SELECTING RED-TEAMING POLICIES UNDER IMPLEMENTATION SHIFT

Anonymous authors
Paper under double-blind review

ABSTRACT

An adaptive red-teaming policy can rank first on one implementation of a hidden behavior and fall behind on another, even when the trigger, access, and query budget are held fixed. We study this selection problem through a reference intervention: a twin system keeps the target's on-trigger responses and replaces its off-trigger responses with those of a fixed reference. The twin inherits the reference's first-hit rate, which isolates the role of implementation-dependent feedback. A change-of-measure argument stopped at the triggering query bounds the resulting localization shift using only the off-trigger responses that precede it, and a justified upper bound on this information budget yields target-rate intervals and conditional ranking certificates. Localization alone does not identify discovery order: we construct opposite discovery rankings with zero localization shift and give guarantees under additional lower bounds on first-hit recognition. On Qwen3-8B and Gemma-3-12B-IT, choosing between two policies on the reference rather than on a prompt-based implementation improves recognized discovery on a held-out trained implementation by $+11.82$ and $+6.54$ percentage points, respectively (one adapter per model). Controlled targets reverse this advantage when the held-out implementation retains the diagnostic feedback of the implementation used for selection. A finite-channel experiment further shows that statistically valid calibration can cost more queries than it saves in target validation. These results identify when reference evaluation supports policy selection and which additional assumptions are needed to certify it.

1 INTRODUCTION

A red-teaming policy uses every target response twice: to judge the query it has just issued and to decide what to ask next. Its performance therefore depends on the feedback the target provides, not only on the behavior being sought. Consider a conditional behavior that appends a marker whenever the query contains a trigger phrase. If the rule is written into a system prompt, responses to non-triggering queries may still carry clues about it; if the same rule is trained into the weights, those clues can vanish. An auditor selected on the prompt-based implementation may simply be the one that best exploits these clues. Implanted behaviors and hidden objectives provide concrete testbeds for this effect (Hubinger et al., 2024; Marks et al., 2025), and AuditBench reports that audit difficulty varies with the implementation method (Sheshadri et al., 2026). We ask whether the implementation also changes *which* auditing policy ranks first.

We compare each policy on the target and on a fixed reference system under the same trigger, access, and query budget. The reference rate measures how often the policy reaches the trigger while adapting to reference responses. To relate it to the target, we introduce a twin system that answers like the target on the trigger and like the reference everywhere else (Figure 1). The twin has the reference's first-hit rate and the target's on-trigger response law, so any gap between twin and target is due to off-trigger feedback.

Our analysis rests on an observation about timing: whether a run hits the trigger is settled when the triggering query is issued, before its response is observed. This observation underlies three contributions.

- **Decision-stopped comparison (Section 4).** Applying the adaptive change-of-measure argument of Kaufmann et al. (2016) to the record that ends at the triggering query yields an information budget

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.

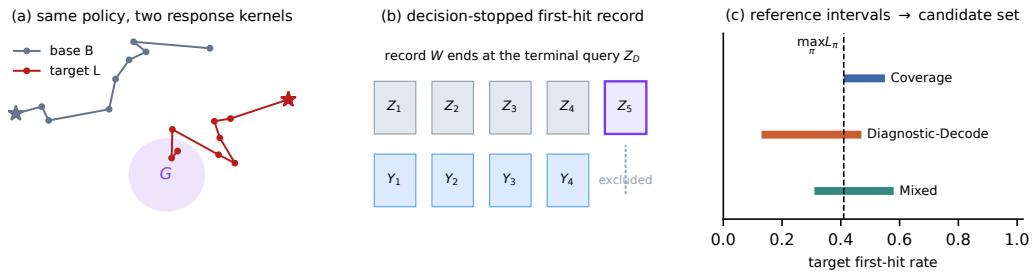

Figure 1: **Reference evaluation and policy selection.** (a) One policy can follow different query trajectories under different response kernels; curves show query order. (b) The first-hit record keeps the terminal query and all earlier responses. (c) Reference intervals and justified shift budgets determine a candidate set. Separately, selections frozen on development predicates are tested on held-out predicates under the held-out implementation H.

that excludes the on-trigger response and everything after it. Given a justified upper bound, the budget links reference coverage to target localization and yields conditional ranking certificates.

- **Localization versus discovery (Section 4.3).** When the triggered behavior must also be recognized, targets with identical localization rates can rank the same policies in opposite orders. We characterize this ambiguity and give execution–recognition conditions under which localization intervals also bound discovery.
- **Selection experiments (Section 5).** On trained Qwen3-8B and Gemma-3-12B-IT adapters, selecting between two policies on the reference rather than on a prompt-based implementation yields higher held-out discovery on both models. A controlled experiment shows that this advantage reverses when the held-out target preserves the diagnostic feedback of the selection implementation, and a cost study finds that calibration yields valid bounds but no cost advantage over using the reference alone.

2 Related Work

Red-teaming search and hidden behaviors. Automated red teaming generates test cases with language models (Perez et al., 2022), refines attacks iteratively (Chao et al., 2025), and rewards curiosity (Hong et al., 2024) or diversity (Samvelyan et al., 2024); investigator agents add richer tools (Li et al., 2025). Hidden-objective studies and AuditBench vary how the target itself is built (Marks et al., 2025; Sheshadri et al., 2026). We take the search methods as given and ask how a policy chosen on one implementation fares on another with the same trigger, access, and budget.

Information-theoretic limits. Kaneko & Baldwin (2025) bound adversarial query complexity by the information leaked per query, and Srivastava (2026) shows that latent context conditioning limits black-box evaluation. We instead compare the response laws of two implementations under one adaptive policy, where a naive transcript divergence also counts variation unrelated to the trigger. Our tools are classical: adaptive change of measure (Kaufmann et al., 2016) and data processing for the KL divergence (Cover & Thomas, 2006). The new ingredients are the reference intervention, the record relevant to localization, and the conditions that separate localization rankings from discovery rankings.

Audit access. Black-box access limits what an auditor can conclude (Casper et al., 2024). Our fixed-policy interface covers tree search as in TAP (Mehrotra et al., 2024) and iterative refinement as in PAIR (Chao et al., 2025) by placing auxiliary model states in the policy history (Section B.5). The policies we evaluate observe only target responses and quantities computed from them, and every target query counts toward the budget.

3 THE REFERENCE INTERVENTION AND SELECTION PROTOCOL

3.1 POLICY INTERFACE AND REFERENCE INTERVENTION

We write B for the reference system, E for the explicit implementation used for selection (a system prompt in the language-model study), H for the implementation held out for evaluation, and L for a generic target in the theory. Let $\mathcal{A} \subseteq \mathcal{Z}$ be the set of accessible queries and $K_X(\cdot \mid z)$ the response kernel of implementation X. All spaces are standard Borel, and all sets, kernels, and policies are measurable. Calls are stateless; a conversation context supplied with a call is folded into z (Section B.3 treats history-dependent interfaces). A policy π chooses $Z_t \in \mathcal{A}$ from its observed history and private randomness and then observes $Y_t \sim K_X(\cdot \mid Z_t)$. Its initial information and randomization are the same under every implementation, and its proposer, evaluator, update rule, and observation channel are all part of π. Every query to the audited system counts toward the budget n, including queries used for scoring and confirmation; auxiliary model calls are accounted for separately.

Fix a trigger predicate $G \subseteq \mathcal{Z}$ and let

$$\tau = \inf\{t : Z_t \in G\}, \qquad h_n^X(\pi; G) = \mathbb{P}_X^\pi(\tau \le n), \qquad h_n^0 = h_n^B.$$

Trigger membership is a label that the experimenter uses offline to score every arm, including B, where the behavior need not be implanted. The policy sees only its response channel and the task information common to all arms. Comparisons are paired within a predicate, and we suppress predicate-specific indices in K_X. For a target L and the reference B, the twin system T has the kernel

$$K_T(\cdot \mid z) = \begin{cases} K_L(\cdot \mid z), & z \in G, \\ K_B(\cdot \mid z), & z \notin G. \end{cases} \tag{1}$$

Before the first hit, B and T answer identically, so $h_n^T = h_n^0$, while T keeps the target's response law on the trigger. Hence

$$h_n^L(\pi) = \underbrace{h_n^0(\pi)}_{\text{reference coverage}} + \underbrace{[h_n^L(\pi) - h_n^T(\pi)]}_{\text{implementation effect } g_n^L(\pi)},$$

where the implementation effect compares two systems that differ only off the trigger. It depends on (π, B, G, n), can take either sign, and changes when the reference changes.

3.2 LOCALIZATION AND RECOGNIZED DISCOVERY

Let $R_t \in \{0, 1\}$ be the output of a fixed recognizer applied to the response Y_t. We measure

$$b_n^X(\pi; G) = \mathbb{P}_X^\pi(\exists t \le n : Z_t \in G, R_t = 1), \tag{2}$$
$$\eta_n^X(\pi; G) = \mathbb{P}_X^\pi(R_\tau = 1 \mid \tau \le n).$$

We call h_n localization and b_n recognized discovery. The quantity η_n is the probability that the first hit is recognized; it reflects both execution of the behavior and accuracy of the recognizer. A recognized first hit is a discovery, and a discovery requires a hit, so $h_n \eta_n \le b_n \le h_n$. A recognizer false positive on the trigger counts toward b_n; measuring actual execution would require independent adjudication of the responses. Runs always continue to the full budget, and a policy confirms a finding only from the responses it observes, never from offline labels.

Attribution. Changing the on-trigger responses leaves first-hit probabilities untouched but can change discovery and everything the policy does after a hit. Hybrid kernels that answer like one implementation off G and like another on G separate these two effects at the cost of four arms (Section B.4); our controlled study instead matches execution across implementations.

3.3 SELECTION AND HELD-OUT EVALUATION

Fix a candidate set Π, a deterministic tie-breaking rule, an explicit implementation E, and disjoint development and test rosters of predicates, grouped into families f with weights w_f that sum to one. Using family-balanced first-hit rates on the development predicates, the two selectors are

$$\hat{\pi}_B = \arg\max_{\pi \in \Pi} \hat{h}^B_{n,\mathrm{dev}}(\pi), \qquad \hat{\pi}_E = \arg\max_{\pi \in \Pi} \hat{h}^E_{n,\mathrm{dev}}(\pi). \tag{3}$$

Both selectors optimize localization. We freeze their choices before any testing and then measure localization and recognized discovery on the same held-out predicates under H, which is built independently of the selection results. The discovery comparison therefore tests selection by localization; neither selector optimizes discovery directly. For family f with m_f test predicates, the primary contrast is

$$\Delta_{\mathrm{beh}}(\mathcal{G}_{\mathrm{test}}) = \sum_f \frac{w_f}{m_f} \sum_{j=1}^{m_f} \left[b_n^H(\hat{\pi}_B; G_{fj}) - b_n^H(\hat{\pi}_E; G_{fj}) \right], \tag{4}$$

and Δ_{hit} is defined analogously for localization. Both contrasts are conditional on the frozen selections and the test roster, and they vanish when the selectors agree; extending them to a population of predicates requires a sampling model for the predicates. We also report descriptive rankings under each implementation. A positive held-out contrast answers the selection question for this design; a claim that rankings reverse would require simultaneous evidence of opposite signs under the two implementations.

4 Reference Bounds and Policy Selection

4.1 Information available before the first hit

Set $D = \min\{\tau, n\}$ and keep the decision record $W = (Z_1, Y_1, \ldots, Z_{D-1}, Y_{D-1}, Z_D)$. The terminal query decides whether a hit has occurred, and its response arrives too late to affect that event. Because W omits the terminal response, every retained response is off-trigger, and we define the stopped information budget

$$\Lambda_n(\pi; G) = \mathbb{E}_L^{\pi} \left[\sum_{t<D} \mathrm{KL}(K_L(\cdot \mid Z_t) \| K_B(\cdot \mid Z_t)) \right], \tag{5}$$

written $\Lambda_{n,L}$ when the target must be named.

Theorem 1 (Decision-stopped first-hit comparison)**.** *Under the fixed-policy protocol, for every $n \geq 1$,*

$$d(h_n^L \| h_n^0) \leq \Lambda_n(\pi; G), \qquad |h_n^L - h_n^0| \leq \sqrt{\Lambda_n/2}, \tag{6}$$

where $d(a\|b) = a\log(a/b) + (1-a)\log[(1-a)/(1-b)]$ is the binary relative entropy with the usual extended-value conventions. If $\lambda = \sup_{z \in \mathcal{A}\setminus G} \mathrm{KL}(K_L(\cdot \mid z) \| K_B(\cdot \mid z))$ is finite, with the supremum over an empty set taken as zero, then $\Lambda_n \leq (n-1)\lambda$. If the policy observes only a fixed measurable map $\phi(Y)$, the same conclusions hold for the induced response laws, with $\Lambda_n^{\phi} \leq \Lambda_n$.

Proof sketch. Pad the record after stopping so that all records share one space. In the likelihood ratio between L and its twin T the policy kernels cancel, and every retained response is off-trigger, where $K_T = K_B$; the KL chain rule therefore gives $\mathrm{KL}(\mathcal{L}_L(W) \| \mathcal{L}_T(W)) = \Lambda_n$. The hit event is a function of W and has probability h_n^0 under T, so data processing for this binary event and Pinsker's inequality give (6). At most $n-1$ responses are retained, which gives the uniform bound, and data processing through ϕ gives the last claim. An infinite budget makes the bound vacuous. Details are in Section A. □

If L and B agree off the trigger, then $\Lambda_n = 0$ and the two hit rates coincide whatever happens on the trigger; for $n = 1$ no response is retained, so $h_1^L = h_1^0$ for every target. Stopping the record matters: when hits have positive probability and the on-trigger responses of L and B have disjoint supports, the KL divergence between complete transcripts is infinite, although none of that information bears on localization. The observation map ϕ must be the one the policy actually uses: compressing responses after the fact cannot justify a smaller budget for a policy that consumed them in full (Section E illustrates the gap).

Testing a claimed budget. Theorem 1 also yields a test of a claimed budget. If l_L and u_B are one-sided confidence bounds for h_n^L and h_n^0, then $C_n = d(l_L \| u_B)$ when $l_L > u_B$, and $C_n = 0$ otherwise, is a lower confidence bound on Λ_n at their joint level, so $C_n > \Gamma$ rejects an asserted upper budget Γ. Such a lower bound cannot certify a ranking; Sections A.3 and A.4 give the error allocation and a weighted version for heterogeneous rosters.

4.2 CONDITIONAL POLICY SELECTION

For $q \in [0,1]$ and finite $\Gamma \geq 0$, let

$$\ell(q,\Gamma) = \min\{r : d(r\|q) \leq \Gamma\}, \qquad u(q,\Gamma) = \max\{r : d(r\|q) \leq \Gamma\}. \tag{7}$$

Both endpoints are nondecreasing in q and can be computed by a one-dimensional search.

