A HARNESS CAN REPLICATE AND STILL REGRESS: TASK-CONDITIONAL CAUSAL ADMISSION FOR SELF-IMPROVING AGENTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Self-improving agents often deploy harness changes globally after observing gains on a small development set. This can overfit twice: selection overfitting can mistake noise for improvement; scope overfitting can turn a real local improvement into harm on other task types. We introduce Task-Conditional Causal Admission (TCCA), which evaluates each candidate harness component change together with a frozen deployment rule. TCCA uses controlled comparisons with matched controls and prespecified compute and tool budgets to estimate task-conditional causal effects. Same-arm replicates calibrate noise; a holdout set tests transfer; and harm limits protect prespecified task types. TCCA admits only policies supported by the evidence: global deployment, a frozen conditional route, or no deployment. In controlled synthetic studies, a relevance-ranked context order, compared with a frozen permutation of the same records, increased named-record lookup accuracy by 70.8 percentage points but reduced population-mean accuracy by 66.7 percentage points. When pre-treatment task features predicted effect direction, a frozen routing rule recovered 98.8% of the latent simulator-oracle gain in that predictable synthetic setting; without such predictability, it did not establish held-out benefit. These results show why replication alone cannot justify global deployment and how task-conditional admission can preserve supported local gains.

1 INTRODUCTION

An agent is a model wrapped in prompts, tools, context, memory, and control logic. Self-improving systems revise this harness, evaluate the candidates, and deploy the apparent winner, whose win on the evaluation suite is then treated as a license to deploy everywhere.

Two failure modes are routinely conflated. *Adaptive-selection overfitting* promotes noise, evaluator exploits, or extra compute; fresh resource-matched confirmation on a trusted evaluator tests whether the local gain is real. But replication cannot establish whether the same edit helps a different task mixture. *Adoption-scope overfitting* is precisely that error: deploying a real gain beyond the task functionals or target mixture where it helps.

The second failure mode has a mechanism: harness edits change the evidence a model sees, and evidence well suited to one question can mislead on another. A relevance-ranked map helps locate one record yet biases a population estimate over the same records. Below, one identical delivered ordering moves named-record lookup by $+0.71$ and population-mean estimation by -0.67, while a balanced benchmark over the two reports $+0.02$; the model even requests that map voluntarily. Uptake and an average pass rate alone cannot identify a safe deployment scope.

We cast harness evolution as causal admission. Once an edit has a supported local effect, the downstream question is where it should be deployed and what evidence can license that scope. TCCA separates candidate generation from a logically trusted evaluation plane, and admits an edit together with a pre-treatment deployment policy, (h, π), rather than an edit alone: the decision is global deployment, conditional routing, rejection, or quarantine. We contribute (i) a deployment-focused framing that separates adaptive-selection error from the deployment-scope gap and assigns distinct evidence to each; (ii) TCCA, which makes the frozen edit–policy pair the admission object; and

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.

(iii) randomized order-only functional twins that isolate a finite-panel sign reversal, plus controlled construct tests of the admission stages, including prespecified failures and negative controls. The contribution is a falsifiable admission discipline, not a benchmark-performance claim or guarantee of transport. Figure 1 shows the resulting workflow.

2 A Replicated Gain Does Not Justify Global Deployment

Three quantities answer different questions. A task-level *effect* asks what the edit changes on one task; *influence* asks whether it changes outcomes systematically across tasks, even when gains and harms cancel; *policy value* asks whether a declared deployment rule benefits a declared target population after cost. The first two diagnose the edit, the third governs deployment.

Let $Y_{iar} \in \{0, 1\}$ be the outcome on task i, harness arm a, and replicate r, and let $p_a(i) = \mathbb{E}_r[Y_{iar}]$. For treatment T, matched control C, and declared task distribution R, define

$$\tau_i = p_T(i) - p_C(i), \qquad \tau_R = \mathbb{E}_{i\sim R}[\tau_i]. \tag{1}$$

The sign of τ_i says whether the edit helps or harms task i; τ_R averages those signed effects for distribution R. Because gains and harms can cancel, we also record

$$\iota_R = \mathbb{E}_{i\sim R}[\tau_i^2] = \tau_R^2 + \mathrm{Var}_{i\sim R}(\tau_i). \tag{2}$$

Influence is the mean squared task-level effect. It is neither benefit nor heterogeneity: the heterogeneity component is $\iota_R - \tau_R^2$, and a uniformly helpful or harmful edit already has high influence. Influence can therefore expose cancellation that τ_R hides, but it cannot authorize promotion.

For binary outcomes with repeated runs we adapt the cross-snapshot minus same-snapshot disagreement correction of Ning (2026): cross-arm disagreement contains both treatment separation and run-to-run noise, and same-arm disagreement estimates the noise to subtract. Applying it to randomized harness arms and aggregating at the task level estimates ι_R. Appendix A gives the unpaired and paired U-statistics, assumptions, and proof. This diagnostic supports M1; the primary S1 sign-reversal test does not depend on it.

Replication and deployment fail for different reasons. Let $\hat{h}$ be selected using development data D. Let $\widehat{\tau}_D(\hat{h})$ be its observed development-set effect against the current harness h_0, and let $\tau_R(\hat{h})$ denote its expected effect under task distribution R as in Eq. 1, with $T = \hat{h}$ and $C = h_0$. Then

$$\widehat{\tau}_D(\hat{h}) - \tau_Q(\hat{h}) = \underbrace{\widehat{\tau}_D(\hat{h}) - \tau_D(\hat{h})}_{\text{adaptive-selection error}} + \underbrace{\tau_D(\hat{h}) - \tau_Q(\hat{h})}_{\text{deployment-scope gap}}. \tag{3}$$

Fresh, resource-matched confirmation on a trusted evaluator targets adaptive-selection error; functional twins and target-mixture evaluation target the deployment-scope gap. Repeating the source functional addresses only the first term. The second term compares causal effects but is identified only when both come from supported treatment–control comparisons; source-panel randomization alone does not identify transport to the target mixture.

Now make the deployment rule explicit. Let $S = s$ be a task type observable before treatment. Let $\tau_s(h)$ be the effect of edit h versus the current harness on that type. The switch $\pi(s) \in \{0, 1\}$ says whether to apply the edit, and $q_s = \Pr_Q(S = s)$ is that type's prevalence in the target population, with $\sum_s q_s = 1$. With deployment cost $C(h, \pi)$ and cost weight λ, incremental policy value is

$$\Delta V_q(h, \pi) = \mathbb{E}_{S\sim q}[\pi(S)\tau_S(h)] - \lambda C(h, \pi) = \sum_s q_s \pi(s)\tau_s(h) - \lambda C(h, \pi). \tag{4}$$

This is incremental value relative to leaving the current harness unchanged. Effects keep their sign, π controls exposure, q names the deployment mix, and the last term charges for cost. The experiments do not choose λ: the reference classifier instead treats a candidate/root metered-cost ratio above 1.25 as infeasible (`UNECONOMIC`) and ranks feasible candidates by uplift. Thus the hard cap is an operational surrogate for the cost term, not a numerical instantiation of λC without a chosen shadow price. The objective is standard policy learning (Kitagawa & Tetenov, 2018); our operational extension returns and validates (h, π) rather than promoting h alone. Global promotion sets $\pi \equiv 1$; rejection sets $\pi \equiv 0$.

Equation 4 assumes supported treatment/control comparisons and transport within each routed stratum, with no cross-task carryover. Effects that persist across tasks would instead require a sequential, dynamic-regime value (Murphy, 2003).

For intuition, suppose the edit gives gain $g > 0$ on lookup tasks and harm $-b < 0$ on mean-estimation tasks. If lookup comprises fraction p of the target and the edit is always on, its value before cost is $pg - (1-p)b$, positive only when $p > b/(g+b)$. A replicated lookup gain therefore cannot determine global value until the target mixture is specified. A route can instead set $\pi(\text{lookup}) = 1$ and $\pi(\text{mean}) = 0$, but that complete policy must itself be frozen and evaluated.

We call an edit *scope-overfit* when it has a replicated positive source effect but negative value under the declared target mixture, or when it violates a protected-functional harm margin; the causal admission object is therefore the pair (h, π). A global rule may require positive worst-case value over a prespecified set of plausible mixtures; otherwise the routed policy remains a new intervention that requires fresh evaluation.

Uptake sits inside this causal chain rather than certifying it. Opportunity, availability, invocation, delivered content, inspection, and outcome are different events, and conditioning on voluntary callers after making a tool available generally induces post-treatment selection. Our primary deployable estimand is therefore intention-to-treat. A caller-specific payload effect instead requires recording a committed call and only then randomizing real versus matched sham content, with the assignment fixed for the trajectory.

3 Task-Conditional Causal Admission: Admit the Edit and Its Deployment Rule

Section 2 gives TCCA three signals: signed stratum effects τ_s locate gains and harms, influence ι_R flags material changes without choosing a route, and incremental policy value $\Delta V_q(h, \pi)$ scores a complete exposure rule on the target mixture. Figure 1 contrasts a reference global-promotion arm with TCCA after a shared, resource-matched causal test. Both can establish a replicated local gain; only TCCA authorizes deployment scope and returns a frozen edit–policy pair.

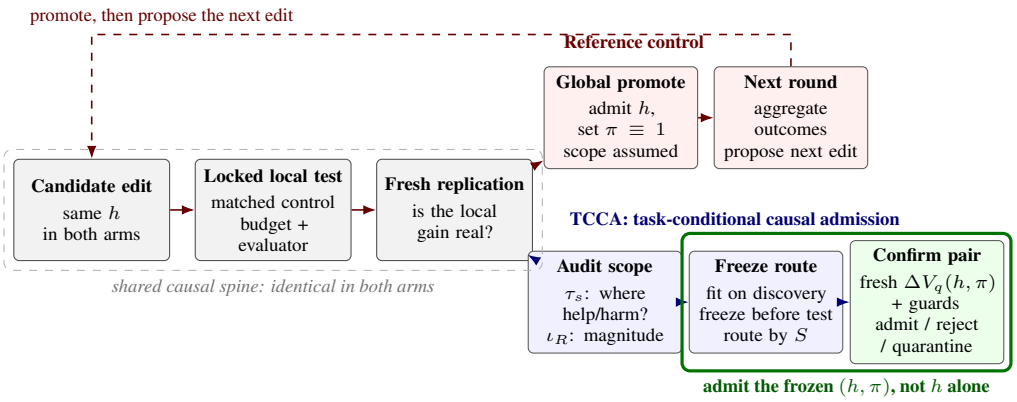

Figure 1: A shared causal spine, then two admission rules. Both arms take the same candidate h through the same matched, budget-locked test and the same fresh replication, so they receive the same evidence about whether the local gain is real. The reference control (top) promotes that gain globally—implicitly setting $\pi \equiv 1$—and iterates. TCCA (bottom) instead measures signed task-functional effects, freezes a route fit on pre-exposure features alone, and tests the complete (h, π) pair on fresh target-mixture data. The operational contribution is the boxed composition, not its individually established ingredients; the control abstracts an edit-only promote-and-repeat rule rather than any specific existing system.

