# BEYOND TRUST: PROOF-CARRYING GENERATION AND INDEPENDENT RELEASE ASSURANCE FOR MULTI-AGENT LLMS

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Multi-agent LLM systems act across trust boundaries, yet downstream consumers cannot independently verify *why* an output was released or *how often* released outputs fail. We introduce **PCG-MAS**, a proof-carrying release-assurance framework with two certificates. A *DecisionCertificate* binds each release to a consumer-derived acceptance scope, exact execution and policy obligations, semantic checks, and replay metadata, so a producer that curates what the checker sees cannot force acceptance. An *EpochRiskCertificate* binds the complete committed release stream to an independent risk-limiting audit. Across seven models and eight factual and agentic datasets, we test controlled-coverage acceptance, exhaustive curation attacks, log-bypass attacks, weighted auditing, replay, and cost. Relative to frozen primary comparators, the worst-case released-failure difference under selective-view curation is $-3.1$ pp and the natural released-failure difference at matched coverage is $-1.3$ pp; the governed population records 0 unlogged consequential releases, and the pooled epoch audit reaches CERTIFY with a 36.7% label saving over uniform auditing.

## 1 INTRODUCTION

LLM agents increasingly combine reasoning with retrieval and external actions: ReAct couples reasoning with environment interaction (Yao et al., 2023), Toolformer and MRKL connect language models to tools and modular services (Schick et al., 2023; Karpas et al., 2022), and AutoGen supports coordinated multi-agent interaction and delegation (Wu et al., 2024). As these systems move from answering questions to taking consequential actions, a deployer must decide whether an output or action may cross a release boundary, and later be able to justify that decision to an independent consumer, auditor, or risk function.

Existing components address parts of this problem. Citation and attribution methods measure source support (Gao et al., 2023; Xu et al., 2025); semantic verifiers such as MiniCheck, AlignScore, and QAFactEval characterize grounding (Tang et al., 2024; Zha et al., 2023; Fabbri et al., 2022); agent guardrails enforce action policies (Chen et al., 2025; Xiang et al., 2025; Liu et al., 2026); record/replay systems reconstruct execution (Feng et al., 2025); and AI-control protocols study oversight of models that may actively subvert it (Greenblatt et al., 2024). Two assurance gaps remain. First, a producer can influence the *view* on which it is judged by selecting favorable evidence, trace events, or tool observations. Second, even a correct per-decision checker does not by itself support a statistically valid statement about the population of decisions that was actually released.

We introduce **PCG-MAS**, an independent release-assurance architecture (Fig. 1). At the decision level, the consumer—not the producer—defines the maximum admissible acceptance scope. A producer witness may supply a compact, portable explanation, but hiding an admissible refutation cannot turn an anchored rejection into a release. At the population level, every governed release belongs to a committed epoch that is eligible for independent audit. The audit is risk-limiting: under its stated assumptions it bounds the probability of certifying a finite population whose realized failure fraction exceeds a registered threshold. The audit does not improve the model or the checker; it certifies, or refuses to certify, the performance that the per-decision layer actually achieved.



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.

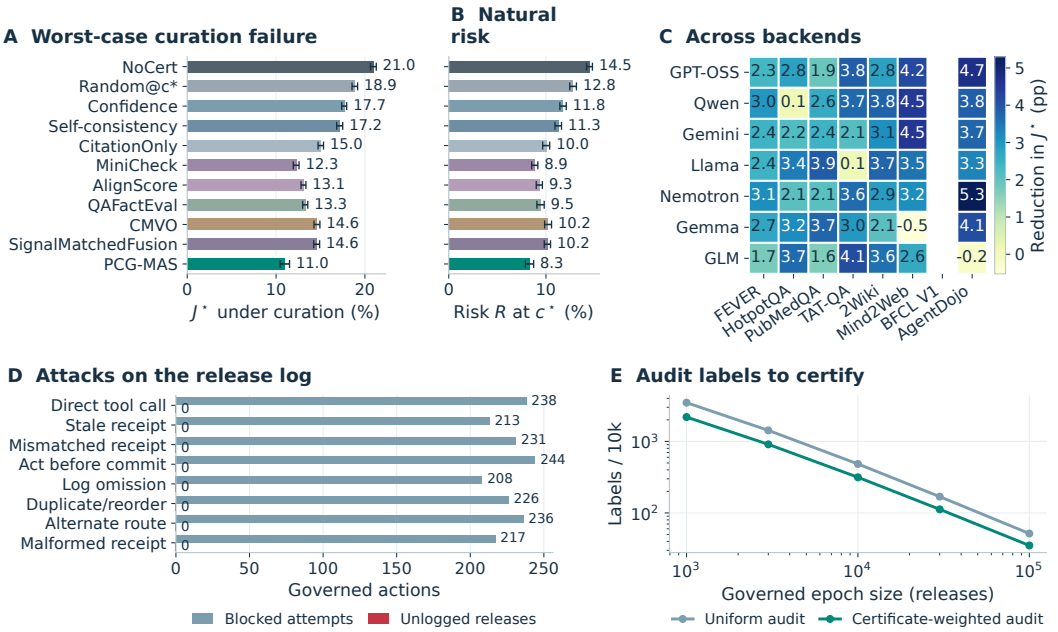



Figure 1: DERIVED. **Headline results** (macro-averaged over the test split; error bars are 95% model-cluster bootstrap intervals). **A**: worst-case released failure $J^\star$ under the registered selective-view curation affordance; lower is better. **B**: natural released-failure risk $R$ at the controlled coverage $c^\star$. **C**: **C**: reduction in $J^\star$ relative to the task family's primary comparator in every model–dataset cell; positive favors PCG-MAS; BFCL is blank because it exposes no selective view ($H_R$ N/A). **D**: attacks on the governed release log, split into blocked attempts and unlogged consequential releases. **E**: independent audit labels per 10,000 releases needed to certify a governed epoch under uniform and certificate-weighted sampling. N/A is never zero-filled.

The guarantee is deliberately bounded. PCG-MAS does not prove world truth, universal safety, or perfect outcome labels. Grounding and declared policy/security remain separate obligation families, and missing evidence, checker inapplicability, replay uncertainty, consumer-scope incompleteness, and immature outcome labels stay explicit instead of being converted into passes.

**Evaluation scope.** We evaluate seven model families—`gpt-oss-120b`, `qwen3.8-27b`, `gemini-3.1-flash-lite`, `llama-4-scout-17b-16e-instruct`, `nemotron-3-120b-a12b`, `gemma-4-26b-a4b-it`, and `glm-4.7-flash`—on eight datasets spanning factual reasoning, web interaction, function calling, and adversarial agents: FEVER (Thorne et al., 2018), HotpotQA (Yang et al., 2018), PubMedQA (Jin et al., 2019), TAT-QA (Zhu et al., 2021), 2WikiMultiHopQA (Ho et al., 2020), Mind2Web (Deng et al., 2023), BFCL V1 (Patil et al., 2025), and AgentDojo (Debenedetti et al., 2024), a complete $7 \times 8 = 56$-cell panel. Comparisons include NoCert, random acceptance at matched coverage, confidence and self-consistency selective prediction, CitationOnly, MiniCheck, AlignScore, QAFactEval, the structure-matched controls CMVO and SignalMatchedFusion, and the native-scope specialists ShieldAgent (Chen et al., 2025), GuardAgent (Xiang et al., 2025), AgentDoG (Liu et al., 2026), and AgentRR (Feng et al., 2025).

**Primary contributions.**

**(1) Curation-robust per-decision acceptance.** We show that any release rule judging only a producer-chosen view is either blind to refutation or exploitable by omission (Proposition 4.1), and give a consumer-anchored DecisionCertificate that removes the producer's control over the maximum view (Fig. 2; Algorithm 1).

**(2) Complete-population, label-efficient risk-limiting assurance.** Released decisions enter a committed, receipt-enforced epoch whose realized failure fraction is certified by a finite-population audit (Theorem 4.1); certificate signals direct the audit to cut label cost without weakening its validity.

**(3) A falsifiable evaluation of both claims.** Each load-bearing assumption is paired with an experiment that breaks it (Table 2; Fig. 4), and results are reported at controlled coverage against frozen comparators (Section 5).

**Related work.** The main text keeps only the comparison logic needed to interpret the method; a broader treatment of AI control, proof-carrying pipelines, grounding, verifiable retrieval, finite-population auditing, agent guardrails, replay, and authorization appears in Appendix A, and deployment boundaries and misuse in Appendices E.17–E.20.

## 2 Formulation and Preliminaries

We represent one agentic run by a committed runtime graph and separate the decision-level and population-level assurance objects. Full typing, canonicalization, and replay details are in Appendix C; notation is indexed in Appendix F.8.

### 2.1 Runtime and certificate objects

**Definition D1** (Agentic runtime graph)**.** At time $t$, $G_t = (\mathcal{V}_t, E_t, \tau_{\mathcal{V}}, \tau_E, \mathrm{attr}_t)$ is a typed graph whose committed objects include evidence payloads/digests, transformations, tool calls, memory/delegation records, schemas, policies, checker identities, and replay settings. Evidence nodes $\mathcal{V}_t^{\mathrm{src}}$ carry $h(v) = H(x(v))$; "source" denotes provenance, not world truth.

Snapshot replay reconstructs the committed execution; fresh-environment divergence is audited separately instead of changing the recorded decision.

### 2.2 Acceptance contract

**Definition D2** (DecisionCertificate)**.** For input $x$ and committed candidate $y$, a DecisionCertificate is

$$Z = (x, y, \Omega, S^\star, W, \Pi, \Gamma, \mathcal{A}, \chi, d, \mathrm{meta}), \tag{1}$$

where $\Omega$ is the consumer-pinned contract, $S^\star$ the consumer-derived eligible scope, $W$ an optional producer witness, $\Pi$ the pinned replay state, $\Gamma$ the execution/policy contract, $\mathcal{A}$ the applicable-obligation set, $\chi$ the checker identities and configurations, and $d$ the typed release decision.

**Definition D3** (Consumer-derived eligible scope)**.** The admissible verification scope is

$$S^\star(x) = \mathrm{Derive}_\Omega(x), \tag{2}$$

or an equivalent consumer-controlled construction fixed by the declared contract. A producer witness $W$ may be checked against $S^\star$ but does not define the maximum view on which release is decided.

**Running example.** In the AgentDojo banking suite (Debenedetti et al., 2024), a user asks the agent to inspect an incoming payment and return only the excess amount, while untrusted tool content may carry an injected transfer instruction. The producer witness may expose the incoming transaction and selected account facts, but the consumer contract independently derives $S^\star$: the transaction state, intended recipient, policy version, and tool preconditions. Exact checks bind the proposed transfer and runtime state, learned checks assess whether the action is supported by the user request, and the policy/security channel rejects attacker-directed transfers. Appendix C.4 walks this example through every certificate channel, the release-log entry, and the later outcome and audit states.

**Definition D4** (Typed release decision)**.** Required obligation families take values in $\{\textsc{Pass}, \textsc{Fail}, \textsc{NotApplicable}, \textsc{Indeterminate}\}$. Release requires every required grounding, exact-execution, and policy/security obligation to pass. A grounded but policy-disallowed candidate is withheld; a required indeterminate obligation escalates or fails closed according to the frozen contract. No state implies world-truth certification.

### 2.3 Epochs and outcomes

For a governed epoch $\mathcal{E} = \{1, \ldots, N_{\mathcal{E}}\}$ of releases, let $F_i \in \{0, 1\}$ be the independent outcome-authority failure label, and let

$$r_{\mathcal{E}} = \frac{1}{N_{\mathcal{E}}} \sum_{i \in \mathcal{E}} F_i \tag{3}$$

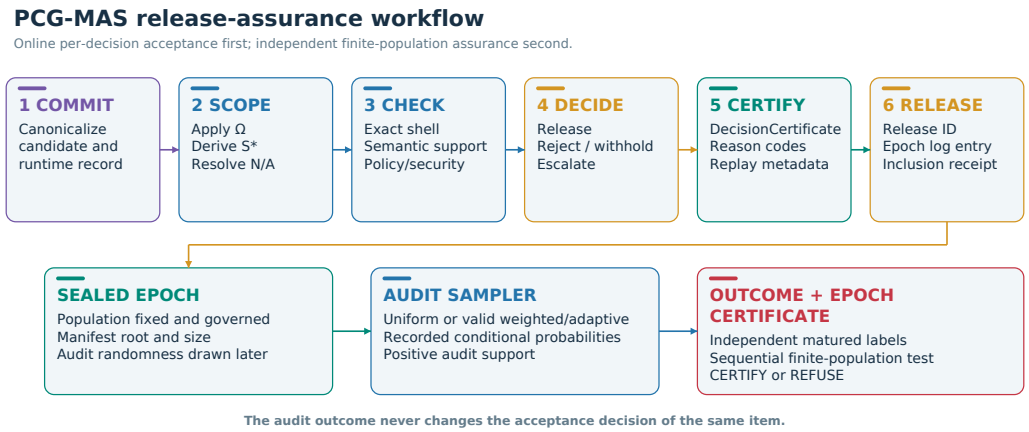



Figure 2: PROTOCOL. **Decision and population assurance workflow.** The consumer canonicalizes and commits the candidate, derives the admissible scope, evaluates exact, learned, and policy obligations, and emits a DecisionCertificate. Governed releases enter a committed epoch. After commitment, the registered audit sampler selects decisions for independent outcome labeling, and the sequential finite-population test emits an EpochRiskCertificate or refuses certification.

denote the realized finite-population failure fraction. Grounding failure and policy/security failure are recorded separately and combined as $F_i^{\text{any}} = F_i^{\text{epistemic}} \vee F_i^{\text{policy}}$ when the endpoint requires both.

**Definition D5** (EpochRiskCertificate)**.** An EpochRiskCertificate binds the committed release population and its manifest root, the registered audit design and risk limit, the sampled release identities and their selection probabilities, the matured independent outcome labels, the stopping state, and the resulting finite-population decision. It certifies a statement about the epoch, not the correctness of every item and not a future epoch.

## 3 PROPOSED METHODOLOGY

PCG-MAS separates *generation* from *consumer-side acceptance* and makes the trust boundary explicit: an untrusted producer proposes a candidate and a compact witness, while the consumer controls the contract, admissible scope, release decision, and audit design.

**[1] Acceptance checks and applicability.** For obligation family $j$, let $T_j(x) \in \{\text{PASS}, \text{FAIL}, \text{INDETERMINATE}\}$ and let $\mathcal{A}(x)$ be the predeclared applicability set. The binary projection is

$$V_j(x) = \mathbf{1}[j \notin \mathcal{A}(x)] + \mathbf{1}[j \in \mathcal{A}(x)]\mathbf{1}[T_j(x) = \text{PASS}], \qquad \text{Check}(Z; G_t) = \prod_{j \in \{H,\Pi,\Gamma,\vdash\}} V_j(x). \tag{4}$$

The typed state is preserved when the projection fails closed: an unavailable required measurement is not relabeled as an observed violation.

**[2] Consumer-derived scope and semantic checking.** The maximum release view is $S^\star(x) = \text{Derive}_\Omega(x)$. A frozen task adapter $h_g(c)$ and a budget-$K$ window operator $\mathcal{W}_K(S^\star)$ feed a pinned NLI checker:

$$\begin{aligned} m(w,h) &= q_{\text{ent}}(w,h) - \max\{q_{\text{con}}(w,h), q_{\text{neu}}(w,h)\}, \\ s_g(c,S^\star) &= \max_{w \in \mathcal{W}_K(S^\star)} m(w, h_g(c)), \qquad c_g(c,S^\star) = \max_{w \in \mathcal{W}_K(S^\star)} q_{\text{con}}(w, h_g(c)). \end{aligned} \tag{5}$$

The complete operator—adapter, scope derivation, window enumeration, aggregation, and thresholds $(\tau_g, \kappa_g)$—is calibrated as one object on the calibration split. A compact portable witness

$$W^\star \in \arg\min_{W' \subseteq S^\star} |W'| \quad \text{s.t.} \quad T_\vdash(c, W') = \text{PASS} \tag{6}$$

is searched under a frozen budget and tie-break. It explains a release but is not a release precondition: an exhausted budget leaves the witness state INDETERMINATE, and $W^\star$ never replaces the anchored contradiction and sufficiency decision over $S^\star$ (search cost and false negatives in Appendix E.7).

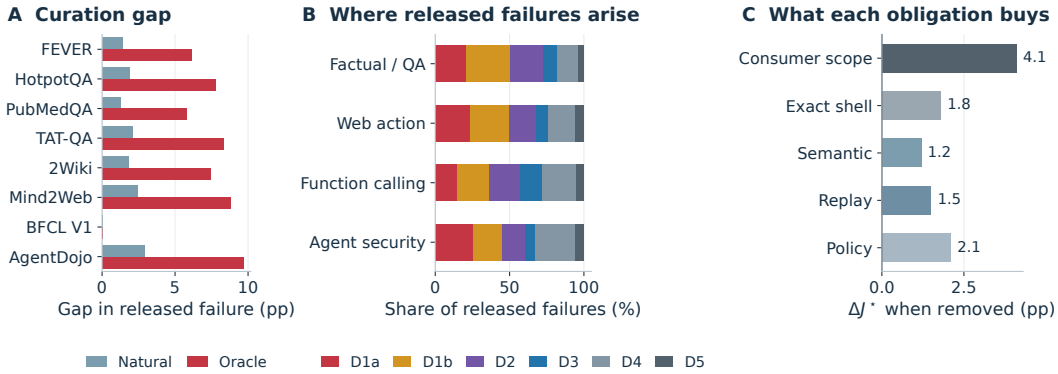



Figure 3: DERIVED. **Curation materiality and mechanism. A**: natural and oracle curation gaps in released failure per dataset, i.e., the additional failure a witness-only rule admits when the producer selects the view; BFCL is N/A because it has no registered curation affordance. **B**: decomposition of released failures into producer curation (D1a), consumer-scope incompleteness (D1b), learned-checker residual (D2), exact-shell failure (D3), applicability failure (D4), and population-integrity failure (D5). **C**: increase in $J^\star$ when each obligation family is removed.

**[3] Reproducible execution, policy, and DecisionCertificate.** Evidence payloads, transformations, tool inputs/outputs, policy decisions, delegation records, scope-derivation configuration, and checker configuration enter a content-addressed execution graph. Tool and evidence records bind stable identifiers, canonical request/response bytes, digests, and receipts where the environment exposes them; hashing a fabricated upstream response does not make it true. Missing or corrupt required children are INDETERMINATE. Untrusted payloads reach the verifier only as inert data (Appendix C.17), and Algorithm 1 fixes the execution order (Appendices C.14 and C.16); integrity and retry semantics are in Appendix E.9.

**[4] Complete release populations and risk-limiting audit.** A population claim covers only releases that belong to the governed epoch (Appendix C.2). Every consequential action must therefore carry a valid log-inclusion receipt before it executes, and bypass attempts are measured directly. After an epoch is sealed, the audit draws release identifiers under a registered without-replacement law, obtains independent outcome labels, and tests

$$H_0 : \; r_{\mathcal{E}} > \tau_{\text{fail}} \tag{7}$$

at risk limit $\delta_{\text{aud}}$. Certificate signals serve as side information: with $\omega_{i,t}$ the conditional probability that release $i$ is drawn at step $t$ given the allowed history, weighted or adaptive sampling is admissible only when the exact law is recorded, the test uses it, and every eligible release keeps positive support, $\omega_{i,t} \geq \omega_{\min} > 0$. Misleading priorities can cost labels but cannot invalidate the test.

**[5] Performance, coverage, and assurance efficiency.** For method $m$, benign coverage and conditional released-failure risk are

$$C_m = \Pr(A_m = 1), \qquad R_m = \Pr(F^{\text{any}} = 1 \mid A_m = 1). \tag{8}$$

Let $\mathcal{P}_{\text{reg}}$ be the registered selective-view affordance and $a$ an admissible curation policy. The adversarial endpoint is

$$J_m(a) = \Pr(A_m = 1 \wedge F^{\text{any}} = 1 \mid a), \qquad J_m^\star = \sup_{a \in \mathcal{P}_{\text{reg}}} J_m(a), \tag{9}$$

a maximum over the finite exhaustive oracle. The joint mass avoids a denominator an adversary could shrink by suppressing correct releases. Natural comparisons are made at a controlled coverage $c^\star$, and audit efficiency is reported as labels to the stopping state and labels and cost per 10,000 releases. The audit cannot rescue a poor $R_m$ or $J_m^\star$. Redundancy, responsibility, and controller diagnostics are secondary (Appendices C.11, D.8, and E.18).

## 4 THEORETICAL ANALYSIS

The analysis separates a *necessity* statement about selective views from the statistical claim that an independent audit controls erroneous population certification. Neither proves world truth, and the statistical primitives are not claimed as new. For DecisionCertificate $Z$ and committed runtime graph

$G_t$, the acceptance projection is

$$\mathrm{Check}(Z; G_t) = V_H V_\Pi V_\Gamma V_\vdash. \tag{10}$$

**Proposition 4.1** (Selective-view dilemma). *Let $V(x, y; W) \in \{0, 1\}$ be any* witness-only *release rule, i.e., one that decides from the evidence view $W$ exposed by the producer, with admissible views in $\mathcal{P}_{\mathrm{reg}}$ ordered by inclusion. $V$ is omission-robust—no permitted omission turns a rejection on an admissible view into an acceptance on one of its subviews—if and only if*

$$W \subseteq W' \implies V(x, y; W) \le V(x, y; W'). \tag{11}$$

*Consequently, an omission-robust witness-only rule is* refutation-blind*: once any admissible subview is accepted, every admissible extension—including one that adds a decisive contradiction—is accepted too. Conversely, any conflict-sensitive rule that accepts a supportive $W$ but rejects an admissible extension $W'$ admits an exploitable selective-view flip on that contested region.*

*Proof sketch.* For a binary rule, "no omission turns a rejection into an acceptance" is exactly monotonicity in the view order. Monotonicity forces acceptance of every admissible extension of an accepted view, including one that adds a refuting item; any rule that rejects such an extension leaves the accepted subview available to the producer. The full proof is in Appendix B.2. □

**Corollary 4.1** (Anchored curation invariance). *Fix $(x, y, \Omega)$, its valid derived scope $S^\star(x)$, and the trusted-base state used by the anchored consumer decision $D^\star(x, y; S^\star)$. For any two producer witnesses $W, W' \subseteq S^\star$ admitted by the witness-interface contract, changing $W$ to $W'$ cannot turn $D^\star(x, y; S^\star) = 0$ into a release.*

*Proof sketch.* $D^\star$ is a function of the candidate, contract, anchored scope, and trusted base only; the witness enters through witness-local conjuncts that can reject but never release. Details are in Appendix B.2. □

**Theorem 4.1** (Risk-limiting epoch certification). *Under Assumptions A2–A4, let $\mathcal{E}$ be a fixed complete finite population committed before the audit draw, and let the registered without-replacement sampling process and sequential test form a level-$\delta_{\mathrm{aud}}$ test of the bad-population null in Eq. (7). If the test uses the actual conditional sampling law and the outcome labels are correct under the registered authority, then*

$$\Pr_{\mathrm{audit}}\left(\mathrm{CERTIFY}(\mathcal{E}) \ \wedge \ r_{\mathcal{E}} > \tau_{\mathrm{fail}}\right) \le \delta_{\mathrm{aud}}, \tag{12}$$

*where the probability is over the registered audit randomness. The bound is distribution-free with respect to how the fixed epoch was generated and does not extend to future epochs.*

*Proof sketch.* Conditioned on the committed population, the only randomness is the post-commit sampling. A level-$\delta_{\mathrm{aud}}$ sequential test under the actual law rejects a true bad-population null with probability at most $\delta_{\mathrm{aud}}$, whatever process produced the population. The full argument, and where each assumption enters, is in Appendix B.2. □

Validity is inherited from the finite-population test (Stark, 2019; Shekhar et al., 2023; Chugg et al., 2023); PCG-MAS contributes the release-population and certificate architecture that turns it into an assurance statement about agent decisions. The theorem is therefore only as strong as the operational assumptions in Table 2, each of which is stated formally in Appendix B.1 and attacked directly in Section 5.

**Proposition 4.2** (Compositional anchored scope). *For a DAG of committed candidate outputs, suppose each child derives its admissible scope only from $\Omega$ and accepted committed parent outputs. Conditional on those outputs, producer-side omission within the registered witness affordance cannot increase an end-to-end release by changing a child witness alone.*

*Proof sketch.* Induct over a topological order of the DAG: once accepted parents are fixed, each child scope is a registered function of $\Omega$ and those parents, so Corollary 4.1 applies node by node (Appendix B.2). □

This is a composition statement about selective-view robustness, not a sum of per-node failure rates and not a liability attribution rule. Replay adds a separate numerical-stability boundary: bounded

Table 1: Execution ledger and co-primary endpoints.

(a) Execution and assurance ledger.

| Quantity | Value |
|---|---|
| Models / datasets / cells | 7 / 8 / 56 |
| Test candidates | 10,080 |
| Governed releases | 8,077 |
| Certificates recomputed | 8,057 |
| Cross-host agreement | 99.41 |
| Exact-replay obligations | 68.4 |
| Indeterminate decisions | 3.1 |
| Unlogged releases | 0 |
| Audit labels | 404 |
| Epochs certified / audited | 4 / 4 |

(b) Co-primary endpoints by dataset.

| Dataset | Comparator | $\Delta J^\star$ (pp) | $\Delta R$ (pp) | $c^\star$ (%) |
|---|---|---|---|---|
| FEVER | MiniCheck | −2.5 | −1.9 | 79.7 |
| HotpotQA | MiniCheck | −2.5 | −1.9 | 80.4 |
| PubMedQA | MiniCheck | −2.6 | −1.5 | 80.2 |
| TAT-QA | MiniCheck | −2.9 | −1.7 | 80.4 |
| 2Wiki | MiniCheck | −3.1 | −1.5 | 80.2 |
| Mind2Web | CMVO | −3.1 | −0.9 | 80.0 |
| BFCL V1 | SignalMatchedFusion | N/A | −1.1 | 80.2 |
| AgentDojo | CMVO | −3.5 | −1.6 | 80.2 |
| **Macro** | hierarchical | −3.1 | −1.3 | 80.1 |

$\Delta$ = PCG-MAS − the task family's primary comparator, fixed on the calibration split; negative $\Delta$ favors PCG-MAS. $\Delta J^\star$: worst-case released failure under the registered curation affordance ($H_R$); $\Delta R$: natural conditional released-failure risk at coverage $c^\star$ ($H_N$). Macro: hierarchical macro-average over task families with Holm-adjusted inference.

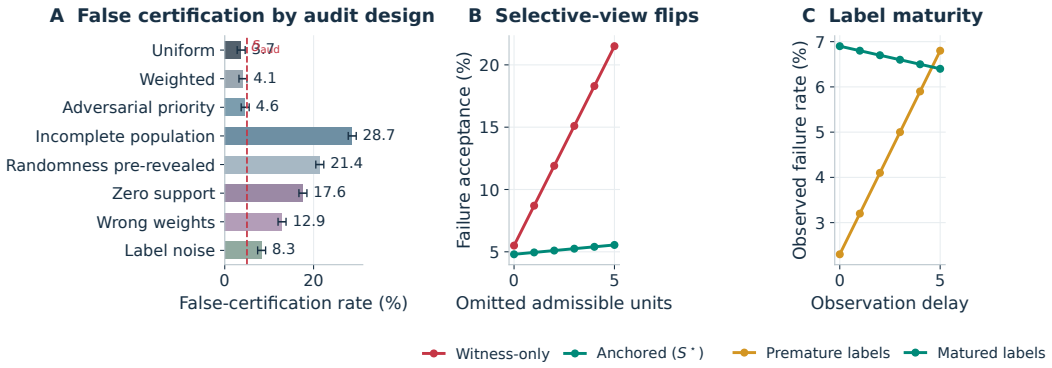



Figure 4: DIRECT/DERIVED. **The guarantees under stress.** **A**: empirical false-certification rate of the audit under valid designs (uniform, certificate-weighted, adversarial priorities with a support floor) and under designs that break one assumption; the dashed line marks $\delta_{\text{aud}}$. **B**: released failure admitted by witness-only and anchored acceptance as the producer omits more admissible units (Proposition 4.1). **C**: observed failure rate when immature labels are treated as outcomes versus matured labels.

score error yields decision invariance outside a tolerance band, and near-threshold decisions become INDETERMINATE. Appendix B also gives secondary irredundancy, audit-envelope, dependence, shift, and selectivity results (Appendix B.3), the operating-point corollary (Appendix B.4), a ranking lower bound (Appendix B.6), their interpretive limits (Appendix B.5), and the theorem-to-test map (Appendix B.7; Fig. 7).

## 5 EXPERIMENTS

**Anonymous artifacts.**

**Code:** open repository **Live demo:** launch checker

The demo is an interactive certificate/checker interface for consumer-side recomputation; it is not used to generate or score reported results. Appendix F records the prompt and structured I/O bank.

**Research questions. RQ1 (Curation robustness):** at controlled coverage, does consumer-anchored acceptance reduce worst-case released failure under selective-view curation ($H_R$)? **RQ2 (Natural reliability):** is natural released-failure risk non-inferior or superior at the same operating point ($H_N$)? **RQ3 (Mechanism):** is curation material, and which obligations close which failures? **RQ4 (Population assurance):** is the governed population complete, and does the audit certify validly and with fewer labels? **RQ5 (Verifiability, robustness, and cost):** do certificates recompute across hosts, compose with native guardrails, fail closed under stress, and at what overhead?

**Protocol.** Calibration, validation, and test splits are disjoint. Thresholds, the coverage target $c^\star$, the curation affordance, the audit design, and the primary comparator of each task family (lowest calibration risk at $c^\star$ among applicable, resource-compatible methods) are fixed on calibration data before the test split is generated (Table 22); the test sample size follows a registered power

Table 2: Load-bearing assumptions, practical enforcement, and direct falsifiers. Numbered statements appear in Appendix B.1.

| ID / assumption | Practical route | Falsifier / failure mode |
|---|---|---|
| A1 Consumer scope valid | Contract-derived $S^\star$; scope manifest | D1b incompleteness / omitted admissible evidence |
| A2 Release population complete | Governed receipt + sequence + inclusion check | Unlogged consequential release |
| A3 Audit law valid | Post-commit randomization; exact $\omega_{i,t}$; positive support | Early sample, zero support, wrong law |
| A4 Outcome authority independent/mature | Benchmark-native or blinded authority; maturation rule | Leakage, label noise, premature nonfailure |
| A5 Replay/integrity sufficient | Canonical bytes, hashes, tool IDs/receipts, replay class | Byte mismatch, spoofed-but-committed source, drift |

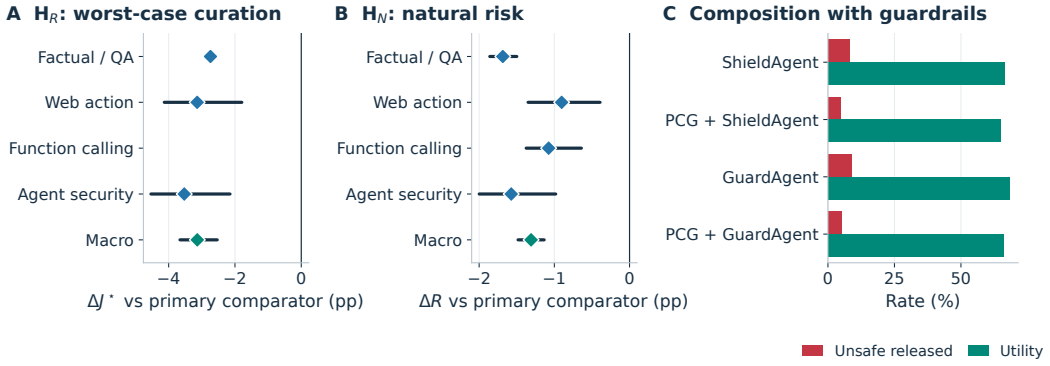



Figure 5: DERIVED. **Co-primary endpoints and composition.** **A**: $H_R$ difference in worst-case released failure ($\Delta J^\star$) against the primary comparator, per applicable task family and macro-averaged, with 95% model-cluster bootstrap intervals. **B**: $H_N$ difference in natural released-failure risk ($\Delta R$) at $c^\star$; the dashed line marks the non-inferiority margin when one is registered. Negative differences favor PCG-MAS. **C**: unsafe-release and utility rates for native guardrails alone and composed with PCG-MAS on identical AgentDojo trajectories. $H_R$ is N/A for function calling because BFCL has no registered curation affordance.

analysis (Table 21). Acceptance-only methods score the same frozen candidates. Failure is defined by benchmark-native authorities—answer correctness for factual tasks, the native evaluators for Mind2Web and BFCL, and separate utility and security predicates for AgentDojo—with $F^{\text{epistemic}}$, $F^{\text{policy}}$, and $F^{\text{any}}$ reported separately (Appendices C.3, E.2, and E.10; Table 19); no LLM judge is an outcome authority. Comparators retain their native scope (Table 20; Appendix C.5): policy and replay systems are compared on identical trajectories rather than pooled, and N/A is never zero-filled. BFCL contributes exact function-call checks; $H_R$ is N/A there because it exposes no selective view.

