (C4) versus 0.2917 with the baseline condition C1 (fB1–fB4, $n = 8$ solver-instances per condition; C1 is not a K_0-only control); it is excluded from confirmatory analysis. **[CORRECTED 2026-09-24: prior manuscript erroneously reported 0.1875/0.0625; reaggregated values from `galois-e2z3-pilot-reaggregation-20260924` are 0.3333/0.2917.]** Details and provenance are in Appendix J.

Context-reset cost audit (exploratory proxy). An offline audit of 32 Family B pilot episodes (four worlds, two seeds, four policies) finds mean fresh discovery calls per episode of 6.0, 14.5, 14.875, and 17.875 for R, R+S, R+B, and V, respectively. Their mean total discovery prompt-input tokens are 1,576.625, 5,587.25, 5,238.625, and 6,563.375. The corresponding modeled input-token costs are USD 0.000914, 0.003241, 0.003038, and 0.003807 per episode, using the runner's fallback prompt-token price rather than provider billing. The source JSON, world-cluster bootstrap intervals, and analysis code are in `experiments/gx2_context_reset_cost/`. These are prompt-burden proxies, *not* measurements of re-sent prefix tokens: the logs contain neither prefix boundaries nor explicit reset events, and the runner has no matched retained-context arm. Therefore this audit does not test the context-reset cost ratio predicted by Theorem (II).

GX-2 local-frame operational comparison (exploratory). We additionally ran a paired, application-local opaque binary marked-leaf checker with Azure DeepSeek-V4-Pro proposing the next leaf: depths $d \in \{2, 3, 4\}$, 10 chains, three seeds, and fresh versus retained checker frames (180 episodes, 30 pairs per depth). The fresh checker replays the depth-d path for each

Table 5: **GX-3 replay accounting, not provider-side restoration** (90 saved orders; 30 per depth). LLM fresh/cache is the global-prefix-cache ratio of means with chain-cluster 95% CI; fresh/stack uses only the current-path popped-frame stack on the same orders. Random, lexicographic, and worst-target columns use the original cache accountant; first-hit is pointwise; LCP is adjacent longest-common-prefix, LLM / random.

d	fresh/cache [CI]	fresh/stack	random	lex.	worst	first-hit	LCP	>p95
2	1.190 [1.183, 1.198]	1.053	1.201	1.250	1.333	1.00	0.17 / 0.34	0%
3	1.446 [1.434, 1.461]	1.151	1.452	1.588	1.714	1.00	0.50 / 0.57	0%
4	1.746 [1.730, 1.761]	1.246	1.748	2.000	2.133	1.00	0.87 / 0.74	23%

proposal; the retained checker restores previously checked prefix frames without charging another checker call. All 180 episodes reached the marked leaf. The ratio of mean charged checker calls, fresh over retained, was 1.292 (95% chain-cluster bootstrap CI $[1.016, 1.600]$) at $d = 2$, 1.780 $[1.332, 2.366]$ at $d = 3$, and 1.897 $[1.532, 2.324]$ at $d = 4$ (10 chain clusters; seeds and arms resampled together). The full grid used 947 model calls and approximately USD 0.094 at the runner's fallback tariff. This is an *operational local-oracle contrast*, not evidence that Azure retains model context: its chat-completions endpoint is stateless and the retained arm resends its message history. Prompt formats can also change the model's proposal order. Thus these ratios do not identify whether a deployed LLM satisfies Theorem 4.2's interface assumptions or test its minimax bound. Protocol, episode-level counts, and chain-cluster intervals are in `experiments/gx2_context_reset_cost/RESULTS_20260925.md` and `results/gx2/retained_arm_summary.json`.

GX-3: proposal-order replay under two charging rules. GX-2 compared fresh and retained arms under different prompt formats, so the proposal order was confounded with the interface. GX-3 holds the proposal order fixed in a replay calculation under two stipulated charging rules: for each depth $d \in \{2, 3, 4\}$, chain (10) and seed (3), i.e. 90 cells, Azure DeepSeek-V4-Pro emits one target-blind proposal order under a single prompt format, and that same order is replayed against every one of the 2^d marked leaves under a fresh accountant (d charged calls per proposal) and a retained accountant (new prefix edges only). All 90 cells completed with no API failures (USD 0.064; invalid outputs fall back to lexicographic order in 26/5/16 turns at $d = 2/3/4$). Table 5 reports (a) the uniform-target ratio of means with a 5,000-draw chain-cluster bootstrap 95% CI, (b) the worst-target ratio, which equals $dm/(2m - 2)$ under this replay accountant at every depth, and the first-hit ratio, which is 1 *pointwise* (a target found on the first proposal costs the same d calls in both arms), and (c) the same ratio and adjacent longest-common-prefix (LCP) statistic for 2000 uniform random permutations and for lexicographic order. Under the global-prefix-cache accountant, the LLM order is close to the uniform-random reference at each depth (Figure 6a) (fresh/cache ratios $1.190/1.446/1.746$ vs $1.201/1.452/1.748$). Replaying those same 90 orders with a current-path popped-frame stack instead gives fresh/stack $1.053/1.151/1.246$; this is a different accounting rule, not evidence that the provider restores frames. At $d = 4$ its adjacent LCP (0.873) exceeds the random mean (0.737), and 7 of 30 cells (23%) exceed the random 95th percentile. So the ratios are uniform-target averages or worst-target values and are *not* an individual agent's per-episode cost ratio; the regime with a non-uniform LLM order is at $d = 4$ only, and we do not claim that provider-side frame restoration is free (the endpoint is stateless). Operational correspondence to Theorem 4.2 has not been established for arbitrary replay orders; equality of an accounting ratio to its formula does not establish the theorem's interface assumptions for a deployed agent. Raw orders and the summary (`results/gx3/summary.json`, sha256 prefix `6e8946b4`) and code (`experiments/gx3_policy_order_replay/`) are in the repository.

Consumer burden across 96 cells. The burden, exact-graph proxy, and G-5/GX-1 exploratory plots are in Appendix I; the main-text figures are reserved for the two restricted analyses and the paired held-out utility result. We pool the 72 G-4 finite-consumer cells (arms K_0, z_R, z_V, gold; worlds cB1–cB4, fB1–fB2; three seeds) with the 24 paired cells above, and record for every non-K_0 cell the context tokens $\mathrm{tok}(K)$ that its library adds to the prompt (API-recorded prompt tokens minus those of the paired K_0 cell on the same test items; a cl100k recount of the library string gives the same picture) and $\Delta U = U(K) - U(K_0)$ on the same world and seed. The 18 K_0 anchors are $\Delta U = 0$ by construction and are excluded, leaving 78 cells in 6 worlds. A *descriptive* linear

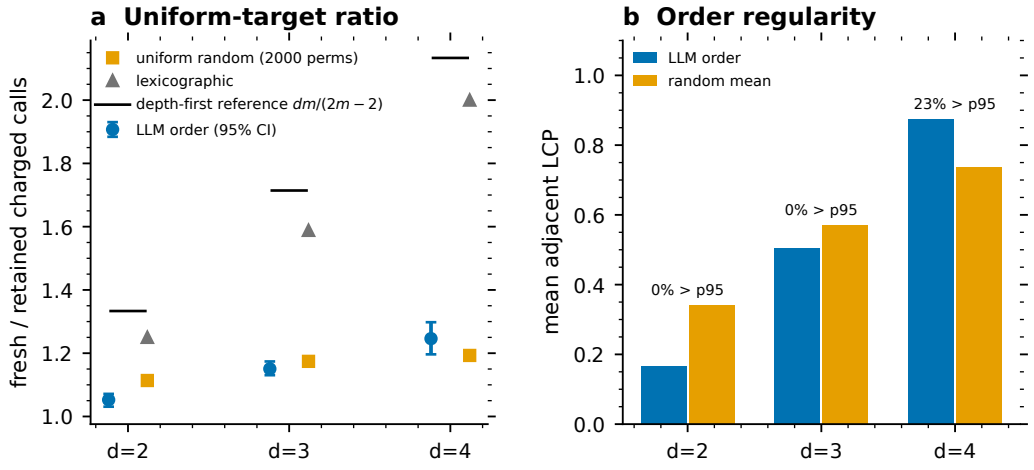


Figure 6: **GX-3 global-cache replay and order structure.** (a) Uniform-target fresh/cache ratio for the LLM order (blue), 2000 random permutations (orange), lexicographic order (grey), and worst-target cache-accountant value (bar). The same orders yield fresh/stack $1.053/1.151/1.246$ at depths $2/3/4$ (Table 5); the plotted ratios are not stack ratios. (b) Mean adjacent LCP against the random mean. Chain-cluster $n = 10$; provider-side frame restoration is untested.

association of ΔU with context length has slope $+0.31 \times 10^{-3}$ per token (world-cluster bootstrap 95% CI $[-0.15, +0.56] \times 10^{-3}$, 2,000 resamples, fixed seed), intercept -0.142 ($[-0.22, -0.02]$), and $R^2 = 0.03$. This slope is neither an estimate of nor an empirical counterpart to the baseline-horizon burden $\beta_T(t) = S_T(K_0, b) - S_T(K_0, b - ct)$ in Remark D.27: the same K_0 has not been evaluated at both horizons and c has not been measured. Conditions with about the same context length differ in mean ΔU (gold -0.009 at 304 tokens, installed-V -0.097 at 312, z_V -0.056 at 330), yet their world-cluster intervals overlap, so neither a length-only account nor a content-quality account is resolved by these data. Six worlds and one consumer limit any statement beyond this artifact class. A content-free padding control (G-4b; 18 additional cells, same worlds and seeds, the K_0 prompt plus a filler block of repeated "pad" tokens sized to the API-recorded length of the z arms, realized mean 198 tokens; K_0 and z cells reused from G-4, not re-sampled) gives $\Delta U(K_0^{\text{pad}} - K_0) = +0.028$ (world-cluster 95% CI $[-0.019, +0.074]$), against -0.056 ($[-0.130, +0.028]$) for z_R and -0.056 ($[-0.185, +0.093]$) for z_V. Length alone does not reproduce the negative point estimates, but all three intervals contain 0 and overlap; this padding comparison does not identify the baseline-horizon burden either. The filler is unnatural repeated text, and one consumer and six worlds bound the claim. Figure 9 shows all cells.

Frozen-artifact utility audit, paired design ($n = 12$ pairs): null result. The paired audit gives $\overline{\Delta}_{sp} = -0.0833$ ($p = 0.2188$ over the 12 pairs; treating the 4 base worlds as the independent unit, the exact sign-flip $p = 0.25$), a non-significant, directionally non-positive exploratory result. Details and provenance are in Appendix J.

Theorem-faithful stack: mechanism readout only (exact counts, cost $0). Job `galois-theorem-faithful-stack-20260925` (registration `prereg/theorem_faithful_stack_mechanism_20260926.json`; launcher `code/queue/launch_theorem_faithful_stack.sh`, commit `931f47f`) compares three charging rules on fixed proposal orders: A (fresh: every step rebuilds its prefix), D (LIFO-strict stack: popped frames are discarded and rebuilt on re-entry, the stack model of the theory), and F (arbitrary-order global prefix cache). Two tree-search families are used, `bin_d4` ($m = 2$, $d = 4$) and `ter_d3` ($m = 3$, $d = 3$)—tree shapes, not the Lean families of the registered full design—each with 2 calibration and 4 unseen worlds (hidden marked leaf), under a deterministic `lex_dfs` policy and `random_perm` policies (seeds 1–3); 48 rows in total, plus the 90 saved Azure proposal orders of GX-3 (`results/gx3/orders`; 30 each at $d = 2, 3, 4$). Both preregistered checks pass on every row: executor and closed-form D costs agree exactly, and `lex_dfs` gives $D = F$ on complete

Table 6: **Theorem-faithful stack mechanism readout (exact operation counts, cost $0).** Mean complete-enumeration counts under charging rules A (fresh), D (LIFO-strict stack), F (arbitrary-order prefix cache), and the mean excess $D - F$ on complete enumeration and to first hit; "exact" marks executor-vs-closed-form agreement on every row. Deterministic accounting on fixed orders; no LLM policy, no utility, no SD outcome. The lower block replays the 90 saved GX-3 Azure orders by depth; the pooled row averages the three depths. Source: `results/theorem_faithful_stack/full/RESULT.json`; depth rows from `results/tfs_depth_stratified_r115/full/RESULT.json`.

family	policy / split	n	A	D	F	$D-F$ (complete)	$D-F$ (first hit)	exact
bin_d4	lex_dfs / calibration	2	64.0	30.0	30.0	0.0	0.0	yes
bin_d4	lex_dfs / unseen	4	64.0	30.0	30.0	0.0	0.0	yes
bin_d4	random_perm / calibration	6	64.0	51.2	30.0	21.2	5.2	yes
bin_d4	random_perm / unseen	12	64.0	53.1	30.0	23.1	7.1	yes
ter_d3	lex_dfs / calibration	2	81.0	39.0	39.0	0.0	0.0	yes
ter_d3	lex_dfs / unseen	4	81.0	39.0	39.0	0.0	0.0	yes
ter_d3	random_perm / calibration	6	81.0	72.2	39.0	33.2	13.5	yes
ter_d3	random_perm / unseen	12	81.0	70.0	39.0	31.0	16.4	yes
GX-3 Azure orders, $d = 2$		30	8.0	7.5	6.0	1.50	–	–
GX-3 Azure orders, $d = 3$		30	24.0	20.5	14.0	6.47	–	–
GX-3 Azure orders, $d = 4$		30	64.0	50.9	30.0	20.90	–	–
GX-3 Azure orders, pooled		90	–	–	–	9.62	–	–

enumeration (checkpointed-DFS identity). Under `random_perm` the LIFO stack pays a mean excess $D - F$ of 27.2 (calibration) and 27.0 (unseen) operations on complete enumeration and 9.3 / 11.8 to first hit; on the 90 saved GX-3 Azure orders the pooled mean excess is 9.62, the average of 1.50, 6.47, and 20.90 at $d = 2$, 3, and 4 (chain-cluster 95% CIs $[1.30, 1.70]$, $[6.00, 6.87]$, $[18.97, 23.00]$; depth-stratified recount `results/tfs_depth_stratified_r115/full/RESULT.json`, commit `5c5efa1`), so the pooled value is not a single-depth figure (Table 6). These are deterministic accounting consequences of proposal order, not evidence of agent capability, useful discovery, or provider-side state restoration. The live-agent closed-loop readout and the SD/consumer-utility readout of the registered design were NOT EVALUATED; a reduced live run is reported in the paragraph "Live stack execution" (§E.1). Full run: $N = 48$, cost $0, exit 0 (`results/theorem_faithful_stack/full/receipt.json`; table generated by `code/make_tfs_table.py`).