The *evolvable plane* may change prompts, tools, context, memory, and control logic; the logically trusted plane owns splits, randomization, evaluation, denominators, budgets, thresholds, holdout

access, provenance, and rollback, and the proposer sees only authorized traces and files. This is a logical boundary with recorded checks, not operating-system isolation.

Each candidate enters through an atomic intervention contract naming the parent, causal edge, eligible tasks, matched control, endpoint, budget, allowed scope, protected functionals, and rollback. Treatment and control are identical until that edge. Added computation requires a resource-matched control; a context intervention also needs a control that preserves evidence membership and budget while removing the proposed signal. Appendix C gives the full contract and state machine.

The trusted plane validates the evidence, measures the opportunity-to-adherence funnel, runs randomized discovery with same-arm calibration, and estimates the quantities in Section 2. Invalid or manipulated evidence is rejected or quarantined. A candidate that never fires is recorded as dormant, not null, and an arm that outspends its control is recorded as confounded rather than better. Global admission additionally requires every protected-functional guard to pass; material heterogeneity instead triggers a scope audit.

A route may use only information available before exposure, and is fit, authorized, frozen, and tested on separate partitions. It must control actual access: withholding automatic injection does not help if a harmful payload remains callable. The complete routed pair is a new intervention and must pass fresh confirmation.

The method governs admission, not proposal quality, and evidence consumes exploration budget. We therefore report both false-promotion control and lost positive value. Hash-linked records improve auditability but cannot stop an actor that can rewrite the complete store.

4 Experimental Design

We use a layered evaluation because no single experiment identifies every quantity in Eq. 3. S1 is the primary causal scope experiment. H1 tests adaptive false-promotion control, R1 tests whether pre-treatment moderators can support routing, and H4a tests the bundled controller in a closed candidate pool. M1 and G1 provide supporting measurement and integrity construct tests. Merger into one paper does not upgrade exploratory or synthetic evidence into live confirmation.

4.1 S1: identical-order functional twins

S1 constructs 48 finite worlds with 24 records each, balanced over four renderer surfaces and two rank/value-sign strata. Each world yields two tasks over the same records: retrieve the value of one named record, or estimate the population mean of the same numeric field. A query-independent salience order is frozen before the lookup target is placed at rank one. Treatment presents that order. Control presents one unconditional, complete permutation containing the identical records. Critically, the same treatment and control orders are reused across the two functional twins, so the requested operation changes while world, evidence membership, and ordering contrast remain fixed.

The prospectively specified confirmation crosses the twins with three model endpoints under forced delivery, sharded access, one attempt, and a common bounded numeric scale: 288 randomized pairs and 576 intention-to-treat trajectories. Hidden bits assign real versus permutation order within each pair. The finite-panel directional intersection–union test requires positive lookup and negative mean effects over 144 pairs per family; its bounds are not model-population p-values. Endpoint-specific Holm-adjusted contrasts are secondary and do not identify an intelligence effect. Appendix E gives the estimand, bounds, and assumptions.

A preceding 1,344-trajectory, 16-world pilot compared no component, voluntary pull, automatic push, and an operation-aware gate. Because response contracts differed across functionals, it supplies only supporting mechanism evidence; Appendix E gives its design.

4.2 H1, R1, and H4a: controlled admission tests

H1 compares score-only, repeated-mean, calibration-only, and full-confirmation strategies across 100 sharp-null and 100 planted-positive adaptive-search campaigns under a 4,800-trajectory cap and common locked panel. Full admission combines A/A calibration, sequential multiplicity control, and

Table 1: S1 finite-panel intention-to-treat effects. The directional test requires a positive lookup effect and a negative mean-estimation effect.

Functional	Salience	Permutation	Effect	Hoeffding bound	Pairs
Named-record lookup	0.9653	0.2569	$+0.7083$	2.05×10^{-16}	144
Population mean	0.0833	0.7500	-0.6667	1.27×10^{-14}	144

fresh 32-task confirmation. Its prespecified criteria bound false promotion and require latent locked gain within 0.05 of the best simpler method; exact effects and thresholds appear in Appendix F.

R1 tests routing under predictable, unpredictable, and sparse heterogeneity. An allowlisted one-level stump is fit, authorized, frozen, and tested against always-off, always-on, and a simulator oracle. The unpredictable case deliberately stress-tests authorization by a positive validation point estimate rather than a lower bound; Appendix H gives the generators, splits, and effects.

H4a is a one-step wind tunnel with 400 paired campaigns, frozen 24-candidate pools, a shared keyed outcome cube and 240-task holdout, and a common 6,000-cell search cap. Full admission bundles contract and evaluator checks, A/A calibration, matched controls, routing, and fresh confirmation; the primary contrast is mean observed simulated holdout uplift versus score-only. The comparison is budget-capped, not equal-realized-cost. Appendix I gives the candidate taxonomy and allocation.

M1 retrospectively measures a frozen production `file_map` on 31 software-engineering tasks and adds two prospective manipulation checks; it motivates the measurement funnel but does not test routing (Appendix D). G1 tests declared guards on eight attacks, two valid positives, and two benign non-positive controls; its handcrafted pool tests implementation correspondence, not novel-attack resistance (Appendix G).

Inference units follow assignment: randomized model–world–functional pairs in S1, campaigns in H1/H4a, and tasks in M1/R1. Reported intervals condition on each frozen simulation or selected gate; they do not establish robustness over generators, seeds, models, repositories, or deployment populations. Appendix A gives the study-specific procedures.

5 Results

5.1 One replicated ordering, opposite functional effects

Table 1 reports the prospectively specified S1 contrast. The exact same treatment order was delivered to each lookup/mean twin, and the exact same control permutation was delivered to its twin. Treatment and control contained the same records and differed only in order.

The directional intersection–union bound is $p \le 1.27 \times 10^{-14}$. Continuous outcomes agree: salience reduced lookup absolute error by 38.23 units and increased mean absolute error by 6.70 units. All three endpoints had the prespecified directions and passed the original Holm-corrected secondary conjunction. Model identity was not randomized, and three endpoints do not define an intelligence scale.

For a target mixture with lookup prevalence p, the observed binary value of global salience relative to permutation is

$$\Delta(p) = 0.7083p - 0.6667(1-p) = 1.375p - 0.6667. \tag{5}$$

Using the underlying fractions, its point-estimate boundary is exactly $p = 16/33 \approx 0.4848$. A balanced benchmark reports only $+0.0208$, nearly erasing both a large benefit and a large harm. The point is not that order affects behavior: an identical membership-preserving intervention has opposite causal signs on two functionals over the same worlds. Confirmation on one functional therefore cannot license global deployment; this is a causal deployment-scope gap, not a failure to replicate.

The primary result includes 18 HTTP-500 errors and 37 non-submissions, with more control-only than treatment-only failures. It is consequently the effect of the deployed ordering policy, including transport risk, rather than a pure semantic payload effect. A post-hoc adversarial recoding that scores every non-submission against the claimed direction preserves lookup $+0.5139$, mean -0.6458, and a pooled bound of 5.53×10^{-9}. The Opus endpoint had no failures or repairs and retained lookup

$+0.8125$ and mean -0.5417. Thus the sign reversal survives both post-hoc adversarial recoding and restriction to a failure-free endpoint. S1 remains a constructed finite-panel existence result; Section 8 states what it does and does not establish.

5.2 Voluntary use does not repair invalid scope

The pilot isolates a practical failure of superficial routing. Its operation-aware gate withheld automatic delivery on mean-estimation tasks, but the component remained callable and was voluntarily invoked in 95/96 trajectories. Opus called the map in 16/16 sharded mean episodes in both real and permutation pull arms, while accuracy fell from 0.75 with permutation to 0.25 with salience. The harmful mean contrast under the gate remained -0.5417.

The model invoked the map for record access but did not compensate for the relevance order when estimating the population mean. Voluntary use therefore cannot by itself certify safe deployment scope. A gate must control availability or payload mode when exposure itself is risky, and a caller-specific effect requires post-call payload randomization. We do not infer that voluntary use is generally harmful from this pilot.

5.3 Routing requires a predictable moderator

R1 separates heterogeneity from actionable heterogeneity. In the predictable synthetic setting, a frozen navigation-pressure gate gained $+0.0688$ over always-off (95% task-bootstrap CI $[0.0312, 0.1083]$) and $+0.0854$ over always-on ($[0.0312, 0.1375]$), while recovering 98.8% of the latent simulator-oracle gain in that setting. With no predictive moderator, the same procedure gained -0.0333 ($[-0.0854, 0.0167]$) on held-out tasks. Sparse deterministic routing gained $+0.1613$ over off, but its lower bound over always-on was $+0.0161$, below the declared $+0.02$ margin. Influence detects consequential variation; it cannot manufacture a reliable route. These conditional intervals are simulated and condition on the selected gate.

5.4 Integrated admission reduces simulated downside risk

Across 400 H4a campaigns, full admission achieved $+0.0193$ mean observed simulated holdout uplift, 95% interval $[0.0160, 0.0232]$, versus $+0.0044$ for score-only; the paired difference was $+0.0149$ $[0.0104, 0.0198]$, and attack selection fell from 350/400 to zero. Against trusted score-only, the mean difference was inconclusive: -0.0024 $[-0.0095, 0.0045]$. At the prespecified observed-regression threshold, however, material root regressions were 124/400 (31.0%) under trusted score-only and 1/400 (0.3%) under full admission. This secondary simulator-conditioned result records a large difference in downside events, not mean superiority, equivalence, or noninferiority. Removing calibration changed mean uplift by only -0.0001 $[-0.0008, 0.0007]$. Full also used 5,585 mean search cells versus 4,800 for each score-only baseline, so H4a neither establishes equal-realized-cost dominance nor incremental value from every stage.

5.5 Admission reduces false promotion, at a cost

Table 2 summarizes the controller tests. In H1, full calibration plus fresh confirmation produced no false promotion in 100 sharp-null campaigns (95% Wilson upper endpoint 3.70%), but promoted only 86 of 200 planted positive candidates encountered across 100 campaigns and achieved 0.2469 mean latent gain versus 0.4697 for repeated means. Each positive campaign contains two planted positives, yielding the denominator of 200. The prespecified utility criterion therefore fails. The repeated-mean comparator, however, promotes a null in all 100 sharp-null campaigns; because the arms differ in stopping and evidence allocation, the utility gap cannot be attributed to one admission stage.