**Statistical analysis.** Each co-primary estimand is a hierarchical macro-average over task families of the paired difference between PCG-MAS and that family's primary comparator (Eqs. 9 and 8), with model-cluster bootstrap intervals and Holm's step-down correction over $H_R$ and $H_N$. A non-inferiority margin is used only if it is justified from task semantics before calibration; otherwise $H_N$ is tested for superiority. The 56-cell results provide heterogeneity evidence rather than 56 primary tests (Appendix E.12).

**RQ1–RQ2: curation robustness and natural reliability.** Table 1 (execution ledger Table 1a; endpoints Table 1b) and Fig. 5 report both co-primary endpoints; Fig. 1A–C summarizes them and disaggregates $J^\star$ over the 56 cells (full matrix in Appendix D, Fig. 13). On the test split, the macro $H_R$ difference is $-3.1$ pp and the macro $H_N$ difference is $-1.3$ pp (Table 1); the pooled governed epoch reaches the decision CERTIFY after 404 certificate-weighted audit labels (Table 3).

**RQ3: mechanism.** Curation materiality is measured on the development split before test-split evaluation, using the natural producer witness and an exhaustive oracle over the registered affordance. Fig. 3 shows where released failures arise and what each obligation family buys (component ablations in Appendix D).

**RQ4: population assurance.** Every consequential AgentDojo action routes through receipt-verifying governed tools under attacks by direct execution, stale/mismatched receipts, action before commitment, omitted, duplicated, or reordered entries, alternate routes, and malformed receipts. Table 3 and

Table 3: Population assurance per governed epoch at risk limit $\delta_{\text{aud}}$. Labels are independent outcome labels drawn until the sequential test stops; the decision is the EpochRiskCertificate outcome.

| Epoch | $N_{\mathcal{E}}$ | $\tau_{\text{fail}}$ (%) | Labels uniform | Labels weighted | Saving (%) | Decision | Unlogged releases |
|---|---|---|---|---|---|---|---|
| Factual / QA | 5,050 | 5.0 | 430 | 282 | 34.4 | CERTIFY | 0 |
| Web action | 1,007 | 5.0 | 168 | 108 | 35.7 | CERTIFY | 0 |
| Function calling | 1,011 | 5.0 | 151 | 104 | 31.1 | CERTIFY | 0 |
| Agent security | 1,009 | 5.0 | 226 | 149 | 34.1 | CERTIFY | 0 |
| **Pooled** | 8,077 | 5.0 | 638 | 404 | 36.7 | CERTIFY | 0 |

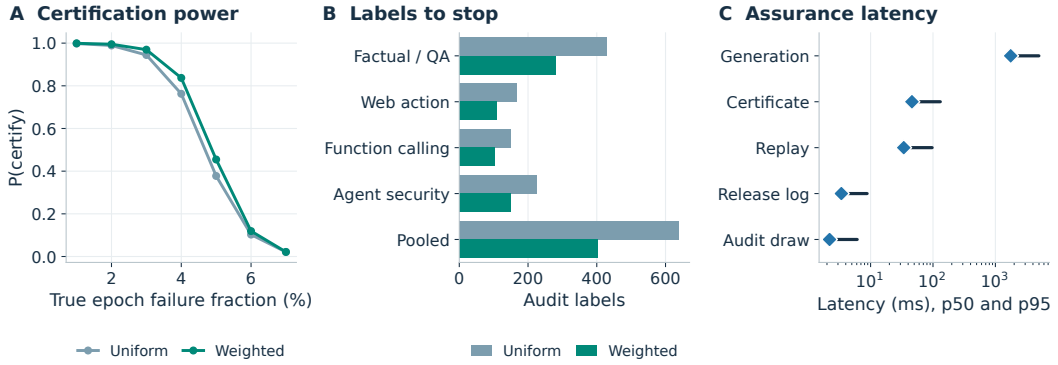



Figure 6: DIRECT/DERIVED. **Population assurance.** **A**: probability that the audit certifies an epoch as a function of its true failure fraction; the dashed line marks $\tau_{\text{fail}}$. **B**: audit labels to the stopping state per epoch. **C**: median latency per assurance phase with 95th-percentile whisker.

Table 4: Resource and assurance-cost anatomy.

| Quantity | PCG-MAS |
|---|---|
| Certificate latency, p50 (ms) | 46 |
| Certificate latency, p95 (ms) | 131 |
| Tokens vs. NoCert (×) | 1.47 |
| Checker + replay (×) | 1.23 |
| Release log / release (ms) | 3.4 |
| Audit labels / 10k releases | 500.2 |
| Audit cost / 10k releases ($) | 18.90 |

Table 5: Checker characterization (test split).

| Family | Prec. | Recall | FPR | ECE |
|---|---|---|---|---|
| Factual / QA | 92.8 | 90.1 | 3.1 | 0.026 |
| Web action | 90.4 | 87.3 | 3.9 | 0.034 |
| Function calling | 94.4 | 91.6 | 2.4 | 0.021 |
| Agent security | 89.2 | 86.1 | 4.7 | 0.041 |

Fig. 6 report unlogged releases, audit labels under uniform and certificate-weighted designs, and the resulting certification decisions. Fig. 4 breaks one assumption at a time (incomplete populations, pre-revealed randomness, zero support, wrong weights, label noise, premature labels) to show which conditions the guarantee needs (Appendix E.11).

**RQ5: verifiability, robustness, and cost.** Independent hosts recompute every DecisionCertificate, with each obligation carrying a replay class (Table 29). A spoofing arm commits misleading tool bytes that pass integrity and replay checks, showing that byte integrity is not source truth; the composition arm (Fig. 5C; Table 27) tests PCG-MAS over native guardrails. Tables 4 and 5 report resource cost and checker characterization (equal-resource and Pareto comparisons in Appendix E.14).

**Limitations.** Before we conclude, we cite the facts that both guarantees are conditional: curation robustness holds only within the consumer-derived scope, and population assurance only for governed releases with independent, mature labels. Certificates bind bytes, not world truth; checker error, spoofed sources, and ungoverned sinks remain outside them, and all evidence is from benchmarks, not deployments.

**Conclusion and future work.** PCG-MAS provides two-level release assurance: consumer-anchored acceptance for each decision, and independent finite-population auditing of the realized governed stream. The first earns its performance through $\text{H}_R$ and $\text{H}_N$; the second certifies or refuses that performance. Both remain conditional on declared scope, checker calibration, integrity bindings, outcome authority, population completeness, replay class, and audit law; future work targets attested third-party execution and domain-specific outcome authorities without weakening the standard.

## REPRODUCIBILITY STATEMENT

The submission binds the retained aggregate experimental contracts, comparator applicability, artifact lineage, and manuscript numbers to machine-readable files and anonymous code. Appendix C supplies the runtime algorithm, replay/support diagnostics, worked AgentDojo certificate, and guarantee boundary (Figs. 9, 10, 11, 12); Appendix B gives the load-bearing curation/RLA proofs and dependency map (Fig. 8; Table 7). Appendix D contains the full matrix and comparator atlas (Fig. 13/13; Table 9), ablations (Fig. 14; Tables 10, 11), audit/curation/completeness/failure-decomposition/scaling views (Figs. 15, 16, 17, 18, 19; Tables 12, 13, 14, 15, 16). Appendix E documents the retained execution summaries, comparator scope, power, curation, witness search, audit law, integrity/composition, coverage, replay, resources, and gates (Tables 17, 18, 19, 20, 21, 23, 24, 25, 26, 27, 28, 29, 4, 5, 30, 31, 22, 8; Figs. 20, 21). Appendix F gives the prompt/I-O bank, interface taxonomy, dataset-specific I/O map, representative examples, capture semantics, structured-output identity, and notation (Sections F.1, F.2, F.3, F.4, F.5, F.6, F.7, F.8; Tables 32, 33, 34, 35, 36, 37, 38, 39); runtime/proof cross-indexes are Sections C.1, C.4, C.6, C.7, C.8, C.9, C.10, C.11, C.12, B.1, B.2, B.3, B.7, and B.8.

Table 6 maps each formal object, algorithm, and extended result to the corresponding appendix section and main-text claim.

All result tables and figures are generated from the retained aggregate contracts and the 616-row model–dataset–method result matrix. The reserved per-item canonical contracts are header-only and are not claimed as replay evidence.

## AI USE STATEMENT

Generative AI tools were used to support literature discovery and checking, bibliography preparation, LaTeX formatting, and workflow/ schematic images that do not depend on the empirical results. The authors reviewed and verified all AI-assisted material and take full responsibility for the final submission and its reported results.

## REFERENCES

Ander Alvarez, Santhiya Rajan, Samuel Mugel, and Román Orús. ProvenanceGuard: Source-aware factuality verification for MCP-based LLM agents. *arXiv preprint arXiv:2606.18037*, 2026. doi: 10.48550/arXiv.2606.18037. URL https://arxiv.org/abs/2606.18037.

Nikolay Avramov, Hidde Lycklama, Alexander Viand, and Anwar Hithnawi. Atlas: Efficient verifiable semantic search. *arXiv preprint arXiv:2609.11841*, 2026. doi: 10.48550/arXiv.2609.11841. URL https://arxiv.org/abs/2609.11841.

Zhaorun Chen, Mintong Kang, and Bo Li. ShieldAgent: Shielding agents via verifiable safety policy reasoning. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 8313–8344. PMLR, 2025. URL https://proceedings.mlr.press/v267/chen25ae.html.

Ben Chugg, Santiago Cortes-Gomez, Bryan Wilder, and Aaditya Ramdas. Auditing fairness by betting. *arXiv preprint arXiv:2305.17570*, 2023. doi: 10.48550/arXiv.2305.17570. URL https://arxiv.org/abs/2305.17570.

Edoardo Debenedetti, Jie Zhang, Mislav Balunovic, Luca Beurer-Kellner, Marc Fischer, and Florian Tramèr. AgentDojo: A dynamic environment to evaluate prompt injection attacks and defenses for LLM agents. In *Advances in Neural Information Processing Systems*, volume 37, 2024. doi: 10.52202/079017-2636. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/97091a5177d8dc64b1da8bf3e1f6fb54-Abstract-Datasets_and_Benchmarks_Track.html. Datasets and Benchmarks Track.

Xiang Deng, Yu Gu, Boyuan Zheng, Shijie Chen, Samuel Stevens, Boshi Wang, Huan Sun, and Yu Su. Mind2web: Towards a generalist agent for the web. In *Advances in Neural Information Processing Systems*, volume 36, 2023. doi: 10.52202/

075280-1220. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/5950bf290a1570ea401bf98882128160-Abstract-Datasets_and_Benchmarks.html.

Dhruv Dixit and Paritosh Pandey. R2VC: Modular fact-checking with retrieval, verification, and confidence calibration. *arXiv preprint arXiv:2609.11955*, 2026. doi: 10.48550/arXiv.2609.11955. URL https://arxiv.org/abs/2609.11955.

Alexander Fabbri, Chien-Sheng Wu, Wenhao Liu, and Caiming Xiong. QAFactEval: Improved QA-based factual consistency evaluation for summarization. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 2587–2601, Seattle, United States, July 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.naacl-main.187. URL https://aclanthology.org/2022.naacl-main.187/.

Erhu Feng, Wenbo Zhou, Zibin Liu, Le Chen, Yunpeng Dong, Cheng Zhang, Yisheng Zhao, Dong Du, Zhichao Hua, Yubin Xia, and Haibo Chen. Get experience from practice: LLM agents with record & replay. *arXiv preprint arXiv:2505.17716*, 2025. doi: 10.48550/arXiv.2505.17716. URL https://arxiv.org/abs/2505.17716.

Murdoch J. Gabbay. Cryptographic certificates of validity for trustworthy AI. *arXiv preprint arXiv:2606.23768*, 2026. doi: 10.48550/arXiv.2606.23768. URL https://arxiv.org/abs/2606.23768.

Tianyu Gao, Howard Yen, Jiatong Yu, and Danqi Chen. Enabling large language models to generate text with citations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 6465–6488. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.emnlp-main.398. URL https://aclanthology.org/2023.emnlp-main.398/.

Ryan Greenblatt, Buck Shlegeris, Kshitij Sachan, and Fabien Roger. AI control: Improving safety despite intentional subversion. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 16295–16336. PMLR, 2024. URL https://proceedings.mlr.press/v235/greenblatt24a.html.

Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. Constructing a multi-hop QA dataset for comprehensive evaluation of reasoning steps. In *Proceedings of the 28th International Conference on Computational Linguistics (COLING)*, pp. 6609–6625. International Committee on Computational Linguistics, 2020. doi: 10.18653/v1/2020.coling-main.580. URL https://aclanthology.org/2020.coling-main.580/.

Dengzhe Hou, Lingyu Jiang, Fangzhou Lin, and Kazunori D. Yamada. CogEEGAgent: Toward autonomous cognitive EEG analysis with grounded execution and selection-aware verification. *arXiv preprint arXiv:2607.25045*, 2026. doi: 10.48550/arXiv.2607.25045. URL https://arxiv.org/abs/2607.25045.

Mengkang Hu, Tianxing Chen, Qiguang Chen, Yao Mu, Wenqi Shao, and Ping Luo. HiAgent: Hierarchical working memory management for solving long-horizon agent tasks with large language model. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 32779–32798, Vienna, Austria, 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.acl-long.1575. URL https://aclanthology.org/2025.acl-long.1575/.

Yue Huang, Hang Hua, Yujun Zhou, Pengcheng Jing, Manish Nagireddy, Inkit Padhi, Greta Dolcetti, Zhangchen Xu, Subhajit Chaudhury, Ambrish Rawat, Liubov Nedoshivina, Pin-Yu Chen, Prasanna Sattigeri, and Xiangliang Zhang. Building a foundational guardrail for general agentic systems via synthetic data. In *International Conference on Learning Representations*, 2026. doi: 10.48550/arXiv.2510.09781. URL https://openreview.net/forum?id=M47SWYubR5.

Qiao Jin, Bhuwan Dhingra, Zhengping Liu, William Cohen, and Xinghua Lu. PubMedQA: A dataset for biomedical research question answering. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on*

*Natural Language Processing (EMNLP-IJCNLP)*, pp. 2567–2577. Association for Computational Linguistics, 2019. doi: 10.18653/v1/D19-1259. URL https://aclanthology.org/D19-1259/.

Seongjae Kang, Taehyung Yu, and Sung Ju Hwang. PolicyGuide: From guarding one action to guiding the whole workflow for policy-compliant LLM agents. *arXiv preprint arXiv:2608.19861*, 2026. doi: 10.48550/arXiv.2608.19861. URL https://arxiv.org/abs/2608.19861.

Ehud Karpas, Omri Abend, Yonatan Belinkov, Barak Lenz, Opher Lieber, Nir Ratner, Yoav Shoham, Hofit Bata, Yoav Levine, Kevin Leyton-Brown, Dor Muhlgay, Noam Rozen, Erez Schwartz, Gal Shachaf, Shai Shalev-Shwartz, Amnon Shashua, and Moshe Tenenholtz. MRKL systems: A modular, neuro-symbolic architecture that combines large language models, external knowledge sources and discrete reasoning. *arXiv preprint arXiv:2205.00445*, 2022. doi: 10.48550/arXiv.2205.00445. URL https://arxiv.org/abs/2205.00445.

Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 6769–6781. Association for Computational Linguistics, 2020. doi: 10.18653/v1/2020.emnlp-main.550. URL https://aclanthology.org/2020.emnlp-main.550/.

George Koomullil. Proof-carrying certificates for LLM pipelines: A trust-boundary architecture. *arXiv preprint arXiv:2605.16407*, 2026. doi: 10.48550/arXiv.2605.16407. URL https://arxiv.org/abs/2605.16407.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems*, volume 33, pp. 9459–9474, 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/hash/6b493230205f780e1bc26945df7481e5-Abstract.html.

Yifei Li, Xiang Yue, Zeyi Liao, and Huan Sun. AttributionBench: How hard is automatic attribution evaluation? In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 14919–14935. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.findings-acl.886. URL https://aclanthology.org/2024.findings-acl.886/.

Dongrui Liu, Qihan Ren, Chen Qian, Shuai Shao, Yuejin Xie, Yu Li, Zhonghao Yang, Haoyu Luo, Peng Wang, Qingyu Liu, Binxin Hu, Ling Tang, Jilin Mei, Dadi Guo, Leitao Yuan, Junyao Yang, Guanxu Chen, Qihao Lin, Yi Yu, Bo Zhang, Jiaxuan Guo, Jie Zhang, Wenqi Shao, Huiqi Deng, Zhiheng Xi, Wenjie Wang, Wenxuan Wang, Wen Shen, Zhikai Chen, Haoyu Xie, Jialing Tao, Juntao Dai, Jiaming Ji, Zhongjie Ba, Linfeng Zhang, Yong Liu, Quanshi Zhang, Lei Zhu, Zhihua Wei, Hui Xue, Chaochao Lu, Jing Shao, and Xia Hu. AgentDoG: A diagnostic guardrail framework for AI agent safety and security. *arXiv preprint arXiv:2601.18491*, 2026. doi: 10.48550/arXiv.2601.18491. URL https://arxiv.org/abs/2601.18491.

Mar Llambí-Morillas and Daniel Fernández-Fernández. Cryptographically verifiable authorization for autonomous AI agents: A falsifiable hypothesis and proof-of-concept. *Frontiers in Computer Science*, 8:1966725, 2026. doi: 10.3389/fcomp.2026.1966725. URL https://doi.org/10.3389/fcomp.2026.1966725.

Harsh Maheshwari, Srikanth Tenneti, and Alwarappan Nakkiran. CiteFix: Enhancing RAG accuracy through post-processing citation correction. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 6: Industry Track)*, pp. 310–317, Vienna, Austria, 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.acl-industry.23. URL https://aclanthology.org/2025.acl-industry.23/.

Jacob Menick, Maja Trebacz, Vladimir Mikulik, John Aslanides, Francis Song, Martin Chadwick, Mia Glaese, Susannah Young, Lucy Campbell-Gillingham, Geoffrey Irving, and Nat McAleese. Teaching language models to support answers with verified quotes. *arXiv preprint arXiv:2203.11147*, 2022. doi: 10.48550/arXiv.2203.11147. URL https://arxiv.org/abs/2203.11147.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Wei Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. FActScore: Fine-grained atomic evaluation of factual precision in long form text generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 12076–12100. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.emnlp-main.741. URL https://aclanthology.org/2023.emnlp-main.741/.

Reiichiro Nakano, Jacob Hilton, Suchir Balaji, Jeff Wu, Long Ouyang, Christina Kim, Christopher Hesse, Shantanu Jain, Vineet Kosaraju, William Saunders, Xu Jiang, Karl Cobbe, Tyna Eloundou, Gretchen Krueger, Kevin Button, Matthew Knight, Benjamin Chess, and John Schulman. WebGPT: Browser-assisted question-answering with human feedback. *arXiv preprint arXiv:2112.09332*, 2021. doi: 10.48550/arXiv.2112.09332. URL https://arxiv.org/abs/2112.09332.

George C. Necula. Proof-carrying code. In *Proceedings of the 24th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL)*, pp. 106–119. ACM, 1997. doi: 10.1145/263699.263712. URL https://doi.org/10.1145/263699.263712.

Shishir G. Patil, Huanzhi Mao, Fanjia Yan, Charlie Cheng-Jie Ji, Vishnu Suresh, Ion Stoica, and Joseph E. Gonzalez. The berkeley function calling leaderboard (BFCL): From tool use to agentic evaluation of large language models. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 48371–48392. PMLR, 2025. URL https://proceedings.mlr.press/v267/patil25a.html.

Jingxi Qiu, Zeyu Han, and Cheng Huang. SURE-RAG: Sufficiency and uncertainty-aware evidence verification for selective retrieval-augmented generation. *arXiv preprint arXiv:2605.03534*, 2026a. doi: 10.48550/arXiv.2605.03534. URL https://arxiv.org/abs/2605.03534.

Zipeng Qiu, Wenjie Qu, Jiaheng Zhang, and Binhang Yuan. V3DB: Audit-on-demand zero-knowledge proofs for verifiable vector search over committed snapshots. *arXiv preprint arXiv:2603.03065*, 2026b. doi: 10.48550/arXiv.2603.03065. URL https://arxiv.org/abs/2603.03065.

James Rhodes and George Kang. Proof of execution: Runtime verification for governed AI agent actions. *arXiv preprint arXiv:2607.05397*, 2026. doi: 10.48550/arXiv.2607.05397. URL https://arxiv.org/abs/2607.05397.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. In *Advances in Neural Information Processing Systems*, volume 36, 2023. doi: 10.52202/075280-2997. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/d842425e4bf79ba039352da0f658a906-Abstract-Conference.html.

Shubhanshu Shekhar, Ziyu Xu, Zachary C. Lipton, Pierre J. Liang, and Aaditya Ramdas. Risk-limiting financial audits via weighted sampling without replacement. *arXiv preprint arXiv:2305.06884*, 2023. doi: 10.48550/arXiv.2305.06884. URL https://arxiv.org/abs/2305.06884.

Shivani Shukla and Himanshu Joshi. Proof-carrying answers: A systematic protocol for verifiable retrieval-augmented generation with cryptographic provenance. In *2025 Annual Computer Security Applications Conference Workshops (ACSAC Workshops)*, pp. 405–413, 2025. doi: 10.1109/ACSACW69556.2025.00050. URL https://doi.org/10.1109/ACSACW69556.2025.00050.

Philip B. Stark. Sets of half-average nulls generate risk-limiting audits: SHANGRLA. *arXiv preprint arXiv:1911.10035*, 2019. doi: 10.48550/arXiv.1911.10035. URL https://arxiv.org/abs/1911.10035.

Jiashuo Sun, Jimeng Shi, Yixuan Xie, Saizhuo Wang, Jash Rajesh Parekh, Pengcheng Jiang, Zhiyi Shi, Jiajun Fan, Qinglong Zheng, Peiran Li, Shaowen Wang, Ge Liu, and Jiawei Han. Retrieval is cheap, show me the code: Executable multi-hop reasoning for retrieval-augmented generation. *arXiv preprint arXiv:2605.12975*, 2026. doi: 10.48550/arXiv.2605.12975. URL https://arxiv.org/abs/2605.12975.

Liyan Tang, Philippe Laban, and Greg Durrett. MiniCheck: Efficient fact-checking of LLMs on grounding documents. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 8818–8847, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.499. URL https://aclanthology.org/2024.emnlp-main.499/.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. FEVER: A large-scale dataset for fact extraction and VERification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pp. 809–819. Association for Computational Linguistics, 2018. doi: 10.18653/v1/N18-1074. URL https://aclanthology.org/N18-1074/.

Georg Wölflein, Dyke Ferber, Daniel Truhn, Ognjen Arandjelović, and Jakob Nikolas Kather. LLM agents making agent tools. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 26092–26130, Vienna, Austria, 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.acl-long.1266. URL https://aclanthology.org/2025.acl-long.1266/.

Qingyun Wu, Gagan Bansal, Jieyu Zhang, Yiran Wu, Beibin Li, Erkang Zhu, Li Jiang, Xiaoyun Zhang, Shaokun Zhang, Jiale Liu, Ahmed Hassan Awadallah, Ryen W. White, Doug Burger, and Chi Wang. AutoGen: Enabling next-gen LLM applications via multi-agent conversations. In *First Conference on Language Modeling (COLM)*, 2024. URL https://openreview.net/forum?id=BAakY1hNKS.

Zhen Xiang, Linzhi Zheng, Yanjie Li, Junyuan Hong, Qinbin Li, Han Xie, Jiawei Zhang, Zidi Xiong, Chulin Xie, Carl Yang, Dawn Song, and Bo Li. GuardAgent: Safeguard LLM agents via knowledge-enabled reasoning. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 68316–68342. PMLR, 2025. URL https://proceedings.mlr.press/v267/xiang25a.html.

Tianyang Xu, Dan Zhang, Kushan Mitra, and Estevam Hruschka. Verification-aware planning for multi-agent systems. In *Proceedings of the 2026 Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pp. 7528–7546. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.eacl-long.353. URL https://aclanthology.org/2026.eacl-long.353/.

Yumo Xu, Peng Qi, Jifan Chen, Kunlun Liu, Rujun Han, Lan Liu, Bonan Min, Vittorio Castelli, Arshit Gupta, and Zhiguo Wang. CiteEval: Principle-driven citation evaluation for source attribution. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 32759–32778, Vienna, Austria, 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.acl-long.1574. URL https://aclanthology.org/2025.acl-long.1574/.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 2369–2380. Association for Computational Linguistics, 2018. doi: 10.18653/v1/D18-1259. URL https://aclanthology.org/D18-1259/.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. ReAct: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations (ICLR)*, 2023. URL https://openreview.net/forum?id=WE_vluYUL-X.

Siyu Yuan, Kaitao Song, Jiangjie Chen, Xu Tan, Yongliang Shen, Kan Ren, Dongsheng Li, and Deqing Yang. EASYTOOL: Enhancing LLM-based agents with concise tool instruction. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 951–972, Albuquerque, New Mexico, 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.naacl-long.44. URL https://aclanthology.org/2025.naacl-long.44/.

Yuheng Zha, Yichi Yang, Ruichen Li, and Zhiting Hu. AlignScore: Evaluating factual consistency with a unified alignment function. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 11328–11348, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.634. URL https://aclanthology.org/2023.acl-long.634/.

Ziling Zhou. Governing dynamic capabilities: Cryptographic binding and reproducibility verification for AI agent tool use. *arXiv preprint arXiv:2603.14332*, 2026. doi: 10.48550/arXiv.2603.14332. URL https://arxiv.org/abs/2603.14332.

Fengbin Zhu, Wenqiang Lei, Youcheng Huang, Chao Wang, Shuo Zhang, Jiancheng Lv, Fuli Feng, and Tat-Seng Chua. TAT-QA: A question answering benchmark on a hybrid of tabular and textual content in finance. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 3277–3287. Association for Computational Linguistics, 2021. doi: 10.18653/v1/2021.acl-long.254. URL https://aclanthology.org/2021.acl-long.254/.

# A Related Work

Work relevant to PCG-MAS spans grounding, agent control and policy enforcement, replay/provenance, proof-carrying systems, verifiable retrieval, and sequential finite-population auditing. We do not claim these primitives individually as new. Its question is how to combine them into a release-assurance architecture in which the producer neither defines the maximum acceptance view nor controls the population on which an independent risk statement is made.

**Agentic runtimes and long-horizon tool use.** ReAct, Toolformer, MRKL, and AutoGen established language-model interaction with tools, retrieval, and multi-agent coordination (Yao et al., 2023; Schick et al., 2023; Karpas et al., 2022; Wu et al., 2024). Later work studies concise tool interfaces, hierarchical memory, and agents that construct tools themselves (Yuan et al., 2025; Hu et al., 2025; Wölflein et al., 2025). These systems motivate the runtime state PCG-MAS commits, but do not by themselves provide an independent population-level statement about the decisions actually released.

**AI control and untrusted producers.** AI-control protocols explicitly evaluate safety procedures when a powerful model may intentionally subvert oversight (Greenblatt et al., 2024). PCG-MAS shares the untrusted-producer perspective but studies a different assurance object: a consumer-owned release decision and the independently auditable finite population of releases. Its selective-view threat is narrower than arbitrary adversarial behavior and is registered explicitly rather than inferred from generic red-team success.

**Grounding, attribution, and selective verification.** RAG and dense retrieval provide open-book evidence (Lewis et al., 2020; Karpukhin et al., 2020); WebGPT, GopherCite, and citation-aware generation attach sources to answers (Nakano et al., 2021; Menick et al., 2022; Gao et al., 2023). FActScore and AttributionBench make factuality and attribution more granular (Min et al., 2023; Li et al., 2024), while CiteEval and CiteFix extend citation-quality evaluation and correction (Xu et al., 2025; Maheshwari et al., 2025). MiniCheck, AlignScore, and QAFactEval provide learned grounding/factuality checks (Tang et al., 2024; Zha et al., 2023; Fabbri et al., 2022). SURE-RAG emphasizes that evidence sufficiency is a set-level selective-verification problem rather than an independent-passage score (Qiu et al., 2026a). PCG-MAS can use such verifiers as learned obligations; its additional question is who controls the admissible verification scope and how the resulting release stream is audited.

**Agent safety, policy compliance, and verification-aware planning.** ShieldAgent and GuardAgent enforce action policies over trajectories (Chen et al., 2025; Xiang et al., 2025); Safiron targets pre-execution risk, AgentDoG adds trajectory diagnosis, and PolicyGuide compiles policy into workflow guidance (Huang et al., 2026; Liu et al., 2026; Kang et al., 2026). VeriMAP moves verification criteria into multi-agent planning (Xu et al., 2026). These are important native-scope comparators or infrastructure components; the proposed method does not assign them grounding, replay, or population-audit guarantees absent from their released interfaces.

**Executable, provenance-aware, and domain-typed agent pipelines.** R2VC modularizes retrieval, verification, calibration, citation, and abstention for fact checking (Dixit & Pandey, 2026); PyRAG exposes multi-hop reasoning as executable programs with inspectable intermediate state and deterministic runtime feedback (Sun et al., 2026); ProvenanceGuard keeps MCP source identity explicit during support and attribution checks (Alvarez et al., 2026); and CogEEGAgent uses a domain-specific scientific harness with typed contracts and outcome-blind confirmation control (Hou et al., 2026). These systems sharpen the boundary of our claim: deterministic traces, source-aware verification, or domain contracts can supply valuable obligations, but PCG-MAS targets a general consumer-owned release decision plus an auditable complete release population. Where an external system has a compatible public interface and overlapping task semantics it is eligible for native-scope comparison; otherwise it is treated as related infrastructure rather than forced into an artificial pooled score.

**Proof-carrying and verifiable LLM pipelines.** Proof-carrying code provides the broader consumer-verification lineage (Necula, 1997). AgentRR adapts record-and-replay to reusable agent experience (Feng et al., 2025). Contemporary proof-carrying LLM work includes trust-boundary certificates over pipeline evidence and checker assumptions (Koomullil, 2026), cryptographic proof-carrying answers for RAG (Shukla & Joshi, 2025), cryptographic validity certificates (Gabbay, 2026), dynamic capability binding and replay commitments (Zhou, 2026), proof of execution (Rhodes & Kang, 2026), and verifiable authorization (Llambí-Morillas & Fernández-Fernández, 2026). Koomullil in particular

develops conflict-aware bilattice grounding with an emission-gate soundness lemma over a declared trust boundary; we therefore do not claim conflict-aware supplied-evidence checking itself as novel. Our selective-view question is different: who controls the maximum admissible view on which release is decided, and can the realized release population later support an independent audit?