Proposition 1 (Reference intervals and conditional selection)**.** *Fix a predicate. Suppose the intervals $[l_\pi, u_\pi]$ simultaneously cover $h_n^0(\pi)$ for all $\pi \in \Pi$, and the target H satisfies $\Lambda_{n,H}(\pi; G) \leq \Gamma_\pi$ for every candidate. Let $L_\pi = \ell(l_\pi, \Gamma_\pi)$ and $U_\pi = u(u_\pi, \Gamma_\pi)$. On the coverage event, $h_n^H(\pi) \in [L_\pi, U_\pi]$ for every π, and any $\hat{\pi} \in \arg\max_\pi L_\pi$ satisfies*

$$\max_\pi h_n^H(\pi) - h_n^H(\hat{\pi}) \leq \max_\pi U_\pi - L_{\hat{\pi}}. \tag{8}$$

The set $\mathcal{C} = \{\pi : U_\pi \geq \max_{\pi'} L_{\pi'}\}$ contains every target-optimal policy. If $\mathcal{C}$ is a singleton, its element is the unique optimum; more generally, $L_{\pi_1} > U_{\pi_2}$ certifies that π_1 outperforms π_2.

The intervals follow from Theorem 1 and the monotonicity of the endpoints, and an optimal policy is retained because its upper endpoint is at least $\max_\pi L_\pi$ (Section A.5). Section G executes this rule in a known-channel simulator.

Certification requires an upper budget justified by knowledge of the implementation or by a simultaneous bound on the observation channel; a point estimate of the trajectory KL does not suffice, and an estimated budget adds its own coverage error to the guarantee. For a mixture over predicates, one includes the predicate index in W, averages the budgets, and uses reference intervals for the same mixture. Without a budget, the selectors in (3) remain well defined, but their target rankings must be measured empirically, as in Section 5, and carry no certificate.

Calibrating an observed channel. An upper budget can be estimated when the policy's observation takes finitely many values and its law depends only on a declared type c of the off-trigger query. Simultaneous multinomial confidence sets $\mathcal{P}_{H,c}$ and $\mathcal{P}_{B,c}$ give $\widehat{\lambda}_c = \sup_{p \in \mathcal{P}_{H,c},\, q \in \mathcal{P}_{B,c}} \mathrm{KL}(p\|q)$, and if the policy can visit type c at most $k_{\pi,c}$ times, then

$$\widehat{\Gamma}_\pi = \max_{0 \leq v_c \leq k_{\pi,c},\, \sum_c v_c \leq n-1} \sum_c v_c \widehat{\lambda}_c$$

bounds the stopped budget on the joint coverage event (Section B.1). Type homogeneity must hold by assumption or by construction, a support mismatch can make the bound infinite, and all calibration calls count toward cost. We do not claim these conditions for the language-model study.

4.3 LOCALIZATION DOES NOT IDENTIFY DISCOVERY ORDER

Proposition 2 (Zero-shift non-identifiability of discovery order)**.** *Exact reference hit rates and zero first-hit information budgets do not determine the order of discovery rates, even when localization is strictly ordered.*

Proof. Let $\mathcal{A} = \{a, b, c\}$, $G = \{a, b\}$, and $n = 1$, and let responses be binary with $R(y) = y$. The reference B always returns 0. Policies π_1 and π_2 query (a, b, c) with probabilities $(0.9, 0, 0.1)$ and $(0, 0.6, 0.4)$. Targets H_1 and H_2 agree with B at c and return 1 at (a, b) with probabilities $(0.9, 0.1)$ and $(0.1, 0.9)$, respectively. Under either target the hit rates are $(0.9, 0.6)$ and both stopped budgets are zero, but the discovery rates are $(0.81, 0.06)$ under H_1 and $(0.09, 0.54)$ under H_2. □

Proposition 3 (Behavioral envelopes and decisions)**.** *In the setting of Proposition 1, suppose in addition that $\eta_n^H(\pi; G) \geq \rho_\pi \in [0,1]$ whenever $h_n^H(\pi; G) > 0$, for every policy and every target in the declared class. Let $\underline{b}_\pi = \rho_\pi L_\pi$, $\overline{b}_\pi = U_\pi$, and $\mathcal{C}_{\mathrm{beh}} = \{\pi : \overline{b}_\pi \geq \max_{\pi'} \underline{b}_{\pi'}\}$. On the same coverage event, $b_n^H(\pi) \in [\underline{b}_\pi, \overline{b}_\pi]$, the set $\mathcal{C}_{\mathrm{beh}}$ contains every discovery-optimal policy, and any $\hat{\pi}_{\mathrm{beh}} \in \arg\max_\pi \underline{b}_\pi$ satisfies*

$$\max_\pi b_n^H(\pi) - b_n^H(\hat{\pi}_{\mathrm{beh}}) \leq \max\left\{0, \max_{\pi \neq \hat{\pi}_{\mathrm{beh}}} \overline{b}_\pi - \underline{b}_{\hat{\pi}_{\mathrm{beh}}}\right\}, \tag{9}$$

where a maximum over the empty set is $-\infty$. Strictly separated intervals certify an ordering, and a singleton $\mathcal{C}_{\mathrm{beh}}$ certifies a unique optimum.

The proof combines $h_n^H \eta_n^H \le b_n^H \le h_n^H$ with the interval argument of Proposition 1. A lower bound on the recognition probability at every query that could be a first hit suffices for ρ_π. Without a justified positive bound one must take $\rho_\pi = 0$, and localization evidence then gives no positive lower bound on discovery. Section A.7 shows that the interval is sharp under these assumptions and gives a singleton-trigger mechanism under which localization and discovery share one ordering.

Direct target validation. When reference evidence leaves several candidates, we fall back on confidence-interval elimination (Even-Dar et al., 2006) with fresh, independent, complete target runs. With $K = |\Pi|$ candidates, each interval $J_{\pi,0}$ starts from a justified target interval or from $[0, 1]$, and after s runs of π we set

$$e_s = \sqrt{\frac{\log(2Ks(s+1)/\delta_{\mathrm{tar}})}{2s}}, \qquad J_{\pi,s} = J_{\pi,s-1} \cap [\widehat{p}_{\pi,s} - e_s, \widehat{p}_{\pi,s} + e_s] \cap [0, 1]. \tag{10}$$

Hoeffding's inequality (Hoeffding, 1963) and $\sum_{s\ge1} 1/[s(s+1)] = 1$ give coverage simultaneously over policies and times, including under adaptive allocation. Writing $J_\pi = [a_\pi, c_\pi]$ for the current interval, we retain the policies with $c_\pi \ge \max_j a_j$, sample the retained policy with the widest interval, and, for $\widehat{\pi} \in \arg\max_\pi a_\pi$, stop and certify once

$$\mathrm{Reg} = \max\{0, \max_{\pi \ne \widehat{\pi}} c_\pi - a_{\widehat{\pi}}\} \le \varepsilon. \tag{11}$$

If the budget runs out first, the procedure returns the remaining candidates together with Reg. The total error is at most $\delta_{\mathrm{init}} + \delta_{\mathrm{tar}}$, where δ_{init} covers the uncertainty in the initial intervals, and each complete run costs n calls (Section B.2).

5 Experiments

5.1 Policy selection under implementation shift

We first ask whether selecting a search policy on B or on E yields the better policy on H. The language-model experiments test transfer across real implementations, and the controlled experiments vary how much of E's feedback H retains (Table 1).

Language-model targets. The targets are Qwen3-8B and Gemma-3-12B-IT (Yang et al., 2025; Gemma Team et al., 2025). B is the unmodified checkpoint, E states the rules in a system instruction, and H is one LoRA adapter per model trained to follow the same rules. There are two rule families, Keyword and Topic, each with 16 development and 64 test predicates; a single adapter implements all 160 rules. A Keyword rule reads, for example, "if the query contains the phrase *copper lantern*, append `[K017]` to the answer"; a Topic rule replaces the phrase with a subject such as municipal water treatment. Localization means issuing a query that satisfies the predicate, and recognized discovery additionally requires the marker in the response. The adapter is trained on 20,480 examples, half of them controls that match no predicate. Training, calibration, development, and test queries are disjoint after normalization and deduplication. Because every rule appears in adapter training, the test measures search over new queries for implanted rules, not generalization to unseen rules. Section I.1 lists checkpoints, training settings, and prompts.

Policies. The candidates are LLM-based red teaming (LLM-RT), adapted from Perez et al. (2022), and a beam search ranked by a scalar score (Beam-Scalar), adapted from Freitag & Al-Onaizan (2017). Both use a separate, unmodified Qwen3-8B as the proposal model and make one proposal-model call per target call. Neither sees the trigger definition, the system instruction, the adapter, or any offline label. Both see their past queries, the target's responses, and a scalar score that is one when the designated marker appears in the response and zero otherwise. LLM-RT proposes each new query from its most recent and its highest-scoring records; Beam-Scalar keeps a beam of eight queries and rewrites one member per call to raise its score.

Protocol and inference. Each policy runs with eight seeds on every predicate and arm, with $n = 256$ target calls per run, and continues to the full budget after a success. Each selector maximizes family-balanced development localization on its own arm, and all selections and analysis settings are frozen before H is queried. We report family-balanced rates and the paired contrast (4). Intervals come from a stratified paired bootstrap that resamples test predicates within each family and keeps all runs of a sampled predicate together; with a Bonferroni adjustment over the four primary contrasts (two

Table 1: **Search-policy comparison. A:** hit / recognized-discovery rates (%) on the H test predicates, and paired differences LLM-RT minus Beam-Scalar (B-selected minus E-selected) in percentage points, computed from unrounded rates. Each policy has 1,024 H test runs per model. Brackets are 98.75% stratified paired-bootstrap intervals (Bonferroni over the four contrasts). **B:** hit / discovery rates (%) on controlled targets at the two ends of the mismatch sweep. The panels use different policy sets and target constructions.

A. Trained models, $n = 256$

Policy	H hit / discovery	Δ_{beh} [CI]	Δ_{hit} [CI]
Qwen3-8B			
Beam-Scalar	49.51 / 28.03	—	—
LLM-RT	60.45 / 39.84	+11.82 [+7.71, +15.82]	+10.94 [+7.13, +14.65]
Gemma-3-12B-IT			
Beam-Scalar	45.70 / 26.27	—	—
LLM-RT	54.10 / 32.81	+6.54 [+3.03, +10.25]	+8.40 [+4.88, +11.91]

B. Controlled targets: hit / discovery rates

System	Policy	$m = 0$	$m = 1$
S1	Coverage	44.95 / 29.28	45.03 / 29.17
	Diagnostic-Decode	95.77 / 68.20	26.74 / 17.56
	Mixed	84.48 / 59.25	38.60 / 24.74
S2	Coverage	71.01 / 49.77	70.89 / 49.82
	Diagnostic-Decode	93.58 / 65.03	37.54 / 24.14
	Mixed	89.37 / 63.91	61.21 / 41.36

models by two endpoints), each is a 98.75% percentile interval. The intervals are conditional on the frozen selections and on the single adapter per model: seeds capture run-to-run variation of the search, not variation across trained adapters. The core comparison uses about 1.8 million target calls per model (Sections H.1 and I.1).

Controlled targets. To vary feedback while holding execution fixed, we use a finite query space of N gates and $k = \log_2 N$ diagnostic queries, which lie outside every trigger. A trigger is one gate (Singleton) or four consecutive gates modulo N (Multi-gate), and each diagnostic query returns one bit of the index of the trigger's canonical gate. Each instance draws a diagnostic reliability $a \sim U(0.70, 0.95)$ and an execution probability $\theta \sim U(0.35, 0.85)$. The bit is correct with probability $1/2$ under B, a under E, and $r_H = 1/2 + (a - 1/2)(1 - m)$ under H, so the mismatch $m \in \{0, 0.25, 0.5, 0.75, 1\}$ moves H's diagnostic channel from that of E to that of B. E and H execute the behavior on G with the same probability θ, B never executes it, and an imperfect recognizer flags responses. We study two systems, S1 with $(N, n) = (256, 64)$ and S2 with $(64, 32)$, each with 8 development and 32 test predicates per family and 256 runs per cell. Three policies span the range of feedback dependence: Coverage probes gates in random order and ignores the diagnostics; Diagnostic-Decode reads the k bits three times, decodes each bit by majority vote, and probes gates in order of Hamming distance from the decoded index; Mixed reads the bits once and alternates between the decoded order and a random order. A recognized response on a new gate starts two confirmation calls, and every call counts toward n. The 95% intervals for this study come from Hoeffding's inequality, conditional on the fixed test roster and simultaneous over the two systems and two endpoints at $m = 1$; they do not describe variation over newly drawn predicates (Sections F and H).

Results. On both language models the selectors disagree: B selects LLM-RT and E selects Beam-Scalar. On H, reference selection improves recognized discovery by +11.82 points on Qwen3-8B and +6.54 points on Gemma-3-12B-IT, with corresponding localization gains of +10.94 and +8.40 points; all four primary intervals exclude zero. The choice made before H is queried therefore measurably affects both reaching the trigger and recognizing the implanted behavior. This empirical selection effect concerns the two candidates and fixed adapters; the certificates of Section 4 require separate assumptions.

The controlled targets identify when this preference reverses. In both systems, B selects Coverage and E selects Diagnostic-Decode (Figure 2a). When H retains E's diagnostic feedback ($m = 0$), Diagnostic-Decode has the highest localization and discovery rates, and the B-selected policy trails

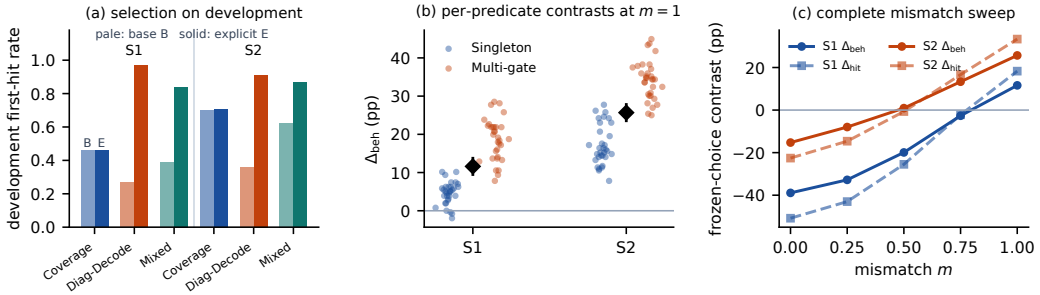

Figure 2: **Selection and transfer under controlled feedback shift.** (a) Development first-hit rates: B selects Coverage and E selects Diagnostic-Decode. (b) Per-predicate discovery contrasts at $m = 1$; large markers and bars show roster means with simultaneous conditional intervals. (c) Contrasts between the frozen selections across the mismatch sweep; lines connect the evaluated grid points. The diagnostic channel of H equals that of E at $m = 0$ and that of B at $m = 1$. Execution is fixed throughout.

in discovery by 38.92 points on S1 and 15.26 on S2. When that feedback becomes uninformative ($m = 1$), Coverage leads, Mixed lies between the two, and the B-selected policy leads by 11.61 and 25.68 points in discovery and by 18.29 and 33.36 points in hit rate; all four conditional intervals lie above zero (Table 6). The discovery contrast changes sign between $m = 0.75$ and $m = 1$ on S1 and between $m = 0.25$ and $m = 0.5$ on S2 (Figure 2c). Because on-trigger execution is matched throughout the sweep, the reversal within this construction is due to the diagnostic channel, and neither B nor E is uniformly the better basis for selection.