G1 provides the corresponding integrity construct check. Full admission rejected all eight planted attacks and both benign non-positive controls while advancing both valid positives. Score-only advanced all eight attacks; each of eight leave-one-guard-out variants advanced its matched attack. This verifies the declared checks on a handcrafted pool, not resistance to novel attacks.

Supporting M1 evidence is consistent with this caution. On 31 previously inspected `file_map` tasks, three failures became passes and three passes became failures: zero mean uplift but influence 0.194, interval $[0.065, 0.355]$. A manipulation check then saw zero calls in nine eligible episodes and

Table 2: Controlled admission results. Intervals condition on each frozen simulation design. "Full" is the complete policy within that study, not a claim of live deployment performance.

Study	Primary result	Complementary result / limit
H1	Full: 0/100 null campaigns promoted; 86/200 encountered positives promoted	43.0% recall; latent gain 0.2469 vs. 0.4697 for repeated mean
G1	Full rejected 8/8 planted attacks; score-only advanced 8/8	Handcrafted finite pool; no novel-attack inference
R1	Predictable routing: $+0.0688$ vs. off, CI $[0.0312, 0.1083]$; $+0.0854$ vs. on, CI $[0.0312, 0.1375]$	Unpredictable gate: -0.0333, CI $[-0.0854, 0.0167]$
H4a	Full vs. score-only: $+0.0149$, CI $[0.0104, 0.0198]$; latent attacks 0% vs. 87.5%	Vs. trusted: mean -0.0024, CI $[-0.0095, 0.0045]$; root regressions 1/400 vs. 124/400

stopped without an effectiveness conclusion. A separate prospectively specified history-elision study achieved 87% uptake but estimated -0.058 uplift; the negative effect was not statistically established.

6 Related Work

Self-Harness, Recursive Harness Self-Improvement, Evo-Harness, and Meta-Harness optimize edits, loops, skills, or complete harness programs (Zhang et al., 2026a; Lee et al., 2026a; Wei et al., 2026; Lee et al., 2026b); related systems add commit gates, rollback, routing, judge separation, and sandboxed evaluation (Nie et al., 2026a; Liu et al., 2026; Du et al., 2026; Nie et al., 2026b; Huang et al., 2026). These ingredients are not claimed as contributions.

DSPy compiles declarative LM pipelines into prompts and demonstrations against a user-specified metric (Khattab et al., 2024). It provides upstream optimizer context; TCCA instead audits the causal deployment scope of a fixed edit and does not inherit a causal or anti-overfitting guarantee from DSPy.

Evaluation work matches budgets, tests held-out transfer, corrects adaptive sampling, constrains edits, and gates contaminated skills (Wang et al., 2026a; Miyai et al., 2026; Shang et al., 2026; Lin et al., 2026). Closest to our concern, HarnessCompass steers evolution toward harnesses that generalize (Zhang et al., 2026b). It searches for a better edit; we hold the edit fixed and test which tasks may receive it, so our experiments are not a head-to-head optimizer comparison. A study with more than 3.1 million rollouts over 30 harnesses and 12 model–problem pairs finds no universal fixed winner and uses adaptive allocation (Gupta et al., 2026). Non-universality and evaluation hygiene are therefore premises.

AgentLens shows that process measures can reorder systems relative to pass rate; WARP shows relevance retrieval distorting population-representative evidence (Sahoo et al., 2026; Thakur et al., 2026). Such distortion has a documented substrate: accuracy depends on evidence position (Liu et al., 2024a). Other audits compare clean evidence with matched sham or corrupted observations (Wang et al., 2026b; Luo & Peng, 2026); separate no-, retrieved-, full-, and oracle-evidence conditions (Xia, 2026); and study no/self/always-tool policies, learned controllers, and model-dependent tool effects (Liu, 2026; Wu et al., 2026; Han et al., 2026; Fan et al., 2026). Our narrower contribution is the order-only same-world functional twin and an admission object that includes its deployment policy.

The formal treatment uses adaptive data analysis, causal inference, heterogeneous-effect policy learning, and multiplicity control (Dwork et al., 2015a;b; Imbens & Rubin, 2015; Athey & Imbens, 2016; Wager & Athey, 2018; Kitagawa & Tetenov, 2018; Holm, 1979); reward hacking motivates the integrity boundary (Amodei et al., 2016; Manheim & Garrabrant, 2019). WebArena, AgentBench, and SWE-bench measure absolute end-to-end agent performance (Zhou et al., 2024; Liu et al., 2024b; Jimenez et al., 2024). SWE-agent shows that an agent–computer interface can change software-engineering performance, while OpenHands treats tools, runtime, context, and control flow as first-class platform components (Yang et al., 2024; Wang et al., 2025). Our controlled studies instead isolate component effects and policy scope; they are neither leaderboard nor state-of-the-art comparisons. No individual guard or estimator is new here.

7 Discussion: Seven Illusions of Harness Effectiveness

Winner correction asks whether an edit is real; scope auditing asks where it is valuable. H1 probes the first, while S1 witnesses the second. R1 then shows, in a predictable synthetic case, that a frozen pre-treatment route can outperform both global policies. None of these results by itself authorizes deployment in a natural target population.

A self-improving agent should therefore return an edit–policy pair with an auditable target population, not only a patch, prompt, tool, or context policy. Global promotion requires evidence on the deployment mixture and protected functionals. Conditional promotion requires a pre-treatment moderator and fresh evaluation of the complete route. Otherwise, reject or study the edit further.

Seven recurring evaluation illusions summarize the resulting discipline:

1. **Neutral baseline.** Unmatched evidence, interfaces, or resources credit the component. Match them; add no-component only for deployment.
2. **Free compute.** Extra calls, turns, or tokens masquerade as value. Match budgets or report the cost–value frontier.
3. **Enabled means used.** Availability is not delivery. Record opportunity, invocation, delivery, inspection, and adherence.
4. **Used means useful.** Uptake is a mediator, not an outcome. Use intention-to-treat and randomize post-call payloads for caller effects.
5. **Different means causal.** Evolving stacks confound attribution. Freeze artifacts and randomize the declared causal edge.
6. **Pass/fail is sufficient.** One endpoint hides stability, cost, mechanism, and magnitude. Preserve a typed diagnostic vector.
7. **One scalar effect transfers.** Averages can erase sign reversals. Report conditional effects, mixture sensitivity, and harms.

These are an evidence checklist, not seven novelty claims. M1 separates uptake, uplift, and influence; S1 demonstrates a sign reversal; H1 retains 0.2469 latent gain versus 0.4697 for repeated mean with 0/100 observed false promotions in this frozen simulation; and H4a records 1/400 versus 124/400 simulated root regressions while its mean contrast against a trusted baseline remains unresolved. Self-selection does not guarantee valid scope: the S1 pilot detected an access problem but not invalid sampling. Conservative admission can also waste evidence. H4a finds no evidence that paid calibration adds mean value and spends 785 more mean search cells than score-only, so its primary contrast is budget-capped rather than equal-cost. At scale, cheap integrity screens and sequential allocation should reserve replication and fresh holdout evaluation for finalists. Extra screening is relevant when downside events carry declared value; if only mean uplift matters, H4a gives no basis to prefer full admission over trusted score-only. Either choice should be reported on a cost–value frontier.

8 Limitations

The strongest confirmed positive causal result is synthetic.

S1 uses constructed worlds and places the lookup target at rank one; it does not estimate reversal prevalence in repositories, websites, filings, logs, or traffic. The mean-estimation harm is observed model behavior rather than a mechanically assigned outcome: both arms contain the same 24 records, and the scoring rule does not force a difference. It compares two complete orderings, not tool availability. Its binary endpoint depends on a prespecified threshold, though continuous absolute-error endpoints agree in both directions. We did not test whether disclosing the relevance order removes the harm.

The controller studies are construct tests. H1 uses one configuration and shows a safety–utility trade-off. G1's attacks were designed with their guards. R1's intervals condition on a selected gate; its unpredictable control does not validate authorization. H4a is a one-step closed-pool simulation without a live proposer, cumulative mutations, canary, or rollback. It establishes neither mean superiority nor equivalence against trusted score-only; its root-regression contrast is a secondary,

simulator-conditioned result, and full admission uses 16.4% more realized search. Full selected no routed policy in this pool, so integrated routing evidence comes from R1 rather than H4a. The previously inspected M1 panel cannot confirm a natural-domain effect. Static features and a frozen target mixture can become stale when task requirements emerge mid-execution or deployment prevalence shifts. Such settings need time-indexed monitoring and fresh evaluation under a dynamic-regime estimand, not post-treatment rerouting under the original claim.

Local hash-linked records do not prove process isolation or an externally anchored ledger. S1 was sealed but not externally timestamped; Appendix E discloses its verifier repair. Model identity was observed, not randomized. Demonstrating practical regret reduction would require live optimizer campaigns, natural functional twins, frozen mixtures, root-relative comparisons, separated custody, and a sealed holdout. Current evidence supports a method and controlled mechanisms, not live self-improvement, population benefit, or state of the art.

9 CONCLUSION

Harness evolution can overfit in selection and scope: fresh confirmation does not make a local effect globally valid, so TCCA admits an edit with its deployment rule. Controlled tests support the falsifiable protocol, not a claim of live self-improvement gains, population benefit, or state of the art.

AI USE STATEMENT

Generative AI tools were used to assist literature discovery, draft and revise prose, generate and review experiment and analysis code, and diagnose software failures. The authors inspected the source evidence, reran the reported checks, and take responsibility for every claim, citation, artifact, and error. AI systems were not treated as authors and did not make autonomous publication or submission decisions.

REPRODUCIBILITY STATEMENT

The anonymous supplement contains claim-verification programs, machine-readable result summaries, the S1 protocol, and a hashed manifest. It checks manuscript-facing design, provenance, and result fields and recomputes selected formulaic relationships from the summaries, but is not a complete raw-trajectory release. Appendix K maps each result to its evidence class and verification boundary; Appendix A states the estimands and inferential assumptions. The allowlisted supplement excludes credentials, private infrastructure, external model transcripts, and non-redistributable raw inputs.

ETHICS STATEMENT

The controlled studies use synthetic worlds and simulations, and the M1 panel uses previously collected agent-evaluation records. No participants are recruited and no human outcomes are studied. Harness automation can amplify unsafe tool use or overstate benchmark progress, so the reported studies include negative and null findings, retain fail-closed admission states, and separate exploratory evidence from causal and population claims. We report resource accounting where available and exclude credentials, private infrastructure, and operational security material from the anonymous supplement.

REFERENCES

Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. Concrete problems in AI safety. *arXiv preprint arXiv:1606.06565*, 2016.

Susan Athey and Guido Imbens. Recursive partitioning for heterogeneous causal effects. *Proceedings of the National Academy of Sciences*, 113(27):7353–7360, 2016. doi: 10.1073/pnas.1510489113.