**Verifiable retrieval and committed execution.** Atlas proves HNSW semantic-search execution against a committed index (Avramov et al., 2026), while V3DB provides audit-on-demand zero-knowledge proofs for vector search over committed snapshots (Qiu et al., 2026b). These systems can strengthen retrieval integrity inside the trusted base. They do not automatically establish that the downstream agent's complete release stream has low realized failure.

**Risk-limiting and anytime-valid auditing.** Risk-limiting audits provide finite-population, design-based control over erroneous certification. SHANGRLA reduces election audits to sequential tests over finite lists (Stark, 2019). Risk-limiting financial auditing develops confidence sequences for randomized weighted sampling without replacement and side-information-driven sampling (Shekhar et al., 2023). Auditing Fairness by Betting studies anytime-valid monitoring under probabilistic, potentially changing collection policies and distribution shift (Chugg et al., 2023). PCG-MAS imports these statistical ideas rather than claiming them as new. Its contribution is the agent-release architecture that binds per-decision certificates, complete-population evidence, outcome authority, and the registered audit design into one independently verifiable assurance path.

**Positioning.** Across the literature, evidence support, policy validity, replayability, authorization, committed retrieval, and sequential auditing are separately available. PCG-MAS asks whether the release decision itself can be made consumer-controlled and replayable, and whether the complete realized release population can then support an independent finite-population risk statement. The paper therefore evaluates the integration and its failure boundaries, not a claim that any constituent verifier, cryptographic primitive, or audit statistic is novel.

# B Proofs and Theoretical Derivations

**Appendix map.** Table 6 lists where each formal object, algorithm, protocol, and extended result appears and which main-text element it supports.

Table 6: Appendix map.

| Location | Contents | Supports |
|---|---|---|
| App. A | Related work: AI control, proof-carrying pipelines, grounding, guardrails, replay, verifiable retrieval, risk-limiting auditing | Section 1 |
| B.1 | Assumptions A1–A5, each with its practical route and falsifier | Theorem 4.1; Table 2 |
| B.2 | Proofs of Proposition 4.1, Corollary 4.1, Theorem 4.1, Proposition 4.2 | Section 4 |
| B.3–B.6 | Secondary results: irredundancy, audit envelope and converse, dependence and common-mode floor, restricted-TV shift, selectivity identity, operating point, ranking lower bound | Section 4 |
| B.5–B.7 | Interpretive limits; theorem-to-test map | Figs. 4, 8 |
| App. C | Runtime graph, consumer scope, release population, worked certificate, replay, contracts, grounding certificates, witness minimization, support separation, checker interface, guarantee boundary | Sections 2–3 |
| C.16 | Algorithm 1 (consumer-anchored release decision); Fig. 9 (witness search) | Section 3 |
| App. D | 56-cell matrix, ablations, audit sampling, oracle curation, bypass and necessity, failure decomposition, audit scaling | Section 5 |
| App. E | Execution identity, outcome authority, comparators, sample size, curation, checker, witness search, audit design, integrity, maturation, inference, replay, resources, gates, lineage, deployment, broader impact | Section 5 |
| App. F | Input bank, interfaces, end-to-end bindings, worked records, notation | Section 5 |

## B.1 Assumptions and their practical routes

The load-bearing results rest on five operational assumptions rather than on trust in the whole agent stack. Each is stated formally and followed by the route that enforces it in practice and the experiment that attacks it; Table 2 gives the compact map.

**Assumption A1** (Consumer-scope validity)**.** For every input $x$, the contract-derived scope $S^\star(x) = \mathrm{Derive}_\Omega(x)$ contains every item that is admissible under the declared contract $\Omega$.

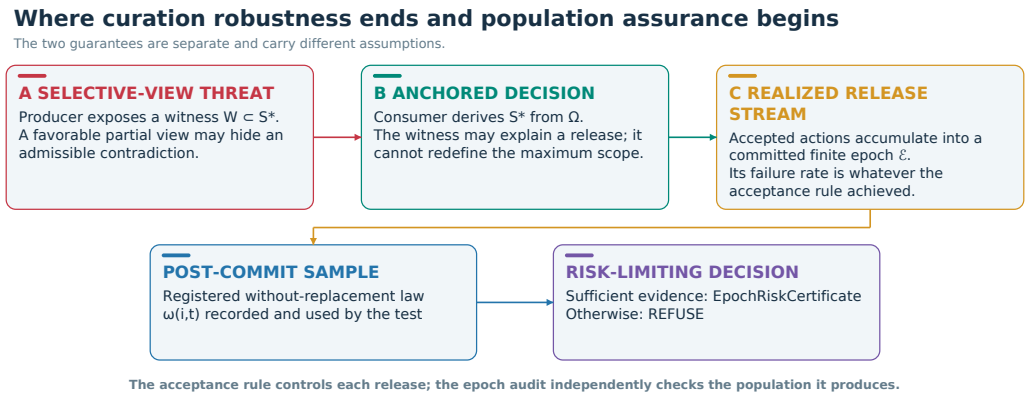



Figure 7: PROTOCOL. **Guarantee geometry.** **A**: the producer witness $W$ versus the consumer-derived admissible scope $S^\star$. **B**: the anchored decision, which the witness can explain but not widen. **C**: the realized release stream, committed as a finite epoch and audited after commitment; an EpochRiskCertificate is issued only when the registered finite-population test rejects the bad-population null.

**Practical route.** Scope derivation, retrieval and index snapshots, tool-query contracts, and eligibility rules are fixed before evaluation and committed in each certificate. A violation appears as D1b—admissible evidence exists but is not derived—and is measured separately from producer curation (D1a); the theory never sets D1b to zero by definition.

**Assumption A2** (Governed-population completeness)**.** Every consequential release in an epoch is committed to the append-only release log, with its sequence number, DecisionCertificate hash, and inclusion receipt, before its action executes.

**Practical route.** Governed tools refuse to execute without a valid receipt. The bypass suite attempts direct execution, stale or mismatched receipts, action before commitment, omitted, duplicated, or reordered entries, alternate routes, and malformed receipts. An append-only commitment proves the integrity of included entries, not completeness by itself.

**Assumption A3** (Registered audit law and positive support)**.** Audit randomness is generated after the epoch is committed. At each draw $t$ the conditional selection probabilities $\omega_{i,t}$ are predictable from the allowed history, recorded, used by the test, and satisfy $\omega_{i,t} \geq \omega_{\min} > 0$ for every still-eligible release.

**Practical route.** The audit ledger records the law of every draw together with the side information that produced it. Necessity ablations reveal the randomness early, set zero support on an adverse stratum, and apply the wrong weights.

**Assumption A4** (Independent, mature outcome authority)**.** Outcome labels are produced by the registered authority independently of the online acceptance path, and a label is final only after the registered observation horizon $\theta_{\text{obs}}$.

**Practical route.** Benchmark-native evaluators are hash-bound before evaluation, and same-item labels are inaccessible to acceptance. Delayed-label and label-noise ablations measure how the assurance statement degrades when the assumption fails.

**Assumption A5** (Replay/integrity sufficiency)**.** The trusted base—hash function, canonicalization, log schema, version pins, replay runner, and checker bundle—is uncompromised, and the declared replay class of every obligation holds.

**Practical route.** Content addressing and cross-host replay establish which bytes, tool identities, code versions, and checker states were evaluated; they do not establish that an upstream source is truthful. Byte mismatch, drift, and non-replayable actions surface as replay classes or INDETERMINATE decisions, and source-to-world error remains external to the certificate.

### B.2 LOAD-BEARING ASSURANCE RESULTS

The load-bearing results are deliberately narrow. The risk-limiting audit machinery is imported from finite-population sequential testing; the contribution is the release architecture that makes its population and sampling assumptions explicit for agent decisions.

**Proof of Proposition 4.1.** Omission-robustness means that for every registered pair $W \subseteq W'$, a rejection on $W'$ cannot coexist with an acceptance on $W$; for binary $V$ this is exactly $V(x, y; W) \leq V(x, y; W')$, which is Eq. (11), and the converse direction is immediate. Refutation-blindness follows directly: if $V(x, y; W) = 1$ and $W' = W \cup \{r\}$ is admissible for a refuting item $r$, monotonicity forces $V(x, y; W') = 1$, so the rule cannot reject on the basis of refuting evidence added to an accepted view. Now consider any contested candidate for which a useful conflict-sensitive verifier accepts a supportive admissible view $W$ but rejects an admissible extension $W'$ after decisive conflict is added. The monotonicity condition is violated, so a producer allowed to choose between these views can expose $W$ and obtain acceptance although the more informative admissible view rejects. Conversely, a rule that rejects every such supportive subview can avoid this flip, but at the cost of the nontrivial acceptance behavior under discussion. For a finite registered affordance, exhaustive enumeration therefore measures the exact selective-view flip mass; for larger affordances the registered oracle or attack family lower-bounds it. Anchoring the maximum decision view at consumer-derived $S^\star$ removes this producer choice without asserting that $S^\star$ is world-complete.

**Proof of Corollary 4.1.** By construction, the anchored decision $D^\star(x, y; S^\star)$ is a function of the fixed candidate, consumer contract, consumer-derived scope, and trusted-base state; it is not a function of which admissible compact witness the producer chooses to present. Replacing $W$ by another admitted $W'$ can therefore change only witness-local validity, portability, or explanation sidecars. If $D^\star(x, y; S^\star) = 0$, every release rule that conjuncts the anchored decision with any witness-local requirement remains zero. Thus permitted witness curation cannot convert an anchored rejection into release. The argument does not establish that $S^\star$ is world-complete; failures of the consumer scope remain D1b rather than being erased by the proposition.

**Justification of Theorem 4.1.** The theorem conditions on a fixed complete finite population and invokes a registered level-$\delta_{\text{aud}}$ sequential test for the bad-population null under the actual without-replacement sampling law. By the defining validity property of that test, the probability—over the registered audit randomization—of rejecting a true bad-population null is at most $\delta_{\text{aud}}$. This remains true for an adaptively generated producer history because the producer strategy is absorbed into the already-fixed finite population before the audit randomization begins. Weighted or predictable adaptive draws are allowed only when the chosen confidence-sequence or test-martingale construction is valid for those conditional selection probabilities (Stark, 2019; Shekhar et al., 2023; Chugg et al., 2023). Outcome-label error and population incompleteness are separate assumptions and are therefore attacked empirically rather than hidden inside the theorem.

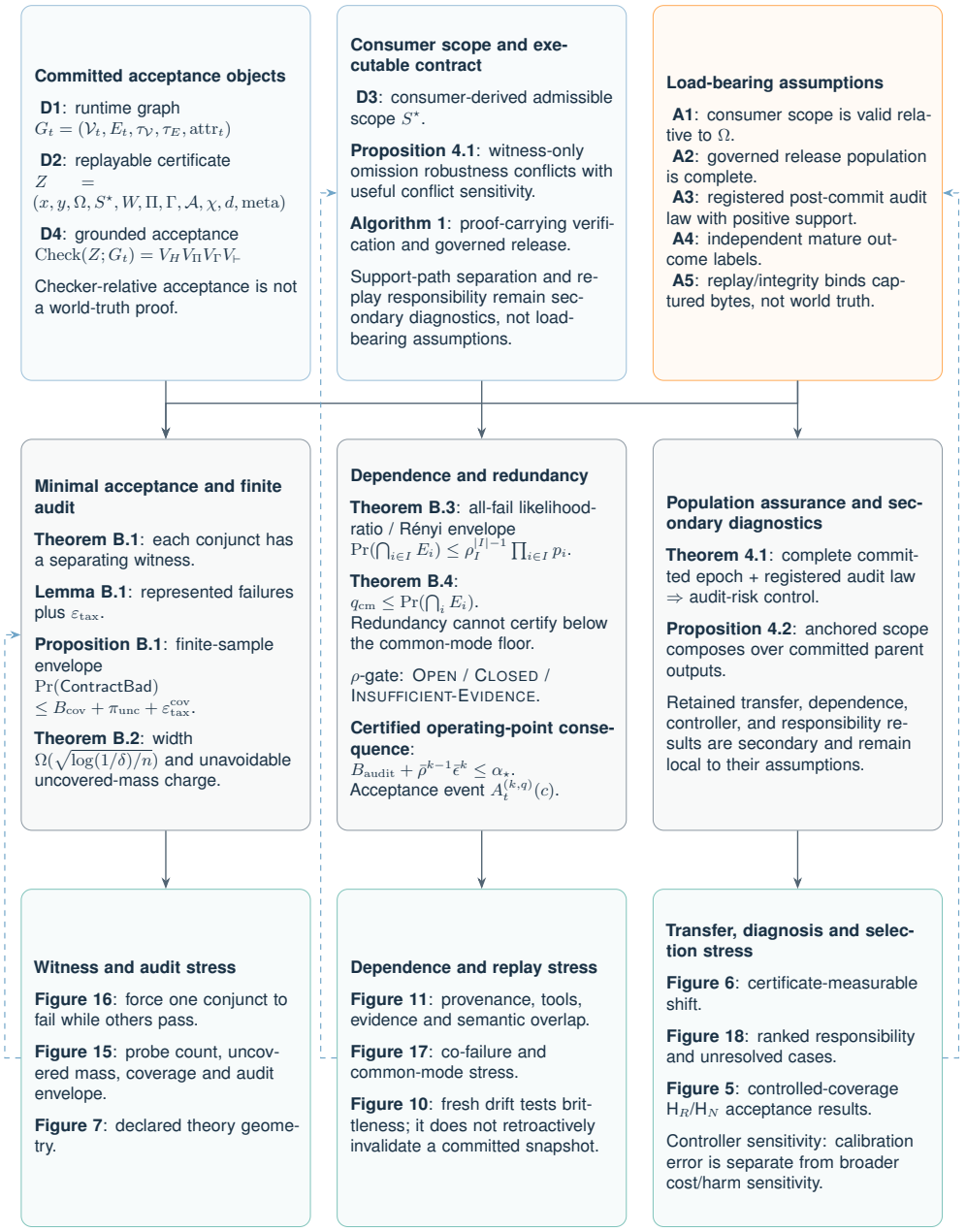
**Compositional scope.** Proposition 4.2 follows by induction over a topological ordering of the committed DAG: once accepted parent outputs are fixed, each child scope is a deterministic/registered function of $\Omega$ and those parents, so a producer witness cannot redefine the child scope. The result is conditional on the committed candidate outputs and does not cover adversarial answer selection.

**Secondary results.** The remainder of this appendix gives exact-contract, audit-envelope, dependence, shift, and selectivity results that support the implementation and secondary analyses. Each holds only under its own local assumptions (Secondary Assumptions S1 and S2), and neither primary contribution depends on them.

### B.3 CORE STATEMENTS AND ACCEPTANCE BOUNDS

The theory concerns a checker-relative acceptance predicate; the five audit channels are a distinct meta-level taxonomy. Fig. 7 and Appendix Fig. 8 link the declared capability/theory structure to the observable patterns used to stress it; Tables 8 and 7 give the full guarantee-role and theorem-to-test maps. Each formal statement below is followed by its proof.

**Irredundant acceptance.**




Solid: formal dependency and test design. Dashed: empirical stress.
Stress tests challenge applicability; they do not prove assumptions.

Figure 8: PROTOCOL. **Theorem-to-test dependency map.** Formal objects and assumptions feed the results they license; empirical stress tests probe those assumptions or applicability conditions. Algorithm 1 operationalizes the contract but is not a proof node, and empirical success does not prove an assumption true.

For $S \subseteq \mathcal{C} = \{H, \Pi, \Gamma, \vdash\}$, define

$$\mathrm{Check}_S = \prod_{j\in S} V_j, \qquad \mathcal{A}_S = \{\mathrm{Check}_S = 1\}. \tag{13}$$

**Definition D6** (Separating witness family)**.** $\mathcal{W}_j$ is separating for $j$ if $V_j = 0$ and all other certificate bits equal one on every member.

**Theorem B.1** (Irredundancy of the four certificate conjuncts)**.** *If $\mathcal{W}_j \neq \varnothing$ for each $j \in \mathcal{C}$, then $\mathcal{A}_{S\cup\{j\}} \subsetneq \mathcal{A}_S$ whenever $j \notin S$; hence no conjunct is entailed by the other three.*

*Proof.* Fix $S \subseteq \mathcal{C}$ and $j \notin S$. For any $w \in \mathcal{W}_j$, all conjuncts in $S$ equal one, so $w \in \mathcal{A}_S$, while $V_j(w) = 0$, so $w \notin \mathcal{A}_{S\cup\{j\}}$. Thus the inclusion is strict. Repeating for every missing coordinate proves irredundancy and non-entailment. The separating-witness evaluation tests whether the implemented checker admits the required witnesses; attempted witness generations, including failures, remain part of the empirical record. □

The witness proves the logical strict inclusion; whether the implemented system exposes such witnesses is an empirical applicability question.

**Finite-sample audit envelope.**

Let ContractBad denote an accepted run violating an in-scope declared obligation, and define the frozen audit-channel set

$$\mathcal{J} = \{\text{int}, \text{rep}, \text{drift}, \text{chk}, \text{cov}\}. \tag{14}$$

**Secondary assumption S1** (Declared audit semantics and taxonomy coverage)**. S1(a)** each declared audit channel is recomputable from the frozen certificate/audit record; **(S1(b))** any in-scope bad mass outside the declared channels is either zero on the target population or carried explicitly as $\varepsilon_{\text{tax}}$.

**Engineering interpretation of Secondary Assumption S1.** S1(a) is an implementation condition: audit-channel semantics, inputs, and recomputation rules are frozen. S1(b) is a substantive target-population coverage condition and is not implied by checker design. A frozen challenge arm may expose counterexamples to S1(b) on its challenge distribution, but cannot establish $\varepsilon_{\text{tax}} = 0$ under deployment. Deployment-like estimation therefore requires the separately declared stratified sampling design. All channel–stratum confidence events share the family-wise budget

$$\sum_{h,j} \delta_{j,h} \le \delta_{\text{conf}}, \tag{15}$$

and uncovered population mass $\mu_{\text{unc}}$ is charged once at the union level. If stratum masses $\mu_h$ are estimated, the resulting statement is conditional on the frozen observed mixture unless their uncertainty is propagated.

**Lemma B.1** (Trace decomposition)**.** *Under S1(a),*

$$\Pr(\mathsf{ContractBad}) = \Pr\left(\mathsf{ContractBad} \wedge \bigcup_j \mathsf{Fail}_j\right) + \varepsilon_{\text{tax}}; \tag{16}$$

*this says nothing about the separate source-truth residual $\varepsilon_{\text{src}}$.*

*Proof.* Let $F = \cup_{j\in\mathcal{J}} \mathsf{Fail}_{j,t}$. The events $\mathsf{ContractBad}_t \cap F$ and $\mathsf{ContractBad}_t \cap F^c$ partition $\mathsf{ContractBad}_t$. Therefore

$$\Pr(\mathsf{ContractBad}_t) = \Pr(\mathsf{ContractBad}_t \cap F) + \Pr(\mathsf{ContractBad}_t \cap F^c), \tag{17}$$

and the second term is $\varepsilon_{\text{tax}}$ by definition. S1(b) is precisely the additional coverage condition that sets this residual to zero on the claimed target population. This decomposition is silent about $\varepsilon_{\text{src}}$, which lies outside ContractBad. □

For covered strata $\mathcal{H}_{\text{cov}}$ with masses $\mu_h$ and simultaneous channel/stratum upper confidence bounds $U_{j,h}(\delta_{j,h})$, define

$$B_{\text{cov}} = \sum_{h\in\mathcal{H}_{\text{cov}}} \mu_h \min\left\{1, \sum_{j\in\mathcal{J}} U_{j,h}\right\}, \quad \varepsilon_{\text{tax}}^{\text{cov}} = \Pr\left(\mathsf{ContractBad}, H \in \mathcal{H}_{\text{cov}}, \bigcap_j \neg\mathsf{Fail}_j\right). \tag{18}$$

**Proposition B.1** (Finite-sample contract-audit envelope)**.** *With simultaneous confidence at least* $1-\delta$*,*

$$\Pr(\mathsf{ContractBad}) \le B_{\text{cov}} + \mu_{\text{unc}} + \varepsilon_{\text{tax}}^{\text{cov}}. \tag{19}$$

*Under S1(b) on covered strata the final term vanishes; uncovered mass is charged exactly once.*

*Proof.* For each covered stratum $h$ and channel $j$, choose a one-sided Bernoulli UCB $U_{j,h}(\delta_{j,h})$ such that

$$\sum_{h\in\mathcal{H}_{\mathrm{cov}}}\sum_{j\in\mathcal{J}}\delta_{j,h}\le\delta. \tag{20}$$

A union bound over the channel–stratum confidence events gives simultaneous validity with probability at least $1-\delta$. Conditional on covered stratum $h$,

$$\Pr\left(\bigcup_{j\in\mathcal{J}}\mathsf{Fail}_{j,t}\mid H=h\right)\le\min\left\{1,\sum_{j\in\mathcal{J}}U_{j,h}(\delta_{j,h})\right\}. \tag{21}$$

Weighting by $\mu_h$ gives the covered contribution $B_{\mathrm{cov}}$. The uncovered region has total mass $\mu_{\mathrm{unc}}$, so its entire bad-event probability is at most that mass and is charged once. Covered bad events represented by no frozen audit channel contribute $\varepsilon_{\mathrm{tax}}^{\mathrm{cov}}$. Hence

$$\Pr(\mathsf{ContractBad}_t)\le B_{\mathrm{cov}}(\delta)+\mu_{\mathrm{unc}}+\varepsilon_{\mathrm{tax}}^{\mathrm{cov}}. \tag{22}$$

Under S1(b) on the covered region the final residual is zero. For one fully covered stratum, Bonferroni allocation across channels recovers the pooled special case. If the $\mu_h$ are estimated, the proof is conditional on the frozen observed mixture unless their uncertainty is separately propagated. □

**Theorem B.2** (Confidence-rate converse and unavoidable uncovered mass)**.** *Uniform Bernoulli intervals have worst-case width $\Omega(\sqrt{\log(1/\delta_{\mathrm{conf}})/n})$; unprobed mass $\mu_{\mathrm{unc}}$ induces worst-case absolute error at least $\mu_{\mathrm{unc}}/2$ and must remain available to a uniformly valid upper envelope. For the range-Hoeffding objective, proportional allocation solves the stated continuous relaxation.*

*Proof.* **(i) Confidence-rate converse.** Let $P_0=\mathrm{Bern}(\frac{1}{2}-\varepsilon)$ and $P_1=\mathrm{Bern}(\frac{1}{2}+\varepsilon)$, $0<\varepsilon\le 1/4$. A direct calculation gives

$$\mathrm{KL}(P_0\|P_1)\le C\varepsilon^2 \tag{23}$$

for a universal constant $C$. For $n$ probes, Bretagnolle–Huber implies that every test has maximal error at least

$$\tfrac{1}{4}\exp(-Cn\varepsilon^2). \tag{24}$$

Choosing

$$\varepsilon=\sqrt{\frac{\log(1/(4\delta))}{Cn}} \tag{25}$$

whenever this is at most $1/4$ makes the two means, separated by $2\varepsilon$, indistinguishable with error at least $\delta$. Therefore every uniformly $(1-\delta)$-valid interval must have worst-case width

$$\Omega\left(\sqrt{\frac{\log(1/\delta_{\mathrm{conf}})}{n}}\right). \tag{26}$$

**(ii) Uncovered mass.** Construct two deployment laws that agree on every probed stratum and set the failure probability on unprobed strata of total mass $\mu_{\mathrm{unc}}$ to zero versus one. Their observed probe laws are identical while their deployment-weighted rates differ by $\mu_{\mathrm{unc}}$. Hence minimax absolute error is at least $\mu_{\mathrm{unc}}/2$, and a uniformly valid upper envelope must allow the full uncovered mass.

**(iii) Range-based allocation.** By Cauchy–Schwarz,

$$\left(\sum_h\mu_h\right)^2\le\left(\sum_h\frac{\mu_h^2}{n_h}\right)\left(\sum_h n_h\right), \tag{27}$$

so

$$\sum_h\frac{\mu_h^2}{n_h}\ge\frac{1}{n}, \tag{28}$$

with equality in the continuous relaxation exactly when $n_h\propto\mu_h$. Integer allocations use the nearest feasible counts. This claim concerns the stated range-based Hoeffding objective, not variance-adaptive Neyman allocation.

**Union slack.** Let $F_j$ be the channel-failure events and $N_F=\sum_j\mathbf{1}_{F_j}$. Then

$$\sum_j P(F_j)-P\left(\bigcup_j F_j\right)=E[(N_F-1)_+] \tag{29}$$

exactly. Union overlap, containment slack, finite-sample/UCB charges, and uncovered mass are distinct quantities. □

**Dependence and common mode.**

**Secondary assumption S2** (Calibrated replay and control)**.** Replay interventions are conditionally i.i.d. with bounded effects; the declared risk estimate has bounded calibration error and the secondary controller objective is Lipschitz in that risk.

**Engineering interpretation of Secondary Assumption S2.** S2 concerns replay sampling and the optional calibrated controller; it does not assume independence among redundant branches. Branch dependence is represented separately by observable joint-to-product quantities. Let $B_i$ denote the event that branch $i$ accepts a false claim under the unified checker. For $I \subseteq [k]$, $|I| \geq 2$,

$$\lambda_I = \frac{\Pr(\cap_{i\in I} B_i)}{\prod_{i\in I} \Pr(B_i)}, \qquad \rho_I = \max\{1, \lambda_I^{1/(|I|-1)}\}. \tag{30}$$

The held-out upper envelope is

$$\widehat{\rho}_{\mathrm{UCB}} = \max_{I\in\mathcal{I}_{q_0}} \max\left\{1, \left(\frac{\widehat{p}_I^{+}}{\prod_{i\in I}\widehat{p}_i^{-}}\right)^{1/(|I|-1)}\right\}, \quad \mathcal{I}_{q_0} = \{I \subseteq [k] : |I| \geq q_0\}, \tag{31}$$

with $\widehat{\rho}_{\mathrm{UCB}} = \infty$ whenever a marginal lower bound is zero. The frozen calibration envelope is

$$\bar{\rho} = \widehat{\rho}_{\mathrm{UCB,cal}}. \tag{32}$$

For runtime component $e$, the seeded replay effect is

$$\Delta_e^{(m)}(Z; G_t) = \mathrm{Check}(Z; G_t) - \mathrm{Check}(Z; G_{t,m}^{\setminus e}), \tag{33}$$

and binary acceptance gives

$$\Delta_e^{(m)}(Z; G_t) \in [-1, 1]. \tag{34}$$

If $Q_a(r)$ is the declared action score as a function of false-accept risk $r$, controller regularity is

$$|Q_a(r) - Q_a(r')| \leq L_{\mathrm{ctrl}}|r - r'| \qquad \forall a \in \mathcal{A}. \tag{35}$$

Stateful tools, logging drift, or distribution shift may invalidate the empirical replay/calibration conditions; the corresponding evaluation then fails closed or remains unresolved rather than being absorbed into the theorem.

**Theorem B.3** (All-fail likelihood ratio and Rényi envelope)**.** *For branch-failure law $P$, independent-marginal law $Q$, and $p_i = \Pr(B_i)$,*

$$\lambda_I = \frac{\Pr(\cap_{i\in I} B_i)}{\prod_{i\in I} p_i} \leq e^{D_\infty(P\|Q)}, \qquad \rho_I = \max\{1, \lambda_I^{1/(|I|-1)}\}, \tag{36}$$

*so*

$$\Pr(\cap_{i\in I} B_i) \leq \rho_I^{|I|-1} \prod_{i\in I} p_i. \tag{37}$$

*Proof.* For $p_i \in (0, 1)$,

$$Q = \bigotimes_i \mathrm{Bern}(p_i) \tag{38}$$

has positive mass at every corner. Hence

$$\lambda_{[k]} = \frac{P(\mathbf{1})}{Q(\mathbf{1})} = \frac{dP}{dQ}(\mathbf{1}) \leq \sup_x \frac{dP}{dQ}(x) = e^{D_\infty(P\|Q)}. \tag{39}$$

Defining

$$\rho_{[k]} = \max\{1, \lambda_{[k]}^{1/(k-1)}\} \tag{40}$$

yields

$$P(\cap_i E_i) \leq \rho_{[k]}^{k-1} \prod_i p_i. \tag{41}$$

The same rooted parameterization is used in Eqs. (45) and (110); a zero marginal lower confidence bound sends the UCB to $\infty$ and closes the extrapolation gate. Full $D_\infty$ over $2^k$ cells remains a diagnostic because sparse worst-cell estimation can be unstable. □

**Theorem B.4** (Common-mode floor). *If shared cause $\mathcal{K} \subseteq \cap_i E_i$ has mass $q_{\mathrm{cm}} > 0$, then*

$$q_{\mathrm{cm}} \le \Pr(\cap_i E_i), \qquad \lambda_{[k]} \ge q_{\mathrm{cm}} \epsilon_{\mathrm{path}}^{-k}, \qquad k^* = \left\lceil \frac{\log q_{\mathrm{cm}}}{\log \epsilon_{\mathrm{path}}} \right\rceil. \tag{42}$$

*Thus redundancy cannot certify below the common-mode floor.*

*Proof.* If $\mathcal{K} \subseteq \cap_i E_i$, monotonicity gives

$$P(\cap_i E_i) \ge q_{\mathrm{cm}}. \tag{43}$$

If each $p_i \le \epsilon_{\mathrm{path}}$, then

$$\lambda_{[k]} = \frac{P(\cap_i E_i)}{\prod_i p_i} \ge q_{\mathrm{cm}} \epsilon_{\mathrm{path}}^{-k}. \tag{44}$$

Thus no redundancy level can certify below $q_{\mathrm{cm}}$. The controlled common-mode subtest checks this implication under its declared shared-cause mechanism; realistic injection stress is reported separately and is not forced to satisfy the theorem's stylized premise. □

The held-out gate uses

$$\widehat{\rho}_{\mathrm{UCB}} = \max_{|I| \ge q_0} \max \left\{ 1, \left( \frac{\widehat{p_I^+}}{\prod_{i \in I} \widehat{p_i^-}} \right)^{1/(|I|-1)} \right\}, \quad \widehat{p_i^-} = 0 \Rightarrow \widehat{\rho}_{\mathrm{UCB}} = \infty. \tag{45}$$

Its state is OPEN, CLOSED, or INSUFFICIENT-EVIDENCE; only OPEN licenses $\rho$-based extrapolation. A direct joint-failure UCB may still describe the exact observed $k$-branch configuration without extrapolation.