Live stack execution, reduced budget (exploratory, $N = 24$, $0.18). Job `galois-tfe-sd-minimal-20260926` (attempt 2, launched 09:13 JST, receipt and design at commit `247d057`) runs the three retention arms of the registered design as live agents: C1 (nothing retained; every node rebuilds its prefix by real re-execution), C4 (theorem-faithful LIFO stack; popped frames are discarded and rebuilt on re-entry), and GPC (global prefix cache with arbitrary-order restore). The research policy and the consumer are both Azure DeepSeek-V4-Pro. The design crosses 4 existing Family B worlds (cB1–cB4) with 2 research seeds and the 3 arms, giving 24 trajectories. Within a (world, seed) block, the three arms share the same seeded proposal schedule (a depth-3 binary revision tree per target). The run finished all 24 cells at 10:33:50 JST: 639 provider calls, $0.18 against a $50 cap, no budget stop, and no Lean timeouts, provider failures, or wall-clock caps. *Deviations from the registered design* (recorded in `results/tfe_sd/minimal/launch_receipt.json`): unseen worlds tW01–tW64 were not generated, so existing worlds are used; the sealed set has 10 theorems per world (4 validation, 6 test), not 16; the consumer is the Azure model, not the local prover; each trajectory has 16 checker calls, 16,384 generated tokens, 600 s of discovery wall-clock, and a 300 s Lean timeout; the artifact is frozen only at full budget (the 1/4 and 1/2 freezes were dropped); and the consumer uses 1 seed. The 16-call cap binds in 11 of 24 trajectories (C1: 1, C4: 5, GPC: 5). *Cost readout (M4).* Mean theorem actions (executed frames) per trajectory are 10.50 (C1), 14.25 (C4), and 13.88 (GPC). C1 re-executes 11 frames in total, and C4 and GPC restore 22 frames each. Paired within the 8 blocks, C4−C1 is $+3.75$ actions and GPC−C1 is $+3.375$ (each positive in 6 of 8 blocks), and GPC−C4 is -0.375 (3 negative, 2 zero, 3 positive). This is not a test of Theorem L3, which compares executors on the same trajectory. Provider sampling is not reproducible: the same empty-library consumer evaluation returned 0.2 and 0.6 on cB4, as reported below. Hence the three arms' trees diverge after the first

Table 7: **Live retention arms at reduced budget (`galois-tfe-sd-minimal-20260926`, $N = 24$, $0.18).** Per-arm means over 8 (world, seed) blocks: theorem actions, total re-executed and restored frames, the number of trajectories that exhausted the 16 checker calls, accepted research targets (of 6), sealed-set solve rate with the frozen full-budget library $\hat{s}(K)$, and its difference from the empty library. The lower block shows paired differences with sign counts. The research policy and consumer are Azure DeepSeek-V4-Pro with a single consumer seed; the K_0 evaluations alone vary by up to 0.4 within a world. The table is descriptive and gives no SD estimate. Source: `results/tfe_sd/minimal/rows.jsonl`.

arm	n	actions	re-exec	restores	cap hit	accepted	$\hat{s}(K)$	$\hat{s}(K) - \hat{s}(K_0)$
C1	8	10.50	11	0	1	5.88	0.250	-0.200
C4	8	14.25	0	22	5	4.88	0.263	-0.137
GPC	8	13.88	0	22	5	5.12	0.350	-0.038
paired over 8 (world, seed) blocks: mean difference [#neg / #zero / #pos]								
C4$-$C1	$+3.750$ [1/1/6]					-1.000		$+0.0125$ [4/1/3]
GPC$-$C1	$+3.375$ [1/1/6]					-0.750		$+0.1000$ [2/3/3]
GPC$-$C4	-0.375 [3/2/3]					$+0.250$		$+0.0875$ [3/0/5]

differing output, and C1 resolved more targets early (all 6 research targets in 7 of 8 blocks, versus 3 for C4 and 4 for GPC; mean accepted 5.88, 4.88, and 5.12). *Utility and pairing (M1/M6).* The consumer interface was not measured in this run (δ_Γ is not computed), and the M6 paired arm contrast below is preliminary evidence only: it is not a held-out frozen-artifact utility comparison, and it does not measure an intervention-arm effect. At full budget, the mean sealed-set solve rate with the frozen library, $\hat{s}(K)$, is 0.250 (C1), 0.263 (C4), and 0.350 (GPC); the mean difference from the empty library K_0 is -0.200, -0.137, and -0.038. Paired library solve-rate differences are $+0.0125$ (C4$-$C1; 4 negative, 1 zero, 3 positive), $+0.100$ (GPC$-$C1; 2/3/3), and $+0.0875$ (GPC$-$C4; 3/0/5). The indicator $\hat{s}(K) - \hat{s}(K_0) \geq \gamma = 0.05$ holds in 8 of 24 trajectories (2, 2, and 4 per arm). We do not treat this fraction as an estimate of SD. Nine independent K_0 evaluations of the same empty library differ within a world by up to 0.4 (cB1: 0.5/0.6/0.6; cB3: 0.3/0.3; cB4: 0.2/0.2/0.6). This consumer noise is as large as every contrast above, and with $n = 8$ blocks and 10 sealed theorems per world, none of the contrasts is distinguishable from zero. These arm outcomes are therefore supplementary and preliminary; they do not supply the held-out frozen-artifact utility comparison or the intervention-arm outcome that the restricted analyses would require. Table 7 gives the arm summary, and the per-cell values are in Appendix Table 9. Source: `results/tfe_sd/minimal/rows.jsonl`; aggregation by `code/make_tfe_sd_minimal_table.py` (`agg_summary.json`). *Event-log replay (supplementary, exploratory).* A CPU-only replay (commit `5c5efa1`, cost $0; `results/tfe_sd_eventlog_replay_r115/full/RESULT.json`) re-derives the accounting from the 24 live event logs (SHA256-checked snapshot, no integrity failures). It applies the charging rules A, D, and F of the theorem-faithful stack readout to each realized extension sequence. In all 8 C4 cells the D count equals the recorded theorem actions, and in all 8 GPC cells the F count does; C1 has no matching rule and is only reported. The replay also reproduces the per-arm means of `agg_summary.json`. On these trajectories no extended node recurs, so the live D$-$F excess is 0 in every cell. The live runs therefore check that the executors charge as specified, but they never enter the regime in which the stack pays more than the cache. This is supplementary evidence on the accounting only: it does not connect the interface quantities of Section 4 to the evaluated agent's utility, and the 4-world summaries are descriptive.

Exact finite-graph panel (Family B). Table 10 (appendix) places, for one family and the same 40 worlds, the state-DP value, the independent enumeration value, and the search-effort contrast between exhaustive expansion (no return; the C1 side of the keep×return contrast) and rollback search (return; the C4 side). All 40 worlds were enumerated to completion (0 truncated), and the per-world difference DP $-$ enumeration is exactly zero for cost and for both bounds of $p^\star$ (Figure 13). Optimal success ($p^\star$) and truncation (none) are reported next to the expansion counts. This is a structural E1 proxy, not the E2 estimand: no C1 or C4 solve rate is reported here.

For the separately filtered 51 exact-enumeration worlds with an accepting vertex, Figure 10 displays the finite-graph search-budget contrast by family and the underlying two threshold values.

Certification is not held-out utility (exploratory G-5/GX-1). Figure 11 juxtaposes two diagnostic results rather than an E2 intervention estimate. In G-5, 100 completed trials span 10 worlds, five seeds, and two target-selection policies (50 trials per policy). Both policies proposed a target on every trial. The world-mean certified fraction was 0.36 for LLM selection and 0.68 for random target selection, whereas the useful fraction was 0.14 for both; certified-but-unhelpful fractions were 0.22 and 0.54, respectively. These are descriptive rates, not evidence that supplying a structure improves autonomous discovery. For GX-1, only 23 of 24 planned cells completed; the balanced three-world subset has 32 verified lemmas in 54 attempts in each feedback arm. The archived summary reports zero exposed error slots, so the proposed feedback intervention was inactive. The raw GX-1 cells are absent from this worktree, and neither the incomplete arm contrast nor a causal feedback effect is reported. Artifact hashes and recovery limitations are listed in `EXPERIMENT_AUDIT_20260925.md`.

The zero-difference audit plot is in Appendix J, Figure 13.

E.2 Frozen-artifact transfer to unseen theorems

Freeze. At the end of the research phase, the discovered structure, its checked proofs, its dependency record, and the retrieval settings used are saved as one bundle whose hash is committed before any sealed test is opened; changes after opening are recorded as freeze violations and excluded. Bundles that fail the check are not installed; the follow-up runs then use K_0 and are scored as such. Conversation history, failed candidates, and reasoning traces are not part of the bundle.

Splits. Unseen theorems are released in four splits: new theorems of the same family (S1), larger instances (S2), unseen compositions (S3), and the same theorems handed to a different solver that receives only the bundle (S4). The primary endpoint is the paired C4-versus-C1 difference on S1; S2–S4 are secondary and Holm-corrected.

Prespecified claim reduction. If the S4 portability ratio falls below the committed threshold, the discovered structure is reported as local auxiliary reasoning rather than as a reusable abstraction; if S2 or S3 fail, the claim is restricted to same-family, same-size theorems; if S1 fails, E3 is removed from the main claims. Reductions are one-directional and no split is redefined after the outcomes are seen.

Reporting. For both protocols we release the full budget curves (success at every recorded token snapshot) rather than only the largest-budget endpoint, together with the prespecified NOGO outcomes, so that the empirical claim is falsifiable including when additional budget yields no detectable improvement.

F Positioning: from proving to discoverability

Finite-budget *statistical provability* asks for the probability that a specified proof-search procedure reaches a verified proof. We retain that finite-budget perspective, but move the object of evaluation one level up: the agent must produce a certified abstraction and that abstraction must help on a held-out theorem population. The distinction prevents a solved training instance, an unverified textual suggestion, or a retrieved proof from being counted as a discovered structure.

F.1 Relation to nearby systems and theory

Table 8 records the intended boundary of this work against twelve nearby systems and theory papers. These are scope comparisons, not strict-containment or empirical superiority claims. In particular, the table does not claim that all prior systems presume a supplied structure, nor that a proposed intervention has succeeded before the preregistered experiments are run.

Library-learning systems. Library-learning systems (DreamCoder, LILO, LEGO-Prover, Dream-Prover) report that accumulated abstractions help on held-out tasks, and Berlot-Attwell et al. show that in at least two of them the gain is not attributable to reuse (Berlot-Attwell et al., 2024). We take that critique as the starting point: rather than propose a learner, we define statistical discoverability so that certification, proposal and held-out utility are separated, bound it under a finite search graph, and preregister E2/E3 to test whether retention matters at matched budget.

Bounded discovery and classical limits. Five classical results delimit what discovery can mean under a finite budget; only the last is connected to our measurements formally. Universal approximation (Cybenko, 1989; Hornik et al., 1989) says that a sufficiently expressive model *can* represent a useful structure, but not that a bounded search *finds* one; that gap is the question we test. No-free-lunch (Wolpert & Macready, 1997) implies that no search procedure is best averaged over all problems, so a frozen artifact can help only relative to an inductive bias matched to the theorem family, which is why utility is measured on a held-out population and not asserted in general. Solomonoff induction (Solomonoff, 1964) names the compute-unbounded ideal; our budgeted proof search is a bounded surrogate for it, and we make no claim to approximate it. The Bekenstein bound (Bekenstein, 1981) is a physical ceiling on the information a bounded region can hold; we cite it only as context for the observation that any stored artifact is finite, and do not use it quantitatively. Epiplexity (Finzi et al., 2026) splits the description length of data under a compute bound T into structural information S_T and residual entropy H_T, $\mathrm{MDL}_T = S_T + H_T$. We read a certified frozen artifact as a candidate witness that lowers the *conditional* residual $H_T(Y \mid \text{artifact})$ at a proof budget b (playing the role of T) at a structural cost $S_T(\text{artifact})$, and a null held-out utility as a statement that no reduction was observed at the tested budgets. This is an operational instance of that decomposition, specified as protocol E2-EPX with the experiments, not an identity between the two frameworks.

A gap below an optimum is not a gap below a ceiling. The Bellman analysis controls distance below the optimum of a specified decision problem. The information analysis instead gives an upper bound under a hidden-world interface. Combining them requires the same estimand, initial distribution, admissible policy class, and budget. Without a common problem, write s for achieved success, v for optimal success, g for a gap bound, and c for a ceiling bound: $s \geq v - g$ and $v \leq c$ give $s \geq c - g - (c - v)$. The term $c - v \geq 0$ is unknown; our analysis leaves it uncontrolled. We therefore withdraw the claimed interval $[c - g, c]$ for s and report only $s \geq v - g$ for the specified decision problem.