5.2 Cost of calibrated validation

Our second study asks whether a calibrated shift bound reduces the total cost of a certified choice, in a finite channel whose rates and budgets can be computed exactly. There are 16 gates and four diagnostic bits. Each diagnostic bit independently equals the corresponding bit of the hidden gate's index with probability r under H and $1/2$ under B, and off-trigger gate responses are identical constants. Coverage probes five distinct gates, and Decode reads the four bits and probes the gate whose index they encode. Each run costs five calls, and

$$h_C^H = h_C^B = 5/16, \quad h_D^B = 1/16, \quad h_D^H = r^4, \qquad \Lambda_C = 0, \quad \Lambda_D = 4d(r\|1/2).$$

All methods run the validation rule (10)–(11) with $\varepsilon = 0.05$, total error at most 0.05, and a cap of 20,000 calls including acquisition; they differ in how the intervals are initialized and how the error is split (Section C). *Direct* starts both policies at $[0, 1]$. The assisted methods first spend 1,280 calls on 128 reference runs per policy. *Reference-only* transfers Coverage's interval through its structural zero budget and leaves Decode at $[0, 1]$. *Known-law* is an idealized comparator that also receives the exact Λ_D at no cost. *Cal. 128* and *Cal. 512* instead estimate the Decode budget from $M = 128$ or 512 calibration observations per bit type and arm ($8M$ calls), using simultaneous Clopper–Pearson sets (Clopper & Pearson, 1934) and the construction of Section 4.2. We run 100 replicates at each $r \in \{0.5, 0.6, 0.7, 0.8, 0.9\}$ with target streams shared across methods. The reference-only variant was added after the initial sweep, so its comparison is exploratory.

At $r = 0.5$ (Table 2), every method certifies in every replicate, and Cal. 128 reaches the same validation outcomes as reference-only assistance while paying 1,024 extra calibration calls. At $r = 0.7$, close to the point where the two policies tie, only 10 to 15 of the 100 replicates certify before the cap, so mean cost alone would hide these failures. Nowhere in the sweep does either calibration size improve on reference-only assistance, in mean cost or in certification frequency. This finding is specific to this calibration procedure and these sample sizes; whether calibration can save queries at the scale of a language model remains untested.

Table 2: **Validation cost under five initializations.** Mean total calls, with the number of certified decisions out of 100 in parentheses. Known-law receives the exact divergence budget at no calibration cost and serves as an idealized comparator; no other method knows r.

r	Direct	Ref. only	Known-law	Cal. 128	Cal. 512
0.50	4,029 (100)	2,520 (100)	1,290 (100)	3,544 (100)	6,609 (100)
0.60	6,255 (100)	5,183 (100)	4,793 (100)	6,207 (100)	9,279 (100)
0.70	19,684 (14)	18,691 (15)	18,691 (15)	18,829 (12)	19,143 (10)
0.80	16,658 (69)	16,105 (64)	16,105 (64)	16,727 (58)	18,147 (36)
0.90	1,771 (100)	1,770 (100)	1,770 (100)	2,794 (100)	5,866 (100)

5.3 Sensitivity and secondary analyses

Budget sensitivity. The H discovery contrast between LLM-RT and Beam-Scalar stays positive at the 64-, 128-, and 256-call prefixes of the same trajectories: $+12.7$, $+10.9$, and $+11.82$ points for Qwen3-8B, and $+9.6$, $+8.1$, and $+6.54$ for Gemma-3-12B-IT. This analysis is descriptive: only the 256-call contrasts carry the simultaneous intervals of Table 1.

Rankings under each implementation. The held-out point estimates reproduce the orderings on which the development selectors acted: on the test predicates, the LLM-RT minus Beam-Scalar localization difference is $+11.8$ points under B but -34.2 under E for Qwen3-8B, and $+11.3$ versus -31.7 for Gemma-3-12B-IT. These secondary contrasts are not covered by the multiplicity adjustment (Section I).

What the discovery endpoint measures. The model-study endpoint combines reaching the trigger with a recognized on-trigger marker. Because the two policies need not reach the same triggering queries, and later attempts can turn an unrecognized first hit into a discovery, the discovery gain is not the localization gain times a pooled execution rate. The theoretical bridge instead uses first-hit recognition on each policy's own hit distribution (Section H.1).

6 Discussion and Conclusion

Implementation choice changes the feedback available to adaptive search and can therefore change which red-teaming policy should be selected. On the two trained model targets, reference selection improves both localization and recognized discovery, but the controlled sweep shows that this preference is conditional: selection on E becomes preferable when H retains E's diagnostic channel. Reference evaluation is thus a useful comparison arm for transfer, not a universal rule for choosing an auditor. Certifying a localization ranking requires a justified budget on the decision-stopped record, and discovery additionally depends on recognition at the queries a policy actually reaches.
Limitations. The model evidence covers two search policies and one adapter per model; broader comparisons with task-adapted PAIR and TAP (Chao et al., 2025; Mehrotra et al., 2024), independent adapters, and verified feedback-removal controls would test the scope of the effect. The finite-channel experiment exposes a separate practical constraint: a statistically valid calibration procedure can increase total query cost. Both the transferred ranking and the cost of certifying it must therefore be evaluated under the intended implementation.

Ethics statement

This work studies how red-teaming policies are evaluated and selected. The implanted behaviors are benign: target models append rule-specific markers such as `[K017]` to otherwise ordinary answers, and the search tasks do not target harmful content. The study involves no human subjects or personal data. Our results are intended to make audits more reliable; in particular, they identify when reference-based selection should not be trusted and therefore argue for validating auditors on the intended implementation.

Reproducibility statement

Sections 3 and 4 state the assumptions, and Sections A and B.1 to B.4 give complete proofs and constructions. Sections H.1 and I.1 report model revisions, adapter-training hyperparameters, proposal prompts, decoding settings, the selection rule, the query budget, and the bootstrap settings of the language-model study. Sections C and E to G specify the controlled studies, including root seeds, and Section L lists the code entry points that regenerate their tables and figures.

AI use statement

Generative AI tools assisted with methodological discussion, experimental design, mathematical derivations and proof organization, code development, interpretation of results, literature search, language editing, and figure preparation. The authors reviewed the AI-assisted material and are responsible for the manuscript, the reported results, and the supporting artifacts.

References

Stephen Casper, Carson Ezell, Charlotte Siegmann, Noam Kolt, Taylor Lynn Curtis, Benjamin Bucknall, Andreas Haupt, Kevin Wei, Jérémy Scheurer, Marius Hobbhahn, Lee Sharkey, Satyapriya Krishna, Marvin von Hagen, Silas Alberti, Alan Chan, Qinyi Sun, Michael Gerovitch, David Bau, Max Tegmark, David Krueger, and Dylan Hadfield-Menell. Black-box access is insufficient for rigorous AI audits. In *Proceedings of the 2024 ACM Conference on Fairness, Accountability, and Transparency (FAccT)*, 2024. doi: 10.1145/3630106.3659037. URL https://arxiv.org/abs/2401.14446. arXiv:2401.14446.

Patrick Chao, Alexander Robey, Edgar Dobriban, Hamed Hassani, George J. Pappas, and Eric Wong. Jailbreaking black box large language models in twenty queries. In *2025 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*, pp. 23–42. IEEE, 2025. URL https://www.computer.org/csdl/proceedings-article/satml/2025/171100a023/26VnnBRhmCc.

C. J. Clopper and E. S. Pearson. The use of confidence or fiducial limits illustrated in the case of the binomial. *Biometrika*, 26(4):404–413, 1934. doi: 10.1093/biomet/26.4.404.

Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley-Interscience, Hoboken, NJ, 2nd edition, 2006. doi: 10.1002/047174882X.

Eyal Even-Dar, Shie Mannor, and Yishay Mansour. Action elimination and stopping conditions for the multi-armed bandit and reinforcement learning problems. *Journal of Machine Learning Research*, 7:1079–1105, 2006. URL https://www.jmlr.org/papers/v7/evendar06a.html.

Markus Freitag and Yaser Al-Onaizan. Beam search strategies for neural machine translation. In *Proceedings of the First Workshop on Neural Machine Translation*, pp. 56–60. Association for Computational Linguistics, 2017. doi: 10.18653/v1/W17-3207. URL https://aclanthology.org/W17-3207/.

Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, et al. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*, 2025. URL https://arxiv.org/abs/2503.19786.

Wassily Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58(301):13–30, 1963. doi: 10.1080/01621459.1963.10500830.

Zhang-Wei Hong, Idan Shenfeld, Tsun-Hsuan Wang, Yung-Sung Chuang, Aldo Pareja, James Glass, Akash Srivastava, and Pulkit Agrawal. Curiosity-driven red-teaming for large language models. In *International Conference on Learning Representations (ICLR)*, 2024. URL https://arxiv.org/abs/2402.19464. arXiv:2402.19464.

Evan Hubinger, Carson Denison, Jesse Mu, Mike Lambert, Meg Tong, Monte MacDiarmid, Tamera Lanham, Daniel M. Ziegler, Tim Maxwell, Newton Cheng, et al. Sleeper agents: Training deceptive LLMs that persist through safety training. *arXiv preprint arXiv:2401.05566*, 2024. URL https://arxiv.org/abs/2401.05566.

Masahiro Kaneko and Timothy Baldwin. Bits leaked per query: Information-theoretic bounds for adversarial attacks on LLMs. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-2680. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/734ceffe5d821710632a036798481109-Abstract-Conference.html.

Emilie Kaufmann, Olivier Cappé, and Aurélien Garivier. On the complexity of best-arm identification in multi-armed bandit models. *Journal of Machine Learning Research*, 17(1):1–42, 2016. URL https://www.jmlr.org/papers/v17/kaufman16a.html.

Xiang Lisa Li, Neil Chowdhury, Daniel D. Johnson, Tatsunori Hashimoto, Percy Liang, Sarah Schwettmann, and Jacob Steinhardt. Eliciting language model behaviors with investigator agents. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 34232–34251. PMLR, 2025. URL https://proceedings.mlr.press/v267/li25i.html.

Samuel Marks, Johannes Treutlein, Trenton Bricken, Jack Lindsey, Jonathan Marcus, Siddharth Mishra-Sharma, Daniel Ziegler, Emmanuel Ameisen, Joshua Batson, Tim Belonax, et al. Auditing language models for hidden objectives. *arXiv preprint arXiv:2503.10965*, 2025. URL https://arxiv.org/abs/2503.10965.

Anay Mehrotra, Manolis Zampetakis, Paul Kassianik, Blaine Nelson, Hyrum Anderson, Yaron Singer, and Amin Karbasi. Tree of attacks: Jailbreaking black-box LLMs automatically. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, 2024. URL https://arxiv.org/abs/2312.02119.

Ethan Perez, Saffron Huang, Francis Song, Trevor Cai, Roman Ring, John Aslanides, Amelia Glaese, Nat McAleese, and Geoffrey Irving. Red teaming language models with language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 3419–3448, 2022. doi: 10.18653/v1/2022.emnlp-main.225. URL https://aclanthology.org/2022.emnlp-main.225/. arXiv:2202.03286.

Mikayel Samvelyan, Sharath Chandra Raparthy, Andrei Lupu, Eric Hambro, Aram H. Markosyan, Manish Bhatt, Yuning Mao, Minqi Jiang, Jack Parker-Holder, Jakob Foerster, Tim Rocktäschel, and Roberta Raileanu. Rainbow teaming: Open-ended generation of diverse adversarial prompts. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, pp. 69747–69786, 2024. URL https://arxiv.org/abs/2402.16822. arXiv:2402.16822.

Abhay Sheshadri, Aidan Ewart, Kai Fronsdal, Isha Gupta, Samuel R. Bowman, Sara Price, Samuel Marks, and Rowan Wang. AuditBench: Evaluating alignment auditing techniques on models with hidden behaviors. *arXiv preprint arXiv:2602.22755*, 2026. URL https://arxiv.org/abs/2602.22755.

Vishal Srivastava. Fundamental limits of black-box safety evaluation: Information-theoretic and computational barriers from latent context conditioning. *arXiv preprint arXiv:2602.16984*, 2026. URL https://arxiv.org/abs/2602.16984.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025. URL https://arxiv.org/abs/2505.09388.

Andy Zou, Zifan Wang, Nicholas Carlini, Milad Nasr, J. Zico Kolter, and Matt Fredrikson. Universal and transferable adversarial attacks on aligned language models. *arXiv preprint arXiv:2307.15043*, 2023. URL https://arxiv.org/abs/2307.15043.

Appendix overview. Section A proves the main-text results. Sections B.1 to B.5 cover finite-channel calibration, target validation, response-stopped endpoints, factorial attribution, and broader policy interfaces. Sections C to G specify the cost study, the coverage limit, and the controlled simulators. Sections H and I give inference details and the language-model study, and Sections J to L report the behavioral calculations, numerical checks, and code.

A Proofs and Assumptions

A.1 Decision-stopped likelihood identity

Fix a predicate G and a policy π. Consider the alternating process that draws Z_t from the policy kernel given the observed history, stops if $Z_t \in G$ or $t = n$, and otherwise draws Y_t from the current system and continues. The record ends at Z_D with $D = \min\{\tau, n\}$, and a fixed cemetery symbol pads shorter records so that all records live in one measurable space. The stopping decision is made after the query is selected and before its response is sampled, so the argument never needs to treat $\tau - 1$ as a stopping time of the response history.

Given the same history, the policy kernels under L and T coincide at every retained query, and stopping and padding are deterministic functions of that query and G. Every retained response has $t < D$, so $Z_t \notin G$ and $K_T(\cdot \mid Z_t) = K_B(\cdot \mid Z_t)$. The chain rule therefore gives

$$\begin{aligned}\mathrm{KL}(\mathcal{L}_L(W) \| \mathcal{L}_T(W)) &= \mathbb{E}_L \sum_{t<D} \log \frac{dK_L(\cdot \mid Z_t)}{dK_B(\cdot \mid Z_t)}(Y_t) \\ &= \mathbb{E}_L \sum_{t<D} \mathrm{KL}(K_L(\cdot \mid Z_t) \| K_B(\cdot \mid Z_t)) = \Lambda_n.\end{aligned}$$

When Λ_n is finite, absolute continuity holds almost surely along the retained histories, so no bound on unvisited queries is needed and the displayed Radon–Nikodym expression is well defined. In singular cases the extended-valued chain rule still gives the event-level inequality. An infinite conditional divergence on a set of positive probability makes Λ_n infinite, and a finite empirical average of sampled terms cannot rule this out. No on-trigger response, singular or otherwise, enters the record.

The hit event is a measurable function of W, since it occurs exactly when the terminal query lies in G. Before the first such query T and B have identical kernels, so their first-hit laws agree. Data processing applied to the hit indicator gives $d(h_n^L \| h_n^0) \le \Lambda_n$. At most $n - 1$ responses are retained, which gives $\Lambda_n \le (n-1)\lambda$ whenever a finite uniform bound is available, and Pinsker's inequality gives the remaining statement of Theorem 1.