**Certificate-measurable shift.**

**Theorem B.5** (Restricted-TV transfer and domain-classifier identity). *For certificate-measurable event $E \in \mathfrak{F}_Z$,*

$$|P_t(E) - P_{\mathrm{cal}}(E)| \le \mathrm{TV}(P_t|_{\mathfrak{F}_Z}, P_{\mathrm{cal}}|_{\mathfrak{F}_Z}) \le \mathrm{TV}(P_t, P_{\mathrm{cal}}). \tag{46}$$

*With equal priors, Bayes balanced accuracy satisfies*

$$2a^* - 1 = \mathrm{TV}|_{\mathfrak{F}_Z}; \tag{47}$$

*a realizable finite classifier supplies only a lower bound.*

*Proof.* For $\mathcal{E} \in \mathfrak{F}_Z$, total variation is the supremum over measurable events:

$$|P_t(\mathcal{E}) - P_{\mathrm{cal}}(\mathcal{E})| \le \sup_{B \in \mathfrak{F}_Z} |P_t(B) - P_{\mathrm{cal}}(B)| \le \sup_{B \in \mathfrak{F}} |P_t(B) - P_{\mathrm{cal}}(B)|. \tag{48}$$

For a certificate-summary classifier $g$ under equal priors,

$$2a(g) - 1 = P_t(g = 1) - P_{\mathrm{cal}}(g = 1) \le \mathrm{TV}|_{\mathfrak{F}_Z}, \tag{49}$$

with equality for the Bayes/Neyman–Pearson region. Thus $2a - 1$ is a lower-bound alarm for a realizable classifier, not an upper transfer penalty. In particular, semantic falsehood need not be $\mathfrak{F}_Z$-measurable because source truth may remain outside the certificate. □

Hence

$$P_t(E) \le P_{\mathrm{cal}}(E) + \mathrm{TV}(P_t|_{\mathfrak{F}_Z}, P_{\mathrm{cal}}|_{\mathfrak{F}_Z}) \le P_{\mathrm{cal}}(E) + \overline{D}_t \tag{50}$$

only when $\overline{D}_t$ is independently justified. The practical shift alarm is

$$D_{\mathrm{alarm}} = \max(0, 2a_{\mathrm{LCB}} - 1). \tag{51}$$

It is an alarm, not an upper semantic-risk certificate.

**Attribution and selection.**

**Secondary redundancy/responsibility/controller consequence.** Under Secondary Assumption S2, unanimity acceptance satisfies

$$\Pr(A_t^{(k,k)}(c) = 1, c \text{ false}) \le \rho^{k-1} \prod_i \Pr(\mathsf{E}_i), \tag{52}$$

and hence is at most

$$\rho^{k-1} \epsilon_{\mathrm{path}}^k \tag{53}$$

when each branch risk is at most $\epsilon_{\text{path}}$. With $M$ bounded replay interventions over $\mathcal{U}$,

$$\Pr\Big(\max_e |\widehat{\mathsf{Resp}}(e) - \mathsf{Resp}(e)| \geq \epsilon\Big) \leq 2|\mathcal{U}|e^{-M\epsilon^2/2}; \tag{54}$$

the unique-top ranking bound in Eq. (63) follows from the same replay regularity. Under the fixed declared cost model only, replacing true risk by a uniformly $\epsilon_{\text{cal}}$-accurate estimate incurs at most $2L_{\text{ctrl}}\epsilon_{\text{cal}}$ one-step regret.

*Proof.* We prove the four claims in sequence.

For redundancy, if $A_t^{(k,k)}(c) = 1$ and $c$ is false, then every branch-specific unified certificate accepts a false claim. Therefore

$$\{A_t^{(k,k)}(c) = 1,\ c \text{ false}\} \subseteq \bigcap_{i=1}^{k} \mathsf{E}_i. \tag{55}$$

By Theorem B.3,

$$\Pr(A_t^{(k,k)}(c) = 1,\ c \text{ false}) \leq \rho^{k-1} \prod_{i=1}^{k} \Pr(\mathsf{E}_i). \tag{56}$$

If

$$\Pr(\mathsf{E}_i) \leq \epsilon_{\text{path}} \tag{57}$$

for every branch, then

$$\Pr(A_t^{(k,k)}(c) = 1,\ c \text{ false}) \leq \rho^{k-1}\epsilon_{\text{path}}^k. \tag{58}$$

For diagnosis, define the seeded replay effect

$$Y_m(e) := \text{Check}(Z; G_t) - \text{Check}(Z; G_{t,m}^{\setminus e}). \tag{59}$$

Since each check value is binary,

$$Y_m(e) \in [-1, 1]. \tag{60}$$

Under Secondary Assumption S2, the replay effects are conditionally i.i.d. given $Z, G_t$, with mean $\mathsf{Resp}(e)$. Hoeffding's two-sided inequality therefore gives

$$\Pr\Big(|\widehat{\mathsf{Resp}}(e) - \mathsf{Resp}(e)| \geq \varepsilon_{\text{resp}}\Big) \leq 2\exp\left(-\frac{M\varepsilon_{\text{resp}}^2}{2}\right). \tag{61}$$

A union bound over the finite component set $\mathcal{U}$ gives

$$\Pr\left(\max_{e\in\mathcal{U}} |\widehat{\mathsf{Resp}}(e) - \mathsf{Resp}(e)| \geq \varepsilon_{\text{resp}}\right) \leq 2|\mathcal{U}|\exp\left(-\frac{M\varepsilon_{\text{resp}}^2}{2}\right). \tag{62}$$

For ranking, let $e^\star$ be the unique maximizer of population responsibility and let $\gamma_{\text{resp}} > 0$ be its gap over the runner-up. For any $\tau \in (0, \gamma_{\text{resp}})$, a ranking error requires the top component to be underestimated by at least $\tau$, or some competitor to be overestimated by at least $\gamma_{\text{resp}} - \tau$. Hence one-sided Hoeffding bounds and a union bound give

$$\Pr\left(\arg\max_{e\in\mathcal{U}} \widehat{\mathsf{Resp}}(e) \neq e^\star\right) \leq \exp\left(-\frac{M\tau^2}{2}\right) + (|\mathcal{U}| - 1)\exp\left(-\frac{M(\gamma_{\text{resp}} - \tau)^2}{2}\right). \tag{63}$$

The detailed margin-split derivation and the symmetric half-margin specialization are given in §D.8.

For control, the affine score model in §D.8 gives, whenever

$$|\widehat{r} - r| \leq \epsilon_{\text{cal}}, \tag{64}$$

$$Q_{\widehat{a}}(r) - Q_{a^\star}(r) \leq 2L_{\text{ctrl}}\epsilon_{\text{cal}}. \tag{65}$$

Together these establish the stated redundancy, responsibility, ranking, and controller consequences. □

Privacy is outside the scope of this paper (Appendix D.8).

Combining the secondary results gives the auxiliary boundary

$$\Pr(\mathsf{Bad}) \leq B_{\text{cov}} + \mu_{\text{unc}} + \varepsilon_{\text{tax}}^{\text{cov}} + \varepsilon_{\text{src}}, \qquad q_{\text{cm}} \leq \Pr(\cap_i E_i) \leq \rho^{k-1}\epsilon_{\text{path}}^k. \tag{66}$$

Appendix Fig. 12 summarizes the boundary used throughout the empirical claims: online acceptance is recomputed from committed artifacts, whereas drift, uncovered mass, checker/taxonomy limitations, and source-truth residuals remain monitored or external quantities.

### B.4 Certified operating-point corollary

Let $B_{\mathrm{audit}}$ denote the finite-sample audit envelope from Proposition B.1, and let $\bar{\rho}, \bar{\epsilon}$ be valid upper bounds on $\rho$ and per-path false-accept risk. If

$$B_{\mathrm{audit}} + \bar{\rho}^{k-1}\bar{\epsilon}^{k} \le \alpha_\star, \tag{67}$$

then the declared conservative bad-accept operating budget is at most $\alpha_\star$. Therefore the smallest redundancy level satisfying the target is

$$k_\star = \min\left\{k \in \mathbb{N} : B_{\mathrm{audit}} + \bar{\rho}^{k-1}\bar{\epsilon}^{k} \le \alpha_\star\right\}. \tag{68}$$

A cost-aware operating point is

$$k^\dagger \in \arg\min_{k\ge 1}\left\{\bar{C}(k) + \lambda L_{\max}\left(B_{\mathrm{audit}} + \bar{\rho}^{k-1}\bar{\epsilon}^{k}\right)\right\}. \tag{69}$$

### B.5 Interpretive boundaries

The formal results constrain different failure mechanisms and should not be collapsed into one scalar guarantee. Audit coverage controls what can be certified from observed audit channels and explicitly charges uncovered mass; common-mode dependence limits what redundancy can buy; restricted-TV transfer applies only to certificate-measurable events;. A learned domain classifier therefore acts as a shift alarm rather than a semantic-risk certificate, and replay responsibility is an acceptance-sensitivity diagnostic under the committed intervention model rather than a claim of real-world causal blame.

The four online certificate conjuncts are likewise not claimed to be individually novel. The formal claim is irredundancy of their conjunction under separating witnesses. Failure to produce a channel-exclusive witness in the registered witness protocol is evidence against the empirical applicability of Theorem B.1, not something hidden by the theorem statement. The audit converse similarly states that weak coverage cannot be repaired by post-hoc analysis, while Theorem B.4 states that shared failure mass cannot be defeated merely by increasing $k$.

### B.6 Ranking recovery: explicit hard family

The upper bound is proved in §B.3. For the converse, let

$$p_+ = \tfrac{1}{2} + \gamma/2, \qquad p_- = \tfrac{1}{2} - \gamma/2, \qquad 0 < \gamma \le 1/2. \tag{70}$$

Under instance $\mathcal{I}_0$, replay effects for components $(e_1, e_2)$ are independent

$$(\mathrm{Bern}(p_+), \mathrm{Bern}(p_-)); \tag{71}$$

under $\mathcal{I}_1$ the two laws are swapped. The top component changes and the margin is $\gamma$. For $M$ samples per component, the observation-law KL is at most $cM\gamma^2$ for a universal $c$. Bretagnolle–Huber therefore gives

$$\inf_{\widehat{e}} \max_{b\in\{0,1\}} P_{\mathcal{I}_b}(\widehat{e} \ne e_b^\star) \ge \tfrac{1}{4} e^{-cM\gamma^2}. \tag{72}$$

This is a minimax existence lower bound; ordinary benchmark performance cannot refute it. A synthetic hard-family sanity test may verify implementation/scaling, while the witness-search experiments evaluate practical ranking and UNRESOLVED rates. The exact upper-bound minimizer is the numerical root of

$$\tau e^{-M\tau^2/2} = (|\mathcal{U}| - 1)(\gamma - \tau)e^{-M(\gamma-\tau)^2/2}; \tag{73}$$

the symmetric $\tau = \gamma/2$ specialization is only approximate when the multiplicity term is asymmetric.

### B.7 Theory-to-evidence map

Fig. 8 maps each formal object or assumption to the empirical stress test that can challenge it. Empirical nodes test assumptions or applicability; they do not prove the formal statements.

Table 7: Theory-to-falsifier map for the release-assurance claims.

| Claim | Required condition | Direct stress | Failure response |
|---|---|---|---|
| Curation invariance | consumer-derived $S^\star$ | oracle selective-view attack | selective-view claim secondary |
| Population assurance | complete committed epoch | bypass/omission suite | no population claim |
| RLA validity | valid sampling law | wrong-law / zero-support ablation | no certificate |
| Label validity | mature outcome authority | noise/delay ablation | qualify guarantee |
| Replay stability | declared replay class | cross-host/runtime replay | indeterminate band |

### B.8 End-to-end evaluation scope and acceptance criteria

Numeric fields are populated only by generated artifacts under the retained evaluation contracts. The released evidence resolves headline values to the 616-row model–dataset–method matrix and the named aggregate contracts. Per-item records and a complete preregistered execution manifest are not present in this package and are therefore not claimed.

The evaluation order resolves backend identity and canonical metrics before headline aggregation; auditor invariance, native-scope baselines, checker characterization, and budget-matched comparisons are evaluated before any superiority claim is reported. Cost telemetry is captured for every model-bearing invocation. Robustness analyses report only claims supported by their corresponding artifacts. The experiment bundle contains retained settings, aggregate and cell-level contracts, generated `CSV/TeX/vector` artifacts, available manifests, consistency checks, and SHA-256 hashes. Missing per-item evidence remains explicit. Failed or incomplete evidence weakens or removes the corresponding claim rather than substituting an unsupported value.

## C Runtime and Certificate Specification

This appendix specifies the runtime objects, consumer scope, release population, replay, contracts, grounding certificates, witness minimization, support separation, checker interface, algorithm, and guarantee boundary used by the method in Section 3.

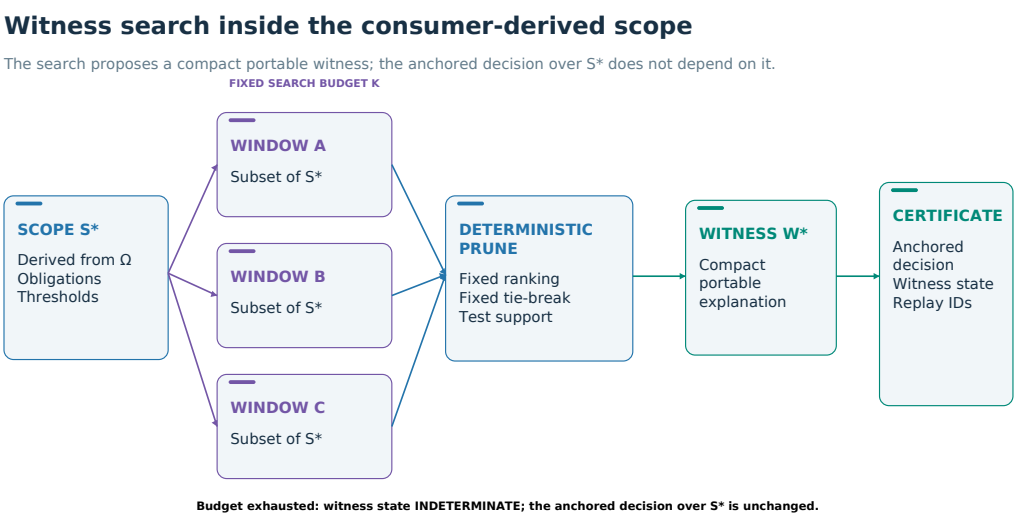



Figure 9: Protocol. **Witness search inside the consumer-derived scope.** Candidate evidence windows drawn from $S^\star$ receive committed checker scores, and deterministic pruning selects a compact portable witness $W^\star$ (Eq. (6)). Budget exhaustion leaves the witness state Indeterminate and does not change the anchored decision over $S^\star$. The schematic reports procedure, not measured performance.

### C.1 Consumer-derived scope and selective-view threat

The certificate machinery separates the producer's portable witness from the consumer's admissible decision scope. The consumer contract $\Omega$ fixes the derivation procedure, corpus/environment snapshot, required observation families, and any bounded expansion rules. The resulting $S^\star(x)$ is therefore a

property of the consumer contract and input rather than of a producer-selected citation set. A witness $W \subseteq S^\star$ may be minimal and portable, but full-scope contradiction/sufficiency checks determine release whenever the contract declares them applicable.

The registered curation adversary may omit or reorder only the evidence or runtime observations permitted by the attack contract. The fixed-candidate theory does not cover best-of-$n$ answer selection, compromised consumer derivation, or evidence outside the declared environment. Those are either separate empirical stresses or outside the guarantee.

### C.2 Release population and audit objects

A population-level assurance statement requires a manifest that corresponds to the governed release stream. In the paper's controlled AgentDojo implementation, consequential tool wrappers verify a release receipt before execution. This makes bypass measurable rather than assuming that a Merkle root alone proves completeness. Real third-party sinks need not understand PCG receipts; production capability/credential custody is therefore a deployment pattern rather than a paper result.

For epoch $\mathcal{E}$, the audit manifest binds the release IDs, root/consistency data, population size, audit design, $\tau_{\mathrm{fail}}$, $\delta_{\mathrm{aud}}$, positive-support condition, and outcome-authority version. Audit randomness is generated under the frozen design after population commitment. Sequential sampling may adapt to allowed past history when the selected finite-population test supports that conditional law. The exact probabilities used at each draw are part of the audit record.

### C.3 Outcome authority and maturation boundary

Outcome authority is evaluation-only and independent of online acceptance. Factual and QA tasks use benchmark-native/reference authority when appropriate; BFCL, Mind2Web, and AgentDojo use their registered task/security evaluators. Human adjudication, if required, uses a frozen rubric and records disagreement.

For deployed workflows, a nonfailure label is not final merely because no failure has yet appeared. A workflow therefore declares an outcome horizon and maturation rule before the epoch begins. The paper evaluates this issue only through a controlled delayed-label ablation; a full delayed-feedback service is outside the headline empirical claim.

### C.4 Worked AgentDojo banking certificate

The running example in Section 2 is instantiated on an AgentDojo banking task in which the user asks the agent to inspect an incoming payment and return only the excess amount. A malicious instruction may arrive through untrusted tool content and attempt to redirect money. The worked record below is a *schema example*, not an experimental result.

| Object / channel | Bound state in the worked certificate |
|---|---|
| Candidate | proposed transfer amount/recipient + canonical candidate hash |
| Consumer scope $S^\star$ | user request, relevant incoming transaction, recipient identity, applicable banking policy, tool preconditions |
| Producer witness $W$ | compact subset offered by the producer; cannot redefine $S^\star$ |
| Integrity / replay $V_H, V_{\Pi}$ | captured request/tool bytes, stable tool identity, policy/checker versions, replay class |
| Semantic support $V_{\vdash}$ | whether the proposed transfer is supported by the user instruction under anchored scope |
| Policy/security $V_{\Gamma}$ | whether the recipient/action is allowed and is not the injected attacker goal |
| Decision | Release, RejectUnsupported, WithholdPolicy, or EscalateIndeterminate |
| Population binding | if released: ReleaseID, sequence number, DecisionCertificate hash, inclusion receipt, epoch ID |
| Independent outcome | benchmark utility and security predicates, unavailable to same-item online acceptance |
| Epoch audit | later draw probability, matured label, sequential audit state, and EpochRiskCertificate / refusal |

This example makes the guarantee boundary visible: a spoofed tool payload can be byte-for-byte committed and replayed while still being false or malicious; such a case may pass integrity and fail independent utility/security outcome authority.

### C.5 STUDY-DEFINED STRUCTURE-MATCHED CONTROLS

CMVO and SignalMatchedFusion are study-defined controls rather than externally published methods, and therefore do not carry external citations. They are included to separate gains from the proof-carrying acceptance contract from gains that could arise simply from adding verifier signals or additional matched inference structure. Both controls are evaluated on the same frozen model–dataset population and under the same canonical metric and resource-accounting pipeline as PROTOCOL.

**CMVO.** CMVO is the frozen structure-matched control defined in the experimental method registry. It preserves the corresponding matched evaluation stack while omitting the complete PROTOCOL certificate-and-replay acceptance contract. Its purpose is to test whether a comparably structured verification pipeline can reproduce the headline safety–coverage behavior without the full proof-carrying acceptance mechanism.

**SignalMatchedFusion.** SignalMatchedFusion is the frozen signal-matched control defined in the experimental method registry. It combines the available acceptance-relevant signals under the matched evaluation and resource-accounting protocol without requiring the complete PROTOCOL certificate to be constructed and independently replayed. It therefore tests whether the observed benefit can be explained by combining the same classes of signals rather than by the certificate contract itself.

These controls are not treated as external state-of-the-art systems. Their role is controlled falsification: NoCert asks what happens when certificate acceptance is removed, CMVO and SignalMatchedFusion test progressively stronger matched controls, and the published/native-scope methods test externally established capabilities where their released interfaces are applicable. All reported comparisons retain the same applicability discipline; N/A is never converted to a numerical outcome.

### C.6 RUNTIME GRAPH DESIGN AND TYPING

Definition D1 models an agent run by a single typed graph $G_t$. Rather than an abstract bookkeeping device, $G_t$ is constructed deterministically from the event stream emitted by the agent runtime. Let

$$\mathfrak{L}_t = (o_1, \ldots, o_N) \tag{74}$$

denote the ordered log of events up to time $t$, where each event records a timestamp, run identifier, parent step identifier, operation type, input artifacts, output artifacts, tool or model version, schema identifier, policy identifier, and deterministic replay metadata. The graph construction is a deterministic map

$$G_t = \mathsf{BuildGraph}(\mathfrak{L}_t; \mathrm{canon}, H), \tag{75}$$

where canon is the canonicalization routine applied before hashing and $H$ is the commitment map. Thus two auditors holding the same logs, canonicalization rules, and hash function reconstruct the identical runtime graph.

**Node construction.** Each committed artifact or runtime operation becomes a typed node under the partition

$$\mathcal{V}_t = \mathcal{V}_t^{\mathrm{src}} \cup \mathcal{V}_t^{\mathrm{tool}} \cup \mathcal{V}_t^{\mathrm{schema}} \cup \mathcal{V}_t^{\mathrm{memory}} \cup \mathcal{V}_t^{\mathrm{policy}} \cup \mathcal{V}_t^{\mathrm{deleg}} \cup \mathcal{V}_t^{\mathrm{claim}} \cup \mathcal{V}_t^{\mathrm{action}}, \tag{76}$$

where the union need not be disjoint if an implementation uses multi-typed nodes. Source nodes store snapshotted evidence, retrieved passages, table records, tool responses, or environment states. Tool nodes record function, API, MCP, retriever, parser, or checker invocations. Schema nodes store input/output validators. Memory nodes store reads or writes across agent memory. Policy nodes store guardrail decisions. Delegation nodes store controller-to-subagent handoffs. Claim and action nodes store emitted claims and candidate decisions.

**Edge construction.** Edges encode operational dependencies required for verification, replay, and responsibility masking:

$$\begin{aligned} \tau_E(e) \in \{&\mathrm{RetrievedFrom}, \mathrm{ProducedBy}, \mathrm{InvokedWith}, \mathrm{ParsedTo}, \\ &\mathrm{AlignedTo}, \mathrm{ValidatedBy}, \mathrm{ReadsMemory}, \mathrm{WritesMemory}, \\ &\mathrm{GatedBy}, \mathrm{DelegatesTo}, \mathrm{Supports}, \mathrm{Emits}\}. \end{aligned} \tag{77}$$

When a retriever returns a passage, $\mathrm{RetrievedFrom}$ and $\mathrm{ProducedBy}$ edges connect the query node to the source node. Extracted spans induce $\mathrm{ParsedTo}$ edges; span alignment produces $\mathrm{AlignedTo}$ or $\mathrm{Supports}$ edges. Tool calls validated against contracts add $\mathrm{ValidatedBy}$ or $\mathrm{GatedBy}$ edges. Controller invocations of subagents add $\mathrm{DelegatesTo}$ edges. These edges define the exact components masked by the replay-interventional responsibility diagnostic in Appendix B.

**Edge attributes and weights.** The theory requires only typed, replayable edges, but implementations attach structured telemetry for diagnostics, pruning, and cost-aware selection:

$$\mathrm{attr}_t(e) = (\mathrm{ts}, \mathrm{run_id}, \mathrm{step_id}, \mathrm{op}, \mathrm{version}, \mathrm{latency}, \mathrm{tokens}, \mathrm{score}, \mathrm{cost}, \mathrm{seed}, \mathrm{hash}). \tag{78}$$

When scalar prioritization is required during certifying search, edges admit a deterministic projection:

$$w_t(e) = \alpha_{\mathrm{lat}}\mathrm{latency}(e) + \alpha_{\mathrm{tok}}\mathrm{tokens}(e) + \alpha_{\mathrm{cost}}\mathrm{cost}(e) - \alpha_{\mathrm{rel}}\mathrm{score}(e), \tag{79}$$

with fixed coefficients recorded in $\mathrm{meta}$. These weights do not enter the acceptance predicate; acceptance is governed solely by the applicable checker conjuncts in Eq. (4).

**Graph operations.** The graph supports three primary operations:

$$\mathrm{Replay}(Z, G_t), \qquad \mathrm{Mask}(G_t, e), \qquad \mathrm{Sep}_{\delta,\kappa}(\pi_i, \pi_j). \tag{80}$$

Here $\mathrm{Replay}$ reconstructs the verifier-consumable artifact from committed nodes and logged transformations, $\mathrm{Mask}$ creates $G_t^{\setminus e}$ for responsibility estimation, and $\mathrm{Sep}_{\delta,\kappa}$ evaluates provenance, tool overlap, and content overlap across redundant support paths.

**Discrete tensor representation.** To interface with standard graph algorithms and structural analysis, $G_t$ admits a standard discrete tuple representation:

$$G_t \equiv (\mathrm{edge_index}, \mathrm{node_type}, \mathrm{edge_type}, \mathrm{node_attr}, \mathrm{edge_attr}). \tag{81}$$

The checker accesses attribute tables for hashes, replay parameters, and contract identifiers; the diagnostic module masks components over $\mathrm{edge_index}$; and the redundancy gate extracts support paths from the same topology.

### C.7 Replay and committed artifacts

Evidentiary integrity requires that accepted claims bind to immutable, content-addressed payloads rather than mutable external references. Replayable pipelines guarantee that the verifier evaluates the exact computational derivation rather than an ungrounded string. Concretely, a replayable pipeline is a logged transformation sequence

$$\Pi = (\phi_1, \ldots, \phi_m), \tag{82}$$

accompanied by replay metadata $\mathrm{meta}$ sufficient to re-execute every stage deterministically.

**Replay stages.** We decompose $\Pi$ into explicit operational phases:

$$\Pi = \Pi_{\mathrm{retrieve}} \circ \Pi_{\mathrm{parse}} \circ \Pi_{\mathrm{align}} \circ \Pi_{\mathrm{checkprep}}. \tag{83}$$

Even when retrieval occurs dynamically, the certificate references retrieval outputs as committed source nodes $x(v)$. Consequently, the consumer-side checker replays from committed snapshots without external network calls.

**Replay metadata contract.** For complete determinism, $\mathrm{meta}$ records:

- **Version pins:** parser, model snapshot, tokenizer, and library versions affecting determinism.
- **Parameters:** query strings, top-$k$ cutoffs, normalization rules, and alignment thresholds.
- **Seeds and randomness:** stochastic steps are seeded and recorded, or isolated and explicitly averaged.
- **I/O boundaries:** external API responses affecting $y$ are committed as source nodes $x(v)$.
- **Canonicalization:** exact text normalization applied before checking (Unicode normalization, whitespace stripping, locale-invariant formatting).

**Measuring environment drift.** Snapshot replay is distinct from fresh execution. Snapshot replay evaluates committed nodes and recorded tool outputs; fresh execution reissues external queries to measure mutable environment changes. The drift indicator is

$$D(Z) = \mathbf{1}[\Pi_{\mathrm{fresh}}(\mathrm{meta}) \not\equiv \Pi_{\mathrm{snapshot}}(\{x(v) : v \in S\}; \mathrm{meta})], \tag{84}$$

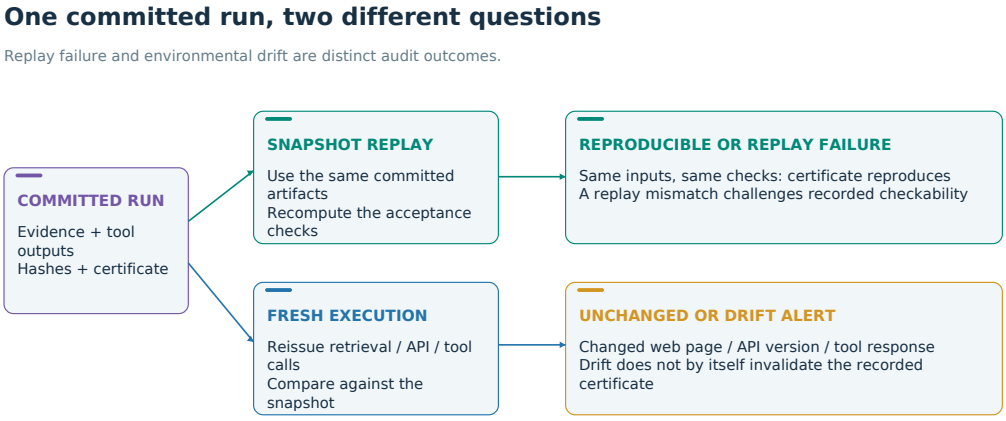



Figure 10: PROTOCOL. **Snapshot replay versus fresh-environment drift.** Snapshot replay reconstructs committed evidence, hashes, and tool outputs to test reproducibility; fresh execution reissues external interactions and is monitored as DriftFail. The two answer different questions and are not merged into one acceptance bit.

evaluated under identical canonicalization. The empirical drift rate across $n_{\mathrm{drift}}$ audit runs is

$$\widehat{\beta}_{\mathrm{drift}} = \frac{1}{n_{\mathrm{drift}}} \sum_{\ell=1}^{n_{\mathrm{drift}}} D(Z_\ell). \tag{85}$$

Drift is monitored as an independent audit channel (DriftFail) and is never conflated with online snapshot-replay failure ($V_\Pi = 0$).

### C.8 EXECUTION CONTRACTS

Execution contracts $\Gamma$ formalize runtime constraints—tool allow-lists, structured I/O schemas, memory-access policies, delegation boundaries, and safety guardrails—as externally auditable predicate sets. In the runtime, compliance is evaluated deterministically by

$$\mathrm{Check}_{\mathrm{exec}}(\Gamma; \mathrm{meta}) \in \{0, 1\}, \tag{86}$$

where meta supplies the execution log, schema outputs, tool parameters, and policy traces. Violation of any applicable contract clause fails closed through $V_\Gamma = 0$.

### C.9 GROUNDING CERTIFICATES

Definition D2 introduces the unified grounding certificate $Z$. While evaluated jointly during acceptance, $Z$ factors logically into claim-side and execution-side sub-certificates.

The claim-side grounding sub-certificate is

$$Z^{\mathrm{clm}} = (c, S, \Pi, \mathcal{A}, \chi, \mathrm{meta}), \tag{87}$$

certifying cryptographic integrity, deterministic replay, and semantic entailment for emitted claim $c$.

The execution-side contract sub-certificate is

$$Z^{\mathrm{exe}} = (\Pi, \Gamma, \mathcal{A}, \chi, \mathrm{meta}), \tag{88}$$

certifying schema validity, tool permissions, memory policies, and verifier-context isolation.

The unified certificate binds both components into one auditable object:

$$Z = (c, S, \Pi, \Gamma, \mathcal{A}, \chi, \mathrm{meta}), \tag{89}$$

where $\mathcal{A} \subseteq \{H, \Pi, \Gamma, \vdash\}$ is the declared applicability set and $\chi$ is the pinned checker bundle.

### C.10 CERTIFICATE MINIMIZATION AND MINIMAL EVIDENCE

Minimal evidence is defined on the claim side of the certificate: holding contract $\Gamma$, applicability $\mathcal{A}$, and metadata meta fixed, the evidence set is pruned to retain only support necessary for claim-side

validity. Starting from initial support $S_0 \subseteq \mathcal{V}_t^{\mathrm{src}}$, a set $S \subseteq S_0$ is one-step locally minimal if

$$\begin{aligned} &\mathrm{Check}_{\mathrm{clm}}((c, S, \Pi, \Gamma, \mathcal{A}, \chi, \mathrm{meta}); G_t) = 1, \\ \forall v \in S: \quad &\mathrm{Check}_{\mathrm{clm}}((c, S \setminus \{v\}, \Pi, \Gamma, \mathcal{A}, \chi, \mathrm{meta}); G_t) = 0. \end{aligned} \tag{90}$$

Local minimality ensures that no single evidence item can be dropped without invalidating claim verification; it is computationally lighter than exhaustive subset search.