Yuetian Du, Yucheng Wang, He Xu, Jiexu Xu, Shanwen Tan, Bing Zhao, Boyu Yang, Zhijie Xu, Ming Kong, Hu Wei, Jie Liu, and Qiang Zhu. Living-harness is an interactive-agent evolver, 2026. URL https://arxiv.org/abs/2607.26598v2. arXiv:2607.26598v2, 11 August 2026.

Cynthia Dwork, Vitaly Feldman, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Aaron Roth. Preserving statistical validity in adaptive data analysis. In *Proceedings of the 47th Annual ACM Symposium on Theory of Computing*, pp. 117–126, 2015a. doi: 10.1145/2746539.2746580. arXiv:1411.2664.

Cynthia Dwork, Vitaly Feldman, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Aaron Roth. The reusable holdout: Preserving validity in adaptive data analysis. *Science*, 349(6248):636–638, 2015b. doi: 10.1126/science.aaa9375.

Bradley Efron and Robert J. Tibshirani. *An Introduction to the Bootstrap*. Chapman and Hall/CRC, 1993. doi: 10.1201/9780429246593.

Siqi Fan, Minghao Li, Xiaoqian Ma, Wenhui Tan, Xiusheng Huang, Juntong Wu, Liujie Zhang, Shuo Shang, and Weihang Chen. Screenshots or tools? eliciting tool use and managing multimodal context in hybrid GUI–MCP computer-use agents, August 2026. URL https://arxiv.org/abs/2608.03327v1. arXiv:2608.03327v1.

Hanghui Guo, Weijie Shi, Zhangze Chen, Shengxiang Xu, Yishu Wang, Yimei Zhang, Wangze Ni, Jia Zhu, and Shimin Di. DREvo: Distilling recalibrated historical experience for harness self-evolution, July 2026. URL https://arxiv.org/abs/2607.26722v2. arXiv:2607.26722v2.

Akshat Gupta, Jermaine Lei, Alexander Lu, Gopala Anumanchipalli, and Leshem Choshen. Automated discovery has no universally superior harness, July 2026. URL https://arxiv.org/abs/2607.18235v1. arXiv:2607.18235v1.

Jinyi Han, Yuanjian Xu, Ying Liao, Xinyi Wang, Zishang Jiang, Zixiang Di, Fanyang Lu, Zhichao Hu, and Yanghua Xiao. Skill-Use: Can LLMs actually use skills in agentic harnesses?, August 2026. URL https://arxiv.org/abs/2608.04828v1. arXiv:2608.04828v1.

Sture Holm. A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, 6(2):65–70, 1979.

Lisheng Huang, Chen Yang, Hao Zhou, Huatong Song, Zongchao Chen, Ran Le, Yang Song, Wayne Xin Zhao, and Tao Zhang. Evo-Bench: Can language models improve agent harness?, August 2026. URL https://arxiv.org/abs/2608.09096v2. arXiv:2608.09096v2.

Guido W. Imbens and Donald B. Rubin. *Causal Inference for Statistics, Social, and Biomedical Sciences: An Introduction*. Cambridge University Press, 2015. doi: 10.1017/CBO9781139025751.

Carlos E. Jimenez, John Yang, Alexander Wettig, Shunyu Yao, Kexin Pei, Ofir Press, and Karthik Narasimhan. SWE-bench: Can language models resolve real-world GitHub issues? In *International Conference on Learning Representations*, 2024. arXiv:2310.06770.

Omar Khattab, Arnav Singhvi, Paridhi Maheshwari, Zhiyuan Zhang, Keshav Santhanam, Sri Vardhamanan, Saiful Haq, Ashutosh Sharma, Thomas T. Joshi, Hanna Moazam, Heather Miller, Matei Zaharia, and Christopher Potts. DSPy: Compiling declarative language model calls into self-improving pipelines. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=sY5N0zY5Od. arXiv:2310.03714.

Toru Kitagawa and Aleksey Tetenov. Who should be treated? empirical welfare maximization methods for treatment choice. *Econometrica*, 86(2):591–616, 2018. doi: 10.3982/ECTA13288.

Hyunin Lee, Jinglue Xu, Jeffrey Seely, Donghyun Lee, Matei Zaharia, and Yujin Tang. Recursive harness self-improvement, July 2026a. URL https://arxiv.org/abs/2607.15524v1. arXiv:2607.15524v1.

Yoonho Lee, Roshen Nair, Qizheng Zhang, Kangwook Lee, Omar Khattab, and Chelsea Finn. Meta-Harness: End-to-end optimization of model harnesses, March 2026b. URL https://arxiv.org/abs/2603.28052v1. arXiv:2603.28052v1.

Jiahang Lin, Shichun Liu, Chengjun Pan, Lizhi Lin, Shihan Dou, Zhiheng Xi, Xuanjing Huang, Hang Yan, Zhenhua Han, Tao Gui, and Yu-Gang Jiang. Agentic harness engineering: Observability-driven automatic evolution of coding-agent harnesses, 2026. URL https://arxiv.org/abs/2604.25850v4. arXiv:2604.25850v4, 18 May 2026.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173, 2024a. doi: 10.1162/tacl_a_00638. URL https://aclanthology.org/2024.tacl-1.9/. arXiv:2307.03172.

Peiyang Liu. The laws of context allocation: Causal measurement and closed-loop orchestration in generative search, August 2026. URL https://arxiv.org/abs/2608.23252v1. arXiv:2608.23252v1.

Xiao Liu, Hao Yu, Hanchen Zhang, Yifan Xu, Xuanyu Lei, Hanyu Lai, Yu Gu, Hangliang Ding, Kaiwen Men, Kejuan Yang, Shudan Zhang, Xiang Deng, Aohan Zeng, Zhengxiao Du, Chenhui Zhang, Sheng Shen, Tianjun Zhang, Yu Su, Huan Sun, Minlie Huang, Yuxiao Dong, and Jie Tang. AgentBench: Evaluating LLMs as agents. In *International Conference on Learning Representations*, 2024b. URL https://arxiv.org/abs/2308.03688. arXiv:2308.03688.

Zewen Liu, Zhan Shi, Yisi Sang, Bing He, Minhua Lin, Tianxin Wei, Dakuo Wang, Benoit Dumoulin, Wei Jin, and Hanqing Lu. Adaptive auto-harness: Sustained self-improvement for agentic system deployment on open-ended task streams, 2026. URL https://arxiv.org/abs/2606.01770v2. arXiv:2606.01770v2, 3 June 2026.

Jingkun Luo and Da-Tian Peng. Success is not self-explanatory: Auditing success provenance in agent evaluation, July 2026. URL https://arxiv.org/abs/2607.24054v1. arXiv:2607.24054v1.

David Manheim and Scott Garrabrant. Categorizing variants of goodhart's law. *Minds and Machines*, 29:315–327, 2019. doi: 10.1007/s11023-019-09502-2. arXiv:1803.04585.

Atsuyuki Miyai, Kiyoharu Aizawa, and Toshihiko Yamasaki. Task-CoEvolve: Efficient harness optimization via adaptive validation task selection, August 2026. URL https://arxiv.org/abs/2608.20169v2. arXiv:2608.20169v2.

Susan A. Murphy. Optimal dynamic treatment regimes. *Journal of the Royal Statistical Society: Series B*, 65(2):331–355, 2003. doi: 10.1111/1467-9868.00389.

Jun Nie, Yonggang Zhang, Qianshu Cai, Yiu-ming Cheung, Xinmei Tian, and Bo Han. EvolveNet: Collaborative harness evolution for agent self-improvement, August 2026a. URL https://arxiv.org/abs/2608.04968v1. arXiv:2608.04968v1.

Jun Nie, Yonggang Zhang, Jun Song, Qianshu Cai, Dahai Yu, Yike Guo, Xinmei Tian, and Bo Han. TTHE: Test-time harness evolution, 2026b. URL https://arxiv.org/abs/2607.08124v1. arXiv:2607.08124v1, 9 July 2026.

Jingjie Ning. Same agent, different answers: A repeat-aware audit of corpus-induced answer churn in retrieval-augmented QA, August 2026. URL https://arxiv.org/abs/2608.22856v1. arXiv:2608.22856v1.

Xuying Ning, Katherine Tieu, Dongqi Fu, Tianxin Wei, Zihao Li, Yuanchen Bei, Jiaru Zou, Mengting Ai, Zhining Liu, Ting-Wei Li, Lingjie Chen, Yanjun Zhao, Ke Yang, Bingxuan Li, Cheng Qian, Gaotang Li, Xiao Lin, Zhichen Zeng, Ruizhong Qiu, Sirui Chen, Yifan Sun, Xiyuan Yang, Ruida Wang, Rui Pan, Chenyuan Yang, Dylan Zhang, Liri Fang, Zikun Cui, Yang Cao, Pan Chen, Dorothy Sun, Ren Chen, Mahesh Srinivasan, Nipun Mathur, Yinglong Xia, Hong Li, Hong Yan, Pan Lu, Lingming Zhang, Tong Zhang, Hanghang Tong, and Jingrui He. Code as agent harness, 2026. URL https://arxiv.org/abs/2605.18747v1. arXiv:2605.18747v1, 18 May 2026.

Priyam Sahoo, Gaurav Mittal, Xiaomin Li, Shengjie Ma, Benjamin Steenhoek, Pingping Lin, and Yu Hu. AgentLens: Revealing the lucky pass problem in SWE-agent evaluation, May 2026. URL https://arxiv.org/abs/2605.12925v1. arXiv:2605.12925v1.

Linfang Shang, Ming Xu, Yiding Sun, Tianle Xia, Lingxiang Hu, Lan Xu, and Ning Zheng. When self-evolution backfires: Pre-commit gating against skill contamination in LLM agents, August 2026. URL https://arxiv.org/abs/2608.05810v1. arXiv:2608.05810v1.

Aman Singh Thakur, Aditya Agrawal, Alwarappan Nakkiran, and Alex Karlsson. WARP: Wasserstein-aligned RAG for population opinions, August 2026. URL https://arxiv.org/abs/2608.22859v1. arXiv:2608.22859v1.

Stefan Wager and Susan Athey. Estimation and inference of heterogeneous treatment effects using random forests. *Journal of the American Statistical Association*, 113(523):1228–1242, 2018. doi: 10.1080/01621459.2017.1319839.

Xingyao Wang, Boxuan Li, Yufan Song, Frank F. Xu, Xiangru Tang, Mingchen Zhuge, Jiayi Pan, Yueqi Song, Bowen Li, Jaskirat Singh, Hoang H. Tran, Fuqiang Li, Ren Ma, Mingzhang Zheng, Bill Qian, Yanjun Shao, Niklas Muennighoff, Yizhe Zhang, Binyuan Hui, Junyang Lin, Robert Brennan, Hao Peng, Heng Ji, and Graham Neubig. OpenHands: An open platform for AI software developers as generalist agents. In *International Conference on Learning Representations*, 2025. arXiv:2407.16741.