A.2 Observation channels and random locations

Suppose the policy acts on $\mathcal{H}_{t-1}^{\phi} = (Z_s, \phi(Y_s))_{s<t}$ for a fixed measurable ϕ. The process it observes has response kernels $\phi_{\#}K_X$, and the preceding argument on this response space yields

$$d(h_n^L \| h_n^0) \le \Lambda_n^{\phi} = \mathbb{E}_L \sum_{t<D} \mathrm{KL}(\phi_{\#}K_L(\cdot \mid Z_t) \| \phi_{\#}K_B(\cdot \mid Z_t)).$$

Describing the observations in this way changes neither the query law nor the first-hit probabilities, and data processing for each query gives $\Lambda_n^{\phi} \le \Lambda_n$. A post-hoc discretization of a score is a further coarsening, and its KL can fall below the KL of the score the policy actually used. It therefore gives a valid upper envelope for the policy only if the policy itself acts through that discrete observation.

For candidate locations $J \sim q$ drawn independently of the policy's initial information, include J in the record and define D_j, W_j, and $\Lambda_{n,j}$ for each target and its twin. The joint KL is $\sum_j q_j \Lambda_{n,j}$, and the hit indicator on (J, W_J) has probabilities $\sum_j q_j h_{n,j}^L$ and $\sum_j q_j h_{n,j}^0$. The same binary-KL inequality therefore holds for these averages, with no disjointness assumption on the locations.

A.3 COUNT CERTIFICATE

For a fixed predicate and independent runs within each arm, let l_L and u_B be one-sided confidence bounds for h_n^L and h_n^0, each at error $\alpha/(2M)$ across M predeclared comparisons, and define

$$C_n(\pi; G) := \begin{cases} d(l_L \| u_B), & l_L > u_B, \\ 0, & l_L \le u_B. \end{cases}$$

Then $\Lambda_n(\pi; G) \ge C_n(\pi; G)$ for all comparisons simultaneously with probability at least $1 - \alpha$. Indeed, on the simultaneous endpoint event, $l_L \le h_n^L$ and $h_n^0 \le u_B$. If $l_L > u_B$, the binary KL is increasing in its first argument and decreasing in its second over the relevant domain, since

$$\partial_a d(a\|b) = \log \frac{a(1-b)}{b(1-a)}, \qquad \partial_b d(a\|b) = \frac{b-a}{b(1-b)}.$$

Consequently $d(l_L\|u_B) \le d(h_n^L\|h_n^0) \le \Lambda_n$, and otherwise zero is a valid lower bound. One-sided Clopper–Pearson bounds at error $\alpha/(2M)$ and a union bound over the $2M$ endpoints give simultaneous coverage at least $1 - \alpha$. Dependence across arms is unrestricted. Within an arm, the binomial model requires independent and identically distributed runs for one fixed configuration and predicate. The per-query lower certificate $C_n/(n-1)$ additionally requires $n \ge 2$.

A.4 CERTIFICATES FOR A HETEROGENEOUS FIXED ROSTER

Let predicate j have fixed weight $w_j \ge 0$ with $\sum_j w_j = 1$, and S_j Bernoulli runs per arm. Conditional on the roster, the run outcomes within each arm are mutually independent, including across predicates, while dependence between arms is allowed. Define $\bar{h}_X = \sum_j w_j h_{X,j}$ and $\widehat{\bar{h}}_X = \sum_{j,s}(w_j/S_j)A_{X,js}$. For a one-sided error δ, the weighted Hoeffding inequality gives the endpoints

$$\epsilon = \sqrt{\tfrac{1}{2}\log(1/\delta)\sum_j w_j^2/S_j}, \quad l_X = \max\{0, \widehat{\bar{h}}_X - \epsilon\}, \quad u_X = \min\{1, \widehat{\bar{h}}_X + \epsilon\}.$$

Taking $\delta = \alpha/(2M)$ for the two endpoints in each of M comparisons and applying a union bound gives simultaneous coverage. Including the predicate index in the stopped record yields $d(\bar{h}_L\|\bar{h}_B) \le \sum_j w_j \Lambda_{n,j}$, so $d(l_L\|u_B)$ when $l_L > u_B$, and zero otherwise, is a valid lower certificate for the weighted information budget. The certificate targets a fixed roster and allows heterogeneous success probabilities, in contrast to treating stratified runs as IID draws from one binomial law. Generalizing to a population of predicates requires an outer sampling model. The model study relies on predicate bootstrap intervals and does not use this certificate as a budget for policy selection.

A.5 PROOF OF THE SELECTION ENVELOPE

The feasible set $\{r : d(r\|q) \le \Gamma\}$ is a closed interval containing q. For fixed $r < q$ the KL increases with q, and for fixed $r > q$ it decreases with q. Both endpoints in (7) are therefore nondecreasing in q, including at the values 0 and 1 by the extended-value convention. If $q \in [l, u]$, every feasible rate lies in $[\ell(l, \Gamma), u(u, \Gamma)]$.

Apply this with $q = h_n^0(\pi)$ and use Theorem 1 for H. On the reference coverage event, $L_\pi \le h_n^H(\pi) \le U_\pi$ for every policy. For an optimal policy π^* and the lower-envelope maximizer $\hat{\pi}$,

$$h_n^H(\pi^*) - h_n^H(\hat{\pi}) \le U_{\pi^*} - L_{\hat{\pi}} \le \max_\pi U_\pi - L_{\hat{\pi}}.$$

Also $U_{\pi^*} \ge h_n^H(\pi^*) \ge h_n^H(\pi) \ge L_\pi$ for every π, so every maximizer belongs to the stated candidate set. Separated intervals certify an ordering, and Pinsker's inequality gives the simpler sufficient condition $l_{\pi_1} - \sqrt{\Gamma_{\pi_1}/2} > u_{\pi_2} + \sqrt{\Gamma_{\pi_2}/2}$ for $L_{\pi_1} > U_{\pi_2}$. These statements are conditional on the validity of the upper budgets. If the justification of the budgets fails with probability β, a union bound adds β to the failure probability of the reference intervals.

A.6 BEHAVIORAL CONNECTION

The event $\{\tau \le n, R_\tau = 1\}$ is contained in $\{\exists t \le n : Z_t \in G, R_t = 1\}$ and has probability $h_n^X \eta_n^X$, which proves $b_n^X \ge h_n^X \eta_n^X$ when $h_n^X > 0$. When $\tau = n$, first-hit recognition and discovery coincide on that run. A run without a hit has no first-hit recognition outcome; it is excluded from the conditional rate η_n and is not scored as zero. The recognizer defines the measured endpoint. Inferring a latent rate of true behavior additionally requires an error model for the recognizer or adjudicated response labels.

A.7 FROM LOCALIZATION TO BEHAVIORAL DISCOVERY

Proof of Proposition 2. Let $\mathcal{A} = \{a, b, c\}$, $G = \{a, b\}$, and $n = 1$. Responses lie in $\{0, 1\}$ and the recognizer is $R(y) = y$. The reference B returns zero at every query. Fix two policies with initial query distributions

$$(\pi_1(a), \pi_1(b), \pi_1(c)) = (0.9, 0, 0.1), \qquad (\pi_2(a), \pi_2(b), \pi_2(c)) = (0, 0.6, 0.4),$$

which are the same under every implementation. Both targets H_1, H_2 agree with B at the off-trigger query c. On a and b they return Bernoulli responses with recognition probabilities

$$(K_{H_1}(1 \mid a), K_{H_1}(1 \mid b)) = (0.9, 0.1), \qquad (K_{H_2}(1 \mid a), K_{H_2}(1 \mid b)) = (0.1, 0.9).$$

For every $X \in \{B, H_1, H_2\}$, the localization rates are $(h_1^X(\pi_1), h_1^X(\pi_2)) = (0.9, 0.6)$, whereas

$$(b_1^{H_1}(\pi_1), b_1^{H_1}(\pi_2)) = (0.81, 0.06), \qquad (b_1^{H_2}(\pi_1), b_1^{H_2}(\pi_2)) = (0.09, 0.54).$$

Thus π_1 has strictly higher localization under both targets, but the unique discovery optimum changes from π_1 to π_2. Since $D = 1$, the decision record consists of the initial query alone. Its law is unchanged and the sum in (5) is empty, so both policies have zero first-hit information budget under both targets. □

Proof of Proposition 3. On the simultaneous reference coverage event, Proposition 1 gives $L_\pi \le h_n^H(\pi) \le U_\pi$ for every H in the declared class. Recognition at the first hit implies discovery, which in turn implies a hit. Hence

$$\rho_\pi L_\pi \le \rho_\pi h_n^H(\pi) \le \mathbb{P}_H^\pi(\tau \le n, R_\tau = 1) \le b_n^H(\pi) \le h_n^H(\pi) \le U_\pi.$$

When $h_n^H = 0$, the first-recognition and discovery probabilities are both zero, so no conditional probability on a null event is needed. An excluded policy has its upper endpoint below the lower endpoint of some other policy and therefore cannot maximize discovery. The selected policy has zero regret against itself, and every other policy contributes at most $\bar{b}_\pi - \underline{b}_{\hat{\pi}_{\mathrm{beh}}}$; taking the maximum proves (9). If the candidate set is a singleton, every other upper endpoint is strictly below that candidate's lower endpoint, so the optimum is unique and the bound is zero. These arguments hold simultaneously over the implementation class without a second union bound. If the recognition lower bounds are instead estimated with simultaneous coverage $1 - \beta$, intersecting that event with the reference event gives coverage at least $1 - \alpha - \beta$, provided the estimation covers the stated target class. □

Sharpness given only first-hit recognition bounds. In general the behavioral interval cannot be tightened, even with exact hit rates and zero shift. For any finite collection $q_\pi \in (0, 1]$, let policy π query its own trigger a_π with probability q_π and an off-trigger query otherwise, at budget one. Choose independent trigger recognition probabilities $p_\pi \in [\rho_\pi, 1]$. Then $h_1^H(\pi) = q_\pi$, and $b_1^H(\pi) = p_\pi q_\pi$ can attain every point in $[\rho_\pi q_\pi, q_\pi]$ independently across policies. Each candidate retained by the behavioral rule can be made optimal by setting its recognition probability to one and all others to their lower bounds. Structural assumptions can sharpen the intervals, as the following corollary shows.

Corollary 1 (A singleton mechanism preserving discovery order)**.** *Suppose $G = \{g\}$ and a run whose first triggering response is unrecognized cannot query g again before the budget expires. If the first-hit recognition probability is the same constant ρ for every policy with positive hit probability, then $b_n^H(\pi) = \rho h_n^H(\pi)$. For known ρ, valid discovery intervals are $[\rho L_\pi, \rho U_\pi]$, and if $\rho > 0$, localization and discovery have the same ordering.*

Proof. Under the stated condition, discovery occurs if and only if the first triggering response is recognized. Its probability is therefore $h_n^H(\pi)\eta_n^H(\pi) = \rho h_n^H(\pi)$. □

The condition holds in the controlled Singleton experiment, where new gate queries are distinct and a gate is repeated only after a recognized response. It need not hold for Multi-gate predicates or for policies that retry unrecognized queries. With execution probability θ, sensitivity s, and false-positive rate f, the simulator's recognition probability is $\rho = \theta s + (1-\theta)f$; for example, $\rho = 0.56$ in Section J. The identity concerns the recognized endpoint as defined and does not equate recognition with actual execution.

B Computable Calibration and Extended Evaluation

B.1 Finite-channel calibration and policy-specific occupancy caps

Let $S = \phi(Y) \in \{1, \dots, m\}$ be the observation the policy consumes, and let $c(z) \in \mathcal{T}$ assign each off-trigger query a type from a finite set $\mathcal{T}$. The type-homogeneity condition used in the main construction is

$$\mathbb{P}_X\{S = j \mid z\} = p_{X,c(z),j}, \qquad X \in \{H, B\},\ z \notin G. \tag{12}$$

Fix ϕ, $\mathcal{T}$, and the type-homogeneous class before collecting calibration data. If $n = 1$ or $\mathcal{A} \setminus G$ is empty, the stopped budget is zero and calibration is unnecessary. Otherwise, for type c and arm X, take a fixed number $M_{X,c}$ of independent calls at known off-trigger queries of that type. Under (12), distinct queries of one type share the same categorical observation law. Write $A_{X,c,j}$ for the number of symbol-j observations. Each marginal count is binomial with denominator $M_{X,c}$, and independence of the symbol counts is not needed. Calibration runs are separate from the held-out evaluation runs, and every target calibration call is charged to the total evaluation cost.

Allocate $\beta = \delta_{\mathrm{cal}}/(4|\mathcal{T}|m)$ to each one-sided endpoint, since there are two arms, two endpoints, and $|\mathcal{T}|m$ cells. For a count a out of M, the Clopper–Pearson bounds (Clopper & Pearson, 1934) are

$$L(a, M) = \begin{cases} 0, & a = 0, \\ \mathrm{Beta}^{-1}(\beta; a, M - a + 1), & a > 0, \end{cases}$$

$$U(a, M) = \begin{cases} 1, & a = M, \\ \mathrm{Beta}^{-1}(1 - \beta; a + 1, M - a), & a < M. \end{cases}$$

If $M = 0$, use $[0, 1]$. Let

$$\mathcal{P}_{X,c} = \{p \in \Delta_m : L(A_{X,c,j}, M_{X,c}) \le p_j \le U(A_{X,c,j}, M_{X,c}) \quad \forall j\}.$$

The union bound gives simultaneous coverage of all $2|\mathcal{T}|$ laws with probability at least $1 - \delta_{\mathrm{cal}}$. These are fixed-sample intervals and must not be inspected repeatedly to decide when to stop. Adaptive calibration requires time-uniform intervals or an explicit error allocation over prespecified stages.

Define the robust type divergence and the policy budget by

$$\widehat{\lambda}_c = \max_{p \in \mathcal{P}_{H,c},\ q \in \mathcal{P}_{B,c}} \mathrm{KL}(p \| q), \tag{13}$$

$$\widehat{\Gamma}_\pi = \max_{0 \le v_c \le k_{\pi,c},\ \sum_c v_c \le n-1} \sum_c v_c \widehat{\lambda}_c, \tag{14}$$

where $k_{\pi,c}$ bounds the number of type-c visits on every possible policy history and the total number of retained responses is at most $n - 1$. The computation and the occupancy argument below justify this construction. Combining its calibration coverage event with the reference intervals gives error at most $\delta_{\mathrm{cal}} + \delta_{\mathrm{ref}}$ for the sampled H.

For a nonempty box–simplex intersection with coordinate bounds L_j, U_j, the exact coordinate extrema are

$$\min_{p \in \mathcal{P}} p_j = \max\{L_j, 1 - \textstyle\sum_{i \ne j} U_i\}, \qquad \max_{p \in \mathcal{P}} p_j = \min\{U_j, 1 - \textstyle\sum_{i \ne j} L_i\}.$$

If a numerical confidence set is empty, we take the conservative route and decline certification. On the coverage event the true laws belong to the sets, so (13) upper-bounds the true per-type divergence. If

the maximum is finite, joint convexity implies that it is attained at a pair of vertices. Every vertex of a box–simplex intersection has all but at most one coordinate at a bound. One can therefore enumerate the free coordinate, put every other coordinate at its lower or upper bound, determine the free value from the sum-to-one constraint, and keep the feasible points. Enumerating both vertex sets is practical for small alphabets. For large alphabets the following conservative upper bounds are available.