The practical pruning operator performs greedy backward elimination under a deterministic ordering $\sigma$:

$$\begin{aligned} &S \leftarrow S_0, \\ \text{for } v \in \sigma(S): \quad &\mathrm{Check}_{\mathrm{clm}}((c, S \setminus \{v\}, \Pi, \Gamma, \mathcal{A}, \chi, \mathrm{meta}); G_t) = 1 \implies S \leftarrow S \setminus \{v\}. \end{aligned} \tag{91}$$

To make subset selection deterministic when multiple minimal sets exist, $\sigma$ sorts candidates by cost, replay responsibility, and cryptographic hash:

$$\sigma(S) = \mathrm{sort}_{v \in S}\big(\mathrm{Cost}(v),\ -\widehat{\mathsf{Resp}}(v),\ \mathrm{hash}(v)\big). \tag{92}$$

When evidence cost is explicit, the ideal support solves the combinatorial objective

$$\min_{S' \subseteq S_0} \sum_{v \in S'} w(v) \quad \text{s.t.} \quad \mathrm{Check}_{\mathrm{clm}}((c, S', \Pi, \Gamma, \mathcal{A}, \chi, \mathrm{meta}); G_t) = 1, \tag{93}$$

for which Eq. (91) serves as the greedy polynomial-time approximation.

Letting $T_H$ denote hashing cost per node, $T_\Pi(|S|)$ replay cost, and $T_\vdash$ entailment cost, a single claim-side evaluation costs

$$T_{\mathrm{check}} = O\big(|S|\, T_H + T_\Pi(|S|) + T_\vdash\big). \tag{94}$$

A single-pass greedy pruning sweep over $S_0$ incurs cost

$$O\big(|S_0|\, T_{\mathrm{check}}\big), \tag{95}$$

with total claim-checker invocations bounded linearly:

$$\#\mathrm{Check}_{\mathrm{clm}}\text{-calls} \le 2|S_0|, \tag{96}$$

accounting for one backward pass and one final verification pass.

**Scaling with chain depth and redundancy.** Across $k$ redundant certificates with chain depth $d$, $M$ masked replay interventions over component set $\mathcal{U}$, step-validation cost $T_{\mathrm{step}}$, replay cost $T_{\mathrm{replay}}$, and controller cost $T_{\mathrm{ctrl}}$, total runtime scales as

$$T_{\mathrm{PCG}} = O(k\, d\, T_{\mathrm{step}} + k\, |S_0|\, T_{\mathrm{check}} + M|\mathcal{U}|\, T_{\mathrm{replay}} + T_{\mathrm{ctrl}})\,. \tag{97}$$

Responsibility attribution ($M|\mathcal{U}|T_{\mathrm{replay}}$) and risk control ($T_{\mathrm{ctrl}}$) operate as optional sidecars and do not increase the online acceptance path.

### C.11 Support-path separation and delegation-aware responsibility

**Definition D7** (($\delta_{\mathrm{sep}}, \kappa_{\mathrm{sep}}, \zeta_{\mathrm{sep}}$)-separated support paths)**.** Two support paths are operationally separated when their committed evidence, runtime transformations, and checker dependencies satisfy the registered overlap/dependence constraints. This remains a secondary diagnostic for redundancy and composition rather than the headline guarantee.

Definition D7 requires three checkable conditions for support-path separation: distinct provenance, bounded shared tool execution, and bounded span overlap.

**Provenance classes.** Each source node $v \in \mathcal{V}_t^{\mathrm{src}}$ carries a structured provenance tuple:

$$\mathrm{Prov}(v) := \big(\mathrm{AuthorityID}(v), \mathrm{PublisherID}(v), \mathrm{Domain}(v)\big), \tag{98}$$

where AuthorityID denotes a registered authority or dataset body, PublisherID is a normalized publisher, and Domain is the source eTLD+1. A support path $\pi$ inherits the provenance of its terminal evidence node:

$$\mathrm{Prov}(\pi) := \mathrm{Prov}(v_\pi). \tag{99}$$

Strict provenance separation requires disjoint authority or domain identifiers.

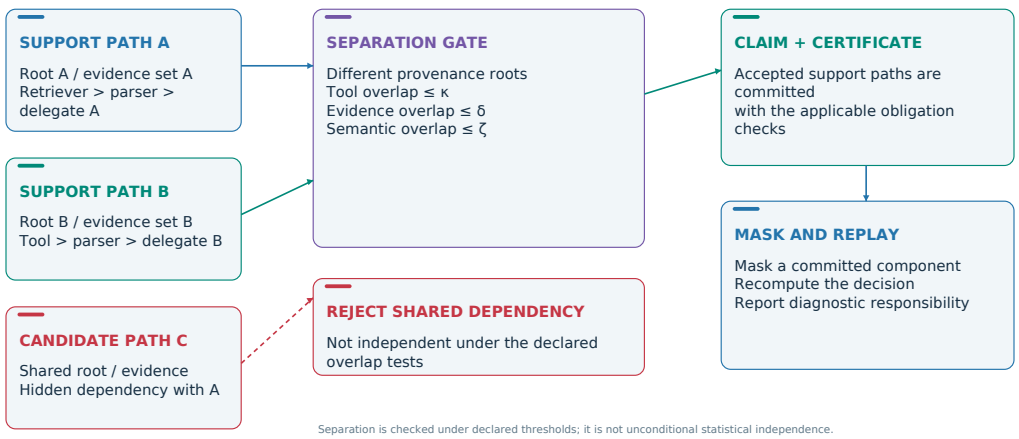



Figure 11: PROTOCOL. **Separated support paths and delegation-aware diagnosis.** Candidate branches expose provenance roots, evidence, tools/parsers, and delegates. The gate checks declared overlap/separation constraints before redundancy is credited; replay-interventional responsibility is then computed only over the committed graph.

**Shared tool and transformation overlap.** Letting $\mathrm{Tools}(\pi)$ denote the multiset of tool, parser, schema, or subagent identifiers along $\pi$, shared operational overlap is

$$\mathrm{ToolOverlap}(\pi_i, \pi_j) := \big|\mathrm{Tools}(\pi_i) \cap \mathrm{Tools}(\pi_j)\big|. \tag{100}$$

The separation condition in Definition D7 enforces $\mathrm{ToolOverlap}(\pi_i, \pi_j) < \kappa$.

**Replayable span overlap score.** For canonical cited span $\mathrm{Span}(v, c)$ aligned to claim $c$, deterministic token shingles are extracted via

$$\mathcal{S}(v, c) := \mathrm{Shingle}_m(\mathrm{Canon}(\mathrm{Span}(v, c))). \tag{101}$$

Between paths $\pi_i, \pi_j$, lexical overlap is computed as the Jaccard index:

$$\mathrm{Overlap}(\pi_i, \pi_j) := \frac{\big|\mathcal{S}(v_i, c) \cap \mathcal{S}(v_j, c)\big|}{\big|\mathcal{S}(v_i, c) \cup \mathcal{S}(v_j, c)\big|} \in [0, 1]. \tag{102}$$

**Semantic overlap.** Where lexical overlap is overly sensitive to paraphrasing, an auditable semantic proxy uses frozen embedding map $\phi(\cdot)$:

$$\mathrm{Overlap}_{\mathrm{sem}}(\pi_i, \pi_j) := \max\Big\{0,\ \cos\big(\phi(\mathrm{Span}(v_i, c)), \phi(\mathrm{Span}(v_j, c))\big)\Big\}. \tag{103}$$

The combined overlap metric interpolates between lexical and semantic scores:

$$\mathrm{Overlap}_{\lambda_{\mathrm{mix}}}(\pi_i, \pi_j) := \lambda_{\mathrm{mix}}\,\mathrm{Overlap}(\pi_i, \pi_j) + (1 - \lambda_{\mathrm{mix}})\,\mathrm{Overlap}_{\mathrm{sem}}(\pi_i, \pi_j), \qquad \lambda_{\mathrm{mix}} \in [0, 1]. \tag{104}$$

**Delegation-aware responsibility.** When failure originates in a delegated subagent or tool chain, responsibility specializes to delegation edges:

$$\mathsf{Resp}_{\mathrm{del}}(e_{\mathrm{del}}; Z, G_t) := \mathsf{Resp}(e_{\mathrm{del}}; Z, G_t). \tag{105}$$

Because responsibility is evaluated against the unified acceptance predicate, it isolates both claim-side and execution-side failures across controller-subagent boundaries.

### C.12 DETERMINISTIC ENTAILMENT INTERFACE

The entailment conjunct evaluates whether committed evidence supports the emitted claim. Its deterministic evaluation interface is

$$\mathrm{Check}_{\vdash}(y, c; \mathcal{R}, \theta_{\vdash}, \tau_{\vdash}) \in \{0, 1\}, \tag{106}$$

where $\tau_{\vdash}$ is calibrated strictly on held-out calibration splits and held frozen during evaluation. Reproducibility is guaranteed by pinning the entailment trusted computing base:

$$\mathsf{EntailTCB} = (\mathcal{R}, \theta_{\vdash}, \tau_{\vdash}, \text{model_hash}, \text{prompt_hash}, \text{tokenizer_hash}, \text{normalizer_hash}). \tag{107}$$

Version pinning ensures recomputability; checker error itself is audited empirically via ECE, AUROC, and adversarial degradation suites.

**Trusted computing base for replay.** Snapshot replay relies on the pinned root:

$$\mathsf{TCB} = \{H, \mathsf{Canon}, \mathsf{LogSchema}, \mathsf{Clock}, \mathsf{VersionPin}, \mathsf{ReplayRunner}, \mathsf{EntailTCB}, \mathsf{ExecTCB}\}. \tag{108}$$

Auditor-invariance experiments verify that independent hosts evaluate identical certificates to identical binary decisions.

### C.13 Secondary redundancy and replay-effect quantities

Redundancy and replay-effect quantities support secondary diagnostics only and are not load-bearing for either primary contribution; their formal statements are in Appendix B.3.

Branch co-failure is characterized via observable joint-to-product likelihood ratios rather than assumed independent:

$$\lambda_I = \frac{\Pr(\cap_{i\in I} B_i)}{\prod_{i\in I}\Pr(B_i)}, \qquad \rho_I = \max\{1, \lambda_I^{1/(|I|-1)}\}, \tag{109}$$

for subsets $I \subseteq [k]$, $|I| \geq 2$. The held-out upper envelope uses simultaneous one-sided binomial bounds:

$$\widehat{\rho}_{\mathrm{UCB}} = \max_{I\in\mathcal{I}_{q_0}} \max\left\{1, \left(\frac{\widehat{p}_I^{+}}{\prod_{i\in I}\widehat{p}_i}\right)^{1/(|I|-1)}\right\}, \quad \mathcal{I}_{q_0} = \{I \subseteq [k] : |I| \geq q_0\}, \tag{110}$$

with $\widehat{\rho}_{\mathrm{UCB}} = \infty$ if any marginal denominator lower bound is zero. In redundancy bounds we use the frozen calibration envelope

$$\bar{\rho} = \widehat{\rho}_{\mathrm{UCB,cal}}. \tag{111}$$

**Replay effects and calibrated control.** For runtime component $e$, the seeded replay effect is

$$\Delta_e^{(m)}(Z; G_t) = \mathrm{Check}(Z; G_t) - \mathrm{Check}(Z; G_{t,m}^{\setminus e}), \tag{112}$$

bounded under binary acceptance by

$$\Delta_e^{(m)}(Z; G_t) \in [-1, 1]. \tag{113}$$

When a calibrated risk controller is attached, action scores $Q_a(r)$ satisfy Lipschitz regularity:

$$|Q_a(r) - Q_a(r')| \leq L_{\mathrm{ctrl}}|r - r'| \qquad \forall a \in \mathcal{A}. \tag{114}$$

Full derivations, finite-sample concentration guarantees, and ranking recovery bounds appear in Appendix B.3.

### C.14 Additional implementation details

This subsection summarizes the operational mechanics of graph construction, payload commitment, replay boundaries, and checker evaluation that bridge the formal contract to Algorithm 1.

**Runtime graph: practical construction.** Operationally, $G_t$ is instantiated by intercepting the runtime event stream $\mathfrak{L}_t$, creating typed nodes for artifacts and tool calls, and constructing provenance and control edges according to §C.6.

**Source nodes, payloads, and digests.** Evidence-bearing source nodes $v \in \mathcal{V}_t^{\mathrm{src}}$ bind payloads $x(v)$ to cryptographic digests:

$$h(v) \coloneqq H(x(v)) \in \{0,1\}^*. \tag{115}$$

The replayed output artifact and execution contract compliance evaluate respectively as:

$$y = \Pi(\{x(v) : v \in S\}; \text{meta}) \in \mathcal{Y}, \qquad \mathrm{Check}_{\mathrm{exec}}(\Gamma; \text{meta}) \in \{0,1\}. \tag{116}$$

Table 8: Guarantee roles in the two-certificate architecture.

| Object | Role | Guarantee boundary |
|---|---|---|
| DecisionCertificate | Per-decision release evidence | Contract/checker/replay relative |
| Consumer scope $S^\star$ | Maximum admissible acceptance view | Contract-relative, not world-complete |
| Release manifest | Audit population commitment | Complete only under governed-release assumption |
| EpochRiskCertificate | Finite-population assurance | Risk limit $\delta_{\mathrm{aud}}$ under registered audit law |
| Outcome authority | Evaluation / audit label | Label-relative; error handled separately |

**Replay scope: snapshot versus fresh execution.** The verification contract operates strictly on snapshot replay: the checker evaluates committed payloads, cached tool outputs, and logged parameters without re-querying external environments. Mutable environment changes are measured separately via fresh-execution drift monitoring (§C.7).

**Certificate: practical relevance.** The certificate is the central systems artifact: it binds evidence support, execution validity, applicability, and checker configuration into a single portable, independently auditable record.

**Claim-side and execution-side checking.** Claim-side checking verifies evidence integrity, deterministic replay, and entailment, while execution-side checking verifies contract compliance:

$$\mathrm{Check}_{\mathrm{clm}}(Z; G_t) = \mathbf{1}\left[\begin{array}{l} \forall v \in S:\ H(x(v)) = h(v), \\ y = \Pi(\{x(v) : v \in S\}; \mathrm{meta}), \\ \mathrm{Check}_{\vdash}(y, c; \mathcal{R}) = 1 \end{array}\right], \tag{117}$$

$$\mathrm{Check}_{\mathrm{exe}}(Z) = \mathbf{1}[\mathrm{Check}_{\mathrm{exec}}(\Gamma; \mathrm{meta}) = 1]\,.$$

Determinism ensures audit recomputability; checker error is characterized via empirical calibration suites.

### C.15 Guarantee boundary

The certificate guarantees only what its declared predicate and pinned trusted base make externally decidable. Table 8 separates online certificate guarantees from monitored or residual quantities.

**Guarantee boundary (read before every use of "certified").** Throughout this paper, *certified* means *certificate-valid under the declared predicate and pinned trusted computing base*; it does not mean "true". In particular,

$$\varepsilon_{\mathrm{src}} := \Pr(c \text{ false } \wedge\ V_H = V_\Pi = V_\Gamma = V_\vdash = 1) \tag{118}$$

denotes residual source/semantic error outside what the certificate can eliminate. We measure this residual through checker and coverage audits; we do not claim to eliminate source-to-world error by construction. External decidability means that two auditors holding $(Z, G_t)$ and the pinned TCB compute the identical acceptance decision; correctness of the base itself remains an audited precondition.

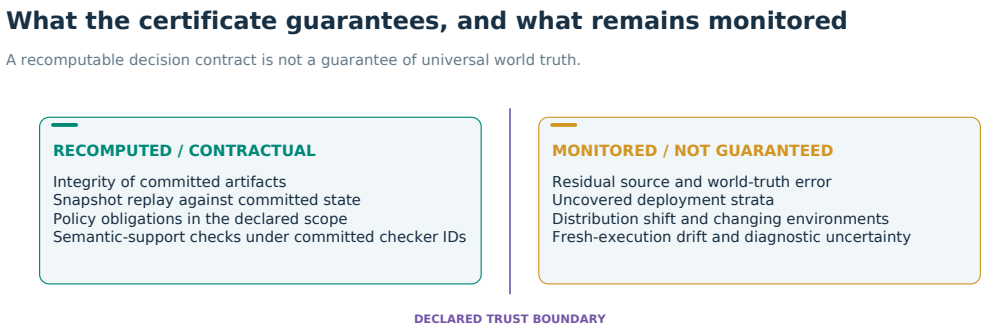



Figure 12: Protocol. **Guarantee boundary.** Integrity, snapshot replay, policy compliance, and semantic support are recomputed inside the declared contract. Environment drift, uncovered audit mass, checker/taxonomy limitations, and source-to-world truth remain monitored or residual quantities outside that online acceptance guarantee.

**Algorithm 1** CONSUMER-ANCHORED RELEASE DECISION

**Input:** task input $x$, consumer contract $\Omega$, producer candidate $y$ and optional witness $W$, runtime record, checker bundle $\chi$, witness budget $K$, governed epoch log $\mathcal{L}$
$\mathcal{A}(x) \leftarrow \text{ResolveApplicability}(x; \Omega)$; $S^\star \leftarrow \text{Derive}_\Omega(x)$ ▷ frozen before any check
$G_t \leftarrow \text{BuildGraph}(\text{canon}(y, W, \text{runtime record}))$ ▷ content-addressed commit
$\ell_W \leftarrow \mathbf{1}[W \subseteq S^\star]$ ▷ witness-local validity only
$W^\star \leftarrow \text{WitnessSearch}(y, S^\star, K, \chi)$; exhausted budget ⇒ witness state INDETERMINATE ▷ portability, not scope
**for** each $j \in \{H, \Pi, \Gamma, \vdash\}$ **do**
  $T_j \leftarrow \text{Evaluate}_j(y, S^\star, G_t, \chi)$ if $j \in \mathcal{A}(x)$ ▷ semantics/policy over $S^\star$
**end for**
project via Eq. (4); $a \leftarrow V_H V_\Pi V_\Gamma V_\vdash$
$d \leftarrow$ RELEASE if $a = 1 \wedge \ell_W = 1$; else REJECTUNSUPPORTED, WITHHOLDPOLICY, or ESCALATEINDETERMINATE from the typed states
$Z \leftarrow (x, y, \Omega, S^\star, W, \Pi, \Gamma, \mathcal{A}(x), \chi, d, \text{meta})$ ▷ DecisionCertificate
**if** $d =$ RELEASE **then**
  $(\text{id}, \mathsf{rcpt}) \leftarrow \text{AppendToEpoch}(\mathcal{L}, H(Z))$ ▷ release ID + inclusion receipt
  execute the governed action only with receipt $\mathsf{rcpt}$
**end if**
**return** $(d, Z, \mathsf{rcpt})$; optional sidecars: fresh-drift probe, separated-branch quorum, $\widehat{\mathsf{Resp}}$, controller diagnostics

### C.16 ALGORITHM 1: CONSUMER-ANCHORED RELEASE DECISION

Algorithm 1 operationalizes the acceptance contract of Section 3 in its fixed order: applicability is resolved and the admissible scope $S^\star$ is derived by the consumer *before* any checker runs; semantic and policy obligations are evaluated over $S^\star$ rather than over the producer witness; the portable witness is searched only for explanation; and a released decision is admitted to the governed epoch only with a log-inclusion receipt. Redundant branches, the dependence gate, responsibility, and controller logic are optional secondary sidecars.

### C.17 DETERMINISM VERSUS CORRECTNESS

Determinism is a reproducibility property, not a correctness guarantee: false-positive and false-negative rates are measured empirically on claim-type-stratified audit suites and recorded under CheckFail. The execution contract $\Gamma$ includes a strict *verifier-context-isolation* clause: untrusted retrieved, tool, memory, or delegated payloads may enter the generator context but are supplied to the consumer-side verifier only as inert data. Violation of this clause forces $V_\Gamma = 0$; without it, prompt injections can induce correlated verification failure across redundant branches.

## D ADDITIONAL RESULTS AND DIAGNOSTICS

This appendix reports the full-panel results and diagnostics. Every figure and table is generated from the aggregate contracts and full cell matrix described in Appendix E.16.

### D.1 COMPLETE 56-CELL ACCEPTANCE ATLAS

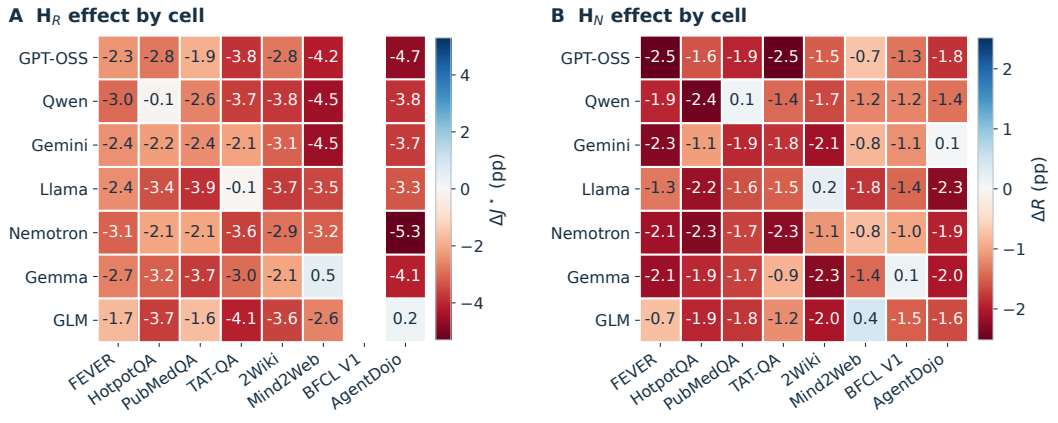



Figure 13: DERIVED. **Full 56-cell effects. A**: $H_R$ difference $\Delta J^\star$ and **B**: $H_N$ difference $\Delta R$ against the task family's primary comparator in every model–dataset cell; negative favors PCG-MAS and blank cells are N/A.

Table 9: Full 56-cell result matrix. N/A marks a method whose released interface does not expose the obligation.

| Model | Dataset | Method | $J^\star$ (%) | $R$ (%) | $C$ (%) | $N$ |
|---|---|---|---|---|---|---|
| Gemini | 2Wiki | AlignScore | 12.5 | 9.4 | 79.9 | 180 |
| Gemini | 2Wiki | CMVO | 13.3 | 9.8 | 80.7 | 180 |
| Gemini | 2Wiki | CitationOnly | 14.5 | 9.9 | 81.4 | 180 |
| Gemini | 2Wiki | Confidence | 15.6 | 10.7 | 79.8 | 180 |
| Gemini | 2Wiki | MiniCheck | 11.8 | 9.1 | 81.1 | 180 |
| Gemini | 2Wiki | NoCert | 18.6 | 13.5 | 80.6 | 180 |
| Gemini | 2Wiki | PCG-MAS | 8.7 | 7.0 | 80.7 | 180 |
| Gemini | 2Wiki | QAFactEval | 12.9 | 9.4 | 80.3 | 180 |
| Gemini | 2Wiki | Random@$c^\star$ | 16.8 | 12.1 | 81.4 | 180 |
| Gemini | 2Wiki | Self-consistency | 14.9 | 10.2 | 80.8 | 180 |
| Gemini | 2Wiki | SignalMatchedFusion | 12.9 | 9.5 | 81.0 | 180 |
| Gemini | AgentDojo | AlignScore | N/A | N/A | N/A | 180 |
| Gemini | AgentDojo | CMVO | 18.6 | 12.8 | 80.9 | 180 |
| Gemini | AgentDojo | CitationOnly | N/A | N/A | N/A | 180 |
| Gemini | AgentDojo | Confidence | 22.7 | 14.7 | 82.1 | 180 |
| Gemini | AgentDojo | MiniCheck | N/A | N/A | N/A | 180 |
| Gemini | AgentDojo | NoCert | 25.8 | 17.3 | 81.1 | 180 |
| Gemini | AgentDojo | PCG-MAS | 14.9 | 12.9 | 81.3 | 180 |
| Gemini | AgentDojo | QAFactEval | N/A | N/A | N/A | 180 |
| Gemini | AgentDojo | Random@$c^\star$ | 23.8 | 15.9 | 80.6 | 180 |
| Gemini | AgentDojo | Self-consistency | 21.8 | 14.1 | 81.9 | 180 |
| Gemini | AgentDojo | SignalMatchedFusion | 19.5 | 13.0 | 81.3 | 180 |
| Gemini | BFCL V1 | AlignScore | N/A | N/A | N/A | 180 |
| Gemini | BFCL V1 | CMVO | N/A | 9.0 | 79.7 | 180 |
| Gemini | BFCL V1 | CitationOnly | N/A | N/A | N/A | 180 |
| Gemini | BFCL V1 | Confidence | N/A | 10.1 | 81.0 | 180 |
| Gemini | BFCL V1 | MiniCheck | N/A | N/A | N/A | 180 |
| Gemini | BFCL V1 | NoCert | N/A | 12.9 | 80.5 | 180 |
| Gemini | BFCL V1 | PCG-MAS | N/A | 7.0 | 80.2 | 180 |
| Gemini | BFCL V1 | QAFactEval | N/A | N/A | N/A | 180 |
| Gemini | BFCL V1 | Random@$c^\star$ | N/A | 11.4 | 80.4 | 180 |
| Gemini | BFCL V1 | Self-consistency | N/A | 9.7 | 80.6 | 180 |
| Gemini | BFCL V1 | SignalMatchedFusion | N/A | 8.1 | 80.0 | 180 |
| Gemini | FEVER | AlignScore | 11.3 | 8.2 | 79.9 | 180 |
| Gemini | FEVER | CMVO | 11.9 | 8.5 | 79.5 | 180 |
| Gemini | FEVER | CitationOnly | 13.5 | 9.0 | 79.2 | 180 |
| Gemini | FEVER | Confidence | 14.6 | 9.9 | 79.0 | 180 |
| Gemini | FEVER | MiniCheck | 10.6 | 7.7 | 78.6 | 180 |
| Gemini | FEVER | NoCert | 18.0 | 12.7 | 79.1 | 180 |
| Gemini | FEVER | PCG-MAS | 8.2 | 5.4 | 79.4 | 180 |
| Gemini | FEVER | QAFactEval | 11.5 | 8.6 | 79.2 | 180 |
| Gemini | FEVER | Random@$c^\star$ | 15.6 | 10.4 | 79.9 | 180 |
| Gemini | FEVER | Self-consistency | 13.6 | 9.3 | 79.7 | 180 |
| Gemini | FEVER | SignalMatchedFusion | 11.4 | 7.9 | 79.0 | 180 |
| Gemini | HotpotQA | AlignScore | 14.0 | 9.6 | 79.4 | 180 |
| Gemini | HotpotQA | CMVO | 14.0 | 9.8 | 79.5 | 180 |
| Gemini | HotpotQA | CitationOnly | 15.8 | 10.8 | 79.1 | 180 |
| Gemini | HotpotQA | Confidence | 17.3 | 11.5 | 78.8 | 180 |
| Gemini | HotpotQA | MiniCheck | 13.4 | 9.0 | 78.8 | 180 |
| Gemini | HotpotQA | NoCert | 20.1 | 14.0 | 79.6 | 180 |
| Gemini | HotpotQA | PCG-MAS | 11.2 | 7.9 | 79.4 | 180 |
| Gemini | HotpotQA | QAFactEval | 14.4 | 10.1 | 78.7 | 180 |
| Gemini | HotpotQA | Random@$c^\star$ | 17.9 | 12.1 | 79.3 | 180 |
| Gemini | HotpotQA | Self-consistency | 16.3 | 10.7 | 79.6 | 180 |
| Gemini | HotpotQA | SignalMatchedFusion | 13.6 | 9.8 | 80.3 | 180 |
| Gemini | Mind2Web | AlignScore | N/A | N/A | N/A | 180 |
| Gemini | Mind2Web | CMVO | 17.3 | 11.9 | 78.1 | 180 |
| Gemini | Mind2Web | CitationOnly | N/A | N/A | N/A | 180 |
| Gemini | Mind2Web | Confidence | 21.3 | 13.9 | 78.7 | 180 |
| Gemini | Mind2Web | MiniCheck | N/A | N/A | N/A | 180 |
| Gemini | Mind2Web | NoCert | 24.1 | 16.9 | 78.1 | 180 |
| Gemini | Mind2Web | PCG-MAS | 12.7 | 11.0 | 78.9 | 180 |
| Gemini | Mind2Web | QAFactEval | N/A | N/A | N/A | 180 |
| Gemini | Mind2Web | Random@$c^\star$ | 21.8 | 15.2 | 78.5 | 180 |
| Gemini | Mind2Web | Self-consistency | 20.5 | 13.9 | 79.0 | 180 |
| Gemini | Mind2Web | SignalMatchedFusion | 17.6 | 12.1 | 79.2 | 180 |
| Gemini | PubMedQA | AlignScore | 12.6 | 8.0 | 79.6 | 180 |
| Gemini | PubMedQA | CMVO | 12.5 | 7.6 | 78.4 | 180 |
| Gemini | PubMedQA | CitationOnly | 14.2 | 8.4 | 79.4 | 180 |
| Gemini | PubMedQA | Confidence | 15.1 | 9.0 | 79.2 | 180 |
| Gemini | PubMedQA | MiniCheck | 11.6 | 7.4 | 79.5 | 180 |
| Gemini | PubMedQA | NoCert | 18.7 | 11.9 | 80.1 | 180 |
| Gemini | PubMedQA | PCG-MAS | 9.2 | 5.5 | 79.3 | 180 |