Extended related work

Proof assistants and hammers. The checker in our restricted models abstracts an interactive proof assistant such as Lean 4 with its mathematics library (de Moura & Ullrich, 2021; The mathlib Community, 2020) or Isabelle/HOL (Nipkow et al., 2002). Hammers connect such assistants to automated provers through premise selection and proof reconstruction (Blanchette et al., 2016; Kaliszyk & Urban, 2014), and learned premise selection improves this pipeline (Irving et al., 2016; Bansal et al., 2019; Mikuła et al., 2024). Thor integrates a hammer into a language-model prover (Jiang et al., 2022a). In our terms, all of these change the oracle interface; the ceiling of Theorem 4.1 is stated for a fixed interface and is agnostic to which prover sits behind it.

LLM-based formal theorem proving. Benchmarks and curricula for Olympiad-level formal proving (Zheng et al., 2022; Polu et al., 2023) enabled a sequence of language-model provers: whole-proof generation with repair (First et al., 2023), in-context agents that query the proof assistant (Thakur et al., 2024; Song et al., 2024), proof-assistant feedback with tree search and subgoal decomposition (Xin et al., 2024b; Ren et al., 2025), and interleaved informal reasoning (Lin et al., 2025). Olympiad-level results with synthetic data and reinforcement learning (Trinh et al., 2024; AlphaProof and AlphaGeometry teams, 2024) are reported as solve rates on sealed problems. Our contribution is orthogonal: we do not propose a prover, and we take the reported solve rate as the utility side of certified-and-useful (Definition 2.4) rather than as evidence of transferable structure.

Search with restorable state and machine-aided discovery. Bandit-based tree search (Kocsis & Szepesvári, 2006; Silver et al., 2018) and its language-model variants (Yao et al., 2023) keep and revisit intermediate nodes; Theorem 4.2 isolates the checker-call cost of rebuilding versus restoring such a frame in an opaque marked-leaf model, without a claim about any particular search policy. Machine-aided discovery of mathematical structure (Davies et al., 2021; Fawzi et al., 2022; Romera-Paredes et al., 2024) reports objects whose value is verified after the fact; statistical discoverability asks, before the fact, how often a frozen procedure returns an artifact that is both certified and useful.

G VERIFIER-SIGNATURE CEILING (APPENDIX)

Proposition G.1 (Restricted-interface Bayes/TV bound)**.** *Under (A1)–(A6), fix a budget B in the hidden-world query model. Then every fixed solver T and $\pi \in \mathcal{P}_B^T$ obey*

$$\| \operatorname{law}(H_{\tau_\pi} \mid U = 0) - \operatorname{law}(H_{\tau_\pi} \mid U = 1)\|_{\mathrm{TV}} \le \delta_\Gamma^B, \qquad \mathbb{P}(\widehat{a}_\pi = U) \le (1 + \delta_\Gamma^B)/2.$$

Moreover $\delta_\Gamma^B \le \delta_\Gamma$ and $B \mapsto \delta_\Gamma^B$ is nondecreasing. This is the standard equal-prior Bayes/TV identity together with the definition of the admissible interface class; it is not an independent computational hardness theorem.

Proof. The policy-class inclusion gives the first bound by definition and Proposition H.3 gives the success bound. Enlarging the budget preserves every lower-cost policy. Each allowed history is a world-independent garbling (in the comparison-of-experiments sense of Blackwell, 1953) of Γ_V, so data processing gives $\delta_\Gamma^B \le \delta_\Gamma$. □

Proposition G.2 (Deterministic initial information)**.** *If $U = g(I)$ almost surely for measurable g, then for every $B \ge 0$,*

$$\delta_\Gamma^0 = \delta_\Gamma^B = \delta_\Gamma = 1.$$

Proof. The all-measurable policy class $\mathcal{P}_B^{\mathrm{all}}$ contains the zero-query policy $\tau = 0$ with retained state $Z_0 = g(I)$ and decoder $\widehat{a} = g(I) = U$ a.s. Hence $\sup_{\pi \in \mathcal{P}_B^{\mathrm{all}}} \mathbb{P}(\widehat{a}_\pi = U) = 1$ already at $B = 0$, so $\delta_\Gamma^0 = \delta_\Gamma^B = \delta_\Gamma = 1$ by the Bayes/TV identity of Proposition G.1. □

Theorem G.3 (Complete verifier-signature ceiling; full form of Theorem 4.1)**.** *Under the causal, no-side-channel interaction model, for every fixed T, budget B, and $\pi \in \mathcal{P}_B^T$, where for* (6) *we assume the correct-action sets $A^*(0)$ and $A^*(1)$ are disjoint (as in Proposition H.3; for binary U one may take $A^*(u) = \{u\}$),*

$$\| \operatorname{law}(H_{\tau_\pi} \mid U = 0) - \operatorname{law}(H_{\tau_\pi} \mid U = 1)\|_{\mathrm{TV}} \le \delta_\Gamma. \tag{5}$$

Therefore $\delta_B^T \le \delta_\Gamma$ and

$$\sup_{\pi \in \mathcal{P}_B^T} \mathbb{P}(\widehat{a}_\pi \in A^*(U)) \le (1 + \delta_\Gamma)/2. \tag{6}$$

The right-hand sides depend on neither the budget B nor the solver T.

Proof. At time zero the visible context is a function of $\Gamma_V(W)$. If the history through time t is generated from $\Gamma_V(W)$ and world-independent randomness, the causal policy's next query has the same property, its response is a coordinate of $\Gamma_V(W)$, and the fixed update rule preserves the induction. Thus H_{τ_π} is a U-independent garbling of Γ_V. Data processing proves (5); Proposition H.3 proves (6). □

The nontriviality condition $\delta_\Gamma < 1$ is a property of the world distribution, verifier, and target—not of the solver. The parity example below has $\delta_\Gamma = 1$, so it does not contradict the theorem.

Scope of the results in this appendix. This ceiling is a data-processing bound on the verifier's information, not a computational hardness theorem. It is uninformative for ordinary mathematical problems whose target U is recoverable from the complete signature Γ_V (for instance when U is a function of the input I), where $\delta_\Gamma = 1$. It is independent of the agent-side statistical lower bounds in §4. Proposition G.2 is not a corollary confined to a special case: it shows that neither a finite query budget nor pass/fail-only verifier responses can make the all-measurable policy class into a class of computationally bounded solvers whenever U is already a measurable function of the declared initial information I. Consequently, the upper bounds of this appendix (Theorem 4.1 and the associated Bayes/TV bounds) hold without any nontriviality assumption; they are simply uninformative when $U = g(I)$ almost surely for a measurable g. Our applications of δ_Γ^B, δ_B^T, δ_Γ as measures of what a checker interface can reveal are stated only for hidden-world query models in which $\mathbb{P}(U \ne g(I)) > 0$ for every measurable g, i.e. the declared I does not already reveal U; this is an applicability note, not a hypothesis of any theorem. Outside this regime the all-measurable class trivializes to $\delta_\Gamma = 1$ at zero budget, and no result in this appendix addresses ordinary public-input mathematics problems in which the target can already be recovered from the given input without

any query; a claim about computationally bounded solvers on such problems requires an additional, explicit computational restriction on the policy class that this appendix does not supply.

This ceiling constrains the verifier's information, not the agent; agent-side limits are the lower bounds of §4.

Proposition G.4 (Robust conflict pair)**.** *If w_+, w_- have different labels and identical complete signatures, then under their equal-prior distribution every finite budget has $\delta_B^T = 0$ and optimal success is at most $1/2$.*

Proof. The restricted signature distribution has $\delta_\Gamma = 0$, so Theorem 4.1 applies. □

Proposition G.5 (Discrete fiber criterion)**.** *For deterministic $\sigma : \mathcal{W} \to \mathcal{S}$ on finite or countable discrete spaces, the following are equivalent: a σ-fiber contains different labels; Φ is not constant on all fibers; and no g has $\Phi = g \circ \sigma$.*

Proof. The first two are definitional; a factor g is well-defined precisely when each fiber has one label. □

For general measurable spaces, fiber constancy alone need not give a measurable factor.

Corollary G.6 (Sparse-exception barrier)**.** *Let the world contain binary sequences and let the target be $\Phi(x) = \mathbf{1}\{\forall n,\ x_n = 1\}$. If the complete verifier signature is invariant under changes on a set of indices of upper natural density zero, then an all-one world and a world modified at one coordinate are a conflict pair. Under their equal-prior distribution, every finite-budget policy has success at most $1/2$.*

Proof. The two worlds differ on a singleton, hence have identical signatures by the assumption and opposite target labels. Proposition G.4 then applies. The invariance condition is essential: a finite-prefix interface need not be invariant under a density-zero perturbation. □

H HIDDEN-WORLD IDENTIFICATION: THE VERIFIER-SIGNATURE CEILING AND FINITE RECOVERY

This appendix retains, unchanged in substance, the identification analysis of earlier drafts: an interaction framework for a hidden world queried through a pass/fail checker (§H.1), the complete verifier-signature ceiling, Theorem M (§G), and the finite-budget tightness constructions and the finite-separator criterion for exact recovery, Theorem S (§H.2). Every statement here is an identity or a data-processing consequence of the declared protocol. The results are necessary conditions on any research procedure whose interaction with the checker is a garbling of the verifier signature; they do not bound the discovery event of Definition 2.4 directly, and their finite instances are audited in Appendix J. Their role is that of a tool: Theorem M is the ceiling used by Proposition H.12 below, and a history-TV quantity is not a surrogate for the discovery event, whose main evaluation is SD (Definition 2.4) and the bound of §C.1.

H.1 INTERACTION FRAMEWORK

Let $W \in \mathcal{W}$ be a hidden world and let $U \sim \mathrm{Bernoulli}(1/2)$ index distributions P_0, P_1, with target $\Phi(W) = U$ almost surely. We impose the following declared-interface assumptions: (A1) $\mathcal{Q}$ is countable, all spaces are standard Borel, and $(w, q) \mapsto r_W(q)$ is measurable; (A2) $I : \mathcal{W} \to \mathcal{I}$ is measurable and is the complete initial world-dependent information; (A3) the verifier is reset to its declared state before each query, and each candidate-proof query returns only $r_W(q) \in \{\mathsf{pass}, \mathsf{fail}\}$, with no trace, proof-state inspection, undeclared retrieval, or hidden state carried between replies; (A4) private randomness R is independent of (W, U); (A5) every query has unit cost $c(q) = 1$, budgets satisfy $B \in \mathbb{N}_0$, and $\tau_\pi \le B$; and (A6) the policy first forms a retained state from its complete history and then uses an independent decoder: $Z_{\tau_\pi} = \mathrm{Retain}(H_{\tau_\pi}, R)$ and $\widehat{a} = d(Z_{\tau_\pi}, R_{\mathrm{dec}})$, where $R_{\mathrm{dec}} \perp (W, U, H_{\tau_\pi}, R)$. Thus the declared verifier-state convention and readout rule exclude inspection of analyst-side data or unretained verifier state. A persistent or stateful verifier is outside

this protocol unless its initial state and transition rule are added to the signature and the theorem is re-instantiated.

The complete verifier signature is the analyst-side random object

$$\Gamma_V(W) := \big(I(W), (r_W(q))_{q\in\mathcal{Q}}\big), \qquad \delta_\Gamma := \|\operatorname{law}(\Gamma_V \mid U=0) - \operatorname{law}(\Gamma_V \mid U=1)\|_{\mathrm{TV}}. \quad (7)$$

It need not be observable in one run or finitely representable. It instead fixes the information boundary of the specified interface, against which a finite interaction policy is compared.

For fixed solver T and budget B, let $\mathcal{P}_B^T$ be its admissible policy class. A policy π stops at $\tau_\pi \le B$ and produces the full tagged history and retained state

$$H_{\tau_\pi} = \big(I, ((q_t, r_W(q_t)))_{t=1}^{\tau_\pi}\big), \qquad Z_{\tau_\pi} = \mathrm{Retain}(H_{\tau_\pi}, R). \quad (8)$$

The tags preserve which query an individual verifier response answers. We use "at most B", so $\mathcal{P}_B^T \subseteq \mathcal{P}_{B'}^T$ for $B \le B'$. Define the history separation

$$\delta_B^T := \sup_{\pi\in\mathcal{P}_B^T} \|\operatorname{law}(H_{\tau_\pi} \mid U=0) - \operatorname{law}(H_{\tau_\pi} \mid U=1)\|_{\mathrm{TV}}. \quad (9)$$

For a fixed policy, $\mathrm{TV}(\operatorname{law}(Z_{\tau_\pi} \mid 0), \operatorname{law}(Z_{\tau_\pi} \mid 1))$ and $\mathbb{P}(\widehat{a} = U)$ are reported separately; the latter is bounded by the former through the declared decoder.

For the interface rather than a particular solver, let $\mathcal{P}_B^{\mathrm{all}}$ be all policies satisfying (A1)–(A6), and define

$$\delta_\Gamma^B := \sup_{\pi\in\mathcal{P}_B^{\mathrm{all}}} \|\operatorname{law}(H_{\tau_\pi} \mid U=0) - \operatorname{law}(H_{\tau_\pi} \mid U=1)\|_{\mathrm{TV}}.$$

Hence $\delta_B^T \le \delta_\Gamma^B$, and δ_Γ^B is nondecreasing in B.