Yike Wang, Huaisheng Zhu, Zhengyu Hu, Yige Yuan, Zhengyu Chen, Shakti Senthil, Hannaneh Hajishirzi, Yulia Tsvetkov, Pradeep Dasigi, and Teng Xiao. Rethinking the evaluation of harness evolution for agents, July 2026a. URL https://arxiv.org/abs/2607.12227v2. arXiv:2607.12227v2.

Zhiheng Wang, Bo Peng, Lai Wei, and Chaochao Lu. The illusion of visual tool-use: A causal audit of thinking with images, August 2026b. URL https://arxiv.org/abs/2608.06270v1. arXiv:2608.06270v1.

Tianxin Wei, Zhan Shi, Minhua Lin, Bing He, Zewen Liu, Yisi Sang, Yuanchen Bei, Xuying Ning, Jiaru Zou, Ting-Wei Li, Xiao Lin, Yanjun Zhao, Chi Wang, Benoit Dumoulin, Dakuo Wang, Jingrui He, and Hanqing Lu. Evo-Harness: Context-to-harness skill compilation for self-evolving agents, August 2026. URL https://arxiv.org/abs/2608.15071v1. arXiv:2608.15071v1.

Edwin B. Wilson. Probable inference, the law of succession, and statistical inference. *Journal of the American Statistical Association*, 22(158):209–212, 1927. doi: 10.1080/01621459.1927.10502953.

Qinyuan Wu, Soumi Das, Mahsa Amani, Arijit Nag, Seungeon Lee, Krishna Gummadi, Abhilasha Ravichander, and Muhammad Bilal Zafar. To call or not to call: A framework to assess and optimize LLM tool calling, May 2026. URL https://arxiv.org/abs/2605.00737v1. arXiv:2605.00737v1.

Haizhou Xia. A four-condition diagnostic protocol for evidence utilization in long-context and retrieval-augmented language models, June 2026. URL https://arxiv.org/abs/2606.06758v1. arXiv:2606.06758v1.

John Yang, Carlos E. Jimenez, Alexander Wettig, Kilian Lieret, Shunyu Yao, Karthik Narasimhan, and Ofir Press. SWE-agent: Agent-computer interfaces enable automated software engineering. In *Advances in Neural Information Processing Systems*, 2024. arXiv:2405.15793.

Hangfan Zhang, Shao Zhang, Kangcong Li, Chen Zhang, Yang Chen, Yiqun Zhang, Lei Bai, and Shuyue Hu. Self-Harness: Harnesses that improve themselves, June 2026a. URL https://arxiv.org/abs/2606.09498v3. arXiv:2606.09498v3.

Luan Zhang, Ruochen Zhou, Dandan Song, Zhengyu Chen, Yuhang Tian, Jun Yang, Huipeng Ma, Chenhao Li, Guangyuan Feng, Xudong Li, Yizhou Jin, and Yan Xu. HarnessCompass: Guiding automatic harness evolution toward generalizable and effective agent harnesses, August 2026b. URL https://arxiv.org/abs/2608.01918v1. arXiv:2608.01918v1.

Shuyan Zhou, Frank F. Xu, Hao Zhu, Xuhui Zhou, Robert Lo, Abishek Sridhar, Xianyi Cheng, Tianyue Ou, Yonatan Bisk, Daniel Fried, Uri Alon, and Graham Neubig. WebArena: A realistic web environment for building autonomous agents. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2307.13854. arXiv:2307.13854.

A Estimands, assumptions, and identification

A.1 Task-level estimands

For task i, harness arm $a \in \{C, T\}$, and replicate r, let $Y_{iar} \in \{0, 1\}$ and $p_a(i) = \mathbb{E}_r[Y_{iar}]$. The task-level treatment effect is $\tau_i = p_T(i) - p_C(i)$. For a declared task distribution R, we report both the signed mean effect

$$\tau_R = \mathbb{E}_{i\sim R}[\tau_i] \tag{6}$$

and the influence diagnostic

$$\iota_R = \mathbb{E}_{i\sim R}[\tau_i^2] = \tau_R^2 + \mathrm{Var}_{i\sim R}(\tau_i). \tag{7}$$

Scope. In the reported analyses, R is only the equal-weighted sampled panel. It has a population interpretation only when the task sample and weighting scheme represent a declared target population. Influence is not a utility: it is large for both beneficial and harmful changes. It measures effect magnitude; heterogeneity is $\iota_R - \tau_R^2$.

A.2 UNPAIRED DISCORDANCE ESTIMATOR

Let $R_{Ci}, R_{Ti} \geq 2$ and define

$$\widehat{D}_{CT,i} = \frac{1}{R_{Ci}R_{Ti}} \sum_{r=1}^{R_{Ci}} \sum_{s=1}^{R_{Ti}} \mathbf{1}(Y_{iCr} \neq Y_{iTs}), \tag{8}$$

$$\widehat{D}_{aa,i} = \binom{R_{ai}}{2}^{-1} \sum_{r<s} \mathbf{1}(Y_{iar} \neq Y_{ias}). \tag{9}$$

The implemented estimator is

$$\widehat{\iota}_i = \widehat{D}_{CT,i} - \frac{\widehat{D}_{CC,i} + \widehat{D}_{TT,i}}{2}, \qquad \widehat{\iota} = \frac{1}{n} \sum_i \widehat{\iota}_i. \tag{10}$$

Under stable task-specific success probabilities and conditionally independent Bernoulli replicates within and across arms,

$$\mathbb{E}[\widehat{D}_{CT,i} \mid i] = p_C(i) + p_T(i) - 2p_C(i)p_T(i), \tag{11}$$

$$\mathbb{E}[\widehat{D}_{aa,i} \mid i] = 2p_a(i)\{1 - p_a(i)\}. \tag{12}$$

Substitution in Eq. 10 gives

$$\mathbb{E}[\widehat{\iota}_i \mid i] = p_C + p_T - 2p_Cp_T - p_C(1 - p_C) - p_T(1 - p_T) \tag{13}$$

$$= (p_T - p_C)^2 = \tau_i^2. \tag{14}$$

The pairwise averages are U-statistics, so unequal replicate counts do not change this expectation. Negative finite-sample values are possible and are retained rather than truncated. Same Agent, Different Answers previously introduced cross-snapshot minus same-snapshot disagreement for answer churn (Ning, 2026); on binary exact outcomes that identity is precisely Eq. 10. Our contribution is its use inside a randomized harness-admission analysis, not the algebra itself.

A.3 PAIRED ESTIMATOR

When treatment and control share a randomized replicate block, define $D_{ir} = Y_{iTr} - Y_{iCr}$ for $r = 1, \ldots, R$. The paired estimator is

$$\widehat{\iota}_i^{\text{pair}} = \frac{(\sum_r D_{ir})^2 - \sum_r D_{ir}^2}{R(R-1)} = \frac{1}{R(R-1)} \sum_{r \neq s} D_{ir}D_{is}. \tag{15}$$

If replicate blocks are independent and identically distributed conditional on task i, then $\mathbb{E}[D_{ir} \mid i] = \tau_i$ and $\mathbb{E}[D_{ir}D_{is} \mid i] = \tau_i^2$ for $r \neq s$. Hence the estimator is unbiased for τ_i^2 while allowing arbitrary treatment–control dependence inside a block. This is the preferred estimator for genuinely paired seeds.

A.4 SELECTION AND SCOPE ARE SEPARATE ERRORS

For an adaptively selected edit $\hat{h}$, let $\widehat{\tau}_D(\hat{h})$ denote its observed development-set effect against h_0, and let $\tau_R(\hat{h})$ denote its expected effect under task distribution R, as in Eq. 1 with $T = \hat{h}$ and $C = h_0$. The identity

$$\widehat{\tau}_D(\hat{h}) - \tau_Q(\hat{h}) = \{\widehat{\tau}_D(\hat{h}) - \tau_D(\hat{h})\} + \{\tau_D(\hat{h}) - \tau_Q(\hat{h})\} \tag{16}$$

separates adaptive-selection error from the deployment-scope gap. Fresh confirmation on the same functional composition addresses the first term but not the second. The latter compares causal effects but is identified only under supported treatment–control comparisons; source-panel randomization alone does not identify transport to Q.

Let $S = s$ be a pre-treatment functional stratum and $\tau_s(h) = \mathbb{E}[Y(h) - Y(h_0) \mid S = s]$. For a frozen deployment rule $\pi(s) \in \{0, 1\}$, target mixture q, and cost $C(h, \pi)$, incremental policy value is

$$\Delta V_q(h, \pi) = \sum_s q_s \pi(s) \tau_s(h) - \lambda C(h, \pi). \tag{17}$$

With gain $g > 0$ in one stratum and harm $-b < 0$ in the other, global value is $pg - (1-p)b$ and changes sign at $p = b/(g+b)$. A robust global rule can require $\inf_{q \in \mathcal{Q}} q^\top \tau(h) - \lambda C(h) > 0$ over a prespecified set of plausible mixtures. Otherwise the routed edit–policy pair must be evaluated as a new intervention.

A.5 Inference conventions

Tasks are the inference unit in M1 and R1; randomized model–world–functional pairs are the unit in S1; independent optimizer campaigns are the unit in H1 and H4a. H1 campaign rates use Wilson intervals (Wilson, 1927). H4a uses paired campaign bootstrap intervals, and R1 uses task-bootstrap intervals with 5,000 percentile draws (Efron & Tibshirani, 1993). The latter condition on the already selected gate and do not propagate gate-fitting or authorization uncertainty. H1 and H4a intervals quantify Monte Carlo variation across independent campaigns under a fixed data-generating process and configuration; keyed pseudorandom draws instantiate those campaigns. They are not robustness intervals over generators, seed choices, evidence allocations, models, repositories, or deployment populations.

B Detailed related-work positioning

Harness evolution is already an established optimization paradigm. DSPy compiles declarative LM programs into prompts and demonstrations by optimizing a user-defined metric over examples (Khattab et al., 2024). It is optimizer context for proposing or tuning a pipeline, not a causal guarantee that a selected change should be deployed to every task; TCCA addresses that downstream admission question. Self-Harness converts failures into scoped edits behind a regression gate; Recursive Harness Self-Improvement revises a prompt-represented multi-agent loop; Evo-Harness compiles failed execution context into reusable skills; and Meta-Harness searches complete harness programs (Zhang et al., 2026a; Lee et al., 2026a; Wei et al., 2026; Lee et al., 2026b). DREvo reweights historical evidence as code changes, and EvolveNet composes program deltas while providing disjoint commit gates, rollback, domain-scoped execution, paired tests, and a budget-aware baseline (Guo et al., 2026; Nie et al., 2026a). Adaptive Auto-Harness selects specialized branches before execution, Living-Harness retrieves task-family state, and TTHE uses separate proposer roles and an agentic judge (Liu et al., 2026; Du et al., 2026; Nie et al., 2026b). Evo-Bench studies long-horizon evolution with disjoint post-freeze evaluation and sandboxed integrity controls (Huang et al., 2026). These systems rule out priority claims for automatic evolution, held-out gates, routing, lineage, rollback, or evaluator separation.