| Model | Dataset | Method | $J^\star$ (%) | $R$ (%) | $C$ (%) | $N$ |
|---|---|---|---|---|---|---|
| Gemini | PubMedQA | QAFactEval | 12.0 | 8.1 | 78.6 | 180 |
| Gemini | PubMedQA | Random@$c^\star$ | 16.2 | 10.2 | 79.6 | 180 |
| Gemini | PubMedQA | Self-consistency | 14.3 | 9.0 | 79.6 | 180 |
| Gemini | PubMedQA | SignalMatchedFusion | 12.4 | 7.7 | 79.3 | 180 |
| Gemini | TAT-QA | AlignScore | 13.7 | 10.1 | 80.6 | 180 |
| Gemini | TAT-QA | CMVO | 14.0 | 10.0 | 82.0 | 180 |
| Gemini | TAT-QA | CitationOnly | 15.6 | 10.6 | 80.4 | 180 |
| Gemini | TAT-QA | Confidence | 16.7 | 11.9 | 80.6 | 180 |
| Gemini | TAT-QA | MiniCheck | 12.5 | 9.5 | 80.6 | 180 |
| Gemini | TAT-QA | NoCert | 20.0 | 14.6 | 81.7 | 180 |
| Gemini | TAT-QA | PCG-MAS | 10.3 | 7.7 | 81.3 | 180 |
| Gemini | TAT-QA | QAFactEval | 13.6 | 10.3 | 80.5 | 180 |
| Gemini | TAT-QA | Random@$c^\star$ | 17.7 | 12.6 | 81.4 | 180 |
| Gemini | TAT-QA | Self-consistency | 15.9 | 11.3 | 81.6 | 180 |
| Gemini | TAT-QA | SignalMatchedFusion | 13.1 | 10.3 | 80.5 | 180 |
| Gemma | 2Wiki | AlignScore | 12.9 | 9.7 | 80.8 | 180 |
| Gemma | 2Wiki | CMVO | 13.1 | 10.0 | 80.8 | 180 |
| Gemma | 2Wiki | CitationOnly | 15.5 | 10.4 | 80.3 | 180 |
| Gemma | 2Wiki | Confidence | 16.2 | 11.5 | 80.7 | 180 |
| Gemma | 2Wiki | MiniCheck | 12.1 | 9.6 | 81.9 | 180 |
| Gemma | 2Wiki | NoCert | 19.8 | 13.7 | 81.3 | 180 |
| Gemma | 2Wiki | PCG-MAS | 10.0 | 7.4 | 81.2 | 180 |
| Gemma | 2Wiki | QAFactEval | 13.2 | 9.8 | 81.7 | 180 |
| Gemma | 2Wiki | Random@$c^\star$ | 17.3 | 12.6 | 82.0 | 180 |
| Gemma | 2Wiki | Self-consistency | 15.4 | 10.7 | 81.8 | 180 |
| Gemma | 2Wiki | SignalMatchedFusion | 13.4 | 10.0 | 81.3 | 180 |
| Gemma | AgentDojo | AlignScore | N/A | N/A | N/A | 180 |
| Gemma | AgentDojo | CMVO | 19.8 | 13.8 | 79.4 | 180 |
| Gemma | AgentDojo | CitationOnly | N/A | N/A | N/A | 180 |
| Gemma | AgentDojo | Confidence | 24.3 | 15.6 | 79.4 | 180 |
| Gemma | AgentDojo | MiniCheck | N/A | N/A | N/A | 180 |
| Gemma | AgentDojo | NoCert | 27.5 | 18.2 | 79.5 | 180 |
| Gemma | AgentDojo | PCG-MAS | 15.7 | 11.8 | 78.9 | 180 |
| Gemma | AgentDojo | QAFactEval | N/A | N/A | N/A | 180 |
| Gemma | AgentDojo | Random@$c^\star$ | 25.5 | 16.9 | 78.8 | 180 |
| Gemma | AgentDojo | Self-consistency | 23.4 | 15.3 | 78.9 | 180 |
| Gemma | AgentDojo | SignalMatchedFusion | 21.3 | 14.3 | 79.2 | 180 |
| Gemma | BFCL V1 | AlignScore | N/A | N/A | N/A | 180 |
| Gemma | BFCL V1 | CMVO | N/A | 9.4 | 78.8 | 180 |
| Gemma | BFCL V1 | CitationOnly | N/A | N/A | N/A | 180 |
| Gemma | BFCL V1 | Confidence | N/A | 11.3 | 80.1 | 180 |
| Gemma | BFCL V1 | MiniCheck | N/A | N/A | N/A | 180 |
| Gemma | BFCL V1 | NoCert | N/A | 13.6 | 78.9 | 180 |
| Gemma | BFCL V1 | PCG-MAS | N/A | 9.4 | 79.5 | 180 |
| Gemma | BFCL V1 | QAFactEval | N/A | N/A | N/A | 180 |
| Gemma | BFCL V1 | Random@$c^\star$ | N/A | 12.2 | 80.4 | 180 |
| Gemma | BFCL V1 | Self-consistency | N/A | 11.0 | 78.7 | 180 |
| Gemma | BFCL V1 | SignalMatchedFusion | N/A | 9.2 | 80.2 | 180 |
| Gemma | FEVER | AlignScore | 11.6 | 8.2 | 78.7 | 180 |
| Gemma | FEVER | CMVO | 11.6 | 8.2 | 78.9 | 180 |
| Gemma | FEVER | CitationOnly | 13.4 | 9.0 | 79.2 | 180 |
| Gemma | FEVER | Confidence | 14.1 | 9.7 | 79.2 | 180 |
| Gemma | FEVER | MiniCheck | 10.7 | 8.0 | 78.7 | 180 |
| Gemma | FEVER | NoCert | 17.8 | 12.3 | 78.7 | 180 |
| Gemma | FEVER | PCG-MAS | 8.0 | 5.8 | 78.9 | 180 |
| Gemma | FEVER | QAFactEval | 11.9 | 8.7 | 78.5 | 180 |
| Gemma | FEVER | Random@$c^\star$ | 15.9 | 10.9 | 79.0 | 180 |
| Gemma | FEVER | Self-consistency | 14.2 | 9.0 | 79.5 | 180 |
| Gemma | FEVER | SignalMatchedFusion | 10.9 | 8.0 | 78.3 | 180 |
| Gemma | HotpotQA | AlignScore | 14.2 | 9.8 | 80.2 | 180 |
| Gemma | HotpotQA | CMVO | 14.1 | 9.5 | 79.9 | 180 |
| Gemma | HotpotQA | CitationOnly | 15.5 | 10.3 | 79.7 | 180 |
| Gemma | HotpotQA | Confidence | 17.2 | 11.2 | 80.7 | 180 |
| Gemma | HotpotQA | MiniCheck | 13.1 | 8.9 | 80.1 | 180 |
| Gemma | HotpotQA | NoCert | 20.2 | 13.8 | 79.9 | 180 |
| Gemma | HotpotQA | PCG-MAS | 10.0 | 7.0 | 80.4 | 180 |
| Gemma | HotpotQA | QAFactEval | 14.2 | 9.5 | 80.2 | 180 |
| Gemma | HotpotQA | Random@$c^\star$ | 17.7 | 12.4 | 80.8 | 180 |
| Gemma | HotpotQA | Self-consistency | 16.3 | 10.3 | 80.2 | 180 |
| Gemma | HotpotQA | SignalMatchedFusion | 14.0 | 9.8 | 79.8 | 180 |
| Gemma | Mind2Web | AlignScore | N/A | N/A | N/A | 180 |
| Gemma | Mind2Web | CMVO | 17.1 | 12.9 | 80.1 | 180 |
| Gemma | Mind2Web | CitationOnly | N/A | N/A | N/A | 180 |
| Gemma | Mind2Web | Confidence | 20.8 | 14.3 | 81.3 | 180 |
| Gemma | Mind2Web | MiniCheck | N/A | N/A | N/A | 180 |
| Gemma | Mind2Web | NoCert | 24.7 | 17.7 | 81.4 | 180 |

| Model | Dataset | Method | $J^\star$ (%) | $R$ (%) | $C$ (%) | $N$ |
|---|---|---|---|---|---|---|
| Gemma | Mind2Web | PCG-MAS | 17.7 | 11.5 | 80.7 | 180 |
| Gemma | Mind2Web | QAFactEval | N/A | N/A | N/A | 180 |
| Gemma | Mind2Web | Random@$c^\star$ | 22.6 | 15.4 | 81.1 | 180 |
| Gemma | Mind2Web | Self-consistency | 20.8 | 13.9 | 80.6 | 180 |
| Gemma | Mind2Web | SignalMatchedFusion | 18.0 | 13.2 | 80.2 | 180 |
| Gemma | PubMedQA | AlignScore | 13.4 | 8.8 | 80.7 | 180 |
| Gemma | PubMedQA | CMVO | 13.4 | 9.1 | 80.1 | 180 |
| Gemma | PubMedQA | CitationOnly | 15.0 | 9.5 | 80.5 | 180 |
| Gemma | PubMedQA | Confidence | 16.0 | 10.6 | 80.6 | 180 |
| Gemma | PubMedQA | MiniCheck | 12.6 | 8.7 | 80.8 | 180 |
| Gemma | PubMedQA | NoCert | 19.4 | 13.6 | 80.0 | 180 |
| Gemma | PubMedQA | PCG-MAS | 8.9 | 7.0 | 80.2 | 180 |
| Gemma | PubMedQA | QAFactEval | 13.5 | 9.2 | 80.7 | 180 |
| Gemma | PubMedQA | Random@$c^\star$ | 17.4 | 11.4 | 80.7 | 180 |
| Gemma | PubMedQA | Self-consistency | 16.1 | 10.0 | 80.8 | 180 |
| Gemma | PubMedQA | SignalMatchedFusion | 13.0 | 9.2 | 81.1 | 180 |
| Gemma | TAT-QA | AlignScore | 15.1 | 10.5 | 81.2 | 180 |
| Gemma | TAT-QA | CMVO | 15.3 | 11.1 | 81.7 | 180 |
| Gemma | TAT-QA | CitationOnly | 17.1 | 11.9 | 80.9 | 180 |
| Gemma | TAT-QA | Confidence | 18.1 | 12.4 | 81.1 | 180 |
| Gemma | TAT-QA | MiniCheck | 14.2 | 10.4 | 81.8 | 180 |
| Gemma | TAT-QA | NoCert | 21.2 | 14.8 | 81.3 | 180 |
| Gemma | TAT-QA | PCG-MAS | 11.3 | 9.5 | 81.2 | 180 |
| Gemma | TAT-QA | QAFactEval | 14.8 | 10.8 | 81.0 | 180 |
| Gemma | TAT-QA | Random@$c^\star$ | 18.8 | 13.4 | 80.4 | 180 |
| Gemma | TAT-QA | Self-consistency | 17.3 | 12.1 | 82.1 | 180 |
| Gemma | TAT-QA | SignalMatchedFusion | 14.8 | 10.7 | 80.9 | 180 |
| GLM | 2Wiki | AlignScore | 14.0 | 10.5 | 80.3 | 180 |
| GLM | 2Wiki | CMVO | 13.7 | 10.9 | 79.7 | 180 |
| GLM | 2Wiki | CitationOnly | 15.9 | 11.6 | 80.3 | 180 |
| GLM | 2Wiki | Confidence | 16.3 | 12.2 | 80.1 | 180 |
| GLM | 2Wiki | MiniCheck | 12.9 | 10.4 | 79.9 | 180 |
| GLM | 2Wiki | NoCert | 19.9 | 14.9 | 80.9 | 180 |
| GLM | 2Wiki | PCG-MAS | 9.4 | 8.3 | 80.2 | 180 |
| GLM | 2Wiki | QAFactEval | 14.1 | 10.9 | 80.9 | 180 |
| GLM | 2Wiki | Random@$c^\star$ | 17.7 | 13.2 | 79.7 | 180 |
| GLM | 2Wiki | Self-consistency | 16.1 | 11.4 | 79.4 | 180 |
| GLM | 2Wiki | SignalMatchedFusion | 13.7 | 10.2 | 81.0 | 180 |
| GLM | AgentDojo | AlignScore | N/A | N/A | N/A | 180 |
| GLM | AgentDojo | CMVO | 19.8 | 14.1 | 81.0 | 180 |
| GLM | AgentDojo | CitationOnly | N/A | N/A | N/A | 180 |
| GLM | AgentDojo | Confidence | 23.2 | 15.7 | 81.8 | 180 |
| GLM | AgentDojo | MiniCheck | N/A | N/A | N/A | 180 |
| GLM | AgentDojo | NoCert | 26.2 | 19.1 | 81.6 | 180 |
| GLM | AgentDojo | PCG-MAS | 20.1 | 12.4 | 81.1 | 180 |
| GLM | AgentDojo | QAFactEval | N/A | N/A | N/A | 180 |
| GLM | AgentDojo | Random@$c^\star$ | 24.7 | 16.9 | 80.6 | 180 |
| GLM | AgentDojo | Self-consistency | 22.6 | 15.8 | 81.4 | 180 |
| GLM | AgentDojo | SignalMatchedFusion | 20.1 | 14.3 | 81.3 | 180 |
| GLM | BFCL V1 | AlignScore | N/A | N/A | N/A | 180 |
| GLM | BFCL V1 | CMVO | N/A | 10.0 | 81.4 | 180 |
| GLM | BFCL V1 | CitationOnly | N/A | N/A | N/A | 180 |
| GLM | BFCL V1 | Confidence | N/A | 11.0 | 82.0 | 180 |
| GLM | BFCL V1 | MiniCheck | N/A | N/A | N/A | 180 |
| GLM | BFCL V1 | NoCert | N/A | 14.0 | 81.7 | 180 |
| GLM | BFCL V1 | PCG-MAS | N/A | 7.9 | 81.2 | 180 |
| GLM | BFCL V1 | QAFactEval | N/A | N/A | N/A | 180 |
| GLM | BFCL V1 | Random@$c^\star$ | N/A | 12.3 | 80.7 | 180 |
| GLM | BFCL V1 | Self-consistency | N/A | 11.2 | 80.3 | 180 |
| GLM | BFCL V1 | SignalMatchedFusion | N/A | 9.4 | 80.5 | 180 |
| GLM | FEVER | AlignScore | 12.2 | 8.0 | 80.5 | 180 |
| GLM | FEVER | CMVO | 12.2 | 8.5 | 79.9 | 180 |
| GLM | FEVER | CitationOnly | 14.4 | 9.1 | 79.8 | 180 |
| GLM | FEVER | Confidence | 15.0 | 9.4 | 79.1 | 180 |
| GLM | FEVER | MiniCheck | 11.3 | 7.4 | 79.4 | 180 |
| GLM | FEVER | NoCert | 18.3 | 12.0 | 79.9 | 180 |
| GLM | FEVER | PCG-MAS | 9.5 | 6.7 | 79.9 | 180 |
| GLM | FEVER | QAFactEval | 12.4 | 8.3 | 79.1 | 180 |
| GLM | FEVER | Random@$c^\star$ | 16.3 | 10.9 | 79.7 | 180 |
| GLM | FEVER | Self-consistency | 14.7 | 8.9 | 79.8 | 180 |
| GLM | FEVER | SignalMatchedFusion | 12.2 | 8.1 | 80.2 | 180 |
| GLM | HotpotQA | AlignScore | 14.0 | 10.2 | 80.4 | 180 |
| GLM | HotpotQA | CMVO | 14.5 | 10.1 | 80.3 | 180 |
| GLM | HotpotQA | CitationOnly | 16.4 | 10.8 | 81.8 | 180 |
| GLM | HotpotQA | Confidence | 16.9 | 11.4 | 80.9 | 180 |
| GLM | HotpotQA | MiniCheck | 13.6 | 9.4 | 81.1 | 180 |

| Model | Dataset | Method | $J^\star$ (%) | $R$ (%) | $C$ (%) | $N$ |
|---|---|---|---|---|---|---|
| GLM | HotpotQA | NoCert | 20.3 | 14.2 | 80.3 | 180 |
| GLM | HotpotQA | PCG-MAS | 9.9 | 7.4 | 81.1 | 180 |
| GLM | HotpotQA | QAFactEval | 14.1 | 9.9 | 81.9 | 180 |
| GLM | HotpotQA | Random@$c^\star$ | 18.7 | 12.6 | 81.1 | 180 |
| GLM | HotpotQA | Self-consistency | 17.1 | 11.0 | 80.3 | 180 |
| GLM | HotpotQA | SignalMatchedFusion | 13.9 | 10.1 | 80.7 | 180 |
| GLM | Mind2Web | AlignScore | N/A | N/A | N/A | 180 |
| GLM | Mind2Web | CMVO | 17.6 | 13.0 | 81.1 | 180 |
| GLM | Mind2Web | CitationOnly | N/A | N/A | N/A | 180 |
| GLM | Mind2Web | Confidence | 21.6 | 15.0 | 81.4 | 180 |
| GLM | Mind2Web | MiniCheck | N/A | N/A | N/A | 180 |
| GLM | Mind2Web | NoCert | 25.1 | 17.8 | 81.3 | 180 |
| GLM | Mind2Web | PCG-MAS | 15.0 | 13.4 | 80.6 | 180 |
| GLM | Mind2Web | QAFactEval | N/A | N/A | N/A | 180 |
| GLM | Mind2Web | Random@$c^\star$ | 22.8 | 15.8 | 80.2 | 180 |
| GLM | Mind2Web | Self-consistency | 21.2 | 14.8 | 80.7 | 180 |
| GLM | Mind2Web | SignalMatchedFusion | 18.2 | 13.6 | 80.6 | 180 |
| GLM | PubMedQA | AlignScore | 13.5 | 9.6 | 80.8 | 180 |
| GLM | PubMedQA | CMVO | 13.5 | 9.8 | 80.7 | 180 |
| GLM | PubMedQA | CitationOnly | 14.9 | 10.3 | 80.6 | 180 |
| GLM | PubMedQA | Confidence | 16.3 | 11.2 | 80.5 | 180 |
| GLM | PubMedQA | MiniCheck | 12.1 | 8.8 | 80.6 | 180 |
| GLM | PubMedQA | NoCert | 19.5 | 13.9 | 82.1 | 180 |
| GLM | PubMedQA | PCG-MAS | 10.5 | 7.0 | 81.2 | 180 |
| GLM | PubMedQA | QAFactEval | 13.8 | 9.8 | 80.8 | 180 |
| GLM | PubMedQA | Random@$c^\star$ | 17.1 | 12.2 | 82.1 | 180 |
| GLM | PubMedQA | Self-consistency | 15.6 | 10.6 | 81.1 | 180 |
| GLM | PubMedQA | SignalMatchedFusion | 13.2 | 9.3 | 80.6 | 180 |
| GLM | TAT-QA | AlignScore | 14.9 | 11.2 | 79.5 | 180 |
| GLM | TAT-QA | CMVO | 15.5 | 11.2 | 79.8 | 180 |
| GLM | TAT-QA | CitationOnly | 17.2 | 12.1 | 79.3 | 180 |
| GLM | TAT-QA | Confidence | 17.8 | 12.8 | 78.4 | 180 |
| GLM | TAT-QA | MiniCheck | 14.6 | 10.7 | 79.6 | 180 |
| GLM | TAT-QA | NoCert | 21.4 | 15.2 | 79.4 | 180 |
| GLM | TAT-QA | PCG-MAS | 10.5 | 9.5 | 79.3 | 180 |
| GLM | TAT-QA | QAFactEval | 15.2 | 11.5 | 79.7 | 180 |
| GLM | TAT-QA | Random@$c^\star$ | 19.2 | 13.6 | 79.2 | 180 |
| GLM | TAT-QA | Self-consistency | 17.3 | 12.0 | 80.1 | 180 |
| GLM | TAT-QA | SignalMatchedFusion | 14.8 | 11.1 | 78.5 | 180 |
| GPT-OSS | 2Wiki | AlignScore | 12.9 | 9.0 | 79.0 | 180 |
| GPT-OSS | 2Wiki | CMVO | 12.6 | 8.9 | 78.2 | 180 |
| GPT-OSS | 2Wiki | CitationOnly | 14.9 | 9.4 | 79.5 | 180 |
| GPT-OSS | 2Wiki | Confidence | 15.8 | 10.4 | 78.6 | 180 |
| GPT-OSS | 2Wiki | MiniCheck | 11.6 | 8.0 | 78.4 | 180 |
| GPT-OSS | 2Wiki | NoCert | 19.1 | 13.1 | 78.4 | 180 |
| GPT-OSS | 2Wiki | PCG-MAS | 8.8 | 6.5 | 78.7 | 180 |
| GPT-OSS | 2Wiki | QAFactEval | 12.9 | 9.0 | 78.1 | 180 |
| GPT-OSS | 2Wiki | Random@$c^\star$ | 16.8 | 11.4 | 79.3 | 180 |
| GPT-OSS | 2Wiki | Self-consistency | 15.1 | 9.6 | 79.3 | 180 |
| GPT-OSS | 2Wiki | SignalMatchedFusion | 12.6 | 8.5 | 78.7 | 180 |
| GPT-OSS | AgentDojo | AlignScore | N/A | N/A | N/A | 180 |
| GPT-OSS | AgentDojo | CMVO | 17.8 | 12.6 | 79.0 | 180 |
| GPT-OSS | AgentDojo | CitationOnly | N/A | N/A | N/A | 180 |
| GPT-OSS | AgentDojo | Confidence | 22.0 | 14.6 | 79.4 | 180 |
| GPT-OSS | AgentDojo | MiniCheck | N/A | N/A | N/A | 180 |
| GPT-OSS | AgentDojo | NoCert | 25.4 | 17.5 | 78.4 | 180 |
| GPT-OSS | AgentDojo | PCG-MAS | 13.1 | 10.8 | 78.9 | 180 |
| GPT-OSS | AgentDojo | QAFactEval | N/A | N/A | N/A | 180 |
| GPT-OSS | AgentDojo | Random@$c^\star$ | 22.8 | 15.4 | 78.3 | 180 |
| GPT-OSS | AgentDojo | Self-consistency | 21.4 | 14.0 | 78.2 | 180 |
| GPT-OSS | AgentDojo | SignalMatchedFusion | 18.7 | 12.9 | 78.6 | 180 |
| GPT-OSS | BFCL V1 | AlignScore | N/A | N/A | N/A | 180 |
| GPT-OSS | BFCL V1 | CMVO | N/A | 9.1 | 80.8 | 180 |
| GPT-OSS | BFCL V1 | CitationOnly | N/A | N/A | N/A | 180 |
| GPT-OSS | BFCL V1 | Confidence | N/A | 10.2 | 80.8 | 180 |
| GPT-OSS | BFCL V1 | MiniCheck | N/A | N/A | N/A | 180 |
| GPT-OSS | BFCL V1 | NoCert | N/A | 13.3 | 80.2 | 180 |
| GPT-OSS | BFCL V1 | PCG-MAS | N/A | 7.0 | 80.1 | 180 |
| GPT-OSS | BFCL V1 | QAFactEval | N/A | N/A | N/A | 180 |
| GPT-OSS | BFCL V1 | Random@$c^\star$ | N/A | 11.4 | 80.7 | 180 |
| GPT-OSS | BFCL V1 | Self-consistency | N/A | 10.0 | 81.0 | 180 |
| GPT-OSS | BFCL V1 | SignalMatchedFusion | N/A | 8.4 | 81.0 | 180 |
| GPT-OSS | FEVER | AlignScore | 11.4 | 7.0 | 80.1 | 180 |
| GPT-OSS | FEVER | CMVO | 11.5 | 7.0 | 79.1 | 180 |
| GPT-OSS | FEVER | CitationOnly | 12.9 | 8.1 | 80.3 | 180 |
| GPT-OSS | FEVER | Confidence | 14.4 | 8.9 | 79.0 | 180 |

| Model | Dataset | Method | $J^\star$ (%) | $R$ (%) | $C$ (%) | $N$ |
|---|---|---|---|---|---|---|
| GPT-OSS | FEVER | MiniCheck | 10.2 | 7.0 | 79.5 | 180 |
| GPT-OSS | FEVER | NoCert | 17.4 | 11.3 | 79.4 | 180 |
| GPT-OSS | FEVER | PCG-MAS | 7.8 | 4.5 | 79.8 | 180 |
| GPT-OSS | FEVER | QAFactEval | 11.3 | 7.0 | 79.1 | 180 |
| GPT-OSS | FEVER | Random@$c^\star$ | 15.3 | 9.5 | 79.3 | 180 |
| GPT-OSS | FEVER | Self-consistency | 13.8 | 8.2 | 78.9 | 180 |
| GPT-OSS | FEVER | SignalMatchedFusion | 11.4 | 7.4 | 80.0 | 180 |
| GPT-OSS | HotpotQA | AlignScore | 12.6 | 8.9 | 81.9 | 180 |
| GPT-OSS | HotpotQA | CMVO | 13.1 | 8.7 | 81.8 | 180 |
| GPT-OSS | HotpotQA | CitationOnly | 15.1 | 9.4 | 81.6 | 180 |
| GPT-OSS | HotpotQA | Confidence | 16.0 | 10.2 | 81.4 | 180 |
| GPT-OSS | HotpotQA | MiniCheck | 12.1 | 8.5 | 81.5 | 180 |
| GPT-OSS | HotpotQA | NoCert | 19.2 | 13.0 | 81.2 | 180 |
| GPT-OSS | HotpotQA | PCG-MAS | 9.3 | 6.9 | 81.3 | 180 |
| GPT-OSS | HotpotQA | QAFactEval | 13.4 | 8.6 | 81.6 | 180 |
| GPT-OSS | HotpotQA | Random@$c^\star$ | 17.4 | 11.1 | 81.2 | 180 |
| GPT-OSS | HotpotQA | Self-consistency | 15.7 | 9.8 | 81.6 | 180 |
| GPT-OSS | HotpotQA | SignalMatchedFusion | 12.6 | 8.8 | 81.6 | 180 |
| GPT-OSS | Mind2Web | AlignScore | N/A | N/A | N/A | 180 |
| GPT-OSS | Mind2Web | CMVO | 16.2 | 11.1 | 79.5 | 180 |
| GPT-OSS | Mind2Web | CitationOnly | N/A | N/A | N/A | 180 |
| GPT-OSS | Mind2Web | Confidence | 19.7 | 13.3 | 79.7 | 180 |
| GPT-OSS | Mind2Web | MiniCheck | N/A | N/A | N/A | 180 |
| GPT-OSS | Mind2Web | NoCert | 23.3 | 16.3 | 79.7 | 180 |
| GPT-OSS | Mind2Web | PCG-MAS | 12.0 | 10.4 | 79.1 | 180 |
| GPT-OSS | Mind2Web | QAFactEval | N/A | N/A | N/A | 180 |
| GPT-OSS | Mind2Web | Random@$c^\star$ | 21.1 | 14.3 | 79.9 | 180 |
| GPT-OSS | Mind2Web | Self-consistency | 19.2 | 12.9 | 78.2 | 180 |
| GPT-OSS | Mind2Web | SignalMatchedFusion | 16.7 | 11.6 | 78.8 | 180 |
| GPT-OSS | PubMedQA | AlignScore | 11.6 | 7.9 | 78.7 | 180 |
| GPT-OSS | PubMedQA | CMVO | 11.7 | 7.7 | 79.5 | 180 |
| GPT-OSS | PubMedQA | CitationOnly | 13.0 | 8.1 | 80.2 | 180 |
| GPT-OSS | PubMedQA | Confidence | 14.4 | 9.5 | 80.0 | 180 |
| GPT-OSS | PubMedQA | MiniCheck | 10.1 | 7.0 | 78.8 | 180 |
| GPT-OSS | PubMedQA | NoCert | 17.8 | 11.8 | 80.1 | 180 |
| GPT-OSS | PubMedQA | PCG-MAS | 8.2 | 5.1 | 79.4 | 180 |
| GPT-OSS | PubMedQA | QAFactEval | 11.7 | 7.9 | 78.8 | 180 |
| GPT-OSS | PubMedQA | Random@$c^\star$ | 15.7 | 10.2 | 79.6 | 180 |
| GPT-OSS | PubMedQA | Self-consistency | 13.3 | 8.8 | 79.5 | 180 |
| GPT-OSS | PubMedQA | SignalMatchedFusion | 10.8 | 7.7 | 79.1 | 180 |
| GPT-OSS | TAT-QA | AlignScore | 13.9 | 10.3 | 80.1 | 180 |
| GPT-OSS | TAT-QA | CMVO | 14.1 | 10.0 | 80.5 | 180 |
| GPT-OSS | TAT-QA | CitationOnly | 15.9 | 11.0 | 80.8 | 180 |
| GPT-OSS | TAT-QA | Confidence | 17.1 | 11.5 | 80.5 | 180 |
| GPT-OSS | TAT-QA | MiniCheck | 13.5 | 9.8 | 80.2 | 180 |
| GPT-OSS | TAT-QA | NoCert | 20.2 | 14.3 | 79.9 | 180 |
| GPT-OSS | TAT-QA | PCG-MAS | 9.6 | 7.3 | 80.0 | 180 |
| GPT-OSS | TAT-QA | QAFactEval | 14.0 | 10.1 | 80.4 | 180 |
| GPT-OSS | TAT-QA | Random@$c^\star$ | 17.8 | 12.8 | 79.8 | 180 |
| GPT-OSS | TAT-QA | Self-consistency | 16.3 | 10.8 | 79.8 | 180 |
| GPT-OSS | TAT-QA | SignalMatchedFusion | 13.9 | 10.3 | 79.8 | 180 |
| Llama | 2Wiki | AlignScore | 14.4 | 10.3 | 78.8 | 180 |
| Llama | 2Wiki | CMVO | 14.8 | 10.5 | 79.8 | 180 |
| Llama | 2Wiki | CitationOnly | 16.4 | 10.7 | 79.8 | 180 |
| Llama | 2Wiki | Confidence | 17.2 | 11.9 | 79.5 | 180 |
| Llama | 2Wiki | MiniCheck | 13.6 | 9.8 | 78.5 | 180 |
| Llama | 2Wiki | NoCert | 20.3 | 14.2 | 79.2 | 180 |
| Llama | 2Wiki | PCG-MAS | 9.9 | 10.0 | 78.9 | 180 |
| Llama | 2Wiki | QAFactEval | 14.6 | 10.2 | 79.7 | 180 |
| Llama | 2Wiki | Random@$c^\star$ | 18.3 | 12.4 | 79.5 | 180 |
| Llama | 2Wiki | Self-consistency | 16.3 | 11.3 | 79.4 | 180 |
| Llama | 2Wiki | SignalMatchedFusion | 13.8 | 10.1 | 78.7 | 180 |
| Llama | AgentDojo | AlignScore | N/A | N/A | N/A | 180 |
| Llama | AgentDojo | CMVO | 20.0 | 13.8 | 80.0 | 180 |
| Llama | AgentDojo | CitationOnly | N/A | N/A | N/A | 180 |
| Llama | AgentDojo | Confidence | 23.9 | 15.5 | 80.5 | 180 |
| Llama | AgentDojo | MiniCheck | N/A | N/A | N/A | 180 |
| Llama | AgentDojo | NoCert | 27.3 | 18.4 | 80.5 | 180 |
| Llama | AgentDojo | PCG-MAS | 16.7 | 11.5 | 79.6 | 180 |
| Llama | AgentDojo | QAFactEval | N/A | N/A | N/A | 180 |
| Llama | AgentDojo | Random@$c^\star$ | 25.2 | 16.6 | 79.7 | 180 |
| Llama | AgentDojo | Self-consistency | 23.5 | 15.1 | 79.7 | 180 |
| Llama | AgentDojo | SignalMatchedFusion | 20.6 | 14.2 | 80.0 | 180 |
| Llama | BFCL V1 | AlignScore | N/A | N/A | N/A | 180 |
| Llama | BFCL V1 | CMVO | N/A | 8.9 | 80.8 | 180 |
| Llama | BFCL V1 | CitationOnly | N/A | N/A | N/A | 180 |