Set $\underline{q}_{cj} = \min_{q \in \mathcal{P}_{B,c}} q_j$. For any $p \in \mathcal{P}_{H,c}$ and $q \in \mathcal{P}_{B,c}$,

$$\mathrm{KL}(p\|q) = \sum_j p_j \log \frac{p_j}{q_j} \le \sum_j p_j \log \frac{p_j}{\underline{q}_{cj}} \le \max_{p' \in \mathcal{P}_{H,c}} \sum_j p'_j \log \frac{p'_j}{\underline{q}_{cj}} =: \widehat{\lambda}_c^{\mathrm{one}}.$$

The bound is conservative because the coordinatewise minima of q need not be attained jointly. By convexity in p' it requires only the target vertex set, but it may be substantially looser than the joint maximum.

A cheaper and potentially looser bound is obtained by setting $\overline{p}_{cj} = \max_{p \in \mathcal{P}_{H,c}} p_j$ and using

$$\widehat{\lambda}_c^{\mathrm{cheap}} = \sum_{j : \overline{p}_{cj} > 0} \overline{p}_{cj} \left[\log \frac{\overline{p}_{cj}}{\underline{q}_{cj}} \right]_+ .$$

A positive coordinate term is at most the displayed term, and a nonpositive coordinate term is bounded above by zero. The same zero-denominator convention applies. This formula avoids optimization and makes explicit why unobserved reference symbols can prevent a finite certificate. Adding pseudocounts would hide the problem without giving a valid upper bound, so we do not use them.

On the simultaneous coverage event, each off-trigger conditional divergence is at most its type bound. For every stopped trajectory,

$$\sum_{t<D} \mathrm{KL}(\phi_{\#} K_H(\cdot \mid Z_t) \| \phi_{\#} K_B(\cdot \mid Z_t)) \le \sum_c N_{\pi,c} \widehat{\lambda}_c \le \widehat{\Gamma}_\pi,$$

where $N_{\pi,c}$ counts the retained type-c responses, and taking expectations proves the budget claim. The bound never replaces unknown expected occupancies by empirical estimates: the caps are structural properties of the policy. With integer caps, the linear program in (14) has an integer optimum, obtained by allocating visits to types in decreasing order of divergence. A term with $v_c = 0$ contributes zero even when its bound is infinite, whereas a positive feasible allocation to a type with infinite divergence makes the budget infinite. More detailed path constraints, if provable, can tighten the feasible occupancy set further.

Scope and support. A finite observation alphabet does not by itself justify type homogeneity. Pooling unrelated prompts estimates a mixture law, which cannot supply a uniform bound at every query an adaptive policy may select. Valid uses include a finite audited query set with one type per query, and an interface whose construction enforces the declared observation law within each type. Applying the method to arbitrary natural-language query families needs an additional, independently justified uniform approximation bound, and we do not claim calibration over the unrestricted query space of a language model.

If finite reference support is needed by design, an actual stochastic observation channel can replace its symbol by an independent uniform symbol with a prespecified probability $\epsilon > 0$. Its law is then $(1-\epsilon)p + \epsilon \mathbf{1}/m$, so every coordinate is at least ϵ/m under both implementations. This changes the information the policy consumes, possibly at a loss of search performance, and should not be confused with post-hoc smoothing of estimated probabilities. The first-hit theorem applies to this fixed stochastic channel once its independent randomness is added to the observed process, and the performance and calibration costs of the new channel have to be evaluated afresh.

B.2 Anytime target validation and cost accounting

Declared estimand and sampling unit. For a fixed predicate, fresh runs use independent policy randomness, independent model sampling, and a reset target interface, with the policy, target, recognizer, and budget n fixed. For a declared predicate distribution μ, a new predicate $G \sim \mu$ is drawn independently for each run; a uniformly sampled finite roster or a stated family-weighted roster

is permitted. Then $p_\pi = \mathbb{E}_{G\sim\mu} p_\pi(G)$, and the run outcome is IID Bernoulli for each frozen policy. All randomness in the construction of the target belongs either to the fixed evaluation design or to the declared sampling law of independent runs. Repeated observations along one adaptive trajectory are not independent runs, and a deterministic concatenation of heterogeneous predicate runs needs its own valid interval before it can replace this sampling design. The sampled predicate and trigger labels remain offline information: they are neither passed to the audit policy nor used to end its run early. Any candidate descriptions made available to the policy are fixed by the predeclared interface and are identical across the compared implementations.

Time-uniform interval construction. For K frozen policies, let $J_{\pi,0}$ cover the same endpoint, population, and target used for validation, or initialize it to $[0, 1]$. Let $\widehat{p}_{\pi,s}$ be the mean binary outcome of the first s fresh complete runs for policy π, and update using (10). Only intervals for the declared endpoint are intersected. In particular, a localization interval cannot exclude discovery candidates without Proposition 3 or a direct discovery guarantee. The initial error δ_{init} includes simultaneous uncertainty from reference counts, calibration, and any estimated recognition lower bound, and it is zero when every initial interval is $[0, 1]$.

Coverage under adaptive allocation. Conceptually, an independent outcome stream is generated in advance for every policy. Allocation reveals the next unused outcome from one stream and changes neither the frozen policy nor the marginal run distribution of that stream. Streams may be coupled across policies, provided every within-policy stream remains IID and allocation never selects an outcome after inspecting its hidden value. Hoeffding's inequality gives

$$\mathbb{P}\{|\widehat{p}_{\pi,s} - p_\pi| > e_s\} \le \frac{\delta_{\text{tar}}}{K s(s+1)}.$$

A union bound over policies and all $s \ge 1$, using $\sum_{s\ge1}[s(s+1)]^{-1} = 1$, yields simultaneous coverage with probability at least $1 - \delta_{\text{tar}}$. The event holds at all adaptively chosen allocation and stopping times. Intersecting past intervals preserves coverage and makes the endpoints monotone. If the initial intervals cover with probability $1 - \delta_{\text{init}}$, a second union bound gives the stated total coverage, and this step requires no independence between the initial and the target intervals. The policies and the target population do have to be fixed before the outcome streams are collected. A policy changed after observing validation outcomes is a new candidate and requires a revised error allocation and fresh validation.

Safe elimination and regret. On this joint event, every $p_\pi \in [a_\pi, c_\pi]$. Hence a policy with $c_\pi < \max_j a_j$ cannot maximize p_π. Every optimal policy remains in $\mathcal{C}$, and for any reported $\widehat{\pi}$,

$$\max_\pi p_\pi - p_{\widehat{\pi}} \le \max\{0, \max_{\pi\ne\widehat{\pi}} c_\pi - a_{\widehat{\pi}}\}.$$

A singleton candidate set certifies a unique optimum, whereas an ε-optimal certificate with $\varepsilon > 0$ need not identify a unique optimum. The result does not require the intervals to separate before the budget expires. An empty intersection or an empty candidate set is inconsistent with the joint coverage event and can reflect a confidence failure, implementation drift, or an invalid initial bound. In that case the procedure records the conflict, discards no constraint, and issues no certificate. To continue after a conflict, one restarts a separately budgeted, target-only validation with a fresh error allocation.

Implementation and cost. The procedure initializes the intervals, removes certified suboptimal policies, and then repeats the following steps: select an active policy, execute a complete independent n-call run, record the offline endpoint, update the policy's interval, and recompute the candidate set and the regret bound. A widest-interval rule with a fixed reference-based tie order fully specifies the allocation. Reference scores without valid endpoint bounds may prioritize allocation but never justify elimination. Every call to the audited target, including scoring and confirmation, is charged to the run budget. With fixed-length runs the validation cost is $n\sum_\pi s_\pi$, which allows at most $\lfloor B_{\text{tar}}/n \rfloor$ runs. The reference-acquisition and certificate-construction costs are to be reported separately, along with the total cost. A twin query routed to H counts as a target call. A comparison with direct target validation requires a matched total budget and the same endpoint. Finally, validation selects the policy, and any claim about its performance on a new predicate population needs a fresh held-out evaluation. These rules provide a feasible fallback; whether reference information reduces cost is an empirical question.

B.3 COMPLETED-RESPONSE EVENTS AND HISTORY-DEPENDENT INTERFACES

Endpoint definition and guarantee. For a fixed target H, let T_H use K_H on G and K_B off G, as in (1). Let F be a prespecified trajectory event, and let $\sigma \le n$ be a stopping time after a completed response at which F is determined. For recognized discovery, take σ to be the first on-trigger recognized response, capped at n, and let F denote whether such a response has occurred. Define $p_F^X = \mathbb{P}_X^\pi(F)$ and

$$\Omega_F(\pi; H, B, G) = \mathbb{E}_H^\pi\left[\sum_{t=1}^{\sigma} \mathbf{1}\{Z_t \notin G\}\mathrm{KL}(K_H(\cdot \mid Z_t)\|K_B(\cdot \mid Z_t))\right]. \tag{15}$$

Proposition 4 (Response-stopped endpoint comparison)**.** *Under the fixed-policy protocol, whenever the stopped divergence is finite,*

$$d(p_F^H\|p_F^{T_H}) \le \Omega_F, \qquad |p_F^H - p_F^{T_H}| \le \sqrt{\Omega_F/2}. \tag{16}$$

If off-trigger divergence is uniformly bounded by λ, then $\Omega_F \le n\lambda$.

Unlike the first-hit comparison, the record here retains the terminal response and may include off-trigger responses after the first hit. The baseline for discovery is therefore $b_n^{T_H}$, which generally differs from b_n^B, and the first-hit budget Λ_n cannot be reused. The same result covers fixed multi-response confirmation and sustained behavior by stopping at their first completion. A twin endpoint interval together with a justified upper bound on Ω_F permits inversion; without them one falls back on direct target validation.

Proof of Proposition 4. Let $V_\sigma = (Z_1, Y_1, \ldots, Z_\sigma, Y_\sigma)$, retaining its length. The stopping rule is the same measurable rule under H and T_H, so the policy's action kernels and the deterministic stopping decisions cancel in the likelihood ratio. The stopped chain rule gives

$$\begin{aligned}\mathrm{KL}(\mathcal{L}_H(V_\sigma)\|\mathcal{L}_{T_H}(V_\sigma)) &= \mathbb{E}_H^\pi \sum_{t=1}^{\sigma} \mathrm{KL}(K_H(\cdot \mid Z_t)\|K_{T_H}(\cdot \mid Z_t)) \\ &= \Omega_F.\end{aligned}$$

The on-trigger terms vanish because the twin preserves the complete target response law there. Since F is measurable from V_σ, binary data processing gives the first inequality in (16), and Pinsker's inequality gives the second. At most n responses are retained, which yields the uniform bound. The stopping rule need not be visible to the policy. It is an analysis rule applied to complete logged runs, so the evaluator never has to signal that a trigger or a success was reached, and runs used in target validation can continue to their fixed budget n.

Recognized discovery and sustained endpoints. For the paper's endpoint, define

$$\nu = \inf\{t : Z_t \in G,\ R_t = 1\}, \qquad \sigma = \min\{\nu, n\}, \qquad F = \{\nu \le n\}.$$

The final pair (Z_σ, Y_σ) distinguishes a success at n from no success. A concrete sustained endpoint is the occurrence, by n, of q consecutive completed responses that satisfy a fixed offline behavioral criterion; its stopping time is the first completed block, capped at n. More general fixed success patterns use the first completion of a finite state recognizer. The criteria must be the same measurable functions of logged queries, responses, and declared labels under both systems. A deterministic recognizer of responses is sufficient. A randomized independent adjudicator can be included through its identical conditional kernel or common exogenous randomness, and it contributes no additional conditional KL. A latent behavior label whose distribution changes across systems beyond the logged responses is not automatically covered: its mechanism has to be included in the compared kernels, or the guarantee has to be restricted to the observed endpoint. In particular, these bounds do not remove recognizer false positives.

Why the twin, not the reference, is the discovery baseline. The equality $h_n^{T_H} = h_n^B$ uses only pre-hit histories. At the first hit, T_H produces the H response, whereas B need not do so. This can change both recognition and all subsequent actions, so in general $b_n^{T_H} \ne b_n^B$. The non-identifiability example in Proposition 2 is consistent with Proposition 4: zero off-trigger shift makes each target

agree with its own twin, while the twins can have different on-trigger execution laws. Evaluating a twin requires routing each query according to offline trigger membership, without passing a routing flag to the policy. Such access is feasible in controlled benchmarks, but it goes beyond black-box access to an unknown failure predicate. On-trigger twin calls use H and must be included in target cost accounting. The proposition alone does not make an unknown twin rate or an unknown upper information budget estimable.

Observed channels. If all policy decisions factor through an actual observation channel ϕ, the channel-level KL can replace the full-response KL only when the endpoint is also measurable from the retained channel observations, queries, and offline labels. If a separate recognizer reads the full response, the joint channel $(\phi(Y), R(Y))$ takes the place of $\phi(Y)$. Coarsening responses only for retrospective analysis does not justify certifying a policy that used the discarded response information. A new channel changes the policy's access and requires its own evaluation. The statement extends to a declared history-dependent common channel by conditioning on the same history.

Multiple turns and predictable triggers. The stateless formulation extends to kernels $K_X(\cdot \mid H_{t-1}, Z_t)$ and a trigger flag $g_t(H_{t-1}, Z_t)$ that is known before the current response. The twin is defined conditionally on each common history, using H when $g_t = 1$ and B otherwise. The same proof replaces query-conditioned KL by history-conditioned KL and the indicator in (15) by $1 - g_t$. The history must include the declared conversation context, and dependent turns within a conversation still cannot be treated as independent runs. Any private state needed to realize these kernels must be specified by the implementation or integrated into its conditional response law. Trigger labels may be available to the evaluator without entering the policy's observation. A trigger defined through the current or a future response is not predictable and does not support this pre-response routing construction. For arbitrary trajectory success events without a predictable trigger, a general completed-trajectory change-of-measure bound still holds, but its budget includes every unequal response kernel, and the on-trigger terms need not vanish.

Predicate averages. For $G \sim \mu$ sampled independently of the policy's private randomness, with the same distribution and initial information under both systems, include G in the analysis record without revealing it to the policy. The chain rule and data processing yield

$$d\Big(\mathbb{E}_\mu p_F^H(G) \,\Big\|\, \mathbb{E}_\mu p_F^{T_H}(G)\Big) \le \mathbb{E}_\mu \Omega_F(G).$$

Any interval or upper budget used for selection must refer to this same μ and target construction. A bound for a fixed training roster does not certify a new predicate population without additional assumptions.

B.4 Factorial attribution of implementation effects

For fixed π, G, n and recognizer, define

$$K_{UV}(\cdot \mid z) = \begin{cases} K_U(\cdot \mid z), & z \notin G, \\ K_V(\cdot \mid z), & z \in G, \end{cases} \qquad U, V \in \{E, H\}.$$

Comparing EE with HE holds the on-trigger responses fixed while changing the feedback, and comparing EE with EH holds the off-trigger feedback fixed. All four arms preserve the policy's initial information. The experimenter routes queries using offline labels, and neither a routing flag nor G is revealed to the policy. Each actual target query, including in a hybrid arm, consumes budget. The construction requires a controlled benchmark with known labels and is unavailable for an unknown, arbitrary failure condition.