Proposition H.1 (History TV; auxiliary to Theorem 4.1)**.** *(a) Under the hypotheses of Theorem 4.1, every $\pi \in \mathcal{P}_B^T$ satisfies $\mathrm{TV}(\operatorname{law}(H_\pi \mid U=0), \operatorname{law}(H_\pi \mid U=1)) \le \delta_\Gamma$, hence $\delta_B^T \le \delta_\Gamma$. (b) If $U = g(I)$ almost surely for a measurable g and H_π contains I, then this TV equals 1 for every π, so $\delta_B^T = 1$ for every nonempty class regardless of computational restrictions.*

Proof. (a) H_π is generated from Γ by a kernel common to both environments, so data processing gives the bound; take the supremum for δ_B^T. (b) The event $\{g(I) = 0\}$ is H_π-measurable and has probability 1 under $U = 0$ and 0 under $U = 1$. History TV is therefore auxiliary: it bounds, but does not replace, the primary quantity SP_B^T of (1). □

H.2 Finite budgets, tightness, and recovery

Lemma H.2 (Retention is a Markov kernel from history to state)**.** *Under (A1)–(A6), and for* discrete *declared information I and response spaces (so that $\mathbb{P}(H_{\tau_\pi} = h, U = u) > 0$ is meaningful at every atom of the history law; the continuous case would require regular conditional distributions and is not proved here), fix $\pi \in \mathcal{P}_B^T$. Then $R \perp U \mid H_{\tau_\pi}$, and hence $Z_{\tau_\pi} = \mathrm{Retain}(H_{\tau_\pi}, R)$ is obtained from H_{τ_π} by a Markov kernel not depending on U.*

Proof. Write a tagged terminal transcript as $h = (i, s)$ with $s = ((q_1, r_1), \dots, (q_k, r_k))$, $k \le B$. Because π is causal – each query and the stopping decision are measurable functions of (H_{t-1}, R) – and each response $r_W(q_t)$ is a deterministic function of W (A1, A3), the event $\{H_{\tau_\pi} = h\}$ factors, for $(W, R) = (w, r)$, as

$$\mathbf{1}\{H_{\tau_\pi}(w, r) = h\} = \mathbf{1}_{C_h}(w) \cdot \mathbf{1}_{D_h(i)}(r),$$

where C_h is the (policy-independent) event that w's declared information and query responses match s coordinatewise, and $D_h(i)$ is the event that, started from initial information i, the private randomness drives the deterministic causal policy through exactly the query and stopping sequence recorded in s; $D_h(i)$ depends on r and i only, since once C_h pins down every response, the policy's remaining behavior is a fixed function of (i, r). Let $\rho = \operatorname{law}(R)$, independent of (W, U) by A4. For any measurable A and u with $\mathbb{P}(H_{\tau_\pi} = h, U = u) > 0$,

$$\mathbb{P}(R \in A \mid H_{\tau_\pi} = h, U = u) = \frac{\mathbb{P}(R \in A,\, C_h,\, D_h(i) \mid U = u)}{\mathbb{P}(C_h,\, D_h(i) \mid U = u)} = \frac{\rho(A \cap D_h(i))}{\rho(D_h(i))},$$

where $R \perp (W, U)$ lets the R-measurable factor $D_h(i)$ (fixed once h, hence i, is fixed) be pulled out of the conditioning on (W, U); on the complementary null event the kernel is set arbitrarily. The right-hand side depends on h only through i and $D_h(i)$, not on u, so $R \perp U \mid H_{\tau_\pi}$. Since $Z_{\tau_\pi} = \mathrm{Retain}(h, R)$ is a fixed measurable function of (h, R), pushing this kernel forward through $\mathrm{Retain}(h, \cdot)$ gives $\mathrm{law}(Z_{\tau_\pi} \mid H_{\tau_\pi} = h, U = u)$ not depending on u. □

Proposition H.3 (Finite-budget TV bound)**.** *Under the discrete-information and response-space condition of Lemma H.2, together with (A1)–(A6), if correct-action sets $A^*(0)$ and $A^*(1)$ are disjoint, then every finite B satisfies*

$$\sup_{\pi \in \mathcal{P}_B^T} \mathbb{P}(\widehat{a}_\pi \in A^*(U)) \le \frac{1 + \delta_B^T}{2}. \tag{10}$$

This is a bound on the declared binary correct-action event within that hidden-world model; it neither estimates nor bounds the frozen held-out utility or discovery event unless an additional identification from those quantities to this event has been established.

Proof. For fixed π, independent decoding is a Markov kernel from Z_{τ_π} to $\widehat{a}_\pi$ by (A6), and retention is a Markov kernel from H_{τ_π} to Z_{τ_π} by Lemma H.2. Data processing bounds output TV by the history TV in (9). If ν_u is the output law under $U = u$, disjointness gives $2\mathbb{P}(\mathrm{success}) \le 1 + \|\nu_0 - \nu_1\|_{\mathrm{TV}}$. Take the supremum. □

Remark H.4 (Fixed weights are not a ceiling). Let X, U be independent fair bits, $Y = X \oplus U$, $W = (X, Y)$, and $\Phi(W) = X \oplus Y$. Queries q_X, q_Y return X, Y. Take I constant. One query gives a fair bit independent of U, hence $\delta_1^T = 0$; two queries reveal the parity, hence $\delta_\Gamma^2 = 1$ (the budget-2 ceiling over the all-measurable class; for a fixed solver T one has $\delta_2^T = 1$ only if T can issue queries on two distinct coordinates, e.g. q_X then q_Y, and $\delta_2^T = 0$ if it can only repeat q_X). The weights and verifier did not change.

Proposition H.5 (Query simulation and budget-limited access)**.** *Consider two interfaces on the same world distribution, with the same initial information and unrestricted causal policies. Suppose the new interface contains every old query, and every new query can be simulated, for every world, by a measurable causal procedure using at most $k \ge 1$ old queries. Both interfaces have unit query costs. Then*

$$\delta_{\mathrm{old}}^B \le \delta_{\mathrm{new}}^B \le \delta_{\mathrm{old}}^{kB}.$$

Their complete signatures are equivalent as information: each is a measurable function of the other.

Proof. An old policy is also available through the new interface, giving the first inequality. Replace each query of a new budget-B policy by its old-interface simulation. The resulting old policy uses at most kB queries, and its history determines the simulated new history. Data processing therefore gives the second inequality. Old responses are coordinates of the new signature. Conversely, each new response is determined measurably by the old signature through its simulation. Since the query families are countable, this determines the new complete signature measurably. □

A signature-redundant shortcut can improve finite-budget access. In the parity construction of Remark H.4, add a unit-cost query $q_\oplus$ returning $X \oplus Y$. This response is already a function of the old complete signature, so the complete-signature TV remains 1. Nevertheless,

$$\delta_{\mathrm{old}}^1 = 0, \qquad \delta_{\mathrm{new}}^1 = 1.$$

The new query is simulated by the two old coordinate queries. Thus complete-signature redundancy does not imply redundancy at a fixed query budget. This is an exact finite-model construction, not an empirical claim about deployed solvers.

Unified tightness construction. For $\delta \in [0, 1]$, let $N \sim \mathrm{Bernoulli}((1 - \delta)/2)$ independently of U, set $Z = U \oplus N$, and take $W = (U, N)$ with constant initial information. Let special query q_Z return Z, while every other verifier response is a function of Z alone. With full history retained and readout $\widehat{a} = Z$, for $B \ge 1$, history TV and retained-state TV are δ and $\mathbb{P}(\widehat{a} = U) = (1 + \delta)/2$. If only the readout is made constant, history TV and retained-state TV remain δ while fixed-readout success becomes $1/2$; if retention is made constant, retained-state TV becomes 0 (history TV remains δ) and success is again $1/2$. This is an existence construction, not a claim about arbitrary deployed solvers.

Proposition H.6 (Fixed-distribution finite recovery)**.** *Let I be constant and unit-cost query q_n return coordinate x_n. Under $U = 0$, set $x_n = 1$ for all n. Under $U = 1$, draw J with $\mathbb{P}(J = j) = 2^{-j}$ and set $x_J = 0$, $x_n = 1$ otherwise. With all histories and arbitrary binary readouts admissible,*

$$\delta_\Gamma = 1, \qquad \delta_\Gamma^B = 1 - 2^{-B}, \qquad \mathrm{Success}_B^* = 1 - 2^{-(B+1)}.$$

Proof. On its all-one path a deterministic budget-B policy reads a set A of at most B coordinates. Under $U = 1$, it has that same transcript precisely with probability $1 - \sum_{j\in A} 2^{-j}$, so TV is $\sum_{j\in A} 2^{-j}$ and is maximized at $A = \{1, \dots, B\}$. Randomized policies are mixtures and cannot exceed this value. The complete signature detects the unique zero. Equal-prior Bayes success is $(1 + \delta_\Gamma^B)/2$. □

Remark H.7 (No saturation prediction). This is one fixed distribution, and $\delta_\Gamma^B \uparrow 1$ without finite attainment. Thus we withdraw any unqualified saturation prediction: finite equality requires an additional screening–recoverability condition.

For a deterministic adaptive query policy κ of depth at most B, write $O_B^\kappa(W)$ for its tagged terminal observation. The interface is observationally complete if $\Phi = g \circ O_B^\kappa$ for some κ, g; it is operationally complete if the composite "run κ, output g" lies in $\mathcal{P}_B^T$.

Lemma H.8 (Finite separator and recovery)**.** *For discrete observation spaces, observational completeness at finite budget is equivalent to a finite-depth admissible adaptive query tree whose terminal-observation fibers are label-homogeneous. If the readout is admissible, it is also equivalent to operational completeness; exact success is then attainable at all larger budgets.*

Proof. A factor g makes every observation fiber homogeneous. Conversely define g by each reachable fiber's common label and arbitrarily elsewhere. Readout admissibility yields the policy; the upper-bound budget convention embeds it at larger budgets. □

Remark H.9 (Recovery is per-world, not almost-sure). Label-homogeneity of a fiber is a property of the deterministic maps Φ and O_B^κ on the declared world space $\mathcal{W}$: a fiber is label-homogeneous exactly when *every* world it contains shares one label, with no exception, not merely a probability-one subset of them. A fiber that contains worlds of opposite label, one of which has probability zero under the declared distribution, is *not* label-homogeneous by this definition, and Lemma H.8 grants no recovery guarantee for it; only $\mathbb{P}(\widehat{a}_\pi = U) = 1$ would hold in that case, which this appendix does not identify with the exact, per-world recovery Theorem S concerns.

The necessary pairwise separation condition is (Sep): opposite-label worlds with identical initial information differ on some admissible query. A usable uniform finite-separator condition is (UF): there is a finite $Q_0 \subseteq \mathcal{Q}$ whose queries separate every opposite-label pair. Querying Q_0 yields finite recovery when executable.

Proposition H.10 (Conditional uniform saturation)**.** *Suppose a statistic Z is recoverable by an admissible policy at budget $B_\star$, and every admissible terminal observation at every $B \geq B_\star$ is a world-independent garbling of Z. Then*

$$\delta_B^T = \| \mathrm{law}(Z \mid U = 0) - \mathrm{law}(Z \mid U = 1) \|_{\mathrm{TV}} \quad (B \geq B_\star).$$

Proof. The assumed garbling gives the upper bound by data processing, while the recovering policy gives the reverse bound. Thus saturation requires explicit screening and recoverability, rather than fixed solver weights alone. □

Remark H.11 (Pairwise versus uniform separation). Let $\mathcal{W} = \{0, 1\}^{\mathbb{N}}$, query q_n reveal x_n, and $\Phi(x) = \mathbf{1}\{\forall n, x_n = 1\}$. Every negative sequence is separable from the all-one sequence by a suitable q_n, but no finite tree separates it from every negative sequence. Thus pairwise separation does not imply a uniform finite budget. Here $\delta_\Gamma = 1$ can hold even though no finite budget attains the signature ceiling.

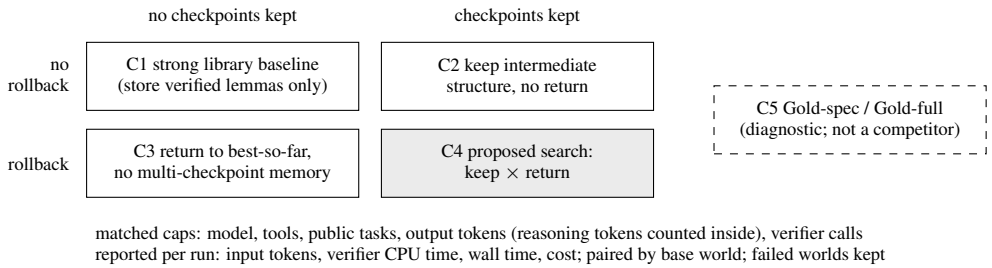


Figure 7: **The main comparison isolates two interventions – keeping intermediate structure and returning from dead ends – under matched generation and verifier budgets.** Conditions C1–C4 form a 2×2 design in which only the two capabilities differ; C5 supplies the structure (specification only, or with proofs) to diagnose where difficulty lies and is excluded from cost comparisons.

H.3 An upper bound on discoverability from the ceiling

The next statement is an auxiliary reduction, not a main result: it is a direct consequence of the definitions and of Theorem M, and it holds only under two environment-design hypotheses that an experiment must verify for its own environments. Let c_0, c_1 be environments sharing (K_0, S, V) and differing in the sealed set, realized as $c = c_U$ for the hidden label $U \sim \mathrm{Bernoulli}(1/2)$ of §H.1. For a threshold $\gamma > 0$ let $Z_u^*(\gamma) = \{z : \mathrm{Cert}_{c_u}(z) = 1,\ U_b(z; c_u) \geq \gamma\}$.

(H-disjoint) $Z_0^*(\gamma) \cap Z_1^*(\gamma) = \varnothing$; for instance $\sup_z \min\{U_b(z; c_0), U_b(z; c_1)\} < \gamma$, or $\mathrm{Cert}_{c_0}(z) = 1 \Rightarrow \mathrm{Cert}_{c_1}(z) = 0$.

(H-garble) The research procedure's entire interaction with the environment is a garbling of the verifier signature $\Gamma_V(W)$ together with randomness independent of (W, U), as in (A1)–(A6) of §H.1.

Proposition H.12 (Conditional information bound on discoverability)**.** *Assume the successful-artifact sets are measurable and satisfy (H-disjoint). Assume the output law conditional on $U = u$ equals the research procedure's output law in environment c_u. Under (H-garble),*

$$\frac{\mathrm{SD}_{B,b}^T(\gamma; c_0) + \mathrm{SD}_{B,b}^T(\gamma; c_1)}{2} \leq \frac{1 + \delta_\Gamma}{2}.$$

No closure of the solver's policy class under utility-based decoding is asserted.