Evaluation work provides further close comparisons. Rethinking the Evaluation of Harness Evolution matches rollout opportunities and tests held-out transfer (Wang et al., 2026a). Task-CoEvolve corrects unequal validation sampling, HarnessCompass restricts evolution to task-agnostic harness changes and uses a large final holdout, and When Self-Evolution Backfires uses cold-state skills, held-out replay, semantic checks, and joint subset selection (Miyai et al., 2026; Zhang et al., 2026b; Shang et al., 2026). Agentic Harness Engineering uses versioned change manifests, protected runner and verifier files, lineage, and rollback in a live campaign (Lin et al., 2026). Our claim is not the first held-out evaluation, matched budget, gate, route, or rollback. It is the joint treatment of adaptive selection and functional deployment scope, with the edit–policy pair as the admission object.

The statistical framing draws on adaptive data analysis and reusable holdouts (Dwork et al., 2015a;b), randomized causal inference (Imbens & Rubin, 2015), heterogeneous-effect estimation and policy learning (Athey & Imbens, 2016; Wager & Athey, 2018; Kitagawa & Tetenov, 2018), and multiplicity control (Holm, 1979). Reward hacking and Goodhart failures motivate the integrity boundary (Amodei et al., 2016; Manheim & Garrabrant, 2019). Agent benchmarks and interfaces establish that context, tools, and control flow can change measured performance (Jimenez et al., 2024; Yang et al., 2024; Wang et al., 2025). Code as Agent Harness independently frames governed mutation through change contracts, evidence bundles, verification, canaries, and rollback (Ning et al., 2026). We claim a concrete synthesis and controlled component tests, not independent novelty for each ingredient.

The closest empirical boundaries further restrict this claim. Large-scale budget-matched evaluation already finds no universally superior fixed harness and proposes adaptive allocation (Gupta et al., 2026); AgentLens already shows that process assessment can change rankings relative to pass rate

(Sahoo et al., 2026); and WARP directly demonstrates population distortion from relevance-ranked retrieval (Thakur et al., 2026). Same Agent, Different Answers removes same-snapshot instability from cross-snapshot churn (Ning, 2026). Related audits already use matched sham or corrupted observations, no/self/always-tool arms, utilization stages, learned gates, and task- or model-dependent context effects (Wang et al., 2026b; Luo & Peng, 2026; Xia, 2026; Liu, 2026; Wu et al., 2026; Han et al., 2026; Fan et al., 2026). Our narrower contribution is the edit–policy admission object and randomized order-only same-world functional twins, not any of those ingredients alone.

C Intervention contracts and admission procedure

C.1 Contract fields and causal atomicity

Each proposed mutation is compiled into a machine-readable contract containing its parent harness, causal hypothesis, eligible state, intervention point, allowed files, predicted mediator and endpoint, uptake prediction, failure risks, cost, matched control, target functional, allowed deployment scope, protected functionals, target-mixture assumptions, evaluation plan, stopping rule, and rollback plan. Atomicity denotes one causal edge rather than one source line. Treatment and control must expose identical model-visible state until the declared intervention point. Changes that add calls, turns, tokens, tools, or computation require a resource-matched control. Context-selection changes additionally require a control that preserves evidence membership, metadata, and budget while removing the proposed ordering or semantic signal when feasible.

The evolvable plane contains prompts, tool descriptions and implementations, context and memory policies, state transitions, recovery, and stopping rules. The logically trusted plane controls splits, randomization, pristine evaluation, estimators, thresholds, resource accounting, provenance, holdout credentials, and rollback. The implementation validates contracts and protected paths, but does not establish live operating-system capability separation.

C.2 Reference procedure

The intended procedure is:

1. validate the contract, provenance, file boundary, evaluator, pass@1 protocol, denominator, joins, and resource ledger;
2. verify eligible opportunities and separately record availability, invocation, payload delivery, inspection, and adherence;
3. conduct randomized discovery comparisons;
4. calibrate observed separation against same-arm A/A variation;
5. estimate signed uplift, influence, reliability, safety, and cost at the task level;
6. estimate functional effects, target-mixture value, and protected harms;
7. apply the ordered state classifier below;
8. freeze a small finalist family and run multiplicity-controlled fresh confirmation; and
9. admit at most one edit–policy pair for sealed evaluation.

The reference state classifier is lexicographic; the first satisfied row wins. Its numerical thresholds are reference-configuration choices, not universal constants. In Table 3, $\widehat{\tau}$ is the candidate-versus-baseline uplift and $\widehat{\tau}_{\text{control}}$ is the resource-matched-control-versus-baseline uplift estimated on the same analysis units.

There is not yet one executable that implements all nine stages. H1, G1, and R1 exercise separate reference mechanisms. H4a combines simplified analogues. Its A/A-derived discovery-screen threshold is $c = \max\{0.015, 0.25Q_{.95}(|A/A|)\}$, where $Q_{.95}(|A/A|)$ is the empirical 95th percentile of the absolute uplifts from 20 paid, one-replicate A/A pairs. A candidate advances as a provisional global candidate when its screen lower bound exceeds c or, after an `INCONCLUSIVE` classification, its point estimate exceeds c; moderation candidates follow a separate routing rule. H4a sets opportunity to one and safety and reliability failures to zero, confirms at most two finalists on 150 fresh tasks

Table 3: Ordered reference admission states.

State	First satisfied condition
INVALID	Contract or evidence-validity check fails.
QUARANTINE	Evaluator manipulation or another declared integrity attack is suspected.
HARMFUL	A safety check fails, reliability regresses by more than 0.02, or the uplift interval is wholly negative.
NO_OPPORTUNITY	Eligible opportunity rate is below 0.20.
DORMANT	Uptake is below 0.20 conditional on opportunity.
UNECONOMIC	Cost ratio exceeds 1.25.
CONFOUNDED	A matched control exists and $\widehat{\tau} - \widehat{\tau}_{\text{control}} \leq 0.01$.
GLOBAL_CANDIDATE	Uplift lower bound is positive and every protected-functional guard passes.
SCOPE_CONFLICT	A protected functional has a material negative bound.
ROUTED_CANDIDATE	Influence lower bound exceeds 0.02, without a protected-functional conflict.
INERT	Influence upper bound is at most 0.02.
INCONCLUSIVE	No earlier state is supported.

with Bonferroni $\alpha = 0.025$, and requires a one-sided lower bound above 0.01. Its development-fitted gate does not use R1's separate fit/authorization split. Sealed live promotion, canaries, and rollback remain prospective.

D M1: FIXED-COMPONENT MEASUREMENT AUDIT

M1 is an exploratory retrospective analysis of a frozen production `file_map` component on 31 software-engineering tasks. It compares treatment with a matched control at the task level and records opportunity, invocation, outcomes, and replicate structure. The component changed six verdicts: three failures became passes and three passes became failures. Mean uplift was therefore zero, while replicate-corrected influence was 0.194 with a 95% task-bootstrap interval of $[0.065, 0.355]$.

Observed self-selection did not reliably separate benefit. The component fired on 10/14 budget-bound opportunities and 28/48 other opportunities (Fisher $p = 0.54$); descriptive conditional effects were $+0.400$ and -0.071. A post-hoc budget-exhaustion rule improved the point estimate but failed to transport to another component, so it is not a validated route.

Two prospective component checks reject opposite shortcuts. In a frontier-model manipulation check, the schema was visible in all nine initial eligible episodes but was invoked zero times; the planned study stopped after 22 of 240 trajectories because treatment was dormant. In a separate prospectively specified 30-task, four-replicate history-elision study, uptake reached 87%, yet observed uplift was -0.058, with one task helped and six harmed. Influence was 0.018 $[-0.003, 0.040]$, and treatment instability was 0.233 versus 0.067. The negative effect was not statistically established. The results distinguish dormancy, high uptake, null average, and heterogeneous influence; they do not demonstrate routing efficacy.

E S1: FUNCTIONAL TWINS AND ROBUSTNESS ANALYSES

E.1 FROZEN PROTOCOL

S1 constructs 48 finite worlds with 24 records each, balanced over four renderer surfaces and two rank/value-sign strata. Each world yields two questions over the same records: retrieve one named value or estimate the population mean of the same numeric field. A query-independent salience order is frozen before the named lookup target is placed at rank one. Treatment delivers the salience order; control delivers one unconditional complete permutation of the identical records. Treatment and control orders are shared across the functional twins.

The prospectively specified confirmation crosses the twins with Muse Spark 1.1, Muse Spark 1.2, and Claude Opus 4.8 under forced delivery, sharded access, one attempt, and a common bounded numeric-response contract. The service endpoints used server-managed decoding defaults; temperature and other provider-side sampling parameters were neither exposed nor pinned, which limits exact external reproduction. The study contains 288 randomized pairs and 576 intention-to-treat trajectories. For

functional family f, let $T_f^{\text{obs}} = n_f^{-1} \sum_j (Y_{Tj} - Y_{Cj})$ over $n_f = 144$ pairs and define $\theta_f = \mathbb{E}_Z[T_f^{\text{obs}} \mid \mathcal{F}]$, where $\mathcal{F}$ fixes the complete finite potential-outcome panel. Independent hidden assignment bits Z place real and permutation orders into the two within-pair execution slots. With no cross-pair interference, the union null is $\theta_{\text{lookup}} \leq 0$ or $\theta_{\text{mean}} \geq 0$. Its conservative randomization-assignment bounds, evaluated at the observed statistics, are

$$\begin{aligned} p_L &= \exp\{-144 \max(T_L^{\text{obs}}, 0)^2/2\}, \quad p_M = \exp\{-144 \max(-T_M^{\text{obs}}, 0)^2/2\}, \\ p_{\text{IUT}} &= \max(p_L, p_M). \end{aligned} \tag{18}$$

They are conditional finite-panel bounds, not unconditional p-values over model or task populations. Endpoint-specific contrasts are Holm-adjusted secondary analyses and do not identify an effect of abstract model intelligence.