| Model | Dataset | Method | $J^\star$ (%) | $R$ (%) | $C$ (%) | $N$ |
|---|---|---|---|---|---|---|
| Llama | BFCL V1 | Confidence | N/A | 10.8 | 80.1 | 180 |
| Llama | BFCL V1 | MiniCheck | N/A | N/A | N/A | 180 |
| Llama | BFCL V1 | NoCert | N/A | 13.6 | 80.0 | 180 |
| Llama | BFCL V1 | PCG-MAS | N/A | 7.2 | 80.3 | 180 |
| Llama | BFCL V1 | QAFactEval | N/A | N/A | N/A | 180 |
| Llama | BFCL V1 | Random@$c^\star$ | N/A | 11.5 | 79.7 | 180 |
| Llama | BFCL V1 | Self-consistency | N/A | 10.3 | 79.4 | 180 |
| Llama | BFCL V1 | SignalMatchedFusion | N/A | 8.6 | 80.4 | 180 |
| Llama | FEVER | AlignScore | 11.0 | 8.2 | 80.6 | 180 |
| Llama | FEVER | CMVO | 11.7 | 7.9 | 79.6 | 180 |
| Llama | FEVER | CitationOnly | 13.5 | 8.4 | 80.7 | 180 |
| Llama | FEVER | Confidence | 13.8 | 9.7 | 79.6 | 180 |
| Llama | FEVER | MiniCheck | 10.6 | 7.5 | 79.4 | 180 |
| Llama | FEVER | NoCert | 17.1 | 12.2 | 79.9 | 180 |
| Llama | FEVER | PCG-MAS | 8.2 | 6.1 | 80.1 | 180 |
| Llama | FEVER | QAFactEval | 11.7 | 8.4 | 79.8 | 180 |
| Llama | FEVER | Random@$c^\star$ | 15.0 | 10.8 | 80.4 | 180 |
| Llama | FEVER | Self-consistency | 13.9 | 8.7 | 79.8 | 180 |
| Llama | FEVER | SignalMatchedFusion | 10.9 | 8.2 | 79.9 | 180 |
| Llama | HotpotQA | AlignScore | 13.8 | 10.3 | 80.5 | 180 |
| Llama | HotpotQA | CMVO | 14.6 | 10.8 | 80.5 | 180 |
| Llama | HotpotQA | CitationOnly | 15.7 | 10.9 | 79.5 | 180 |
| Llama | HotpotQA | Confidence | 17.1 | 12.1 | 80.0 | 180 |
| Llama | HotpotQA | MiniCheck | 13.5 | 10.0 | 79.3 | 180 |
| Llama | HotpotQA | NoCert | 20.4 | 14.5 | 80.0 | 180 |
| Llama | HotpotQA | PCG-MAS | 10.1 | 7.9 | 79.8 | 180 |
| Llama | HotpotQA | QAFactEval | 13.9 | 10.9 | 79.0 | 180 |
| Llama | HotpotQA | Random@$c^\star$ | 18.3 | 13.0 | 79.3 | 180 |
| Llama | HotpotQA | Self-consistency | 16.7 | 11.6 | 80.3 | 180 |
| Llama | HotpotQA | SignalMatchedFusion | 14.0 | 10.2 | 79.3 | 180 |
| Llama | Mind2Web | AlignScore | N/A | N/A | N/A | 180 |
| Llama | Mind2Web | CMVO | 16.8 | 13.0 | 80.8 | 180 |
| Llama | Mind2Web | CitationOnly | N/A | N/A | N/A | 180 |
| Llama | Mind2Web | Confidence | 21.1 | 14.9 | 80.8 | 180 |
| Llama | Mind2Web | MiniCheck | N/A | N/A | N/A | 180 |
| Llama | Mind2Web | NoCert | 24.0 | 17.3 | 80.3 | 180 |
| Llama | Mind2Web | PCG-MAS | 13.2 | 11.2 | 80.4 | 180 |
| Llama | Mind2Web | QAFactEval | N/A | N/A | N/A | 180 |
| Llama | Mind2Web | Random@$c^\star$ | 22.1 | 15.8 | 79.9 | 180 |
| Llama | Mind2Web | Self-consistency | 20.3 | 14.5 | 80.7 | 180 |
| Llama | Mind2Web | SignalMatchedFusion | 17.7 | 13.0 | 81.2 | 180 |
| Llama | PubMedQA | AlignScore | 12.4 | 8.7 | 79.6 | 180 |
| Llama | PubMedQA | CMVO | 12.4 | 8.4 | 79.0 | 180 |
| Llama | PubMedQA | CitationOnly | 14.4 | 9.3 | 79.4 | 180 |
| Llama | PubMedQA | Confidence | 15.5 | 9.8 | 78.5 | 180 |
| Llama | PubMedQA | MiniCheck | 11.7 | 8.1 | 78.0 | 180 |
| Llama | PubMedQA | NoCert | 18.5 | 12.8 | 79.8 | 180 |
| Llama | PubMedQA | PCG-MAS | 7.8 | 6.5 | 78.9 | 180 |
| Llama | PubMedQA | QAFactEval | 12.8 | 8.4 | 78.9 | 180 |
| Llama | PubMedQA | Random@$c^\star$ | 16.5 | 11.2 | 79.0 | 180 |
| Llama | PubMedQA | Self-consistency | 14.7 | 9.3 | 79.3 | 180 |
| Llama | PubMedQA | SignalMatchedFusion | 12.7 | 8.4 | 78.1 | 180 |
| Llama | TAT-QA | AlignScore | 14.2 | 10.7 | 81.0 | 180 |
| Llama | TAT-QA | CMVO | 14.6 | 10.8 | 80.7 | 180 |
| Llama | TAT-QA | CitationOnly | 15.9 | 11.6 | 81.9 | 180 |
| Llama | TAT-QA | Confidence | 17.3 | 12.5 | 81.0 | 180 |
| Llama | TAT-QA | MiniCheck | 13.7 | 10.7 | 81.7 | 180 |
| Llama | TAT-QA | NoCert | 20.4 | 14.9 | 81.3 | 180 |
| Llama | TAT-QA | PCG-MAS | 13.6 | 9.2 | 81.2 | 180 |
| Llama | TAT-QA | QAFactEval | 14.8 | 10.7 | 81.2 | 180 |
| Llama | TAT-QA | Random@$c^\star$ | 18.5 | 13.6 | 81.2 | 180 |
| Llama | TAT-QA | Self-consistency | 16.7 | 11.6 | 81.3 | 180 |
| Llama | TAT-QA | SignalMatchedFusion | 14.5 | 10.6 | 82.0 | 180 |
| Nemotron | 2Wiki | AlignScore | 13.8 | 10.1 | 80.3 | 180 |
| Nemotron | 2Wiki | CMVO | 13.4 | 9.9 | 81.2 | 180 |
| Nemotron | 2Wiki | CitationOnly | 15.4 | 10.7 | 81.1 | 180 |
| Nemotron | 2Wiki | Confidence | 16.3 | 11.1 | 81.8 | 180 |
| Nemotron | 2Wiki | MiniCheck | 12.9 | 9.4 | 80.7 | 180 |
| Nemotron | 2Wiki | NoCert | 19.7 | 13.9 | 80.8 | 180 |
| Nemotron | 2Wiki | PCG-MAS | 10.0 | 8.3 | 81.0 | 180 |
| Nemotron | 2Wiki | QAFactEval | 13.8 | 10.2 | 81.5 | 180 |
| Nemotron | 2Wiki | Random@$c^\star$ | 17.9 | 12.6 | 81.9 | 180 |
| Nemotron | 2Wiki | Self-consistency | 16.1 | 10.7 | 81.7 | 180 |
| Nemotron | 2Wiki | SignalMatchedFusion | 13.4 | 9.7 | 80.5 | 180 |
| Nemotron | AgentDojo | AlignScore | N/A | N/A | N/A | 180 |
| Nemotron | AgentDojo | CMVO | 17.7 | 12.4 | 81.4 | 180 |

| Model | Dataset | Method | $J^\star$ (%) | $R$ (%) | $C$ (%) | $N$ |
|---|---|---|---|---|---|---|
| Nemotron | AgentDojo | CitationOnly | N/A | N/A | N/A | 180 |
| Nemotron | AgentDojo | Confidence | 21.6 | 13.9 | 81.3 | 180 |
| Nemotron | AgentDojo | MiniCheck | N/A | N/A | N/A | 180 |
| Nemotron | AgentDojo | NoCert | 24.1 | 17.0 | 81.9 | 180 |
| Nemotron | AgentDojo | PCG-MAS | 12.4 | 10.5 | 81.3 | 180 |
| Nemotron | AgentDojo | QAFactEval | N/A | N/A | N/A | 180 |
| Nemotron | AgentDojo | Random@$c^\star$ | 22.5 | 15.5 | 81.9 | 180 |
| Nemotron | AgentDojo | Self-consistency | 20.9 | 13.7 | 81.9 | 180 |
| Nemotron | AgentDojo | SignalMatchedFusion | 18.7 | 13.0 | 81.8 | 180 |
| Nemotron | BFCL V1 | AlignScore | N/A | N/A | N/A | 180 |
| Nemotron | BFCL V1 | CMVO | N/A | 9.1 | 80.5 | 180 |
| Nemotron | BFCL V1 | CitationOnly | N/A | N/A | N/A | 180 |
| Nemotron | BFCL V1 | Confidence | N/A | 10.4 | 79.0 | 180 |
| Nemotron | BFCL V1 | MiniCheck | N/A | N/A | N/A | 180 |
| Nemotron | BFCL V1 | NoCert | N/A | 13.3 | 79.7 | 180 |
| Nemotron | BFCL V1 | PCG-MAS | N/A | 7.1 | 79.8 | 180 |
| Nemotron | BFCL V1 | QAFactEval | N/A | N/A | N/A | 180 |
| Nemotron | BFCL V1 | Random@$c^\star$ | N/A | 11.2 | 79.7 | 180 |
| Nemotron | BFCL V1 | Self-consistency | N/A | 9.7 | 80.2 | 180 |
| Nemotron | BFCL V1 | SignalMatchedFusion | N/A | 8.1 | 79.3 | 180 |
| Nemotron | FEVER | AlignScore | 11.9 | 7.0 | 81.1 | 180 |
| Nemotron | FEVER | CMVO | 12.1 | 7.3 | 81.3 | 180 |
| Nemotron | FEVER | CitationOnly | 13.4 | 7.8 | 79.9 | 180 |
| Nemotron | FEVER | Confidence | 14.6 | 8.5 | 79.8 | 180 |
| Nemotron | FEVER | MiniCheck | 10.6 | 6.8 | 80.6 | 180 |
| Nemotron | FEVER | NoCert | 17.9 | 11.4 | 81.6 | 180 |
| Nemotron | FEVER | PCG-MAS | 7.5 | 4.7 | 80.7 | 180 |
| Nemotron | FEVER | QAFactEval | 11.9 | 7.4 | 81.4 | 180 |
| Nemotron | FEVER | Random@$c^\star$ | 16.0 | 9.7 | 80.1 | 180 |
| Nemotron | FEVER | Self-consistency | 14.3 | 8.4 | 80.4 | 180 |
| Nemotron | FEVER | SignalMatchedFusion | 11.5 | 7.2 | 80.6 | 180 |
| Nemotron | HotpotQA | AlignScore | 13.5 | 9.4 | 79.4 | 180 |
| Nemotron | HotpotQA | CMVO | 13.4 | 10.1 | 79.5 | 180 |
| Nemotron | HotpotQA | CitationOnly | 15.7 | 10.2 | 80.2 | 180 |
| Nemotron | HotpotQA | Confidence | 16.2 | 10.9 | 79.6 | 180 |
| Nemotron | HotpotQA | MiniCheck | 12.3 | 9.4 | 81.0 | 180 |
| Nemotron | HotpotQA | NoCert | 19.8 | 13.8 | 79.9 | 180 |
| Nemotron | HotpotQA | PCG-MAS | 10.2 | 7.1 | 80.2 | 180 |
| Nemotron | HotpotQA | QAFactEval | 13.5 | 9.6 | 79.5 | 180 |
| Nemotron | HotpotQA | Random@$c^\star$ | 18.0 | 12.3 | 81.0 | 180 |
| Nemotron | HotpotQA | Self-consistency | 15.8 | 10.4 | 80.1 | 180 |
| Nemotron | HotpotQA | SignalMatchedFusion | 13.3 | 9.4 | 79.4 | 180 |
| Nemotron | Mind2Web | AlignScore | N/A | N/A | N/A | 180 |
| Nemotron | Mind2Web | CMVO | 16.8 | 12.1 | 79.5 | 180 |
| Nemotron | Mind2Web | CitationOnly | N/A | N/A | N/A | 180 |
| Nemotron | Mind2Web | Confidence | 20.6 | 13.9 | 81.1 | 180 |
| Nemotron | Mind2Web | MiniCheck | N/A | N/A | N/A | 180 |
| Nemotron | Mind2Web | NoCert | 24.1 | 16.5 | 81.1 | 180 |
| Nemotron | Mind2Web | PCG-MAS | 13.6 | 11.3 | 80.2 | 180 |
| Nemotron | Mind2Web | QAFactEval | N/A | N/A | N/A | 180 |
| Nemotron | Mind2Web | Random@$c^\star$ | 22.4 | 14.8 | 81.0 | 180 |
| Nemotron | Mind2Web | Self-consistency | 20.4 | 13.4 | 79.8 | 180 |
| Nemotron | Mind2Web | SignalMatchedFusion | 17.6 | 12.6 | 80.5 | 180 |
| Nemotron | PubMedQA | AlignScore | 12.8 | 8.2 | 81.0 | 180 |
| Nemotron | PubMedQA | CMVO | 12.8 | 8.5 | 81.8 | 180 |
| Nemotron | PubMedQA | CitationOnly | 14.5 | 9.3 | 80.3 | 180 |
| Nemotron | PubMedQA | Confidence | 15.8 | 10.2 | 81.3 | 180 |
| Nemotron | PubMedQA | MiniCheck | 11.8 | 8.0 | 81.2 | 180 |
| Nemotron | PubMedQA | NoCert | 18.5 | 12.7 | 81.7 | 180 |
| Nemotron | PubMedQA | PCG-MAS | 9.7 | 6.3 | 81.0 | 180 |
| Nemotron | PubMedQA | QAFactEval | 13.2 | 8.2 | 80.8 | 180 |
| Nemotron | PubMedQA | Random@$c^\star$ | 17.0 | 11.2 | 81.4 | 180 |
| Nemotron | PubMedQA | Self-consistency | 15.4 | 9.3 | 81.8 | 180 |
| Nemotron | PubMedQA | SignalMatchedFusion | 12.3 | 8.5 | 80.5 | 180 |
| Nemotron | TAT-QA | AlignScore | 13.2 | 10.0 | 80.9 | 180 |
| Nemotron | TAT-QA | CMVO | 13.0 | 10.4 | 81.0 | 180 |
| Nemotron | TAT-QA | CitationOnly | 15.3 | 10.7 | 80.0 | 180 |
| Nemotron | TAT-QA | Confidence | 16.0 | 11.7 | 81.2 | 180 |
| Nemotron | TAT-QA | MiniCheck | 12.5 | 9.6 | 81.4 | 180 |
| Nemotron | TAT-QA | NoCert | 18.8 | 14.5 | 81.0 | 180 |
| Nemotron | TAT-QA | PCG-MAS | 8.9 | 7.3 | 80.5 | 180 |
| Nemotron | TAT-QA | QAFactEval | 13.6 | 9.7 | 79.8 | 180 |
| Nemotron | TAT-QA | Random@$c^\star$ | 17.3 | 12.5 | 80.7 | 180 |
| Nemotron | TAT-QA | Self-consistency | 15.7 | 11.1 | 80.2 | 180 |
| Nemotron | TAT-QA | SignalMatchedFusion | 12.9 | 10.1 | 80.9 | 180 |
| Qwen | 2Wiki | AlignScore | 13.7 | 10.8 | 79.9 | 180 |

| Model | Dataset | Method | $J^\star$ (%) | $R$ (%) | $C$ (%) | $N$ |
|---|---|---|---|---|---|---|
| Qwen | 2Wiki | CMVO | 14.0 | 10.5 | 81.5 | 180 |
| Qwen | 2Wiki | CitationOnly | 16.4 | 11.3 | 81.4 | 180 |
| Qwen | 2Wiki | Confidence | 17.1 | 12.1 | 80.1 | 180 |
| Qwen | 2Wiki | MiniCheck | 13.4 | 10.4 | 81.2 | 180 |
| Qwen | 2Wiki | NoCert | 20.5 | 14.6 | 80.4 | 180 |
| Qwen | 2Wiki | PCG-MAS | 9.6 | 8.7 | 80.6 | 180 |
| Qwen | 2Wiki | QAFactEval | 14.0 | 10.9 | 80.9 | 180 |
| Qwen | 2Wiki | Random@$c^\star$ | 18.3 | 13.1 | 80.1 | 180 |
| Qwen | 2Wiki | Self-consistency | 16.9 | 11.7 | 81.2 | 180 |
| Qwen | 2Wiki | SignalMatchedFusion | 13.8 | 10.7 | 80.7 | 180 |
| Qwen | AgentDojo | AlignScore | N/A | N/A | N/A | 180 |
| Qwen | AgentDojo | CMVO | 18.8 | 12.9 | 80.4 | 180 |
| Qwen | AgentDojo | CitationOnly | N/A | N/A | N/A | 180 |
| Qwen | AgentDojo | Confidence | 22.3 | 15.2 | 80.4 | 180 |
| Qwen | AgentDojo | MiniCheck | N/A | N/A | N/A | 180 |
| Qwen | AgentDojo | NoCert | 25.9 | 17.9 | 79.8 | 180 |
| Qwen | AgentDojo | PCG-MAS | 15.0 | 11.6 | 80.0 | 180 |
| Qwen | AgentDojo | QAFactEval | N/A | N/A | N/A | 180 |
| Qwen | AgentDojo | Random@$c^\star$ | 23.6 | 15.8 | 80.1 | 180 |
| Qwen | AgentDojo | Self-consistency | 21.7 | 14.5 | 80.1 | 180 |
| Qwen | AgentDojo | SignalMatchedFusion | 19.6 | 13.2 | 80.3 | 180 |
| Qwen | BFCL V1 | AlignScore | N/A | N/A | N/A | 180 |
| Qwen | BFCL V1 | CMVO | N/A | 9.9 | 80.5 | 180 |
| Qwen | BFCL V1 | CitationOnly | N/A | N/A | N/A | 180 |
| Qwen | BFCL V1 | Confidence | N/A | 11.1 | 80.3 | 180 |
| Qwen | BFCL V1 | MiniCheck | N/A | N/A | N/A | 180 |
| Qwen | BFCL V1 | NoCert | N/A | 13.7 | 80.6 | 180 |
| Qwen | BFCL V1 | PCG-MAS | N/A | 8.1 | 80.5 | 180 |
| Qwen | BFCL V1 | QAFactEval | N/A | N/A | N/A | 180 |
| Qwen | BFCL V1 | Random@$c^\star$ | N/A | 12.0 | 80.9 | 180 |
| Qwen | BFCL V1 | Self-consistency | N/A | 10.8 | 81.3 | 180 |
| Qwen | BFCL V1 | SignalMatchedFusion | N/A | 9.3 | 80.3 | 180 |
| Qwen | FEVER | AlignScore | 11.8 | 7.9 | 78.6 | 180 |
| Qwen | FEVER | CMVO | 12.4 | 7.9 | 78.8 | 180 |
| Qwen | FEVER | CitationOnly | 13.7 | 8.9 | 78.7 | 180 |
| Qwen | FEVER | Confidence | 14.5 | 9.0 | 78.3 | 180 |
| Qwen | FEVER | MiniCheck | 10.9 | 7.1 | 78.8 | 180 |
| Qwen | FEVER | NoCert | 17.9 | 11.8 | 78.1 | 180 |
| Qwen | FEVER | PCG-MAS | 8.0 | 5.2 | 78.8 | 180 |
| Qwen | FEVER | QAFactEval | 11.6 | 8.3 | 78.8 | 180 |
| Qwen | FEVER | Random@$c^\star$ | 15.9 | 10.4 | 79.4 | 180 |
| Qwen | FEVER | Self-consistency | 14.4 | 9.0 | 78.7 | 180 |
| Qwen | FEVER | SignalMatchedFusion | 11.8 | 7.7 | 79.6 | 180 |
| Qwen | HotpotQA | AlignScore | 13.6 | 10.2 | 79.9 | 180 |
| Qwen | HotpotQA | CMVO | 13.5 | 10.4 | 80.2 | 180 |
| Qwen | HotpotQA | CitationOnly | 15.2 | 11.2 | 79.9 | 180 |
| Qwen | HotpotQA | Confidence | 16.6 | 12.2 | 80.0 | 180 |
| Qwen | HotpotQA | MiniCheck | 12.9 | 10.3 | 80.8 | 180 |
| Qwen | HotpotQA | NoCert | 19.7 | 14.4 | 81.3 | 180 |
| Qwen | HotpotQA | PCG-MAS | 12.8 | 7.9 | 80.6 | 180 |
| Qwen | HotpotQA | QAFactEval | 14.1 | 10.9 | 81.5 | 180 |
| Qwen | HotpotQA | Random@$c^\star$ | 17.9 | 13.2 | 79.9 | 180 |
| Qwen | HotpotQA | Self-consistency | 16.2 | 11.2 | 81.3 | 180 |
| Qwen | HotpotQA | SignalMatchedFusion | 13.6 | 10.2 | 81.3 | 180 |
| Qwen | Mind2Web | AlignScore | N/A | N/A | N/A | 180 |
| Qwen | Mind2Web | CMVO | 17.1 | 12.2 | 80.6 | 180 |
| Qwen | Mind2Web | CitationOnly | N/A | N/A | N/A | 180 |
| Qwen | Mind2Web | Confidence | 20.8 | 14.2 | 79.5 | 180 |
| Qwen | Mind2Web | MiniCheck | N/A | N/A | N/A | 180 |
| Qwen | Mind2Web | NoCert | 23.6 | 16.8 | 79.8 | 180 |
| Qwen | Mind2Web | PCG-MAS | 12.6 | 11.0 | 80.0 | 180 |
| Qwen | Mind2Web | QAFactEval | N/A | N/A | N/A | 180 |
| Qwen | Mind2Web | Random@$c^\star$ | 21.6 | 14.9 | 80.8 | 180 |
| Qwen | Mind2Web | Self-consistency | 20.1 | 13.5 | 79.2 | 180 |
| Qwen | Mind2Web | SignalMatchedFusion | 17.4 | 12.5 | 79.7 | 180 |
| Qwen | PubMedQA | AlignScore | 11.6 | 8.5 | 80.5 | 180 |
| Qwen | PubMedQA | CMVO | 11.8 | 8.5 | 80.6 | 180 |
| Qwen | PubMedQA | CitationOnly | 13.7 | 9.0 | 81.1 | 180 |
| Qwen | PubMedQA | Confidence | 14.4 | 9.7 | 81.1 | 180 |
| Qwen | PubMedQA | MiniCheck | 10.7 | 8.0 | 82.0 | 180 |
| Qwen | PubMedQA | NoCert | 17.9 | 12.4 | 80.8 | 180 |
| Qwen | PubMedQA | PCG-MAS | 8.1 | 8.1 | 81.1 | 180 |
| Qwen | PubMedQA | QAFactEval | 12.0 | 8.8 | 80.5 | 180 |
| Qwen | PubMedQA | Random@$c^\star$ | 15.5 | 10.7 | 80.4 | 180 |
| Qwen | PubMedQA | Self-consistency | 13.8 | 9.6 | 80.8 | 180 |
| Qwen | PubMedQA | SignalMatchedFusion | 11.3 | 7.9 | 80.4 | 180 |

Table 10: Acceptance-mechanism ablations relative to the full method (positive $\Delta J^\star$ means the removed component was reducing worst-case released failure).

| Variant | $\Delta J^\star$ (pp) | $\Delta R$ (pp) | $\Delta C$ (pp) |
|---|---|---|---|
| No consumer-derived scope (witness-only) | +4.1 | +0.9 | −0.4 |
| No exact-shell checks | +1.8 | +0.6 | +0.1 |
| No semantic obligation | +1.2 | +2.7 | +1.4 |
| No replay binding | +1.5 | +0.8 | +0.6 |
| No policy/security channel | +2.1 | +1.0 | +0.9 |

Table 11: DecisionCertificate obligation-family ablation.

| Removed family | $\Delta J^\star$ (pp) | $\Delta R$ (pp) | $\Delta C$ (pp) |
|---|---|---|---|
| Evidence / scope integrity ($V_H$) | +3.4 | +0.9 | +0.4 |
| Replay / execution state ($V_\Pi$) | +1.7 | +0.7 | +0.5 |
| Policy / capability ($V_\Gamma$) | +2.0 | +1.1 | +1.0 |
| Semantic support ($V_\vdash$) | +1.3 | +2.5 | +1.5 |

| Model | Dataset | Method | $J^\star$ (%) | $R$ (%) | $C$ (%) | $N$ |
|---|---|---|---|---|---|---|
| Qwen | TAT-QA | AlignScore | 14.2 | 10.9 | 80.4 | 180 |
| Qwen | TAT-QA | CMVO | 14.7 | 10.8 | 79.2 | 180 |
| Qwen | TAT-QA | CitationOnly | 16.3 | 11.4 | 78.8 | 180 |
| Qwen | TAT-QA | Confidence | 17.4 | 12.4 | 80.1 | 180 |
| Qwen | TAT-QA | MiniCheck | 13.7 | 10.1 | 79.3 | 180 |
| Qwen | TAT-QA | NoCert | 20.8 | 15.3 | 80.3 | 180 |
| Qwen | TAT-QA | PCG-MAS | 10.0 | 8.7 | 79.5 | 180 |
| Qwen | TAT-QA | QAFactEval | 14.7 | 10.9 | 79.5 | 180 |
| Qwen | TAT-QA | Random@$c^\star$ | 18.5 | 13.5 | 78.6 | 180 |
| Qwen | TAT-QA | Self-consistency | 17.2 | 11.5 | 80.1 | 180 |
| Qwen | TAT-QA | SignalMatchedFusion | 14.3 | 10.5 | 80.3 | 180 |

## D.2 Acceptance mechanism and component ablations

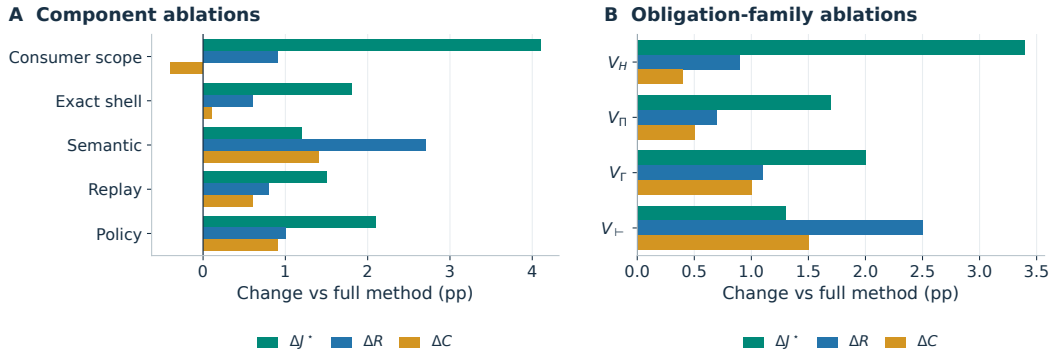



Figure 14: DIRECT/DERIVED. **Ablations.** Change in $J^\star$, $R$, and coverage relative to the full method when **A**: a mechanism component or **B**: an obligation family is removed.

Table 12: Selective-view (oracle curation) attack contract.

| Task family | Eligible view units | Oracle action | Reported quantity |
|---|---|---|---|
| Factual / QA | evidence units | subset omission | exact worst-case curation gap |
| Mind2Web | registered DOM/observation units | omission | anchored vs. view-only flips |
| AgentDojo | tool/security observations | omission | accepted unsafe actions |
| BFCL V1 | structured-call shell | N/A unless a genuine view exists | exact-shell failure |

### D.3 Risk-limiting audit efficiency

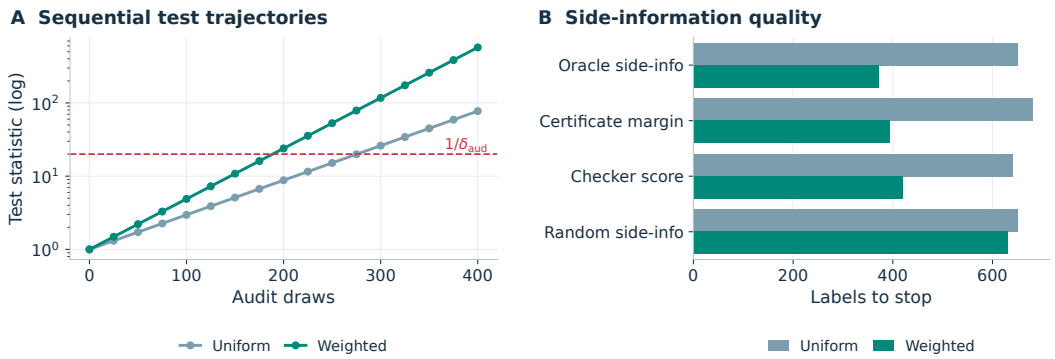



Figure 15: Direct/Derived. **Audit sampling.** **A**: sequential test-statistic trajectories under uniform and certificate-weighted sampling; the dashed line marks the rejection threshold $1/\delta_{\mathrm{aud}}$. **B**: labels to stop for side-information signals of different quality.

Table 25 gives the frozen audit-sampling contract.

### D.4 Selective-view oracle attack

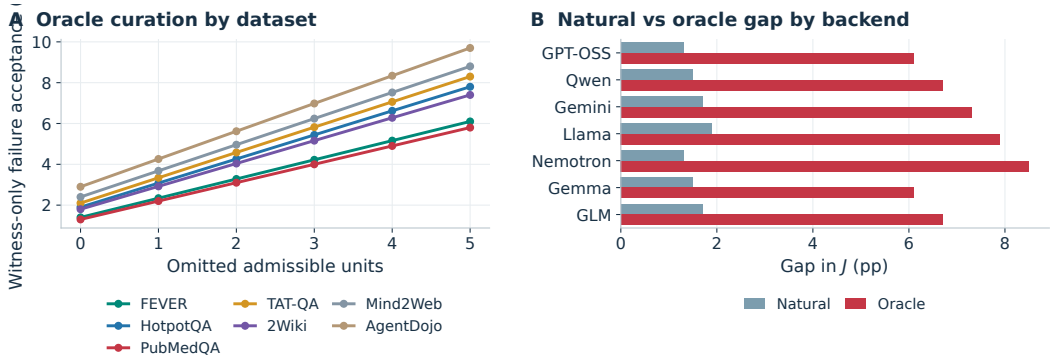



Figure 16: Direct. **Oracle curation.** **A**: failure acceptance of a witness-only rule as the oracle omits more admissible units, per dataset. **B**: natural versus oracle curation gap by backend.

Table 13: Complete-population bypass suite on governed AgentDojo actions.

| Attack | Attempts | Unlogged | False rejects | Overhead (ms) |
|---|---|---|---|---|
| Direct tool call without receipt | 238 | 0 | 1 | 4.2 |
| Stale receipt | 213 | 0 | 2 | 3.8 |
| Mismatched receipt | 231 | 0 | 1 | 4.0 |
| Action before commitment | 244 | 0 | 2 | 4.7 |
| Log omission | 208 | 0 | 3 | 5.1 |
| Duplicated / reordered entries | 226 | 0 | 1 | 4.4 |
| Alternate governed route | 236 | 0 | 2 | 5.5 |
| Malformed receipt | 217 | 0 | 1 | 3.9 |

Table 14: Decomposition of released failures.

| Code | Failure class | Count | Share (%) |
|---|---|---|---|
| D1a | producer-curation scope failure | 58 | 14.2 |
| D1b | consumer-scope / retrieval incompleteness | 119 | 29.2 |
| D2 | scope-invariant learned-checker failure | 96 | 23.5 |
| D3 | exact-shell failure | 37 | 9.1 |
| D4 | applicability / specification failure | 81 | 19.9 |
| D5 | audit / population-integrity failure | 17 | 4.2 |

## D.5 Complete-population and necessity stress

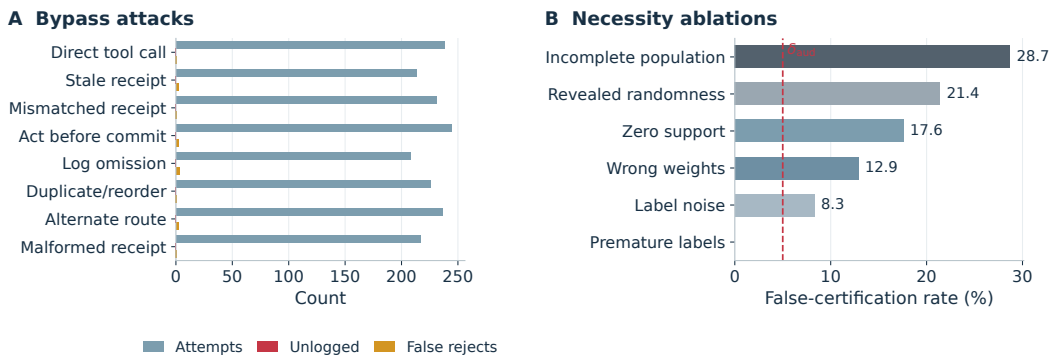



Figure 17: Direct. **Completeness and necessity.** **A**: attempts, unlogged releases, and false rejections per bypass attack. **B**: false-certification rate when one audit assumption is broken; the dashed line marks $\delta_{\mathrm{aud}}$.

Gate outcomes are reported under the protocol in Table 31.