Proof. Let ν_u be the artifact-output law conditional on $U = u$, and let D_u be its measurable successful-artifact set in environment c_u. Disjointness implies

$$\nu_0(D_0) + \nu_1(D_1) \leq 1 + \|\nu_0 - \nu_1\|_{\mathrm{TV}}.$$

Under (H-garble), a single Markov kernel, independent of the world and label, maps the complete signature to the artifact-output law. Data processing therefore gives $\|\nu_0 - \nu_1\|_{\mathrm{TV}} \leq \delta_\Gamma$. Dividing by two proves the claim. □

The right-hand side is at least $1/2$, so the bound is informative only when δ_Γ is small, and (H-disjoint), (H-garble) are strong design assumptions; we have not verified them for the environment pairs of §5. The proposition therefore states that the identification ceiling is a necessary condition on discoverability for environment pairs satisfying them; it is not a rate estimate.

I Protocol design figures (E2, E3)

J Supporting proofs and finite audits

J.1 Discussion boundaries and disclosure rules

What this version shows. It fixes the object – a certified, frozen artifact scored by held-out utility on sealed theorems – and separates it from local correctness and from accepting-vertex reachability.

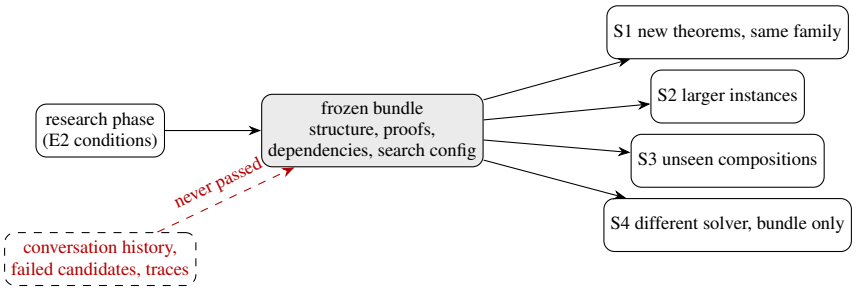


Figure 8: **Transfer is tested only through the frozen artifact: new theorems, larger sizes, unseen compositions, and a different solver that never sees the conversation.** The planned protocol requires the bundle to be hashed and committed before the sealed tests are opened; the different-solver arm receives the bundle and the test statement only, and any prompt containing research-phase history is excluded as a leak.

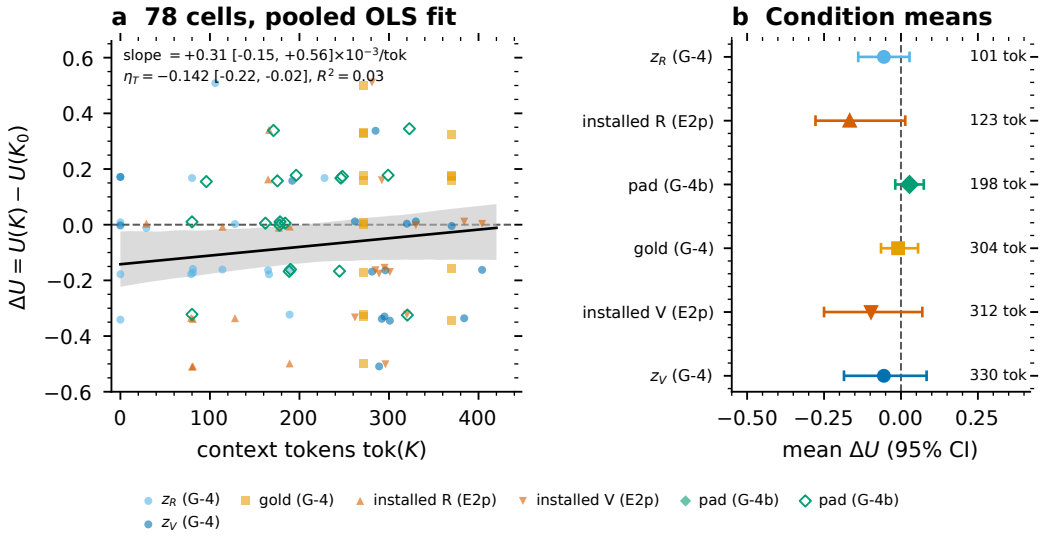


Figure 9: **Descriptive association of ΔU with context tokens $\mathrm{tok}(K)$, not an estimate of baseline-horizon burden.** (a) Each marker is one non-K_0 cell (G-4: z_R, z_V, gold; E2p: installed R, V); the black line is a pooled descriptive OLS fit with world-cluster bootstrap band (95%, 2,000 resamples). The dashed line marks $\Delta U = 0$; open diamonds are G-4b padding cells outside the fit. (b) Condition means with world-cluster 95% intervals. The slope interval contains zero. No paired K_0 horizon intervention or measured token-to-call conversion is available here.

Here "utility" means the population quantity $U_b(z; c)$ of Definition 2.2: for one declared family, base library, fixed solver and proof budget, checker, and sealed finite test distribution, it is the difference in checker-accepted solve probabilities with and without the frozen installation. It is not a score inferred from an accepting trace, a verifier-signature audit, or an unsealed development theorem. The verification-gap bound of §3 constrains traces but cannot by itself distinguish an environment where discovery is nearly certain from one where it is rare; the quantities that would (discovery and system success under the two interventions, transfer when frozen, and accounted cost) are those of the committed protocols in §5, and they are not measured here.

Identification is necessary, not sufficient. Appendix H retains the hidden-world analysis: for a declared pass/fail interface, no admissible policy exceeds the ceiling set by the complete verifier signature, and for discrete observation spaces, exact finite-budget recovery is equivalent to a finite label-homogeneous separator (Lemma H.8). These statements bound what any interaction with the checker can reveal; they are identities and data-processing consequences of the declared protocol, and a history-TV quantity is not a surrogate for the discovery event of Definition 2.4. In particular, if the target is already a measurable function of the public input, the identification quantities trivialize at zero

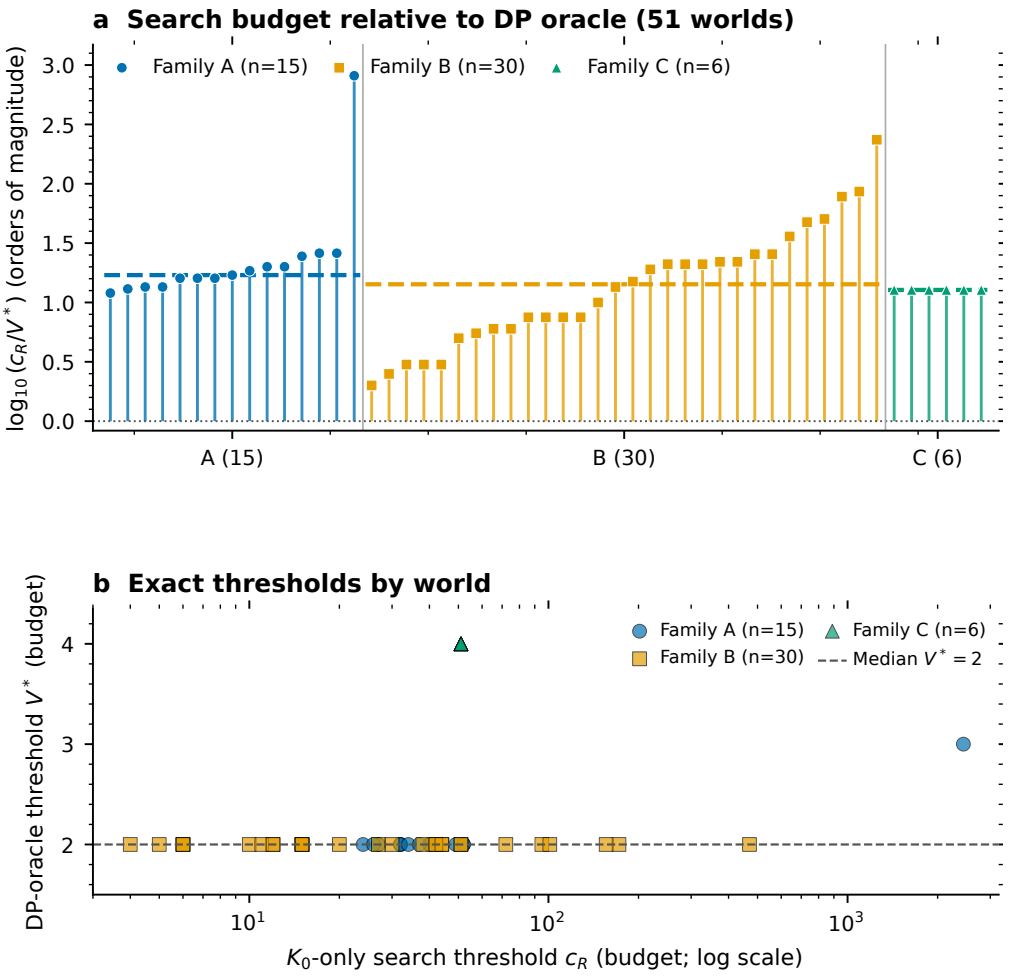


Figure 10: **Finite-graph threshold proxy, not an LLM solve-rate result.** (a) Each lollipop is one world, grouped by family and ordered by c_R within family. Its height is $\log_{10}(c_R/V^*)$, where c_R is the K_0-only rollback-search threshold and V^* is the DP-oracle threshold; zero denotes equal budgets. Dashed segments are within-family medians: A 1.23 ($n = 15$), B 1.15 ($n = 30$), C 1.11 ($n = 6$). (b) The same worlds show c_R on a log axis (4–2439) against V^* on a linear axis (2–4); the dashed line is the pooled V^* median of 2. The reference is a DP oracle, not a supplied certified structure.

budget (Proposition G.2); the main evaluation is therefore placed on actual generation, verification, and use, not on history distinguishability.

Disclosure and negative results. The protocols specify release of full success-versus-budget curves, per-run cost columns, retained failed worlds with fallback outcomes, and the prespecified NOGO conditions under which each claim is reduced, so that the empirical boundary is falsifiable, including when added budget or an intervention yields no detectable improvement.

J.2 Exploratory frozen-artifact pilot and paired audit

Family B exploratory pilot: frozen-artifact utility ($n = 4$ worlds). A four-world Family B pilot (`fB1–fB4`; 8 solver-instances per condition; exploratory, excluded from the confirmatory analysis by preregistered rule) directly measures whether a frozen artifact improves held-out verified-solve rate. Under matched generation-token and verifier-call budgets, the condition that installs the discovered frozen bundle ($K_0 \oplus z_V$) achieves a mean sealed-test verified success rate of $\mathbf{0.3333}$ ($n = 8$) versus $\mathbf{0.2917}$ for the baseline condition C1 (fB1–fB4 only; C1 has accepted lemmas in some cells and is not a K_0-only control). (The earlier values $0.1875/0.0625$ and the verifier-call

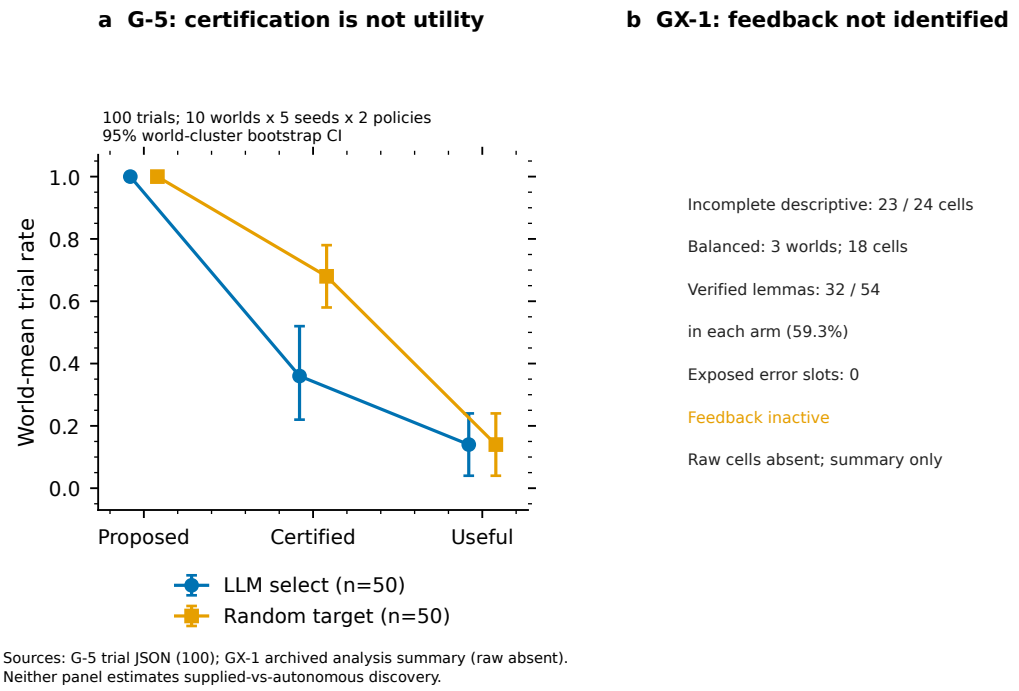


Figure 11: **Exploratory stage-gate diagnostics, not the E2 estimand.** (a) G-5 trial rates for LLM-selected and random targets over 10 worlds and five seeds per policy; bars show 95% world-cluster bootstrap intervals. Certification exceeds measured usefulness, particularly for random targets. (b) GX-1 recovery status from an archived summary only: 23/24 planned cells completed and no error feedback was exposed. The raw cells are unavailable here, so panel (b) is an audit annotation, not an effect size. Neither panel measures C4–C1 frozen-artifact utility.