Protocol and verifier history. The executed V2 protocol retained the exact sealed but unconsumed V1 task draw, seeds, and 288 assignment bits after a predecessor run was externally terminated at 1,508 of 2,448 trajectories. No identical-order call had been launched under V1, and the V2 authorization was outcome-blind to the predecessor's scores and answers. The original frozen verifier then emitted an immutable invalid receipt: at three sites it used `Counter(mapping)` where exact-one checks required `Counter(mapping.keys())`, so every nonempty valid run would fail. V2.1 hash-pinned the original source and made only those three in-memory substitutions. The raw run, invalid receipt, and original analysis code were preserved; the corrected verifier passed, and an independent analysis reimplementation reproduced the results. The hash record is not externally timestamped, so we describe the design as prospectively specified rather than publicly registered.

Table 4: S1 finite-panel intention-to-treat results.

Functional	Salience	Permutation	Effect	Bound	Pairs
Named-record lookup	0.9653	0.2569	$+0.7083$	2.05×10^{-16}	144
Population mean	0.0833	0.7500	-0.6667	1.27×10^{-14}	144

The intersection–union bound is $p \leq 1.27 \times 10^{-14}$. Continuous outcomes agree: salience reduced lookup absolute error by 38.23 units and increased mean absolute error by 6.70 units. All three endpoints had the prespecified directions and passed the original Holm-corrected conjunction. For lookup prevalence p, observed binary global-deployment value is

$$\Delta(p) = 0.7083p - 0.6667(1 - p) = 1.375p - 0.6667, \tag{19}$$

whose point-estimate sign boundary, using the underlying fractions, is exactly $p = 16/33 \approx 0.4848$. A 50/50 mixture yields only $+0.0208$, nearly canceling the large functional effects.

E.2 Failures and post-hoc adversarial recoding

The primary estimate includes 18 HTTP-500 errors and 37 total non-submissions. It is the effect of deploying the ordering policy, including transport risk, not a pure semantic-payload effect. The retained records identify only generic `ModelAPIError` HTTP 500s, not a provider-side root cause, so the cause is unobserved. Errors occurred after 6–8 completed responses; lookup controls also used more requests and tokens than lookup treatments. The arm imbalance is therefore compatible with greater transport exposure, but does not show that permutation order caused server failures. Adversarially scoring every non-submission against the hypothesized direction was a post-hoc sensitivity analysis; it still gives lookup $+0.5139$, mean -0.6458, and a pooled bound of 5.53×10^{-9}. Under this recoding, two of three endpoint-wise Holm conjunctions do not reject at 0.05 (adjusted bound 0.0684); the failure-free endpoint retains lookup $+0.8125$, mean -0.5417, and a rejecting conjunction.

E.3 Voluntary-use pilot

A preceding 1,344-trajectory pilot compared no component, voluntary pull, automatic push, and an operation-aware push gate over 16 worlds. The gate withheld automatic delivery on mean tasks but left the component callable. It was voluntarily invoked in 95/96 gated mean trajectories. In

one endpoint's sharded mean episodes, the component was called in 16/16 real-payload and 16/16 permutation-payload runs, while accuracy was 0.25 under salience and 0.75 under permutation. The mean contrast under the gate remained -0.5417. This is supporting mechanism evidence: the pilot used different response and evaluation contracts across functionals and does not identify a caller-specific content effect. Such an effect requires randomizing real versus matched sham content only after a committed call.

S1 is a constructed synthetic navigation-map analogue. The scoring rule does not mechanically assign mean-estimation harm: both arms contain the same 24 records, so its direction and magnitude are observed model behavior. The study establishes a causal finite-panel existence result under the frozen generator, not natural-domain prevalence, population transport, or harm relative to offering no map.

F H1: ADAPTIVE FALSE-PROMOTION CONTROL

H1 simulates 100 sharp-null and 100 planted-positive campaigns for each of four strategies. Each campaign has a 4,800-trajectory search cap and a common locked panel excluded from that budget. Baseline probabilities follow $\mathrm{Beta}(4, 4)$. The design uses 24 discovery tasks, 32 confirmation tasks, and a 64-task locked panel with four replicates. Each 100-candidate positive campaign places effects $+0.30$ at stream positions 2 and 11.

Full admission uses 20 A/A calibration pairs; discovery looks after one, two, and three replicates with alpha weights $0.1/0.3/0.6$; and threshold $\max\{0.04, Q_{.95}(|A/A|)\}$. Fresh confirmation uses three replicates on 32 tasks, allocates family alpha to candidate i as $0.05/[i(i+1)]$, and applies an exact paired one-sided sign-flip test. The safety criterion requires the upper endpoint of the two-sided 95% Wilson interval for the campaign-level sharp-null false-promotion rate to be below 5%. A separate point-margin utility criterion permits at most 0.05 less latent locked gain than the best simpler strategy; it is not a confidence-interval noninferiority test.

The four strategy arms form a staged diagnostic, not a factorial component ablation. Score-only uses one discovery replicate and no multiplicity correction; repeated-mean uses three discovery replicates but no A/A calibration or fresh confirmation; calibration-only uses A/A-calibrated sequential discovery and consumes no fresh-confirmation outcomes; full uses the same calibrated sequential discovery and adds the fresh panel. Because stopping and reservation change evidence allocation, differences among arms do not identify additive causal contributions of individual stages.

Table 5: H1 campaign-level safety and utility results.

Method	Null campaigns promoted	Positives promoted/reached	Latent locked gain
Full calibration + confirmation	0/100	86/200 (43.0%)	0.2469
Calibration only	1/100	91/200 (45.5%)	0.2605
Repeated mean	100/100	200/200 (100.0%)	0.4697
Score only	100/100	199/200 (99.5%)	0.4679

For full admission, the Wilson upper endpoint is 3.70%, below the recorded 5% target. Calibration without confirmation has an upper endpoint of 5.45% and misses the target. Both naive strategies eventually promote a null in every campaign. Evidence use differs: under the sharp null, full admission inspects 47.2 proposals and consumes 4,536 trajectories per campaign on average, repeated mean inspects 33 and consumes 4,752, and score-only inspects 100 and consumes 4,800. The comparison is budget-capped, not exposure-matched.

The safety result fails the separate utility criterion. Full admission's 0.2469 latent gain is 0.2228 below repeated mean's 0.4697, exceeding the allowed 0.05 point margin. The repeated-mean comparator, however, promotes a null in all 100 sharp-null campaigns. H1 therefore demonstrates false-promotion control and a large utility gap in one frozen reference configuration; because stopping and evidence allocation also differ, the gap cannot be assigned to one safety stage. Planned seed, effect-size, stochasticity, and allocation sweeps were not executed.

G G1: DECLARED INTEGRITY ATTACKS

G1 uses a fixed 12-candidate pool: eight planted attacks, two valid positives, and two benign non-positive controls. The attacks target protected files, hidden artifacts, benchmark identifiers, pass@1 integrity, denominator integrity, resource accounting, the primary endpoint, and result-join identity. Decisions are hash-committed before truth labels are joined, although execution remains in one process rather than a capability-isolated environment.

Table 6: G1 finite-pool integrity results.

Policy	Attacks advanced	Attack identifiers	Valid positives	Precision
Full policy	0/8	—	2/2	100.0%
Without denominator integrity	1/8	`cand_009`	2/2	66.7%
Without hidden-access guard	1/8	`cand_006`	2/2	66.7%
Without ID-specialization guard	1/8	`cand_007`	2/2	66.7%
Without join-identity guard	1/8	`cand_012`	2/2	66.7%
Without pass@1 integrity	1/8	`cand_008`	2/2	66.7%
Without primary-endpoint guard	1/8	`cand_011`	2/2	66.7%
Without protected-path guard	1/8	`cand_005`	2/2	66.7%
Without resource-integrity guard	1/8	`cand_010`	2/2	66.7%
Score only	8/8	`cand_005–cand_012`	2/2	20.0%

The full policy rejects every planted attack and both benign non-positive candidates while advancing both valid positives. Each leave-one-guard-out variant advances the attack constructed for the removed guard. This verifies implementation correspondence under the declared finite threat model. It does not estimate resistance to novel attacks because candidates and guards were designed together, no independent red team authored held-out attack families, and execution lacked syscall or network isolation.

H R1: CONDITIONAL-ROUTING CONTROLS

The predictable and unpredictable generators each contain 600 tasks with four replicates per arm, split 360/120/120 for gate fitting, authorization, and test. One third of tasks receive $+0.24$ and the remainder -0.12, for zero latent population-average uplift. Only the predictable generator links effect sign to an observed pre-treatment feature. Baseline success is $0.50 + 0.12(0.5 - \mathit{difficulty})$. A sparse deterministic generator has 310 tasks with two replicates, split 186/62/62; the highest and lowest deciles of navigation pressure receive effects $+1$ and -1, and other tasks are unchanged.

A one-level decision stump is fit from allowlisted pre-treatment features, authorized on validation data, frozen, and compared once on test with always-off, always-on, and a simulator oracle. The implemented authorization rule requires only a positive validation point estimate. It does not require a confidence bound, a minimum worthwhile gain, or a protected-functional guard.

The retrospective row reuses all 31 previously inspected M1 `file_map` tasks having at least two complete paired control/treatment seed tracks. A deterministic hash split assigned 18/6/7 tasks to fitting, authorization, and test; seven is the complete internal test partition, not a subset selected for having an observed effect. Only public pre-treatment features entered the stump. Nevertheless, the source outcomes had been unblinded and moderator families explored before this analysis, so the internal test split is not prospective or operationally blind.

Table 7: R1 routing results. Intervals condition on the frozen selected gate.

Scenario	Test tasks	Routed gain versus off	Latent oracle recovered
Predictable synthetic	120	$+0.0688$ $[0.0312, 0.1083]$	98.8%
Unpredictable synthetic	120	-0.0333 $[-0.0854, 0.0167]$	10.0%
Sparse deterministic	62	$+0.1613$ $[0.0806, 0.2581]$	100%
Previously inspected retrospective	7	Gate unauthorized; always-off	—

In the predictable case, full-panel estimated uplift is $+0.0067$ and influence is 0.0244. The frozen gate treats 36.7% of test tasks, has latent gain 0.085 versus oracle 0.086, and gains $+0.0854$ $[0.0312, 0.1375]$ over always-on. The unpredictable case has estimated uplift $+0.0058$ and larger influence 0.0310, but no valid moderator. Its gate was authorized by a $+0.0167$ validation estimate despite latent validation gain -0.005, then yielded a test estimate of -0.0333. Its 95% interval, $[-0.0854, 0.0167]$, spans zero, so the run does not establish harmful policy value. It nevertheless exposes a failure mode: influence detects heterogeneity but cannot produce a predictor of its sign, and a noisy positive validation estimate can authorize a route with no established held-out benefit.

The sparse route recovers the full latent oracle value, but its lower bound against always-on is $+0.0161$, below the declared $+0.02$ superiority margin. The seven-task retrospective route was not authorized and therefore never treats; its zero-width interval is tautological. R1 establishes the routing mechanism in the predictable synthetic case, but none of R1 is prospective natural-task routing evidence.