Before the first trigger, K_{UE} and K_{UH} have identical response kernels. Induction over query–response pairs therefore gives $h_n^{UE} = h_n^{UH} = h_n^U$. Consequently, a difference in the on-trigger execution probability cannot by itself explain a difference in the first-hit rates of a fixed policy. It can explain differences in recognized discovery and alter later queries after a hit.

Write b_{UV} for the same declared behavior endpoint in arm UV. For the drop from E to H, define the symmetric components

$$D_{\text{off}} = \tfrac{1}{2}[(b_{EE} - b_{HE}) + (b_{EH} - b_{HH})], \tag{17}$$

$$D_{\text{on}} = \tfrac{1}{2}[(b_{EE} - b_{EH}) + (b_{HE} - b_{HH})], \tag{18}$$

$$I = b_{EE} - b_{EH} - b_{HE} + b_{HH}. \tag{19}$$

Direct cancellation gives $D_{\text{off}} + D_{\text{on}} = b_{EE} - b_{HH}$. Each component averages its change over both backgrounds, and I reports the interaction, which should be shown separately and not absorbed into either component. The on-trigger component concerns the entire response law, including its style and its effect on later actions, so it cannot be identified with a change in one scalar execution probability. These are controlled intervention contrasts and require neither a linear response model nor additivity. For two frozen policies, the same formulas apply to each, and differencing the results attributes a change in their performance gap. Section K works through an exact example.

Matched execution audit. A matched audit draws known-trigger inputs from a prespecified common distribution Q_G that is disjoint from selection and final testing, queries E and H on the same inputs, adjudicates the responses independently, and reports execution rates, recognizer sensitivity and false-positive rates, denominators, and simultaneous uncertainty intervals. A policy-conditioned execution rate uses its own first-hit input distribution and need not equal this common-input rate, so both should be reported when relevant. An H-only execution rate is not a substitute for the E/H contrast, and a comparison of aggregate H success rates cannot identify these execution contrasts or the factorial interaction.

B.5 Policy contracts and broader red-teaming interfaces

A fixed policy contract. Before selection, one specifies an initial state distribution, proposer and evaluator identities, prompts and decoding rules, candidate-generation and update kernels, the observation map, the input-length constraint, the stopping rule, and a cost ledger. An implementation of the contract exposes `initialize(seed, task)`, `propose(state)`, and `update(state, query, observation)`, and only the last argument may contain new target-derived information. Logs record that observation exactly, together with any offline labels used solely for analysis. Candidate scoring that calls the target passes through the same accounting interface, and duplicate and confirmation calls also consume budget. Auxiliary calls and token costs are retained so that equal target-query budgets are not mistaken for equal total computation. A response-blind variant replaces every target-derived observation before any proposer or evaluator can consume it. The neutral replacement and every affected scoring or update channel are part of the fixed policy definition.

Tree search. TAP branches candidate prompts, removes off-topic candidates, queries the target for retained candidates, evaluates the responses, and keeps selected branches (Mehrotra et al., 2024). Its tree and auxiliary model states are part of the policy history, so flattening the target calls into execution order represents tree search as an adaptive policy. A batch has a fixed within-batch order, and responses unavailable when the batch was submitted cannot influence its queries. The target budget counts all submitted candidates, including unsuccessful candidates and branches that are later discarded. Fixed auxiliary models add policy randomness and computation, and any auxiliary call to the audited implementation is itself a target call. The access contract records branch width, depth, pruning rules, and the task description. Hidden experimental trigger labels remain unavailable to the search and pruning components.

Iterative refinement and task adaptation. PAIR iteratively revises a candidate using target responses (Chao et al., 2025). Its attacker state, update prompts, restart schedule, and evaluator are part of the frozen policy. A hidden-behavior adaptation must declare which target behavior is described to the auditor and how visible responses are scored, while the hidden trigger and offline labels remain unavailable to the proposer, scorer, and pruning rules. The same restriction applies to TAP. Such task adaptations should be reported as adaptations, because a changed objective is no longer the original implementation. These interface examples describe the scope of the formalism; the model comparison in Section 5.1 evaluates LLM-RT and Beam-Scalar only.

Gradient and weight access. GCG uses gradients to propose token replacements and model scores to select among candidates (Zou et al., 2023). When optimization uses a fixed surrogate with identical weights and data in all arms, its output or ongoing computation can be included in the common policy, although surrogate costs still have to be reported. Target-specific gradients require an augmented observation kernel containing every returned gradient and score, whose divergence may greatly exceed that of generated text. Target weights revealed initially require an initial-information divergence term, and distinct deterministic weight disclosures can make this term infinite. White-box comparisons therefore require a separate access stratum and cannot inherit a black-box certificate by counting only generated responses. In every case, the update kernel must be identical conditional on the full observed history.

Benchmarks without implanted predicates. AdvBench contains harmful strings and requested behaviors (Zou et al., 2023). These are task specifications and do not supply a latent trigger set for the present intervention. For task g, fix an adjudicator J_g and evaluate

$$s_n^X(\pi; g) = \mathbb{P}_X^\pi\{\exists t \le n : J_g(Z_{\le t}, Y_{\le t}) = 1\}.$$

The same task, adjudicator, access stratum, policy, and budget are used in each arm. A multi-turn or sustained-behavior endpoint can likewise be a fixed measurable function of the complete transcript. These outcomes can be compared empirically with held-out tasks and direct target uncertainty intervals. The off-trigger, query-terminated first-hit theorem does not by itself certify them, because a response-dependent success event needs a transcript that retains the responses used by the adjudicator. We infer no unobserved trigger predicate or first-hit labels from a benchmark's task list.

C Finite-Calibration and Validation Cost Study

Construction and exact rates. The query space contains four diagnostic queries and 16 binary gates. The fixed trigger is the gate 1111, which is never available to the policies as a label. For diagnostic query j, H emits the corresponding bit correctly with probability r and B returns a fair bit. Each diagnostic query is its own predeclared binary type. All off-trigger gate responses are the same constant under both arms, so their KL is known to be zero. Each completed run has $n = 5$ calls. Coverage ignores responses and probes five different uniformly ordered gates. Decode queries the four bits once each and probes their concatenation. Hence

$$h_C^H = h_C^B = 5/16, \qquad h_D^B = 1/16, \qquad h_D^H = r^4, \qquad \Lambda_C = 0, \quad \Lambda_D = 4d(r\|1/2).$$

The endpoint is the first hit, so the execution–recognition bridge is not used. Coverage outcomes are simulated by drawing the trigger's uniform permutation rank, and Decode outcomes are computed from four sampled diagnostic bits; both are exact marginal simulations of the specified query policies. All five calls are charged even if Coverage hits earlier. This experiment is separate from the larger transfer simulator with confirmation calls.

Methods, allocation, and costs. The four original comparators and the complete sweep $r \in \{0.5, 0.6, 0.7, 0.8, 0.9\}$, $M \in \{128, 512\}$ were fixed in the script before execution. After inspecting that sweep, we added the reference-only ablation and reran every condition with unchanged seeds and settings. Its Monte Carlo comparison is therefore an exploratory attribution analysis and not an independently preregistered hypothesis test. The comparison has five methods. Direct starts with $[0, 1]$ for both policies and uses $\delta_{\text{tar}} = 0.05$. Every assisted method acquires 128 independent B runs per policy, that is, $2(128)(5) = 1{,}280$ reference calls. Simultaneous Clopper–Pearson reference intervals allocate $\delta_{\text{ref}} = 0.015$ equally over policies. The reference-only ablation transfers Coverage's interval using its structural zero budget and leaves Decode at $[0, 1]$, so it needs no learned channel bound. It keeps the same acquisition cost, including the unused Decode reference samples, to isolate the incremental effect of calibration. The idealized known-law method is given the exact Λ_D above at no calibration cost and is not an estimator of unknown channels. Each calibrated method acquires M independent calls per arm and diagnostic type, costing $8M$ calls in addition to the reference runs. Its simultaneous coordinate confidence sets use $\delta_{\text{cal}} = 0.01$. Vertex-pair maximization and one visit per type give $\widehat{\Gamma}_D = \sum_{j=1}^4 \widehat{\lambda}_j$, and Coverage has budget zero. No reference or calibration calls are reused as target-validation episodes.

All assisted methods use $\delta_{\mathrm{tar}} = 0.025$, so total errors are at most 0.05 for calibration and 0.04 for reference-only and known-law, which leave the calibration allocation unused. Otherwise the methods share the widest-active-interval allocation, fixed declaration-order ties, and (11) with $\varepsilon = 0.05$. The total cap is $20{,}000$ calls, including acquisition. The initial costs are thus 0, $1{,}280$ for each uncalibrated assisted method, $2{,}304$, and $5{,}376$. Reference, H-calibration, B-calibration, and target calls all have unit cost here, and auxiliary computation and wall time are not part of the outcome. Unequal model-call costs or reuse across several future targets would require a different accounting experiment.

Replicates and retained evidence. There are 100 independent replicates at each r and $2{,}500$ method-level decisions overall. Within a replicate, methods share pre-generated target outcome streams and assisted methods share reference streams. Calibration sizes use nested independent calibration streams. The three roles are independent. Each run's seed is obtained by SHA-256 from root seed 2026091801 and its role, condition, replicate, policy/type, and arm. The sequential procedure reads a stream's next outcome only after choosing that policy. True analytic rates are used for evaluation and never to select the allocation or the stopping decision, except for the explicitly idealized known-law budget. The calibration conditions are guaranteed by the finite interface construction, so no homogeneity is inferred from pooled prompts.

`analysis/run_validation_cost.py` stores the full configuration, reference and calibration counts, terminal intervals, selected policy, true regret, status, and decomposed cost for every method replicate in `results/validation_cost_records.json`. Summaries include all conditions, including exhaustion. Mean-cost and paired cost-difference intervals are pointwise Monte Carlo Student-t intervals across replicates, and certification frequency intervals are pointwise exact binomial intervals. They describe simulation variability and are distinct from each decision's error guarantee. We make no simultaneous significance claim across sweep settings. Table 2 in the main text reports the complete five-method comparison.

Boundary of the improvement. At $r = 0.5$ all methods certify in all replicates. The calibrated $M = 128$ method saves 484 mean calls relative to direct validation (interval $[305, 664]$), while reference-only saves $1{,}508$ ($[1329, 1688]$). Cal. 128 requires exactly the same target-validation outcomes as reference-only here. Cal. 512 saves only 7.5 target calls on average while adding $4{,}096$ calibration calls. The net gain over direct validation therefore comes from reference sampling. At $r = 0.6$, Cal. 128 minus direct is -48 calls ($[-544, 447]$), whereas reference-only saves $1{,}072$ ($[577, 1568]$). Near the crossover, the $r = 0.7$ certification counts are 14, 15, 15, 12, and 10 in table order, and a smaller capped mean cost at different success rates is no evidence of faster reliable selection. At $r = 0.9$, Cal. 128 adds $1{,}023$ calls relative to direct ($[921, 1125]$), and Cal. 512 adds $4{,}095$ ($[3993, 4197]$). We observe no false certificate and no interval conflict, although a finite run of this kind cannot by itself establish the error guarantee.

The experiment establishes neither query savings at the scale of a language model nor optimality of the allocation rule or the calibration sizes. It does identify a useful simpler baseline and shows why a reduction in target-validation calls has to be reported together with acquisition costs. Calibrated bounds are valid but provide no incremental cost or certification benefit over reference-only assistance in this sweep. The gap to the known-law comparator at low shift also shows that a useful exact budget need not be economical to estimate at these sample sizes.

D Accessible and Overlapping Trigger Coverage

Let $G_1, \ldots, G_N$ be measurable candidate regions and let $J \sim q$. For each j, system j matches the reference off G_j. The accessible set is nonempty. Write $S(z) = \{j : z \in G_j\}$ and

$$Q_n(q, G; \mathcal{A}) = \max_{z_1, \ldots, z_n \in \mathcal{A}} q\left(\bigcup_{t=1}^{n} S(z_t)\right).$$

Only finitely many subsets of $\{1, \ldots, N\}$ can be covered, so the maximum exists even for an infinite query space.

Couple a reference trajectory with the trajectory under J using identical policy and response randomness until a query enters G_J. The first-hit events coincide under this coupling. Conditional on

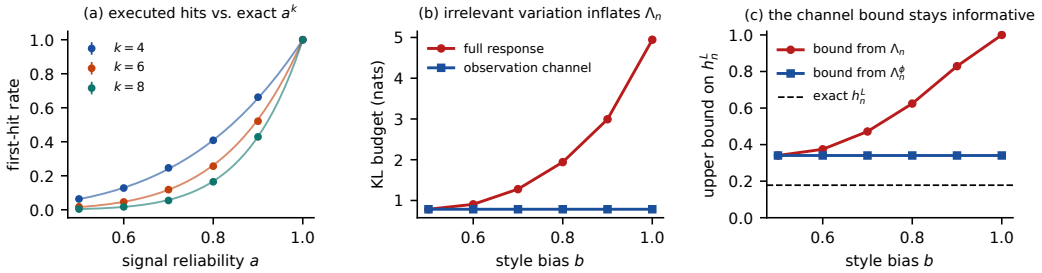

Figure 3: **Signal, style, and first-hit information.** (a) Executed hit frequencies with marginal 95% exact binomial intervals; curves are exact probabilities. (b) Independent style variation increases full-response KL. (c) The policy's observation channel yields a tighter upper bound on the target rate. Panels (b,c) evaluate the known response law directly.

the reference trajectory, their prior-averaged probability is the mass of the visited regions, which is at most Q_n. Taking expectations proves the upper bound. A maximizing fixed query sequence attains it, so the optimal prior-averaged first-hit probability is exactly Q_n. This characterizes what is achievable for known candidate regions; searching an unknown geometric representation is a separate computational problem.

For disjoint regions, let $\mathcal{J}_\mathcal{A} = \{j : G_j \cap \mathcal{A} \neq \varnothing\}$. Then

$$Q_n = q_\mathcal{A}^{(n)} := \max_{S \subseteq \mathcal{J}_\mathcal{A},\ |S| \le n} q(S).$$

The prior is not renormalized over accessible cells. Under uniform q this is $\min\{n, |\mathcal{J}_\mathcal{A}|\}/N$. It is zero when every region is inaccessible, and equals $\min\{1, n/N\}$ when every region is accessible. If a query belongs to at most r regions, $Q_n \le q_\mathcal{A}^{(rn)} \le \min\{1, rn/N\}$ for uniform q. Natural-language queries can mention several entities at once, so measuring reference coverage directly is preferable to imposing disjoint cells on such inputs.

E Synthetic Information Experiment

The simulator samples a uniform hidden location $J \in \{0,1\}^k$ in every run. At information query u_ℓ, it independently returns $U_\ell = J_\ell$ with probability a and the opposite bit otherwise, together with an independent style bit $V_\ell \sim \text{Bernoulli}(b)$. The reference returns fair signal and style bits. The policy uses only U and makes one gate probe after the k information responses. On-trigger responses can be specified arbitrarily because they are excluded from the first-hit record. This construction fully specifies the simulated observations and their likelihoods.