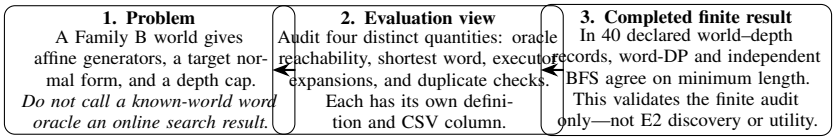


Figure 12: **Family B in three panels: the finite problem, the measurement boundary, and the completed audit.** The result is intentionally narrower than a discovery claim: it reports agreement of two known-world shortest-word calculations, while executor cost remains a separately recorded operational quantity.

means 18.25 vs. 6 did not reproduce from the archived rows and are withdrawn; no verifier-call comparison is claimed.) These are exploratory estimates; they set the confirmatory sample size and are not to be interpreted as confirmatory evidence. All cells and per-world raw values are archived at `experiments/e2_prereg/results/e2p-familyb-frozen-artifact-utility-20260921/`.

Frozen-artifact utility audit, paired design ($n = 12$ pairs): null result. A separate, independently re-aggregated audit measures frozen-artifact utility under a same-world, same-seed, identical-budget-ceiling paired design (condition R, library-only, vs. condition V, installed frozen bundle), with the library frozen (SHA-256 hash and freeze timestamp recorded) before the sealed solve. Across $n = 12$ pairs the mean paired difference in sealed-test verified success rate is $\overline{\Delta}_{sp} = \mathbf{-0.0833}$ (sd 0.1812; 2 pairs positive, 5 negative, 5 tied; exact two-sided sign-flip test over pairs $p = 0.2188$; with the 4 base worlds as the independent unit the world-level test gives $p = 0.25$), i.e. non-significant and directionally non-positive. A second, confirmatory-style set of 12 pairs (24 cells) gives a consistent picture ($\overline{\Delta}_{sp} = -0.0417$; pair-level $p = 0.7793$, world-level $p = 1.0$), though its cells carry no recorded freeze hash or timestamp and so cannot be individually audited for freeze validity. Both point estimates are non-positive; neither

supports a beneficial frozen-artifact utility effect. This audit is exploratory (consistent with the Family B pilot above) and is reported in the same format as a positive result would be, per the disclosure norm stated at the top of this section. Full per-pair rows, provenance checks (hash rehash, freeze-window, budget-ceiling equality), and the disclosed audit limits are archived at `results/galois-e2z1-frozen-artifact-evidence-bundle-20260923/summary.json` (10/10 checks pass).

Reaggregation and control-identity checks. The E2z3 audit recomputes the eight R and eight V pilot cells from `experiments/e2_prereg/results/e2p_fB*_{R,V}_s{1,2}.json`. It obtains mean sealed solve rates R $= 0.2917$ and V $= 0.3333$ and mean discovery verifier calls R $= 6$ and V $= 17.875$. Of the four previously quoted quantities (solve rates $0.0625/0.1875$, verifier calls $6/18.25$), only the R verifier-call mean reproduces. Five of eight R cells contain accepted lemmas, so R is not an empty-library K_0 control. These are the exact subset and values used in the exploratory paragraph above; they do not estimate the effect of installing a particular artifact against its own empty-library counterfactual. The archived E2z3 receipt is `results/e2z3-smoke-20260924/smoke_receipt.json`; it is a read-only reaggregation, not a new model experiment.

The independent E2z2 world-cluster audit of the twelve paired frozen-library comparisons reports world-mean V-minus-R differences -0.0833 (exact four-world sign-flip $p = 0.25$) and -0.0417 ($p = 1.0$) in the two archived sets. Its identity check finds zero cells with a recorded K_0/installed key, zero R–V pairs sharing an identical frozen-library hash, and zero pairs with an empty library. Consequently R–V is not the installed-versus-K_0 contrast; the distinct 24-cell experiment in §5 supplies that contrast. Source: `experiments/e2_prereg/e2z2_world_cluster_k0_identity_audit.py` and its 24 paired input cells. A separate T2z1 verification-gap recomputation is listed in the historical integration ledger, but its primary result file is absent from this repository snapshot; no numerical T2z1 verification claim is made here.

Inventory of archived finite checks. For completeness, the repository also retains finite, mostly seed-free cross-checks that support the implementation audit rather than the held-out utility estimate. The primary files are indexed, with exact paths and SHA-256 hashes, in `EXPERIMENT_AUDIT_20260925.md`. E1b boundary and rollback records contain 11 cases each and are represented by the finite-state discussion and table above. E1c dedup dominance has 360 CSV rows; E1d hardness sweep has 432; E1e state-versus-candidate and E1f order invariance have 144 each; E1g hidden-world exact has 23; E1h independent driver, E1i re-encounter, E1m terminal boundary, and E1p candidate specificity have 40 each; E1t budget-cost reconciliation has 240; E1v interface-cost-TV has 20; and E1x readout-versus-Bayes identity has 39. E1o disagreement, E1q depth monotonicity, E1r cross-driver minimum length, and E1w discovery-versus-usage cost record 40, 10, 40, and 40 rows respectively. E1z4 and E1z5 each cover 40 finite combinations. These counts come from the archived CSV rows or summary keys, not new executions. Several files with `smoke` in the run name contain similar row counts but are launcher or implementation checks, not independent experimental replications. None of these finite checks is substituted for a sealed held-out utility observation or a deployed-LLM interface test.

Proof of Remark C.3. A successful trace ends at an accepting vertex, so its dependency record contains a root-to-accepting path π. By the definition of $g(\pi)$, that path has a checkpoint-to-checkpoint stretch of $g(\pi)$ operations. Since $w^\star(c)$ is the minimum of g over all such paths, $g(\pi) \geq w^\star(c)$. Equality requires a minimizing path. By Definition 3.1, a correct but unhelpful vertex has no eligible continuation to an accepting vertex under the declared remaining resources, so it cannot supply a competing accepting path. □

J.3 Discovery and usage cost audit

Separating discovery cost from usage cost (an E1w audit). On the ten declared Family B worlds crossed with extension length 1–4 (forty world/length combinations), an independent full-state-space breadth-first search – a fresh implementation that does not import or re-run the existing enumeration code – separates three quantities: the true shortest-path discovery cost of a target T_1, the true shortest-path discovery cost of a downstream target T_2 reached without retention, and the marginal usage cost

of extending a retained T_1 to T_2. The retained total cost, $\mathrm{disc}(T_1) + \mathrm{use}(T_1, T_2)$, is strictly greater than the direct discovery cost $\mathrm{disc}(T_2)$ on 10 of the 40 combinations (maximum separation 3, mean separation 0.375 over all forty); on the remaining 30 combinations retention lies on a shortest path, so the two costs coincide. This shows concretely that Remark C.2's claim – that retention lowers a specific accounted attempt sequence – is distinct from, and does not establish, the stronger claim that retention lowers the true shortest-path cost under an optimal oracle: on these 10 combinations the stronger claim is false. This is a purely combinatorial cost-accounting audit of the declared finite Family B word graphs; it is not a utility or confirmatory E2 result, and it makes no claim about LLM search behavior.

Forced-waypoint cost is not an optional-retention effect. Fix one finite directed graph of permitted constructions, with unit cost per extension, a source r, and targets u and v. Let $D(x, y)$ denote the shortest directed-path length from x to y. Whenever the three distances are finite, concatenating shortest paths gives

$$D(r, v) \le D(r, u) + D(u, v).$$

A strict inequality measures the detour imposed by requiring a route through u. It does not show that making retention available increases optimal cost. If a retention-enabled policy class contains every baseline policy and preserves its action costs, its optimal cost $C_1(r, v)$ and the baseline optimal cost $C_0(r, v)$ satisfy

$$C_1(r, v) \le C_0(r, v),$$

because the additional capability may be ignored. All distances in a depth-limited comparison must use the same depth-augmented graph. Shortest-word distance is not online discovery work, and this comparison establishes neither held-out utility nor an empirical retention advantage.

J.4 Family B transition instance and word-oracle audit

Family B finite transition instance. For the repository-recorded Family B instance, a world fixes a modulus n, generators $f_i(x) = a_i x + b_i$ for $i \in [m]$, a target normal form $(A^\star, B^\star)$, and a word cap d. The artifact graph has one vertex for every word $w = i_1 \cdots i_j$, $0 \le j \le d$, and an edge $w \to wi$ for each $i \in [m]$. It is therefore a dependency tree of words, not an execution trace. Its label is computed from the normal form $N(\epsilon) = (1, 0)$ and

$$N(wi) = (a_i A \bmod n,\ a_i B + b_i \bmod n) \quad \text{when } N(w) = (A, B).$$

Every Family B word is well typed; it is terminal exactly when $N(w) = (A^\star, B^\star)$, and otherwise is a checked, nonterminal candidate. Thus a failed check is not silently deleted and does not by itself imply that the word has no successful continuation.

The operational rollback executor is the following finite labelled transition system, whose state is deliberately richer than a word. Write $\mathcal{W}_d = \{w \in [m]^{\le d}\}$, let a frame be $(w, J) \in \mathcal{W}_d \times 2^{[m]}$, and let a stack be a nonempty sequence

$$R = [(w_0, J_0), \ldots, (w_r, J_r)] \quad \text{with} \quad w_0 = \epsilon, \quad w_{k+1} = w_k i_k, \quad |w_k| = k.$$

The legal state space is

$$\mathcal{X} = \{(R, b) : R \text{ is such a stack},\ b \in \{0, \ldots, B\}\} \cup \{\mathsf{success}(w), \mathsf{failure}\}.$$

The initial state is $X_0 = ([(\epsilon, \varnothing)], B)$. The initial word is checked at zero cost; it immediately enters $\mathsf{success}(\epsilon)$ when it is terminal. Otherwise the following transition rules, and only these rules, apply to a nonabsorbing state $X = (R, b)$ with top frame (w, J):

$$\begin{aligned}
\mathsf{extend}_i: \quad & (R, b) \\
& \longmapsto \begin{cases} \mathsf{success}(wi), & N(wi) = (A^\star, B^\star), \\ (R[(w, J) \mapsto (w, J \cup \{i\})] :: [(wi, \varnothing)], b - 1), & N(wi) \ne (A^\star, B^\star), \end{cases} \\
& \qquad i \notin J, \quad |w| < d, \quad b > 0; \qquad \text{(E)} \\
\mathsf{backtrack}: \quad & [(w_0, J_0), \ldots, (w_r, J_r)], b \longmapsto [(w_0, J_0), \ldots, (w_{r-1}, J_{r-1})], b, \\
& \qquad r > 0; \qquad \text{(B)} \\
\mathsf{halt}: \quad & (R, b) \longmapsto \mathsf{failure}. \qquad \text{(H)}
\end{aligned}$$

Here $R[(w, J) \mapsto (w, J \cup \{i\})]$ changes only the top frame, and :: appends a frame. Rule (E) costs one node-expansion/check call, including both successful and failed checks; (B) and (H) cost zero. A failed check is precisely the second branch of (E): the checked word is retained as a new frame rather than deleted, so it may be extended later; (B) discards only the current retained frame and resumes the earlier prefix with its already-tried set intact. At a depth cap, zero budget, or an exhausted top frame, extension is illegal but (B), when available, and (H) remain available. Terminal states have no legal action. The readout is $\mathrm{out}(\mathsf{success}(w)) = w$ and $\mathrm{out}(\mathsf{failure}) = \bot$; thus the output, retained history R, and accumulated cost $B - b$ are distinct objects. Finiteness is immediate: m, d, B are finite, frames range over finite sets, and stack length is at most $d + 1$.

This is not the word oracle in operational notation. For example, if a root has two generators, one nonterminal child and one terminal child, with one expansion available, the oracle below has value $V(\epsilon, 1) = 1$. The transition system nevertheless permits a policy to spend its sole unit on the nonterminal child and then return failure, while another policy selects the terminal child and returns its word. More generally its legal continuation set depends on the retained stack and the sets J_k, which $V(w, b)$ does not contain. Hence oracle reachability, an executor policy's final output, and the executor's action cost are explicitly non-equivalent quantities.

A separate word oracle. For comparison only, define $V(w, b) \in \{0, 1\}$ by

$$V(w,b) = \begin{cases} 1, & N(w) = (A^\star, B^\star), \\ 0, & N(w) \neq (A^\star, B^\star) \text{ and } (b = 0 \text{ or } |w| = d), \\ \max_{i \in [m]} V(wi, b-1), & \text{otherwise.} \end{cases}$$

This is an oracle reachability recurrence: its value is one exactly if a terminal descendant of w of additional length at most b exists. The assertion follows by induction on $\min\{b, d - |w|\}$: the first two lines are the terminal and boundary cases, and every positive-length descendant has a unique first generator, giving the third line. It charges one unit per word extension, not the cost of an online search. In particular, it omits the executor's retained ancestors and tried sets, so it is neither a compression of that executor nor an optimality statement for it. `experiments/e1_enum/e1_enumerate.py` and `e1_dp.py` implement this finite-word calculation; their output must be labelled oracle reachability rather than discovery or rollback cost.

Proposition J.1 (Family B word-DP audit identity)**.** *For a completely enumerated declared Family B world, the recurrence $V(w, b)$ is one exactly when the independently enumerated word tree contains a terminal descendant of w within the residual cap b. Consequently, the DP and breadth-first minimum-word calculation agree whenever both return a finite terminal length.*

Proof. The terminal and boundary clauses give the base cases. Every nonempty terminal descendant has a unique first generator, so induction on the remaining depth proves the recurrence; breadth-first enumeration visits the same word tree in increasing length. □

Proof of Remark B.1. Add and subtract $\max_{a \in C(s)} Q^\star(s, a)$. □

Scope and protocol. These are finite-model implementation checks of the constructions in Appendix H, not empirical evidence for the theorems, not an evaluation of an LLM, and not part of the main experiments of §5. They are numbered Audit 1–3 to keep them distinct from the preregistered protocols E2 and E3.