I H4a: INTEGRATED CLOSED-POOL ADMISSION

Each of 400 paired campaigns contains 24 candidates: 12 named special cases and 12 nulls. The special cases include global effects $+0.085$ and $+0.040$, visible and hidden heterogeneity, a -0.060 harm, development overfitting ($+0.13$ on development and -0.030 elsewhere), a $+0.105$ candidate with cost ratio 1.60, a $+0.060$ candidate with a $+0.055$ resource control, dormancy, invalidity, and zero-effect evaluator and denominator attacks carrying visible bonuses $+0.25$ and $+0.18$.

All methods share a deterministic keyed outcome cube and a 6,000-cell search cap. The simulator creates 100 development, 150 confirmation, and 240 pristine holdout tasks, plus a 25-task development subset. A cell is one task–arm–replicate observation. Full screening uses two replicates, full confirmation four, and A/A calibration 20 one-replicate pairs; full confirms at most two finalists. Holdout estimates use four replicates. The primary estimand is mean observed pristine-holdout uplift and the recorded primary contrast is full minus score-only. Simulator truth appears only in diagnostics. The 6,000-cell cap equalizes maximum search opportunity, not realized search cost. Because no budget sweep or shadow price was prespecified, H4a identifies neither a cost–value frontier nor marginal value per cell; resource differences are descriptive. Its efficacy contrast is therefore a common-cap comparison, not evidence of dominance at equal realized cost. The cost-ratio screen is a feasibility constraint, not a substitute for a frontier.

Trusted score only evaluates every candidate on the 100-task development panel with one replicate per arm, excludes visible-score attack bonuses, and globally deploys the largest positive development mean; it has no contract audit, replication, calibration, routing, or fresh confirmation. *Random audit-valid* spends that same 4,800-cell evaluation budget and then uniformly selects a candidate whose contract validates, recorded audit findings are empty, and metered cost ratio is at most 1.25; it does not select on efficacy. An observed material root regression is a campaign whose observed pristine-holdout uplift is strictly below -0.02; "Root regress." reports the fraction of campaigns meeting that threshold. "Holdout repl." is the fraction whose holdout 95% lower bound exceeds 0.01, and "Repl.|promote" conditions that event on promotion.

Full minus score-only is $+0.0149$ $[0.0104, 0.0198]$. Full selects 0/400 attacks versus 350/400 for score-only, and 6/400 nonbeneficial or disallowed candidates versus 394/400. Full has one observed material root regression, versus 77/400 for score-only. These are descriptive simulator diagnostics, not population rates.

Stronger comparisons constrain the claim. Trusted score-only has mean uplift $+0.0217$; full minus trusted score-only is -0.0024 $[-0.0095, 0.0045]$, which establishes neither a mean difference nor equivalence or noninferiority. At the prespecified observed-regression threshold, full has 1/400 material root regressions versus 124/400 for trusted score-only (0.3% versus 31.0%). This is a secondary descriptive contrast within the frozen simulator, not a population tail-risk estimate. Removing calibration changes uplift from $+0.0193$ to $+0.0194$; the paired difference is -0.0001 $[-0.0008, 0.0007]$. Thus H4a gives no positive evidence that calibration adds average value once other gates and confirmation are present. Removing confirmation lowers uplift to $+0.0067$ and raises nonbeneficial or disallowed selection to 75.8%; this ablation is exploratory and unadjusted.

Table 8: H4a results across 400 paired campaigns. Panel A reports observed outcomes; Panel B labels simulator-truth diagnostics explicitly.

Panel A: observed performance and decisions

Method	Holdout uplift, 95% CI	Promote	Holdout repl.	Repl.|promote
Frozen	0.0000 [0.0000, 0.0000]	0.0%	0.0%	—
Random audit-valid	+0.0071 [0.0034, 0.0109]	100.0%	12.0%	12.0%
Score only	+0.0044 [0.0013, 0.0076]	100.0%	5.5%	5.5%
Repeated mean	+0.0037 [0.0006, 0.0071]	100.0%	6.0%	6.0%
Calibration only	+0.0032 [0.0003, 0.0062]	94.8%	5.3%	5.5%
Trusted score only	+0.0217 [0.0159, 0.0275]	100.0%	31.5%	31.5%
Full without calibration	+0.0194 [0.0161, 0.0229]	26.0%	20.8%	79.8%
Full without confirmation	+0.0067 [0.0023, 0.0113]	100.0%	18.0%	18.0%
Full admission	+0.0193 [0.0160, 0.0232]	25.8%	20.5%	79.6%

Panel B: observed downside and latent diagnostics

Method	Observed root regression	Latent attack	Latent nonbeneficial/disallowed
Frozen	0.0%	0.0%	0.0%
Random audit-valid	21.8%	6.8%	89.3%
Score only	19.3%	87.5%	98.5%
Repeated mean	20.8%	87.3%	98.5%
Calibration only	19.8%	84.3%	93.5%
Trusted score only	31.0%	3.3%	84.5%
Full without calibration	0.3%	0.0%	1.5%
Full without confirmation	32.5%	3.5%	75.8%
Full admission	0.3%	0.0%	1.5%

Full uses 5,585 mean search cells versus 4,800 for score-only, within the common cap, so the comparison is not equal-realized-cost. The primary contrast also bundles evaluator trust, guards, replication, and confirmation. Full selected no routed policy in this pool; integrated routing evidence therefore comes from R1 rather than H4a. H4a characterizes one frozen closed-pool controller; it neither proves live campaign value nor isolates an incremental calibration effect.

J Prospective H4b live evaluation—not executed

Status. H4b is a prospective protocol only. It was not executed for the evidence reported in this paper, contributes no outcome to any table, and supports no claim of live harness-evolution gains, natural-domain prevalence, population benefit, or state-of-the-art performance.

The campaign-level study would connect a real trace analyst and proposer to the admission controller, run multiple independent propose–evaluate–admit cycles, preserve immutable champion lineage, compare every finalist against both its parent and the frozen root, and exercise canary and rollback behavior. The optimizer campaign, not a trajectory, is the primary unit. Under prespecified budgets, the comparison would include: (i) frozen root; (ii) score-only global promotion; (iii) selection-aware global promotion with fresh same-functional confirmation; (iv) causal global promotion with protected-functional guards; and (v) causal scope-aware routing.

Before any outcomes, the protocol must freeze the source development mixture, fresh confirmation panel, same-world functional audit, target mixture, utility weights, protected-functional margins, failure policy, resource ledger, and analysis code. A navigation edit selected on retrieval-heavy tasks should be tested without semantic redesign on natural functional twins such as local lookup/localization/repair versus global count/rate/mean/audit. Repositories, websites, filings, registries, and logs are possible substrates rather than prescribed domains; WebArena and AgentBench are candidate environments only where such twins and a target mixture can be defined (Zhou et al., 2024; Liu et al., 2024b). A separate dynamic-regime study would have to freeze time-indexed feature updates, named target-mixture shifts, sequential assignment, and a corresponding dynamic policy-value estimand. None of these is identified by the static pre-treatment rules studied here.

For voluntary use, the strongest design records the first committed tool call and only then randomizes real versus membership-, metadata-, and token-matched sham content, locking assignment for the trajectory. Availability-level intention-to-treat remains primary. A confirmatory scope result would require the prespecified conjunction of positive fresh source-functional gain, a negative complementary-functional effect, negative global value on the frozen target mixture, superior value for the frozen route, and no material regression on the protected worst functional. Final edit–policy pairs would then be evaluated once on independently custodied holdout tasks.

K Reproducibility and artifact map

The anonymous supplement is a deny-by-default numerical audit bundle, not a complete raw-data release. It contains curated machine-readable summaries, the S1 confirmatory protocol, dependency-free verification programs, and a hashed manifest. The programs check manuscript-facing design, provenance, and result fields and recompute selected formulaic relationships from the summaries. The retained evidence freeze additionally links each summary to a frozen configuration, declared seed or assignment, raw or normalized outcomes, and an analysis record, but those external-model transcripts and non-redistributable inputs are not included. The archive also excludes credentials, private infrastructure, and identity-bearing metadata. This prevents independent qualitative inspection of complete trajectories; a future release should include a sanitized sample where privacy, licensing, and security constraints permit. Table 9 maps each study to its retained evidence and verification boundary.

Table 9: Retained evidence map. The middle column describes the full internal evidence class outside the anonymous review ZIP; the ZIP contains only the numerical-audit materials listed above.

Study	Retained evidence class	Verification boundary
M1	Frozen task-level outcomes, opportunity and uptake records, replicate-aware analysis	Exploratory panel and prospectively specified component checks remain distinguishable
S1	Prospectively frozen protocol, world generator, frozen orders, randomized-pair ledger, raw responses, corrected verifier, preserved analyzer, independent reimplementation	Finite-panel estimand, post-hoc failure recoding, and endpoint-wise multiplicity checks
H1	Campaign generator, candidate stream, A/A calibration, sequential ledger, fresh-confirmation and locked-panel records	Campaign-level false-promotion and utility criteria
G1	Candidate contracts, pre-label decision commitments, truth labels, guard-ablation matrix	Finite declared threat model only
R1	Frozen generators and splits, feature allowlist, fitted gate, authorization record, test outcomes, oracle diagnostics	Test intervals condition on the selected gate
H4a	Candidate pools, keyed outcome cube, method configurations, resource ledger, paired campaign outcomes	Primary contrast separated from exploratory ablations and latent diagnostics
H4b	Protocol and stopping rules only	No execution or outcome is included

The reference offline suite contains 187 declared artifacts and snapshots of 52 source files; it was created after 48 unit and integration tests passed. The later repository-level verification suite passes 93 tests, and a standalone verifier passes 36 checks including deterministic H1 replay and independently recomputed integrity outcomes. These counts establish internal consistency of the frozen artifact, not external timestamping or malicious-actor resistance; these internal-suite materials are not included in the review ZIP. Separately, the review ZIP's numerical-audit verifier passes 77 manuscript-facing checks. The root digest is locally verifiable but is not externally signed; an actor able to rewrite the entire archive could also rewrite its digest. A future live freeze should add external timestamping or public preregistration before outcome access.

Full internal reproduction, outside the anonymous numerical-audit bundle, proceeds in three layers: (1) verify hashes and manifests without executing analyses; (2) regenerate machine-readable summaries, tables, and vector figures in an isolated temporary directory and require byte identity where declared deterministic; and (3) rerun stochastic studies from their frozen configurations, comparing estimands

and tolerances rather than requiring byte identity from external model services. Reviewers can verify the ZIP manifest and run the reported-value consistency checks. Any new natural-task experiment must be reported separately rather than silently updating the studies above.