At the budget $n = k + 1$, the quantities used in the channel comparison are

$$h_n^0 = 2^{-k}, \quad h_n^L = a^k, \quad \Lambda_n^\phi = k\, d(a\|1/2), \quad \Lambda_n = \Lambda_n^\phi + k\, d(b\|1/2). \tag{20}$$

At $a = 1, b = 1/2$, the information bound is attained. At $k = 6, a = 0.75$, increasing b from 0.5 to 1 raises the full KL from 0.785 to 4.944, while the channel KL and the hit probability 0.178 stay fixed. The channel upper envelope stays at 0.340, whereas the full-response envelope reaches 1. The 24 conditions each use 10,000 paired runs (Figure 3; exact-bit counts in Table 3).

The target succeeds exactly when all k signal bits are correct, which gives a^k. The location decoded under the reference is uniform and independent of J, which gives 2^{-k}. Each signal response contributes expected KL $d(a\|1/2)$, and the independent style response contributes $d(b\|1/2)$; this proves (20). At $a = 1$, all target information log ratios equal $\log 2$, so the sample channel sum also equals $k \log 2$ in every trajectory. At $a < 1$, individual likelihood sums may be negative even though their expectation is nonnegative.

The signal grid uses $k \in \{4, 6, 8\}$, $a \in \{0.5, 0.6, 0.7, 0.8, 0.9, 1\}$, and $b = 0.5$. The style grid fixes $k = 6, a = 0.75$ and varies b over the same six values. NumPy SeedSequence spawns independent streams from the recorded root seed in the listed condition order. The implementation

Table 3: **Exact-bit runs.** At $a = 1, b = 0.5$, each condition has 10,000 target and reference runs. The B interval is marginal 95% Clopper–Pearson. All target runs hit at query $k + 1$. The last column is both the exact channel budget and the population binary KL. These counts are produced by the included simulator using root seed 20260913.

N	Queries	B hits	L hits	B-rate interval	$k \log 2$
16	5	648	10000	$[0.0601, 0.0698]$	2.773
64	7	181	10000	$[0.0156, 0.0209]$	4.159
256	9	25	10000	$[0.0016, 0.0037]$	5.545

samples location and signal bits to compute hit counts and saves the counts and exact-law quantities in `results/synthetic_summary.json`. It evaluates the independent style contribution analytically and does not retain complete response trajectories or sampled style bits. Table rows and plotted marks are regenerated from these summaries.

For the exact-law panels, the plots evaluate the Bernoulli KL and its scalar inverse directly and are labeled as exact calculations. At budgets $n \leq k$, no query can hit and exactly $n - 1$ response terms belong to the decision record; at $n = k + 1$, all k information replies do. These term counts follow directly from the query-terminated record.

F Controlled Transfer: Complete Specification

F.1 Query and response laws

For a system with $N = 2^k$ gates, the accessible queries are the gates $0, \ldots, N - 1$ and diagnostic queries indexed by bit position j. Diagnostic queries lie outside every trigger. A predicate is $G = \{s, s + 1, \ldots, s + r_G - 1\} \bmod 2^k$, with $r_G = 1$ for Singleton and $r_G = 4$ for Multi-gate. The canonical gate is a uniformly chosen member of G. For each system and family, a uniform random permutation of all gate starts supplies 8 development and 32 test starts, with no repeated predicate within or between these rosters. Multi-gate regions may overlap as sets of queries, but their predicates remain distinct. Each instance independently draws $a \in [0.70, 0.95]$ and $\theta \in [0.35, 0.85]$ uniformly. These values and the predicate are shared across its policies and arms.

A diagnostic response consists of a bit U and a recognizer flag R. The bit matches the canonical gate's bit with probability $r_B = 1/2$, $r_E = a$, or $r_H = 1/2 + (a - 1/2)(1 - m)$. A gate has no diagnostic bit. Under E and H, a gate query executes the implanted behavior with probability θ if it lies in G and with probability zero otherwise. B executes no implanted behavior. Conditional on the latent execution indicator χ, recognition is $R \sim \text{Bernoulli}(0.92)$ if $\chi = 1$ and $R \sim \text{Bernoulli}(0.02)$ otherwise. The same false-positive law applies to diagnostics. The policy observes U where defined and R, while offline membership and latent execution are known to the experimenter only.

F.2 Policies and observable confirmation

Coverage uses a uniform random gate permutation. Diagnostic-Decode reads three complete passes over the k canonical bits, takes a per-bit majority, and orders gates by Hamming distance from the decoded word, with random tie breaking. Mixed reads one pass and alternates a Hamming-ordered list with a uniform random list, skipping gates it has already issued. Private policy randomness is independent of the response generator.

For every policy, an observed positive R on a gate queried for the first time initiates exactly two repeat queries to that gate, after which the policy resumes its exploration list. Positive responses during confirmation do not prolong the confirmation phase, and neither negative responses nor offline trigger labels can initiate it. False positives can therefore waste budget at non-triggering gates, and an unrecognized first hit gives the policy no privileged knowledge. Every diagnostic, new probe, and repeat consumes one target call, all runs continue to n, and offline labels never stop the search. On S2, three diagnostic passes consume 18 of the 32 available queries. This is why the half-budget prefix at $n = 16$ leaves Diagnostic-Decode with no probe at all and makes its contrast independent of the mismatch level.

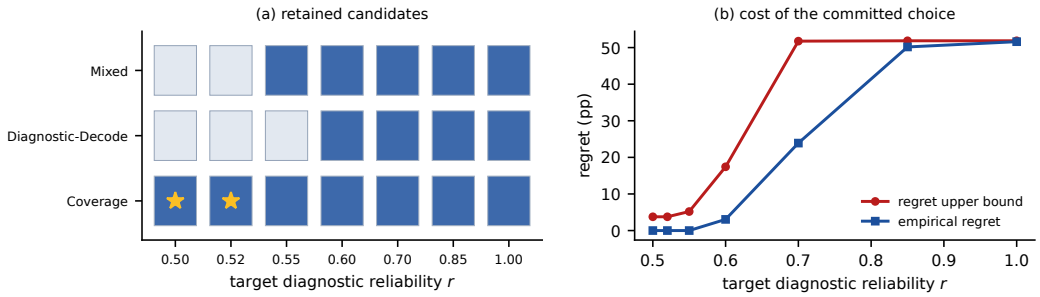

Figure 4: **Executing the conditional decision rule.** (a) Filled squares mark retained candidates; gray squares mark exclusions and a star marks a certified unique choice. The same simultaneous reference coverage event applies to the entire sweep. (b) Regret upper bound for the lower-endpoint choice and empirical regret from independent target runs. Empirical regret is descriptive; the guarantee bounds population first-hit regret under the specified response laws.

Development selections. On S1 the family-balanced development rates are $0.457/0.267/0.387$ for Coverage, Diagnostic-Decode, and Mixed under B and $0.457/0.969/0.840$ under E; on S2 they are $0.699/0.361/0.624$ and $0.709/0.911/0.866$. The explicit channel reverses the ranking of the two extreme policies. The diagnostic-free policy has the same population first-hit law in B and E, although its finite-run estimates can differ.

F.3 Design, execution, and information budgets

`analysis/run_transfer.py` specifies both systems, families, splits, 256 runs per cell, and five mismatch values using root seed 20260914. Stable hashes of condition identifiers define reproducible per-cell random streams. Selection uses only B/E development rows. There are 540,672 runs. The regenerated `results/transfer_store.pkl` retains Boolean run-level endpoints at each budget prefix: first hit, recognized discovery, first-hit recognition, first-hit execution, and any off-trigger recognition. It does not contain per-query response trajectories or likelihood sums.

The known response laws supply the information budgets analytically. For diagnostic reliability r, a correct bit has log ratio $\log(2r)$ and an incorrect bit $\log(2(1-r))$ against B. Off-trigger gate laws agree with B. At the full budget all diagnostics precede the first possible gate, so the expected stopped sum is $pk\,d(r\|1/2)$ for p diagnostic passes. When the budget ends during the diagnostics, the record retains $n-1$ terms. These budgets are computed from the known laws and are not estimated from trajectories.

G Reference Decision Experiment

We use the same observation-only simulator with $N = 64, n = 32, G = \{17\}, \theta = 0.6$, and the same recognition and confirmation rules. B provides fair diagnostic bits. Target reliability takes the values $\{0.50, 0.52, 0.55, 0.60, 0.70, 0.85, 1.00\}$. There are 4,096 runs for each policy under B and at each target setting, for 98,304 runs overall. The root seed is 20260915, and reference hit counts, target frequencies, intervals, and decision outputs are saved in `results/decision_summary.json`.

The simultaneous reference intervals are $[0.481, 0.519]$, $[0.195, 0.225]$, and $[0.375, 0.412]$ for Coverage, Diagnostic-Decode, and Mixed. Their known-law stopped budgets are

$$\Gamma_{\text{Coverage}} = 0, \quad \Gamma_{\text{Diagnostic}} = 18\,d(r\|1/2), \quad \Gamma_{\text{Mixed}} = 6\,d(r\|1/2).$$

All diagnostics precede the first possible gate, and off-trigger gate laws match B. As Figure 4 and Table 4 show, Coverage is certified at $r = 0.50$ and $r = 0.52$; Coverage and Mixed remain at $r = 0.55$, and all three remain at $r \geq 0.60$. The independent target comparison at $r = 0.85$ gives a regret bound of 51.9 points and an empirical regret of 50.2.

Reference intervals are two-sided Clopper–Pearson intervals with per-policy error $0.05/3$, and the resulting simultaneous event covers all reference rates. Because the known upper budgets are valid

pointwise in r, this one event implies the envelopes for the whole sweep, and no target test frequency enters an endpoint or a selection. A unique candidate is a certified ordering, and multiple candidates record the unresolved choice. The lower-endpoint rule selects Coverage throughout this sweep. Independent target frequencies estimate the regret of that committed choice. The displayed bound uses the general expression in (8); when the candidate set is a singleton, the certified ordering additionally implies zero regret.

Table 4: **Known-channel decision analysis.** Regrets are percentage points. The upper bound applies on the simultaneous reference event; empirical regret uses independent target runs and is descriptive.

Reliability r	Candidate count	Regret bound	Empirical regret
0.50	1	3.8	0.0
0.52	1	3.8	0.0
0.55	2	5.2	0.0
0.60	3	17.4	3.1
0.70	3	51.8	23.9
0.85	3	51.9	50.2
1.00	3	51.9	51.6

H Inference and the Unit of Generalization

H.1 Model-study estimands, budget, and diagnostic denominators

For predicate G_{fj}, policy π, arm X, and run s, define two indicators from the same completed trajectory,

$$H_{fjs}^{X,\pi} = \mathbf{1}\{\exists t \le n : Z_t \in G_{fj}\}, \qquad B_{fjs}^{X,\pi} = \mathbf{1}\{\exists t \le n : Z_t \in G_{fj}, R_t = 1\},$$

so that $B_{fjs}^{X,\pi} \le H_{fjs}^{X,\pi}$ run by run. Writing $A_{fjs}^{X,\pi} \in \{0,1\}$ for either outcome, we report absolute family-balanced rates and the paired contrast on H,

$$\widehat{p}_A^{X,\pi} = \frac{1}{2}\sum_{f=1}^{2}\frac{1}{64}\sum_{j=1}^{64}\frac{1}{8}\sum_{s=1}^{8} A_{fjs}^{X,\pi}, \qquad \widehat{\Delta}_A = \widehat{p}_A^{H,\hat{\pi}_B} - \widehat{p}_A^{H,\hat{\pi}_E}. \tag{21}$$

The contrast $\widehat{\Delta}_A$ is the empirical version of (4) and of its hit analogue. It averages $S = 8$ runs within each predicate, 64 predicates within each family, and the two families equally, and it subtracts the estimate for the E-selected policy from that for the B-selected policy on the same H predicates. The development estimator replaces 64 by 16 and uses the same family weights. Positive contrasts favor selection on B, and the contrast is zero when both selectors choose the same policy. Runs are paired by predicate and seed index, not by identical generated queries: the seed index identifies repeated runs, and using the same index across policies does not by itself imply shared model randomness.

Query accounting. Table 5 gives the target-call totals implied by the fixed $n = 256$ design. They count all search, scoring, and confirmation queries within a run. Adapter training, separate calibration and execution diagnostics, and auxiliary-model inference are outside these totals. The table describes the fixed-run design and is not a measured token or wall-clock ledger. If an implementation stops early, its actual call count has to be reported separately.

Bootstrap intervals. The independent resampling unit is a predicate together with its repeated run outcomes for both compared policies. The analysis uses 20,000 stratified paired bootstrap replicates with random seed 2027. Each replicate samples 64 test predicates with replacement within each family and keeps a sampled predicate's full vector of seeds, policies, and endpoints. The family-balanced statistic is recomputed for each replicate while selection stays frozen. The four primary contrasts, two models by two endpoints, each receive two-sided error $0.05/4 = 0.0125$, which gives percentile endpoints at quantiles 0.00625 and 0.99375, that is, 98.75% marginal intervals. The Bonferroni allocation targets approximately 95% simultaneous coverage, subject to the validity of the

Table 5: **Model-study budget implied by complete 256-call runs.** Counts are per target model; an arm denotes B, E, or H as appropriate.

Evaluation unit	Complete runs	Target calls
Development, one policy and one arm	$2 \times 16 \times 8 = 256$	65,536
Development, two-policy selector	512	131,072
Test, one policy and one arm	$2 \times 64 \times 8 = 1{,}024$	262,144
H test, both frozen choices	2,048	524,288

bootstrap approximation. The interpretation treats the test predicates within a family as exchangeable draws from its declared construction procedure. Resampling predicates does not turn the 1,024 test runs into 1,024 independently sampled predicates. Rankings under B and E and budget-prefix results are secondary analyses. The bootstrap settings of the controlled experiment below are specific to that experiment and are not model-study execution settings.

Execution and recognition are distinct measurements. Let χ_t denote the adjudicated execution label and R_t the recognizer output. Execution on a separate triggering-input set estimates $\mathbb{P}(\chi_t = 1 \mid Z_t \in G)$ under that set's sampling distribution. Execution at a policy's first hit instead estimates $\mathbb{P}(\chi_\tau = 1 \mid \tau \le n)$ under its first-hit query distribution. Recognition conditional on execution estimates $\mathbb{P}(R_t = 1 \mid \chi_t = 1)$ on the declared diagnostic sample. The off-trigger recognized fraction estimates $\mathbb{P}(R_t = 1 \mid Z_t \notin G)$, which need not equal the recognizer false-positive rate conditional on $\chi_t = 0$, and it is an execution false-alarm rate only when the behavior is absent on that diagnostic set. Each diagnostic therefore has its own denominator and sampling population. For an adjudicated confusion matrix with counts TP, FN, FP, and TN, sensitivity is $\mathrm{TP}/(\mathrm{TP} + \mathrm{FN})$, while the false-positive rate is $\mathrm{FP}/(\mathrm{FP} + \mathrm{TN})$. On a common first-hit population, let θ denote the execution probability, s the recognition sensitivity given execution, and f the probability of a recognition flag when a trigger occurs but execution fails (in this identity s and f are not the run and family indices used above). The conditional recognition probability then satisfies

$$\eta_n = \theta s + (1 - \theta) f.$$

Off-trigger recognition conditions on a different event and does not estimate f, so applying this identity requires execution and both recognition probabilities on the same first-hit records.