Exact audit. The primary exact-audit record is `experiments/exact_results.csv`, regenerated by `python3 experiments/exact_audit.py` using rational arithmetic. It reports, in separate columns, $\mathrm{TV}(H_\tau \mid 0, 1)$, $\mathrm{TV}(Z_\tau \mid 0, 1)$, fixed-readout success, and Bayes success from history. For the noisy bit with $\delta = 1/2$ and constant retention it asserts $(1/2, 0, 1/2, 3/4)$. It also enumerates the eight depth-two parity decision trees, obtaining $\{0, 0, 1/2, 1/2, 1/2, 1/2, 1, 1\}$, and checks their maximum against the finite Bayes-error dynamic program. The Monte Carlo ledger below is retained only as a secondary sampling-error audit; plug-in TV intervals are not used as confidence intervals for the exact quantity.

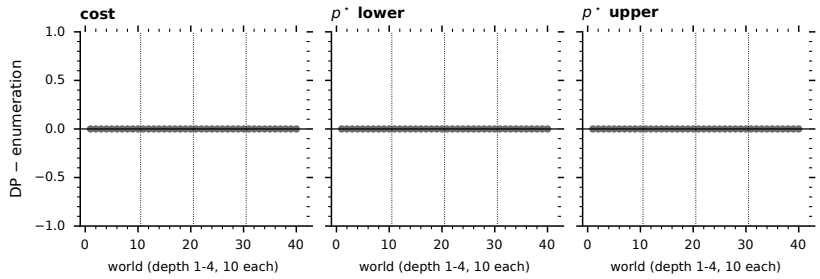


Figure 13: **State DP and independent enumeration differ by exactly zero in every one of the 40 Family B worlds.** Each point is one world (depth 1–4, ten per depth); the vertical axis is DP − enumeration for cost to the nearest terminal and for the lower and upper bounds of $p^\star$ at the reference budget. All worlds are enumerated to completion. Computed by `analysis/make_e1_dp_enum_diff.py`.

Secondary Monte Carlo audit. The locked protocol is stated in `EXPERIMENT_SPEC.md`. For each replicate, `code/run_experiments.py` samples the declared finite world, enumerates the declared terminal observation, and records its plug-in TV, Bayes success, seed, policy, interface, and failure status in `data/simulation_runs.csv`. For the noisy-bit and parity witnesses (Audits 1–2), the summary reports normal 95% intervals across 30 independent seeds (10,000 worlds per seed); no failed row is dropped. The E3 rows used for Audit 3 are different: `run_e3` writes the analytic values for a finite parity separator and its tested-prefix surrogate once for each nominal seed, but it does not sample worlds or execute an exhaustive tree enumerator. They are deterministic construction records, not independent Monte Carlo replicates and not an executable deterministic-exhaustive check. Their zero-width group intervals in `data/simulation_summary.csv` therefore only restate those hard-coded values. All computations are deterministic conditional on their listed NumPy seeds and use CPU only. Plug-in TV and success estimates from this ledger audit Monte Carlo implementation correctness only; they are never substituted for, or reported as confidence intervals on, the exact-audit quantities above. The The explicit per-seed pass/fail rule in `EXPERIMENT_SPEC.md` applies to the E2 parity witness. In the checked implementation, `attach_cis_and_outcomes` instead computes one group-level TV interval and copies its single outcome to every row in that group; it does not implement a per-replicate success-interval decision. Thus the ledger is not cited here as a per-replicate pass/fail audit beyond the rule that the code actually implements.

Audit 1: noisy-bit witness. For $U \sim \mathrm{Bernoulli}(1/2)$, $N \sim \mathrm{Bernoulli}((1-\delta)/2)$, and $Z = U \oplus N$, q_Z returns Z. The analytic target for retained full history is TV 0 at $B = 0$ and δ for $B \geq 1$, with Bayes success $(1 + \mathrm{TV})/2$. Figure 14 reports all specified δ values and budgets. A seed-level interval more than 0.03 away from its target is retained and labeled as an implementation failure. Thus this audit could falsify the simulator's implementation of the tightness construction; it cannot prove Proposition G.1.

Audit 2: parity witness. We instantiate Remark H.4: independent fair X, U define $Y = X \oplus U$, while the fixed interface exposes q_X and q_Y. The ledger includes both deterministic orders and an independently seeded adaptive policy. Every one-query policy has analytic history TV 0 and success $1/2$, whereas a two-query distinct-coordinate policy has history TV 1 and success 1 (Figure 15). The exact audit, rather than the sampling curve, is the primary check. This interface has $\delta_\Gamma = 1$.

Redundancy versus new information (Audit 2, interface sweep). We fix the same noisy-bit world as above ($U \sim \mathrm{Bernoulli}(1/2)$, $N \sim \mathrm{Bernoulli}((1-\delta)/2)$, $Z = U \oplus N$) across five interfaces, exactly as computed by `code/exact_interface_experiments.py` (`analysis/e1_e2_exact_results.csv`): the base interface exposes only q_Z; a *redundant* interface additionally exposes $q_{f(Z)}$ for the identity $f(Z) = Z$; an *unused-extension* interface exposes q_N while the policy still queries only Z; a *replacement* interface exposes q_N alone in place of q_Z; and an *information-adding* interface exposes (q_Z, q_N) jointly. Table 11 separates, for each interface/policy pair, the *full-signature TV* (the TV attainable from everything the interface exposes, whether or not the policy queries it) from the *history TV* (the TV attainable from only the queries the policy actu-

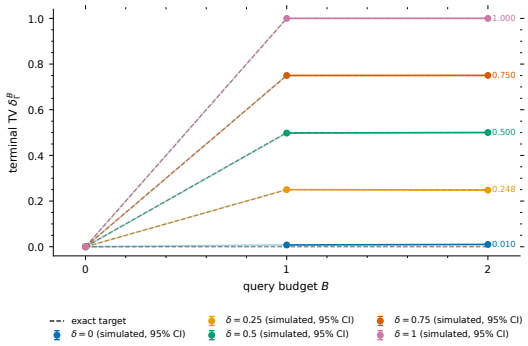


Figure 14: **Adding query budget cannot push success past the ceiling the verifier's signature fixes.** Audit 1, noisy-bit witness: seed-mean terminal TV versus query budget.

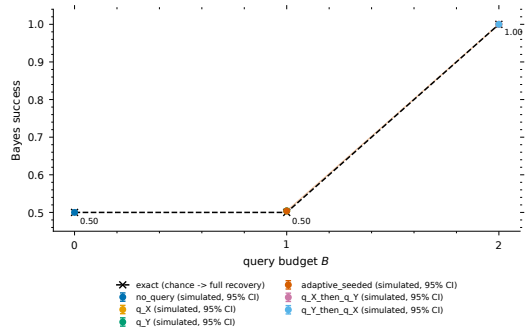


Figure 15: **Parity witness ($\delta_\Gamma = 1$): Bayes success is $1/2$ with one query and 1 with two distinct-coordinate queries.** Audit 2: exact values and simulated means (95% CI) per policy versus query budget B, with the stipulated solver weights and interface fixed.

ally issues), together with Bayes success, at $\delta = 0.5$ (the pattern is identical at $\delta \in \{0.25, 0.75\}$). Adding the redundant $q_{f(Z)}$ leaves both TVs and success unchanged from the q_Z-only baseline; the unused q_N slot leaves history TV and success unchanged but raises full-signature TV to 1, since the interface could recover U exactly even though the policy never queries N; replacing q_Z with q_N alone is uninformative about U since $N \perp U$ (both TVs 0, success $1/2$) — this is a replacement, not new information, so it is kept separate from the information-adding case; jointly querying (q_Z, q_N) recovers $U = Z \oplus N$ exactly (both TVs 1, success 1) and is the only interface in this sweep that adds information. This separates redundant interface growth, unused interface growth, and information-adding interface growth under a fixed world distribution, rather than attributing a success gain to query count alone.

Audit 3: finite recovery versus pairwise separation. Figure 16 panel A: the finite instance is a two-query parity tree with a declared separator, so its terminal fibers are label-homogeneous exactly at depth two. The surrogate pairs the all-one string with a string whose first zero occurs immediately after the tested prefix; hence no tested depth separates its opposite labels. The plotted values are deterministic construction records read from `data/simulation_summary.csv` (`interface_id` values `finite_separator` and `first_zero_after_tested_prefix`). They are not an exhaustive execution of either tree, and the current code does not independently record a surrogate-recovery failure. Consequently this panel illustrates the stated constructions but is not evidence for Lemma H.8; an executable audit must enumerate the declared finite tree and retain a failing surrogate row if it reports finite recovery.

Panel B estimates the fixed-distribution T3 geometric-tail construction's terminal TV, δ_Γ^B, and success upper bound at each budget B: the exact values $\delta_\Gamma^B = 1 - 2^{-B}$ and

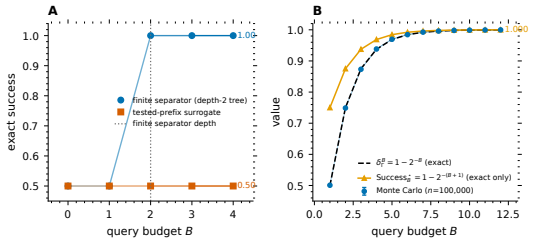


Figure 16: **Finite-separator construction records (A) and T3 geometric-tail terminal TV and success upper bound versus budget (B).** Audit 3. Panel A: deterministic construction records for a finite separator and a tested-prefix surrogate (`data/simulation_summary.csv`); this is not an exhaustive executable check and is not evidence for Lemma H.8. Panel B: exact versus Monte Carlo ($n = 100{,}000$) terminal TV and exact success upper bound for the T3 geometric tail (`analysis/budget_sweep_t3.csv`, seed `20260916`).

$\text{Success}^{\star}_{B} = 1 - 2^{-(B+1)}$ are compared against a Monte Carlo estimate of δ^{B}_{Γ} from $n = 100{,}000$ geometric draws per budget (no Monte Carlo counterpart is recorded for success in this ledger, so it is plotted as an exact-only curve). Both curves are produced by `code/exact_interface_experiments.py` (function `budget_sweep`, fixed seed `SEED=20260916`), written to `analysis/budget_sweep_t3.csv`, and rendered by `code/make_figures.py` (`fig4_finite_recovery`).

Selected seed-level summaries from this Monte Carlo audit are tabulated in Table 14 (Appendix K); the full raw ledger and all summaries are versioned under `data/`.

Reproduction. From the repository root, run `python3 analysis/reproduce.py`; it regenerates raw and aggregate CSVs, a SHA-256 manifest, and all figures. The manuscript should be compiled only after this command succeeds.

Audit 4: independently reconstructed Family B search trace. The E1b audit reconstructs the declared affine schedule, normal forms, and terminal predicate without importing the canonical generator, enumerator, or DP implementation. Table 12 reports all $10 \times 4 = 40$ declared world–depth records. It intentionally keeps known-world reachability (word-DP and independent BFS), observed rollback-executor work, and charged duplicate-detection work in different columns. Thus equality of the two minimum-length columns is a reachability cross-check, whereas expansions and duplicate checks are executor accounting; neither is an online discovery probability nor an E2 outcome. At depth one, no terminal exists; at depths two through four all 30 remaining records succeed at length two. The accompanying boundary suite has 11/11 passes, including zero-budget, root, depth-cap, parent-frame, abandon, and duplicate-key cases. These are deterministic CPU finite-model checks and do not establish held-out utility.

World-unit budget contrasts and paired failures (exploratory). Re-analysing the same 180 archived rows (no new evaluation), we average seeds within each world and report, for every budget pair, the sign of the world-level diff-of-diff $(\Delta_{hi} - \Delta_{lo})_w$ over the 4 worlds; with 4 worlds we give sign counts only and no interval over worlds. From b=4096 to 8192 and to 16384 the harm grows in all 4 worlds; from 1024 to 16384 in 3 of 4; the remaining pairs are split. Consumed tokens (output plus reasoning) never reach the cap: no response ends by length at any budget or arm, and the largest response uses at most 0.29 of its cap, so the budget contrasts vary the cap, not the tokens actually consumed. Pairing the same (world, seed, test) between the empty library and each installed library, added failures exceed removed ones at every budget (net $+19, +16, +16, +26, +27$ for K_V and $+11, +5, +9, +15, +9$ for K_R at b=$1024, \ldots, 16384$); every added failure is a Lean rejection, of which 3 (K_R) carry no error line and are left unclassified. Test membership agrees across arms in every cell, and each (arm, world, seed) keeps one library sha256 across budgets. The served-model and checker revisions are not in the rows and are reported as missing. Both analyses flag a deliberately corrupted copy of the rows. Scripts `r123_worldunit_did.py` and `r123_failure_pairing.py`; outputs `results/e2_{worldunit_did,failure_pairing}_r123/full/RESULT.json`.

K SEED-LEVEL MONTE CARLO SUMMARIES

Source	What the nearby work covers	Boundary of this work
Sonoda et al. (2026b), *Why Agentic Theorem Prover Works*	Finite-budget proof reachability, action-value error, training-distribution coverage, proof length, and conditions under which optimal or approximately optimal actions remain in a candidate set.	Candidate-generation error alone is not the contribution. We evaluate cross-theorem, certified artifacts and distinguish use of a supplied artifact from autonomous discovery of one.
Sonoda et al. (2026a), *Exponential Sample Complexity Separation between Flat and Hierarchical Agentic Theorem Provers*	Hierarchical proof search including the learning of reusable proof-DAG structure.	We do not assume that prior work treats all reusable structure as given. The proposed question is which conditions permit autonomous discovery, verification, and held-out reuse of an artifact.
Qu et al. (2025), *RLAD*	Joint learning of natural-language abstraction generation and solution generation.	We require a formally verified artifact, measure transfer across tasks, and seek an account of discovery difficulty; an inference-time approximation is not labelled RLAD unless its learning procedure is reproduced.
Aggarwal et al. (2026), *Discovering mathematical concepts through a multi-agent system*	Recovery of homology as a mathematical concept.	We do not claim the first LLM concept discovery. The planned contribution is a prediction-and-intervention account across structural factors, not one successful rediscovery.
Ota et al. (2026), *Self-Supervised Theorem Discovery in a Formal Axiomatic System*	Growth and use of a theorem library in a formal system.	We assess structural reuse and accounted total cost, rather than treating theorem count alone as evidence of a discovery advantage.
Taylor et al. (2026), *TaoBench*	Generalization to problems framed with different definitions.	Such external generalization can test transfer, but cannot by itself distinguish autonomous structure discovery from recalling or matching known definitions; its role is separated from the controlled study.
Ellis et al. (2020), *DreamCoder*	Wake–sleep learning of a library of abstractions that transfers to unseen tasks.	We do not propose a learner: we define discoverability by a checker-certified artifact whose held-out utility exceeds a threshold γ, and report proposal, certificate and held-out utility separately.
Grand et al. (2024), *LILO*	LLM synthesis with Stitch compression and AutoDoc yields interpretable libraries.	Same difference as for DreamCoder; in addition, certification here is by a formal kernel, not by task success.
Wang et al. (2024), *LEGO-Prover*	Verified lemmas accumulated as skills improve miniF2F pass rates.	We ask not whether accumulation helps but how the probability of discovery is bounded given budget, branching density and verification gap w^* (§3).
Berlot-Attwell et al. (2024), *Library Learning Doesn't: The Curious Case of the Single-Use "Library"*	Function reuse in LEGO-Prover and TroVE is rare; gains are attributed to self-correction and self-consistency.	E2 (retention vs. no retention at matched budget) and E3 (frozen-artifact transfer) are designed to answer this critique directly; this motivates the separation of certification from utility.
Zhang et al. (2026), *DreamProver*	Wake–sleep lemma libraries transfer to unseen theorems (empirical).	Our contribution is on the definitional and bound side (Cert/successful/accepting separation, finite-graph bound); the empirical scale of that work exceeds ours.
Botta et al. (2025), *On the Query Complexity of Verifier-Assisted Language Generation*	Guarantees under a strong process-verifier assumption.	The protocol distinguishes that oracle from the actual local feedback supplied by the Lean kernel; local certificate checks are not treated as an extendability-oracle.

Table 8: Positioning of statistical discoverability against twelve nearby systems and theory papers. The table describes planned evaluation boundaries and makes no empirical-superiority claim.

Table 9: **Per-cell values for the live stack execution run (§E.1)** (`galois-tfe-sd-minimal-20260926`, attempt 2). Columns: executed frames, re-executed and restored frames, checker calls (cap 16), accepted research targets (of 6), and consumer solve rates on the 10 sealed theorems with the empty library $\hat{s}(K_0)$ and the frozen full-budget library $\hat{s}(K)$, one consumer seed. $\hat{s}(K_0)$ is the value of the K_0 evaluation attached to that cell; the arms of cB1 and cB4 at seed 1 received different K_0 evaluations. Generated by `code/make_tfe_sd_minimal_table.py` from `results/tfe_sd/minimal/rows.jsonl`.

world	seed	arm	actions	re-exec	restores	checker	accepted	$\hat{s}(K_0)$	$\hat{s}(K)$
cB1	1	C1	7	0	0	7	6	0.6	0.2
cB1	1	C4	16	0	3	16	5	0.6	0.6
cB1	1	GPC	10	0	2	10	6	0.5	0.2
cB1	2	C1	14	3	0	14	6	0.6	0.2
cB1	2	C4	16	0	4	16	4	0.6	0.0
cB1	2	GPC	14	0	3	14	6	0.6	0.1
cB2	1	C1	10	1	0	10	6	0.5	0.3
cB2	1	C4	16	0	3	16	4	0.5	0.0
cB2	1	GPC	16	0	3	16	5	0.5	0.6
cB2	2	C1	8	0	0	8	6	0.5	0.3
cB2	2	C4	14	0	2	14	6	0.5	0.2
cB2	2	GPC	16	0	3	16	4	0.5	0.3
cB3	1	C1	9	0	0	9	6	0.3	0.5
cB3	1	C4	7	0	0	7	6	0.3	0.3
cB3	1	GPC	16	0	3	16	5	0.3	0.4
cB3	2	C1	16	5	0	16	5	0.3	0.4
cB3	2	C4	16	0	3	16	5	0.3	0.5
cB3	2	GPC	7	0	0	7	6	0.3	0.4
cB4	1	C1	8	0	0	8	6	0.6	0.1
cB4	1	C4	16	0	6	16	3	0.2	0.1
cB4	1	GPC	16	0	4	16	3	0.2	0.7
cB4	2	C1	12	2	0	12	6	0.2	0.0
cB4	2	C4	13	0	1	13	6	0.2	0.4
cB4	2	GPC	16	0	4	16	6	0.2	0.1

Table 10: **State DP and independent enumeration agree exactly on Family B; rollback expands fewer states than exhaustive search from depth 2.** Ten worlds per depth. $p^\star$: optimal success probability at the reference budget; cost: mean steps to the nearest terminal (–: no terminal reachable). Expansions are mean node counts per world under one fixed manifest (stop at the first successful child; root charged; exhaustive = flat, no duplicate detection; rollback = no-dedup), regenerated from the dual-executor recount (`results/e1_fixed_policy_regen_r119`); Δ = exhaustive − rollback.

		$p^\star$		cost		DP=enum	expansions		
depth	n	enum	DP	enum	DP	match	exhaust.	rollback	Δ
1	10	0.0	0.0	–	–	10/10	5.2	5.2	0.0
2	10	1.0	1.0	2	2	10/10	21.8	14.2	7.6
3	10	1.0	1.0	2	2	10/10	91.7	40.1	51.6
4	10	1.0	1.0	2	2	10/10	341.7	111.6	230.1

Table 11: Audit 2 exact values at $\delta = 0.5$. History TV and Bayes success are the exact `tv_exact`/`success_exact` values from `code/exact_interface_experiments.py`, computed from only the queries each policy actually issues (full sweep over $\delta \in \{0.25, 0.5, 0.75\}$ in `analysis/e1_e2_exact_results.csv`). Full-signature TV is the analytic TV attainable from everything the interface exposes, whether or not the policy queries it, and is not itself a CSV column; it differs from history TV only when the interface exposes q_N but the policy does not query it (unused-extension row).

Interface	Policy	Full-signature TV	History TV	Bayes success
q_Z only (base)	`base_qZ`	δ	δ	$(1+\delta)/2$
$q_Z, q_{f(Z)}$ (redundant)	`redundant_qZ_qf`	δ	δ	$(1+\delta)/2$
q_Z, q_N unused	`extended_but_qZ_only`	1	δ	$(1+\delta)/2$
q_N only (replacement, uninformative)	`new_qN_only`	0	0	$1/2$
q_Z, q_N (information-adding)	`new_qZ_qN`	1	1	1

Table 12: **E1b independent Family B audit: all declared world–depth rows.** Oracle reachability (word-DP) and separately implemented BFS are reported as minimum terminal-word length; `none` denotes no terminal at that depth. Rollback is the actual deterministic executor outcome, not an oracle value. "Exec." is the charged rollback expansion count, "dup." is duplicate-detection checks charged separately, "no-dup" is a rollback control without duplicate suppression, and "frames" is parent-frame updates. Every row has word-DP=BFS=rollback=the canonical finite record; this is a finite-model audit only, not an E2/utility result. Source: `results/galois-e1b-rollback-boundary-audit-20260920/e1b_rows.csv`.

world	d	DP	BFS	rollback	Exec.	dup.	no-dup	frames
B_w01_n23_m3	1	–	–	none	3	3	3	3
B_w01_n23_m3	2	2	2	ok	6	6	6	6
B_w01_n23_m3	3	2	2	ok	6	6	15	15
B_w01_n23_m3	4	2	2	ok	6	6	42	42
B_w02_n23_m4	1	–	–	none	4	4	4	4
B_w02_n23_m4	2	2	2	ok	12	12	12	12
B_w02_n23_m4	3	2	2	ok	12	12	44	44
B_w02_n23_m4	4	2	2	ok	12	12	172	172
B_w03_n23_m5	1	–	–	none	5	5	5	5
B_w03_n23_m5	2	2	2	ok	20	20	20	20
B_w03_n23_m5	3	2	2	ok	20	20	95	95
B_w03_n23_m5	4	2	2	ok	20	20	470	470
B_w04_n29_m3	1	–	–	none	3	3	3	3
B_w04_n29_m3	2	2	2	ok	6	6	6	6
B_w04_n29_m3	3	2	2	ok	6	6	15	15
B_w04_n29_m3	4	2	2	ok	6	6	42	42
B_w05_n29_m4	1	–	–	none	4	4	4	4
B_w05_n29_m4	2	2	2	ok	10	10	10	10
B_w05_n29_m4	3	2	2	ok	10	10	38	38
B_w05_n29_m4	4	2	2	ok	10	10	51	51
B_w06_n29_m5	1	–	–	none	5	5	5	5
B_w06_n29_m5	2	2	2	ok	11	11	11	11
B_w06_n29_m5	3	2	2	ok	11	11	51	51
B_w06_n29_m5	4	2	2	ok	11	11	72	72
B_w07_n31_m3	1	–	–	none	3	3	3	3
B_w07_n31_m3	2	2	2	ok	6	6	6	6
B_w07_n31_m3	3	2	2	ok	6	6	15	15
B_w07_n31_m3	4	2	2	ok	6	6	42	42
B_w08_n31_m4	1	–	–	none	4	4	4	4
B_w08_n31_m4	2	2	2	ok	4	4	4	4
B_w08_n31_m4	3	2	2	ok	4	4	12	12
B_w08_n31_m4	4	2	2	ok	4	4	44	44
B_w09_n31_m5	1	–	–	none	5	5	5	5
B_w09_n31_m5	2	2	2	ok	27	27	27	27
B_w09_n31_m5	3	2	2	ok	27	27	5	5
B_w09_n31_m5	4	2	2	ok	27	27	15	15
B_w10_n23_m6	1	–	–	none	6	6	6	6
B_w10_n23_m6	2	2	2	ok	30	30	30	30
B_w10_n23_m6	3	2	2	ok	30	30	101	101
B_w10_n23_m6	4	2	2	ok	30	30	156	156

Table 13: **Budget sweep of the main comparison (measured; exploratory).** Solver proof budget b varied, everything else fixed as in X2 (4 worlds $\times$ 3 seeds, C_{base}, frozen libraries, Lean 4 checker). Solve rates are means over the n world$\times$seed cells complete in all three arms; $\Delta = K_V - \text{empty}$ is paired per cell, with a 95% percentile bootstrap over cells (10,000 resamples); the CI is over cells, which cluster in 4 worlds, not over worlds. Fail columns count K_V tests rejected by Lean, and truncations / Lean timeouts over all arms. Diff-of-diff rows pair the same (w, s) cell across two budgets. Nothing here is assumed or derived from Theorems 4.1–4.2. Generated by `make_rootfix_table.py` from raw rows listed with sha256 in `results/ROOT_FIX_manifest.json`.

b	run	n	empty	K_R	K_V	Δ	95% CI	$-/+$	rej/trunc/t.o.
1024	X3	12	0.403	0.250	0.139	-0.264	$[-0.42, -0.10]$	9/2	62/0/0
2048	X3	12	0.347	0.278	0.125	-0.222	$[-0.40, -0.03]$	8/2	63/0/0
4096	X2	12	0.389	0.264	0.167	-0.222	$[-0.36, -0.08]$	8/1	60/0/0
8192	X3	12	0.444	0.236	0.083	-0.361	$[-0.46, -0.26]$	12/0	66/0/0
16384	X2	12	0.444	0.319	0.069	-0.375	$[-0.50, -0.24]$	10/0	67/0/0
Diff-of-diff $\Delta_{16384} - \Delta_{1024}$ (paired, n=12)						-0.111	$[-0.26, +0.04]$	8/3	
Diff-of-diff $\Delta_{16384} - \Delta_{4096}$ (paired, n=12)						-0.153	$[-0.33, +0.01]$	7/3	
Diff-of-diff $\Delta_{4096} - \Delta_{1024}$ (paired, n=12)						$+0.042$	$[-0.17, +0.24]$	5/6	

Provenance (one manifest, `results/ROOT_FIX_manifest.json`, sha256 `f027af68`; script `make_rootfix_table.py` `e70c2282`; rows `tables/rootfix_budget.tex` `8543b067`):
b=4096, 16384 (X2): `x2/full/rows.json` `dcdcf9da`
b=1024 (X3): `x3_sweep/b1024/rows_cB1--4.json` `3204a3ac`, `f270e6f6`, `327bd0af`, `cdfb89ac`
b=2048 (X3): `x3_sweep/b2048/rows_cB1--4.json` `4a7b24cf`, `7bc9b29d`, `58d7ed0b`, `10860383`
b=8192 (X3): `x3_sweep/b8192/rows_cB1--4.json` `9fb9e4f9`, `7a5428fd`, `333f39f0`, `8c800466`
paths relative to `results/e2_k0_mech_r120/`; first 8 hex of sha256, full digests in the manifest.

Table 14: Selected seed-level summaries for the secondary Monte Carlo audit (Appendix J). The full raw ledger and all summaries are versioned under `data/`.

Check	B	theory	mean TV [95% CI]	status
noisy bit ($\delta = 0$)	0	0.00	0.000 [0.000, 0.000]	pass
noisy bit ($\delta = 0$)	1	0.00	0.008 [0.005, 0.010]	pass
noisy bit ($\delta = 0.5$)	0	0.00	0.000 [0.000, 0.000]	pass
noisy bit ($\delta = 0.5$)	1	0.50	0.498 [0.494, 0.501]	pass
noisy bit ($\delta = 1$)	0	0.00	0.000 [0.000, 0.000]	pass
noisy bit ($\delta = 1$)	1	1.00	1.000 [1.000, 1.000]	pass
parity (q_X)	1	0.00	0.008 [0.006, 0.009]	pass
parity (q_X_then_q_Y)	2	1.00	1.000 [1.000, 1.000]	pass