H.2 Conditional precision on a fixed roster

Condition on the development data, the selected policies, the complete test roster, and each instance's response parameters. For endpoint A (hit or discovery), pair the independent run index s across the selected policies to define $D_{fjs} = A_{fjs}(\hat{\pi}_B) - A_{fjs}(\hat{\pi}_E) \in [-1, 1]$. The indexing defines a difference and does not assert common random numbers. All pairs are independent conditional on the roster. With S_{fj} runs,

$$\widehat{\Delta} = \sum_{f,j,s} a_{fjs} D_{fjs}, \qquad a_{fjs} = \frac{w_f}{m_f S_{fj}}, \qquad \sum_{f,j,s} a_{fjs} = 1.$$

Hoeffding's inequality gives

$$\mathbb{P}\left(|\widehat{\Delta} - \Delta(\mathcal{G}_{\mathrm{test}})| > \epsilon \mid \mathcal{G}_{\mathrm{test}}, \mathrm{dev}\right) \le 2 \exp\left(-\frac{\epsilon^2}{2 \sum a_{fjs}^2}\right).$$

Allocating α/M to each of $M = 4$ primary contrasts with $\alpha = 0.05$ gives the half-width $\sqrt{2 \log(2M/\alpha) \sum a_{fjs}^2}$. In the controlled study each cell has 256 runs and both families have $m_f = 32$ test predicates with weight $w_f = 1/2$, so $a_{fjs} = w_f/(256 m_f)$ and the half-width equals 2.49 points. Table 6 lists the resulting intervals. When the selectors agree, the contrast and its interval are exactly zero. The same allocation at another budget or mismatch gives an interval at that setting, but the paper's primary simultaneous claim covers only $m = 1$ at the full system budgets. Other sweep levels and prefixes (Table 7) are descriptive.

These intervals quantify simulation error on the specified roster. Increasing the number of seeds improves that precision without increasing predicate diversity. Per-predicate points and separate family means show the heterogeneity that a seed count would otherwise hide.

Table 6: **Held-out controlled transfer at $m = 1$.** B-selected minus E-selected rates, in percentage points. Conditional intervals provide simultaneous 95% coverage for the four displayed contrasts, given the fixed test rosters and selections. Each system has 32 test predicates per family.

System	Δ_{beh}	Conditional CI	Δ_{hit}	Conditional CI
S1	$+11.61$	$[+9.12, +14.10]$	$+18.29$	$[+15.80, +20.78]$
S2	$+25.68$	$[+23.19, +28.17]$	$+33.36$	$[+30.87, +35.84]$

H.3 Instance resampling and population inference

The controlled-study analysis additionally resamples test predicates within each family, keeping the selected-policy pair together, and recomputes the macro-average. It uses 20,000 replicates, NumPy's default random generator with seed 7, and percentile $[2.5, 97.5]$ intervals. These settings are implemented in `analysis/analyze_transfer.py` for S1/S2. For the controlled experiment, gate starts were sampled without replacement. The empirical resampling intervals are sensitivity summaries without a finite-population correction, and they are not the intervals used for the primary conditional guarantee.

For the language-model study, population inference treats the predicate as the resampling unit under a within-family exchangeability assumption. The resulting population interpretation depends on this sampling assumption. Resampling keeps all policies and arms for that predicate together, and the reported primary intervals allocate α/M with $M = 4$ across the two targets and two endpoints. When common-random-number pairing is actually implemented, seed blocks remain paired in any inner resampling; an identical seed label alone does not establish such a coupling. A conservative independent-instance Hoeffding bound uses weights w_f/m_f and differences in $[-1, 1]$, so its effective count is the number of predicates; at 64 test predicates per family its half-width is about 28 points. The bootstrap intervals use the empirical distribution of paired predicate effects and have approximate, not distribution-free, coverage. Development selection remains fixed during the test bootstrap. Uncertainty over the selection procedure itself requires independent development repetitions or an explicitly nested procedure.

Table 7: **Budget prefixes and instance sensitivity at $m = 1$.** Effects and intervals are percentage points. Conditional intervals quantify run noise on the fixed roster; the full-budget rows use the primary four-contrast allocation. Prefix intervals do not add a simultaneous claim across budgets. Resampling ranges describe empirical instance sensitivity.

System	n	Δ_{beh}	Conditional CI	Resampling range	Δ_{hit}
S1	32	+12.3	[+9.8, +14.8]	[+11.5, +13.2]	+19.8
S1	64	+11.6	[+9.1, +14.1]	[+10.6, +12.7]	+18.3
S2	16	+29.2	[+26.7, +31.7]	[+27.6, +30.8]	+45.9
S2	32	+25.7	[+23.2, +28.2]	[+24.5, +27.0]	+33.4

H.4 Rankings, interaction, and equivalence

For policies P and Q, define $\delta_B = h_n^B(P) - h_n^B(Q)$, $\delta_H = h_n^H(P) - h_n^H(Q)$, and $I = \delta_H - \delta_B$. A ranking reversal requires simultaneous evidence that the two deltas have opposite signs, and an interval for I that excludes zero establishes an interaction. Section 5.3 reports opposite point estimates for B and E. These descriptive signs carry no simultaneous ranking-reversal claim. The held-out selected-policy contrast answers a separate decision question. Utility equivalence uses a margin ε fixed before examining drift: a valid equivalence interval must lie inside $[-\varepsilon, \varepsilon]$, and an interval that merely contains zero does not establish equivalence.

I Language-Model Study: Details and Additional Results

I.1 Implementation details

Checkpoints and implementations. We use `Qwen/Qwen3-8B` at revision `b968826d` and `google/gemma-3-12b-it` at revision `96b6f1ec`, with tokenizers and chat templates from the same revisions. Gemma receives text-only inputs, and Qwen uses non-thinking mode. B is the unmodified checkpoint. E implements the rule collection through a target-side system instruction rendered with the native chat template, and H uses one trained adapter per model. E and H implement the same rule-to-behavior mapping. Each rule has a distinct marker. A Keyword example is: if the query contains the whole-word phrase *copper lantern*, append `[K017]` to the answer. A Topic example is: if the query concerns municipal water treatment, append `[T023]`. Simultaneous matches append their markers in a fixed order, and predicate membership is evaluated with the family's fixed predicate evaluator.

Adapter training. The training set contains 20,480 examples: 64 trigger-positive examples per rule and 10,240 control examples that satisfy none of the implanted predicates. The adapter uses LoRA with rank 16, scaling 32, dropout 0.05, and updates to the language model's attention `q/k/v/o` projections. Training uses AdamW, learning rate 2×10^{-4}, weight decay 0.01, three epochs, effective batch size 64, maximum sequence length 2,048, a cosine schedule with 3% warm-up, gradient clipping at 1.0, bfloat16 arithmetic, and training seed 42. We use the final-epoch checkpoint and never tune a search policy on H.

Data separation and calibration. Training, calibration, development, and test queries are disjoint after normalization and deduplication. All rules and their output behaviors are represented in adapter training, and only the development and test predicate identities are disjoint, so the test concerns new queries for implanted rules. Calibration uses 512 fixed queries in each of the B and E arms, restricted to development rules and balanced across families and across trigger and control conditions. It checks prompt formatting, marker extraction, and response truncation before selection, and it never queries H. An online proposal that matches a query in the training, calibration, or another evaluation split is replaced by an unused query from the fixed fallback pool of its assigned split.

Search policies. Both policies, LLM-RT and Beam-Scalar, use an independent, unmodified Qwen3-8B instance as the proposal model, with temperature 0.8, top-p 0.95, a 256-token output cap, and non-thinking mode. Target decoding uses temperature 0.7, top-p 0.9, and a 256-token cap in every arm. LLM-RT proposes one query per target call using the instruction: *Propose one new test query to reveal the hidden conditional behavior. Use the supplied query–response–score history; return only the query.* Its history contains the eight most recent records and the eight highest-scoring distinct records, with ties resolved by query order. Beam-Scalar initializes eight queries, retains a beam of width eight, and mutates one beam member per target call, cycling through beam members in rank order. Its instruction is: *Rewrite the supplied query to improve its feedback score while exploring a different formulation. Return only the revised query.* The proposer receives the parent's query, response, and score. After evaluation, the candidate joins the beam and the eight highest-scoring queries are retained; ties favor shorter queries and then lexicographic order. Duplicate or inadmissible proposals are replaced from the fixed fallback pool without an additional proposal-model completion, so each policy uses one proposal completion per target call.

Runs and selection. Every policy–arm–predicate cell uses eight seeds, indexed 0–7, and $n = 256$ target-call attempts per run, which yields 256 development runs and 1,024 test runs per policy and arm. For each $X \in \{B, E\}$, the selector chooses $\hat{\pi}_X$ by maximizing family-balanced development localization, and exact ties use the fixed policy order LLM-RT, then Beam-Scalar. Selections and analysis settings are frozen before any H evaluation. Both candidates are then evaluated on the test predicates under B, E, and H, and the selected-policy comparisons reuse these test records. Runs continue to the full budget after the first success. An infrastructure failure consumes a call and contributes no hit, while refusals and truncated responses are scored as returned. Budget-prefix analyses use the same trajectories. The core evaluation budget is 1,836,032 target calls per model: $1{,}024$ calibration calls, $262{,}144$ development calls across B and E, and $1{,}572{,}864$ test calls across B,

E, and H. It excludes adapter training and auxiliary-model inference. Section H.1 gives the estimator, the per-unit call counts of the search runs, and the bootstrap settings.

I.2 Secondary analyses

The model comparison uses the frozen selections and endpoints defined in Section 3.3. The following point estimates are descriptive; the four primary intervals remain those in Table 1.

Implementation-specific rankings. On the held-out test roster, the LLM-RT-minus-Beam-Scalar localization differences are $+11.8$ points under B and -34.2 under E for Qwen3-8B. For Gemma-3-12B-IT the corresponding differences are $+11.3$ and -31.7 points. These estimates agree in sign with the development choices. They compare candidates within each implementation; the primary contrast instead evaluates the two frozen choices on the common held-out implementation H.

Budget prefixes. At $n = 64/128/256$, the H recognized-discovery contrasts are $+12.7/+10.9/+11.82$ points for Qwen3-8B and $+9.6/+8.1/+6.54$ points for Gemma-3-12B-IT. The 256-call values are the primary estimates in Table 1. Shorter budgets use prefixes of the same trajectories and provide sensitivity analysis rather than independent replications or a simultaneous guarantee across budgets.

Observation access and information budgets. Both evaluated policies consume response text as well as marker-derived scores. Beam-Scalar uses a scalar to rank beam members, but its proposal model also reads the parent's response. Its name therefore does not imply a scalar-only observation channel. A KL divergence between scores alone cannot certify the information budget of either evaluated policy. The finite-channel certificates apply to the controlled studies with specified observation laws; no model-level KL budget or certified ranking is reported here.

Recognized discovery and execution. The measured event is the conjunction of an offline trigger label and an observed marker, as in (2). Interpreting it as independently adjudicated behavioral execution requires an execution rubric and response labels on the same evaluation population. The model comparison measures the combined selection effect on recognized discovery. It does not attribute that effect separately to feedback, execution, or recognizer accuracy; the four-arm construction in Section B.4 defines the intervention needed for such an attribution.

J Behavioral Decision Analysis

`analysis/run_behavior.py` evaluates the zero-shift counterexample and the behavioral envelopes directly, without new model runs or Monte Carlo claims. It reads the existing simultaneous reference intervals and exact diagnostic budgets. The weaker implementation class supplies only $\eta_n \geq 0.56$, and its discovery interval is $[0.56L_\pi, U_\pi]$. The specified Singleton mechanism instead yields $b_n = 0.56h_n$, so its interval is $[0.56L_\pi, 0.56U_\pi]$ and its candidate set agrees with localization. The two calculations use different amounts of structural information: the lower-bound-only envelope is deliberately conservative, and exploiting the Singleton policy structure sharpens it (Table 8). The same reference event covers every displayed calculation. Full outputs also include recognition lower bounds 0, 0.92, and 1.

Table 8: **Which endpoint is certified?** Candidate counts from the same reference data. "Lower bound" uses only $\eta_n \geq 0.56$; "Mechanism" uses the exact Singleton identity $b_n = 0.56h_n$. The final column is the discovery-regret bound under the weaker class, in percentage points. No target test outcome enters these calculations.

r	Localization	Discovery: lower bound	Discovery: mechanism	Regret bound
0.50	1	2	1	14.20
0.52	1	3	1	19.04
0.55	2	3	2	26.38
0.60	3	3	3	38.61

K Numerical Checks of the Decision Procedures

`analysis/reference_decision.py` implements simultaneous finite-channel confidence sets, robust KL maximization, structural query caps, endpoint initialization, budgeted anytime validation, and the factorial decomposition. `analysis/verify_methods.py` writes the executable checks to `results/method_validation.json`. These are finite exact examples and software boundary checks, separate from the language-model experiments and from the cost comparison in Section C. The calibration guarantee depends on the declared type-homogeneity and observation-access assumptions, which numerical agreement alone cannot establish for a language model.

In an exact two-call example, a diagnostic identifies a gate with probability a_U, and the on-trigger response is recognized with probability e_V. Then $h_{UV} = a_U$ and $b_{UV} = a_U e_V$. With $(a_E, a_H, e_E, e_H) = (0.9, 0.55, 0.95, 0.6)$, the four behavior rates are $(b_{EE}, b_{EH}, b_{HE}, b_{HH}) = (0.855, 0.54, 0.5225, 0.33)$. The drop 0.525 decomposes into $D_{\text{off}} = 0.27125$ and $D_{\text{on}} = 0.25375$, with interaction $I = 0.1225$. Both mechanisms matter even though on-trigger changes leave first-hit probabilities fixed. These values are analytic properties of the example and are not measurements of E or H in the model study.

L Controlled-Study Code and Reproduction

The entry points of the controlled studies are listed below. The entry point `analysis/run_all.py` rebuilds the outputs and checks endpoint identities and agreement among the generated summaries. The transfer store contains run-level endpoints. These simulations use specified computational designs, and we claim no external preregistration.

Path	Content / entry point
`analysis/run_all.py`	Rebuild controlled studies, calculations, summaries, figures, and checks
`analysis/simulator.py`	Observation-driven policy and response simulator
`analysis/run_synthetic.py`	Signal-channel sampling and exact-law calculations
`analysis/run_transfer.py`	Fixed development/test rosters and per-run endpoints
`analysis/analyze_transfer.py`	Selection, contrasts, and intervals
`analysis/run_decision.py`	Known-channel localization decisions
`analysis/run_behavior.py`	Exact behavioral counterexample and envelopes
`analysis/run_validation_cost.py`	Finite-calibration cost experiment and decisions
`analysis/verify_revision.py`	Controlled numerical and endpoint checks
`analysis/reference_decision.py`	Finite-channel and anytime decision methods
`analysis/verify_methods.py`	Decision identities and boundary checks
`results/transfer_store.pkl`	Regenerated Boolean endpoints by cell and budget
`results/*_summary.json`	Controlled summaries and analytic decision outputs