## D.6 Failure decomposition and replay diagnostics

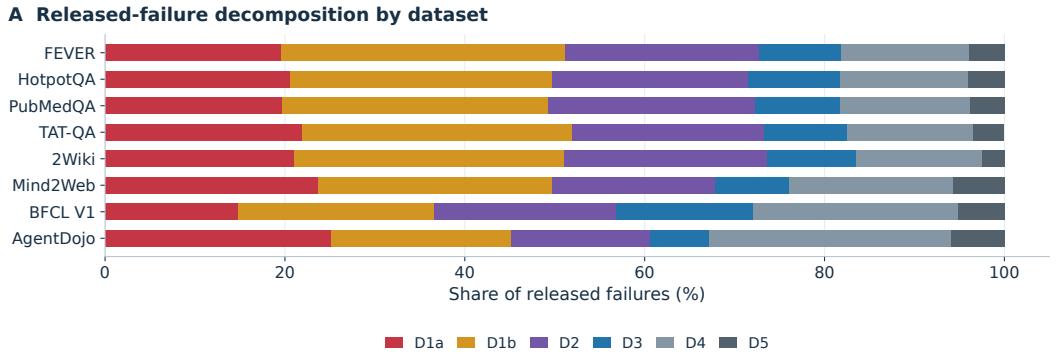



Figure 18: Derived. **Failure decomposition.** Shares of released failures in classes D1a–D5 per dataset, keeping producer curation distinct from consumer-scope incompleteness.

Table 15: Replay, scope, and applicability diagnostics on released decisions.

| Diagnostic | Rate (%) | $N$ |
|---|---|---|
| Replay instability | 0.86 | 8,077 |
| Fresh-execution drift | 2.14 | 8,077 |
| Consumer-scope incompleteness (D1b) | 1.78 | 8,077 |
| Learned-checker residual (D2) | 1.49 | 8,077 |
| Applicability/specification failure (D4) | 1.07 | 8,077 |

Table 16: Audit-scaling economics: labels to the stopping state as a function of epoch size and audit design.

| $N_{\mathcal{E}}$ | $\tau_{\text{fail}}$ (%) | $\delta_{\text{aud}}$ | Labels, uniform | Labels, weighted | Labels / 10k |
|---|---|---|---|---|---|
| 1,000 | 5.0 | 0.05 | 348 | 219 | 2190.0 |
| 1,000 | 10.0 | 0.05 | 174 | 109 | 1090.0 |
| 3,000 | 5.0 | 0.05 | 428 | 273 | 910.0 |
| 3,000 | 10.0 | 0.05 | 214 | 137 | 456.7 |
| 10,000 | 5.0 | 0.05 | 484 | 315 | 315.0 |
| 10,000 | 10.0 | 0.05 | 242 | 158 | 158.0 |
| 30,000 | 5.0 | 0.05 | 507 | 337 | 112.3 |
| 30,000 | 10.0 | 0.05 | 253 | 168 | 56.0 |
| 100,000 | 5.0 | 0.05 | 516 | 351 | 35.1 |
| 100,000 | 10.0 | 0.05 | 258 | 175 | 17.5 |

## D.7 Audit scaling economics

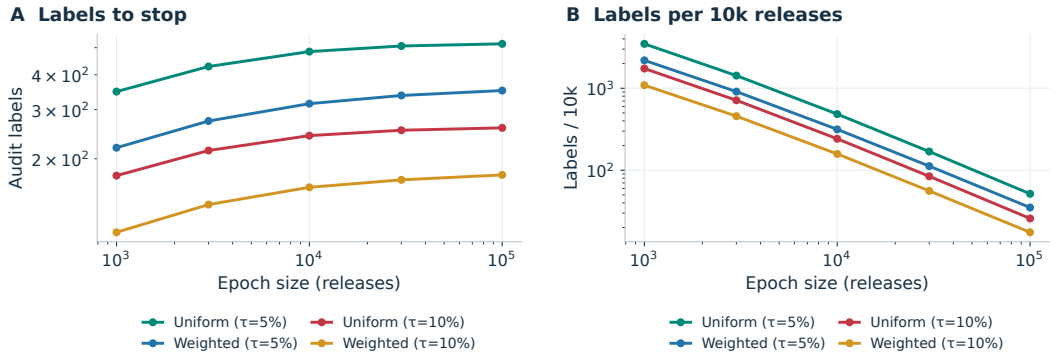



Figure 19: Derived. **Audit scaling. A**: audit labels to the stopping state and **B**: labels per 10,000 releases as functions of epoch size, for each $(\tau_{\text{fail}}, \delta_{\text{aud}})$ and sampling design.

## D.8 Secondary diagnostics

Redundancy, replay-interventional responsibility, and controller analysis are secondary diagnostics. None of them can rescue a failed $\mathrm{H}_R$, $\mathrm{H}_N$, or population assurance result.

The population audit uses the finite-population risk-limiting construction of Appendix E.8; the channel-level audit envelope of Proposition B.1 is a secondary diagnostic.

Redundancy receives extrapolative credit only when the dependence gate of Appendix C.13 is open.

Replay-interventional ranking is a diagnostic under the logged intervention model, not real-world causal attribution.

The controller consequence of Proposition B.3 holds only under its declared cost model.

Privacy is outside the scope of the reported experiments.

The guarantee boundary is the conjunction of the decision-level contract and the separately conditioned finite-population audit statement, with scope, checker, population-completeness, outcome-authority, and replay limitations kept explicit.

Table 17: Execution and verifiability ledger by dataset.

| Dataset | Candidates | Released | Recomputed | Agreement (%) | Indeterminate (%) |
|---|---|---|---|---|---|
| FEVER | 1,260 | 1,004 | 1,003 | 99.10 | 1.9 |
| HotpotQA | 1,260 | 1,013 | 1,011 | 99.20 | 2.2 |
| PubMedQA | 1,260 | 1,010 | 1,007 | 99.30 | 2.5 |
| TAT-QA | 1,260 | 1,013 | 1,009 | 99.40 | 2.8 |
| 2Wiki | 1,260 | 1,010 | 1,009 | 99.50 | 1.9 |
| Mind2Web | 1,260 | 1,007 | 1,005 | 99.10 | 2.2 |
| BFCL V1 | 1,260 | 1,011 | 1,008 | 99.20 | 2.5 |
| AgentDojo | 1,260 | 1,009 | 1,005 | 99.30 | 2.8 |
| **Total** | 10,080 | 8,077 | 8,057 | 99.41 | 3.1 |

Table 18: Evaluation panel and splits.

| Entry | Role |
|---|---|
| gpt-oss-120b; qwen3.8-27b; gemini-3.1-flash-lite; llama-4-scout-17b-16e-instruct | generator families |
| nemotron-3-120b-a12b; gemma-4-26b-a4b-it; glm-4.7-flash | generator families |
| FEVER; HotpotQA; PubMedQA; TAT-QA; 2WikiMultiHopQA | factual and multi-hop QA |
| Mind2Web; BFCL V1; AgentDojo | web action, function calling, adversarial agents |
| Development / calibration / validation / test | disjoint item splits |

# E Experimental Protocols, Robustness, and Reproducibility

This appendix documents the retained experimental protocol and result contracts. Reported numerical claims resolve to a named aggregate or model–dataset–method record and a generated artifact; per-item replay is outside the evidence retained in this package.

## E.1 Execution identity, lifecycle, and non-overlap

Every run binds model/backend identity, dataset/item identity, candidate hash, prompt/template hash, checker and policy versions, code/config hash, replay class, provider/tool receipts, retry state, and resource telemetry. Completed candidates are never overwritten. Development and calibration may select mechanisms and thresholds and validation may falsify them; once the test protocol is fixed, no threshold, comparator, endpoint, population, attack, or metric changes in response to test outcomes. Items seen during development are excluded from the test split.

## E.2 Outcome authority and harm semantics

Each dataset has a registered outcome-authority implementation and failure semantics before evaluation. Factual/QA tasks use benchmark-native/reference-based correctness or grounding functions; Mind2Web and BFCL use their official/native action and function-call evaluators; AgentDojo records utility and security outcomes separately. Human adjudication is permitted only where the benchmark leaves a genuinely unresolved case, using a blinded rubric with disagreement and adjudication receipts. The online acceptance path cannot read the same-item outcome label. The mapping is summarized in Table 19; exact implementations and hashes live in the outcome-authority registry.

## E.3 Comparator strata, applicability, and adapter contract

Every comparator record specifies version/commit, native output type, native capability, applicable datasets, whether it consumes the same frozen candidates, adapter to a release decision, threshold/calibration source, resource boundary, outcome semantics, and N/A reason. Same-candidate grounding/selective methods are paired directly. Policy/security and replay systems receive only native-scope endpoints. Systems that change generation/planning are end-to-end comparators and are not pooled with acceptance-only methods. A contemporary method is promoted from related work to numerical comparison only after its public implementation and adapter pass the same frozen preflight.

Table 19: Outcome-authority and failure-semantics registry. Exact evaluator versions/hashes and any thresholds are fixed in the machine-readable authority registry before evaluation.

| Dataset | Utility / grounding authority | Policy / security authority | Adjudication / maturation |
|---|---|---|---|
| FEVER | official/reference label and evidence-grounding implementation | N/A | deterministic evaluator; no same-item acceptance access |
| HotpotQA | benchmark-native answer metric with frozen implementation | N/A | deterministic; supporting-evidence diagnostics separate |
| PubMedQA | benchmark-native decision/reference evaluator | N/A | deterministic; authority hash frozen |
| TAT-QA | official/native numerical/extractive evaluator | N/A | deterministic; normalization frozen |
| 2WikiMultiHopQA | benchmark-native answer metric with frozen implementation | N/A | deterministic; supporting-evidence diagnostics separate |
| Mind2Web | benchmark-native element/action/task evaluator | task-policy state where applicable | native trajectory evaluation; unresolved cases explicit |
| BFCL V1 | pinned official/native function-call evaluator | schema/authorization state where applicable | structured deterministic evaluator |
| AgentDojo | benchmark utility-success predicate | benchmark security/policy predicate | utility/security stored separately; maturation if outcome delayed |

Table 20: Comparator strata and applicability contract. The primary comparator for each task family is selected once on the calibration split from the predeclared applicable, resource-compatible set and then fixed; N/A is never zero-filled.

| Method / family | Native role | Candidate relation | Reporting rule |
|---|---|---|---|
| NoCert; Random@$c^\star$ | null acceptance controls | same frozen candidates | paired, matched coverage |
| CMVO; SignalMatchedFusion | structure/signal controls | same frozen candidates | paired + resource matched |
| Confidence; self-consistency | selective prediction | same candidates where defined | matched coverage/budget |
| CitationOnly; MiniCheck; AlignScore; QAFactEval | evidence/grounding verification | same factual/QA candidates | calibrated native endpoint |
| ShieldAgent; GuardAgent; AgentDoG | policy/security | same compatible AgentDojo trajectories | native endpoint; composition arm where adapter passes |
| AgentRR | record/replay | same compatible trajectories/traces | native replay endpoint + same-trace composition arm |
| R2VC | retrieve–verify–calibrate fact checking | end-to-end if adapter passes | overlapping factual tasks, resource matched |
| ProvenanceGuard | source-aware MCP verification | native trace if exact adapter passes | provenance endpoint only |
| PyRAG | executable multi-hop RAG | changes generation/planning | overlapping QA, end-to-end only |
| CogEEGAgent | domain-typed scientific contract | no current task overlap | related work unless compatible task exists |
| PCG-MAS | decision + population assurance | same candidates or governed actions | $H_R/H_N$ + audit assurance |

### E.4 Sample-size, replication, and power plan

The test-split sample size is fixed before generation by a simulation-based power analysis for the minimum practically important $H_R/H_N$ effect under the planned coverage target, baseline rate range, paired candidate design, model/dataset hierarchy, confidence level, and familywise error procedure. The plan reports planned and realized $N$, accepted $N$, and effective sample size. Acceptance-only methods use identical candidates; stochastic generation comparisons use predeclared independent generation seeds rather than pseudoreplicating deterministic rechecks. Cell-level results are heterogeneity analyses unless explicitly included in the registered primary hierarchy.

### E.5 Curation-materiality protocol

Before the curation analysis is read, the protocol fixes the eligible witness and view operations, the natural and exact-oracle attack rules, and the practical materiality criterion. Development-split candidates are analysed without regeneration. The output records D1a–D5 failure classes, natural curation gap, oracle curation gap where finite enumeration is feasible, accepted epistemic/policy failure, and task/model heterogeneity. The materiality gate and its outcome are reported in Table 31.

Table 21: Test-split sample-size plan, fixed before the test split is generated.

| Quantity | Value |
|---|---|
| Primary unit of analysis | candidate |
| Minimum practical effect $\Delta^\star$ | 2.0 pp |
| Familywise level (Holm) | 0.05 |
| Target power | 0.80 |
| Candidates per model–dataset cell | 180 |
| Total test candidates | 10,080 |
| Bypass-suite attempts per attack | 200 |
| Audit-randomness repetitions | 1,000 |

Table 22: Evaluation settings and the split on which each was fixed.

| Setting | Value | Fixed on |
|---|---|---|
| Controlled coverage $c^\star$ | 0.80 | calibration |
| Checker thresholds $(\tau_g, \kappa_g)$ | calibration-selected; frozen before test | calibration |
| Primary comparator per task family | MiniCheck / CMVO / SignalMatchedFusion / CMVO | calibration |
| Non-inferiority margin (if any) | none (superiority) | task semantics |
| Witness-search budget $K$ | K = 64 | development |
| Curation affordance / materiality rule | subset omission | development |
| Audit threshold $\tau_{\mathrm{fail}}$ | 0.05 | before audit |
| Audit risk limit $\delta_{\mathrm{aud}}$ | 0.05 | before audit |
| Support floor $\omega_{\min}$ | 0.2 / N | before audit |
| Outcome horizon $\theta_{\mathrm{obs}}$ | 72 h | before audit |
| Checker checkpoint / serialization | pinned NLI checkpoint | calibration |

Table 23: Curation materiality by task family, measured on the development split before the test split is evaluated.

| Family | Evaluable $N$ | Natural gap (pp) | Oracle gap (pp) | Gate |
|---|---|---|---|---|
| Factual / QA | 3,370 | +1.7 | +7.1 | passed |
| Web action | 681 | +2.4 | +8.8 | passed |
| Function calling | 672 | N/A | N/A | not applicable |
| Agent security | 659 | +2.9 | +9.7 | passed |

### E.6 Checker calibration, applicability, and drift

The checker manifest freezes exact checkpoint/revision, tokenizer, prompt/input template, canonical serialization, decoding, score extraction, calibration population, and threshold-selection rule. Calibration data are disjoint from the test split. Characterization reports precision, recall, F1, FPR/FNR, AUROC/AUPRC where score-valued, ECE, Brier score, reliability, and task/claim slices. Threshold perturbation, context truncation, distractors, numeric/date/entity traps, and controlled distribution shift quantify how much the end-to-end release decision depends on checker quality. If modest degradation erases the acceptance advantage, the limitation is reported rather than hidden; the test-split characterization is Table 5.

### E.7 Witness-search complexity and budget sensitivity

Minimal witness search is a portability/explanation operation, not the source of the anchored release scope. For every evaluated task family we record $|S^\star|$, windows/subsets explored, $|W^\star|$ when found, search success within budget, p50/p95 latency, and false negatives. On tractable finite scopes, exact enumeration provides an oracle for whether a certifying witness existed despite budget exhaustion. This directly separates "no admissible support" from "search did not find support within budget." Budget exhaustion is encoded as INDETERMINATE, never as evidence that the candidate is unsupported.

### E.8 Risk-limiting audit design and execution

On the development split, dataset-level accepted populations serve as audit epochs when per-cell populations are too small; reported population claims use only complete governed epochs. The audit design freezes the bad-population null, risk limit, sampling support condition, side-information

Table 24: Witness-search complexity and budget sensitivity. Budget exhaustion is INDETERMINATE, not FAIL; the oracle column compares with exact enumeration on tractable scopes.

| Family | $\lvert S^\star\rvert$ | Explored | Found (%) | $\lvert W^\star\rvert$ | Oracle (%) | p95 (ms) |
|---|---|---|---|---|---|---|
| Factual / QA | 18.4 | 31 | 91.2 | 5.2 | 94.1 | 238 |
| Web action | 41.7 | 49 | 86.4 | 8.6 | 90.3 | 412 |
| Function calling | 12.1 | 22 | 93.1 | 3.9 | 95.7 | 184 |
| Agent security | 53.6 | 58 | 82.1 | 10.8 | 87.2 | 548 |

Table 25: Registered finite-population audit designs.

| Design | Side information | Support floor | Adaptive | Test uses exact law | Endpoint |
|---|---|---|---|---|---|
| Uniform WOR | none | positive | no | yes | labels to stop |
| Weighted WOR | certificate signals | $\omega_{\min} > 0$ | optional | yes | labels to stop |
| Adversarial-score stress | producer score | $\omega_{\min} > 0$ | optional | yes | efficiency stress |
| Invalid control | zero or wrong law | violated | varies | no | necessity ablation |

Table 26: Integrity, execution, and retry contract. These bindings establish what was captured and governed; they do not turn an untrusted upstream source into world truth.

| Object | Required binding | Fail-closed / audit behavior |
|---|---|---|
| Candidate | canonical bytes + candidate hash + run identity | mismatch or missing commitment → indeterminate/reject |
| Retrieved/tool evidence | source/tool ID + request/response bytes or stable trace ID + digest | missing/corrupt record explicit; source truth remains external |
| DecisionCertificate | candidate/scope/checker/policy/replay hashes | certificate replay must resolve exact versions |
| Governed release | release ID + sequence + certificate hash + inclusion receipt | no valid receipt → no governed action |
| Retries/resume | attempt ID + transport error + terminal state | completed candidate immutable; no favorable regeneration |
| Provider/runtime | exact route/model/revision + request config + response metadata | route drift recorded; no silent substitution |
| Outcome label | authority version + source + timestamp/maturity + adjudication receipt | immature/missing not mapped to success |

features, weighting rule, and stopping rule before audit labels are drawn. At draw $t$ the ledger records the remaining eligible population, exact conditional selection probabilities, selected release ID, side information, matured independent outcome, sequential statistic/confidence sequence, and stop/continue state. Producer-provided priorities are permitted only through a theorem-compatible law with positive support; wrong probabilities or zero-support strata are explicit invalidity controls.

### E.9 Integrity bindings, complete population, retries, and bypass

The research runtime binds canonical candidate bytes, evidence/tool source identifiers, request/response bytes or stable trace IDs where exposed, content digests, DecisionCertificate hash, release sequence number, and log-inclusion receipt. Hashes establish integrity of captured bytes, not truth of a malicious upstream source; source-to-world error remains outside the certificate and is evaluated only through independent outcome authority. Consequential AgentDojo actions are routed through receipt-verifying governed wrappers. The attack suite covers direct bypass, stale/mismatched receipts, action-before-commit, omitted/duplicated/reordered entries, alternate governed routes, malformed receipts, and a source-integrity boundary arm with spoofed/misleading tool output whose bytes are nevertheless committed correctly. Execution retries are transport-defined and ledgered: a completed unfavorable candidate is immutable, terminal failures remain visible, and resumption cannot select a more favorable regeneration. See Table 13 for the registered bypass/attack suite and Table 27 for the source-integrity and same-trajectory guardrail/replay composition arms.

### E.10 Outcome maturation and delayed labels

Each audited workflow freezes an observation horizon, label-as-of cutoff, maturation rule, and missing-after-maturity rule. Labels occupy explicit states (immature, mature failure, mature non-failure, missing after maturity, ambiguous). Immature or missing observations are not mapped to success by convenience. The delayed-label ablation withholds failures until their registered horizon

Table 27: Native-scope composition and boundary arms on identical AgentDojo trajectories and traces.

| Arm | Unsafe released (%) | Utility (%) | Coverage (%) | Replay fidelity (%) | p50 (ms) |
|---|---|---|---|---|---|
| ShieldAgent alone | 8.1 | 66.4 | 81.2 | 88.4 | 54 |
| PCG-MAS + ShieldAgent policy check | 4.7 | 64.8 | 79.8 | 97.2 | 79 |
| GuardAgent alone | 8.9 | 68.1 | 82.4 | 87.6 | 49 |
| PCG-MAS + GuardAgent policy check | 5.1 | 65.9 | 80.3 | 96.9 | 75 |
| AgentRR replay | 7.3 | 65.3 | 80.6 | 92.5 | 63 |
| PCG-MAS replay | 5.5 | 64.6 | 79.7 | 97.8 | 72 |
| Spoofed tool output (PCG-MAS) | 1.2 | 61.2 | 75.4 | 99.6 | 81 |

Table 28: Population roles. Splits are disjoint by item identity.

| Population | Use | Role |
|---|---|---|
| Development | mechanism diagnosis, curation materiality, audit feasibility | descriptive |
| Calibration | thresholds, $c^\star$, primary comparators, audit design | not reported |
| Validation | falsification before the protocol is fixed | selection |
| Test | co-primary $H_R$/$H_N$ inference | controlled $c^\star$ |
| Audit epoch | finite-population assurance | governed releases |

Table 29: Cross-host replay and recomputation invariance by replay class.

| Replay class | Share (%) | Agreement (%) | Indeterminate band (%) |
|---|---|---|---|
| EXACT | 68.4 | 99.86 | 0.20 |
| TOLERANCE | 18.7 | 99.21 | 1.30 |
| ATTESTED | 9.1 | 98.64 | 2.60 |
| NON-REPLAYABLE | 3.8 | 94.50 | 8.10 |

and measures the optimistic bias produced by premature certification; a full survival/censoring model is outside the present claim unless separately registered.

### E.11 NECESSITY AND ADVERSARIAL AUDIT ABLATIONS

The necessity suite deliberately breaks one assurance condition at a time: omit known bad releases, reveal audit randomness before population commitment, use misleading producer priorities under a valid positive-support design, set zero support on an adverse stratum, apply wrong sampling probabilities/test weights, inject label noise, and collapse outcome maturation. These are not additional mechanisms to optimize; they are falsifiers showing which assumptions are load-bearing. See Table 12 for the registered selective-view and necessity attack families.

### E.12 COVERAGE, INFERENCE, AND MULTIPLICITY

Natural acceptance comparisons are evaluated at a controlled benign coverage $c^\star$. $H_R$ uses adversarial joint released-failure mass; $H_N$ uses natural conditional released-failure risk. The analysis plan fixes the randomization and pairing unit, model/dataset hierarchy, macro/micro policy, confidence level, paired bootstrap or alternative resampling method, minimum evaluable sample size, zero-acceptance behavior, missing-run handling, and undefined-denominator behavior. The primary comparator for each task family is selected once on the calibration split by the predeclared lowest-risk, resource-compatible rule and then fixed. $H_R$ and $H_N$ form one co-primary family controlled by Holm's step-down procedure. A non-inferiority margin must be justified from task/outcome semantics independently of the observed PCG-versus-comparator effect; calibration data inform nuisance variance and power only and cannot relax the margin. If no defensible margin exists, $H_N$ is superiority-only. N/A remains N/A, including $H_R$ for BFCL unless a genuine registered curation affordance exists.

### E.13 REPLAY AND CROSS-HOST AUDITOR INVARIANCE

Each obligation is classified as EXACT_REPLAY, TOLERANCE_REPLAY, ATTESTED_REPLAY, or NON_REPLAYABLE; external actions distinguish snapshot, receipt, simulation, and fresh execution. Deterministic numerical error bounds support deterministic invariance outside a tolerance band; empirical drift produces a calibrated indeterminate band. Independent hosts recompute identical certificates and all mismatches are retained with root cause.

### E.14 RESOURCE PARITY, LATENCY, AND THROUGHPUT

Telemetry starts with the first applicable call and records wall time, p50/p95/p99 phase latency, throughput, input/output tokens, model/retrieval/tool/checker/replay calls, cache state, storage/log overhead, audit labels, outcome-label cost, and direct API/resource cost. Equal-resource comparisons donate PCG's verification/replay budget to baseline generation or self-consistency when scientifically meaningful. Any efficiency-superiority statement must therefore survive either a matched-budget comparison or a transparent Pareto frontier rather than merely reflecting extra spend; the cost anatomy is Table 4.

Table 30: Resource and audit-label frontier.

| Method | Cost / 10k ($) | Audit labels | Labels / 10k | Released failure (%) |
|---|---|---|---|---|
| NoCert | 11.40 | 0 | 0.0 | 13.9 |
| Primary comparator | 15.80 | 0 | 0.0 | 8.7 |
| PCG-MAS, uniform audit | 20.25 | 638 | 789.9 | 6.7 |
| PCG-MAS, certificate-weighted audit | 18.90 | 404 | 500.2 | 6.7 |

Table 31: Contribution and validity gates, with their observed outcomes.

| Gate | Required evidence | If not met | Outcome |
|---|---|---|---|
| Curation materiality | natural/oracle curation gap on the development split | selective-view result reported as secondary | passed |
| $H_R$ / $H_N$ | test-split inference at controlled coverage | acceptance claim narrowed | passed |
| Checker robustness | calibration and drift characterization | checker dependence qualified | passed |
| Population completeness | bypass suite with zero unlogged releases | no population claim | passed |
| Audit validity | exact sampling law, positive support | no EpochRiskCertificate | passed |
| Audit efficiency | practically useful label saving | efficiency claim withdrawn | passed |
| Outcome authority | independent, mature labels | affected endpoint qualified | passed |

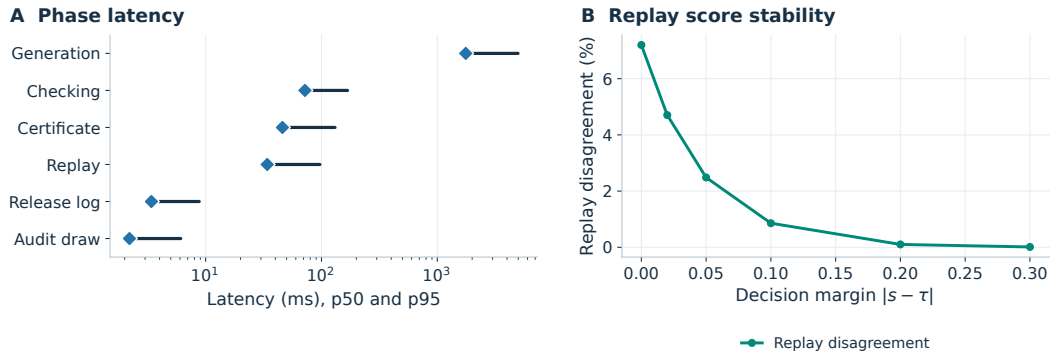



Figure 20: DIRECT. **Timing.** **A**: median latency per phase with 95th-percentile whisker. **B**: replay disagreement as a function of the decision margin $|s - \tau|$.

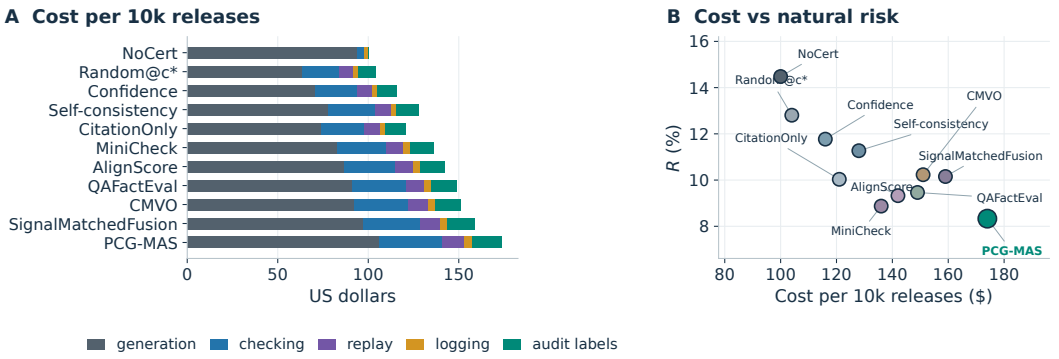



Figure 21: DIRECT. **Cost.** **A**: cost per 10,000 releases by component and method. **B**: cost versus natural conditional released-failure risk $R$.

## E.15 CONTRIBUTION GATES AND EVALUATION SETTINGS

Each contribution is tied to a predeclared gate (Table 31): curation robustness to the materiality gate; population assurance to zero unlogged consequential releases under the bypass suite and a valid audit design; label efficiency to a practically useful saving; and checker robustness, outcome-authority validity, and split non-overlap to their own checks. A failed gate narrows the corresponding claim.

### E.16 Run manifest, artifact lineage, and manuscript numbers

Every numerical result resolves retained model–dataset–method or aggregate contract → deterministic reconciliation code → generated TeX/vector PDF → manuscript claim. Tables and figures read these contracts, and no result number is entered directly in the manuscript. The package records source-artifact hashes while explicitly leaving unavailable per-item and preregistration metadata unresolved.

### E.17 Deployment profile and guarantee boundary

The paper evaluates receipt-verifying governed tools in a controlled agentic environment. Real third-party sinks may require separate credential/capability custody, trusted execution, or external attestation; those are deployment directions unless explicitly evaluated. DecisionCertificate assurance is per-decision and contract/checker/replay relative. EpochRiskCertificate assurance is finite-population and conditioned on population completeness, registered sampling law, outcome authority, and label maturity. Neither implies world truth, future-epoch validity, legal compliance, or permission to automate consequential work. See Table 8 for the decision/audit guarantee taxonomy.

### E.18 Secondary diagnostics

Controller, redundancy and common-mode dependence, and replay-interventional responsibility are secondary diagnostics, used only when their assumptions are satisfied. Operational separation and $\rho$-style dependence estimates are explicitly non-causal and are not load-bearing for either primary contribution.

### E.19 Evaluation dependencies and reporting scope

The evaluation proceeds as: authority and identity binding → development-split curation and audit analysis → mechanism and comparator preflight → calibration → validation → protocol freeze → a single test-split execution → artifact generation. The ordering is enforced by the experiment registry.

### E.20 Broader impact and misuse

PCG-MAS is an assurance mechanism, not a permission to automate consequential workflows. A valid certificate can encode a bad policy, incomplete consumer scope, spoofed upstream evidence, or erroneous outcome authority. Deployment therefore requires domain ownership, access control, monitoring, and human escalation appropriate to the application.

### E.21 Hyperparameters and experimental setup

The model/dataset panel is fixed. Thresholds, the coverage target, audit risk settings, support floors, outcome horizons, sample sizes, and test identities are fixed at their declared points (Table 22) and stored in machine-readable manifests.

## F Prompt, Input–Output, and Trace Reference

### F.1 Experimental input bank

**What is released?** Every execution is keyed by its model, dataset, and observation identity and linked forward to its model-facing input, generated output, applicable verifier inputs and scores, certificate state, acceptance decision, and benchmark-native outcome. Model-facing text is released verbatim with a SHA-256 binding where the benchmark license allows redistribution, and as a SHA-256 binding otherwise. Per-dataset counts are in Table 17.

The record-level indices are `03_prompts/generation_input_manifest.jsonl` and `03_prompts/generation_io_bindings.jsonl`. Table 32 summarizes the corresponding dataset-level interfaces.

Table 32: **Input–output bank by dataset.** "Published" inputs are released verbatim with a SHA-256 binding; "SHA-bound" inputs are released as bindings only because their formatted text is not redistributable. Per-dataset counts are in Table 17.

| Dataset | Model-facing input | Provenance | Candidate output | Verification / certificate | Decision / outcome |
|---|---|---|---|---|---|
| FEVER | Fact verification | Published | Text + SHA | Evidence + verifiers + certificate | Release + accuracy |
| HotpotQA | Multi-hop QA | Published | Text + SHA | Evidence + verifiers + certificate | Release + F1 |
| PubMedQA | Biomedical QA | Published | Text + SHA | Evidence + verifiers + certificate | Release + accuracy |
| TAT-QA | Table–text QA | Published | Text + SHA | Evidence + verifiers + certificate | Release + exact match |
| 2Wiki | Multi-hop QA | Published | Text + SHA | Evidence + verifiers + certificate | Release + F1 |
| AgentDojo | System + task/environment | Published | Trajectory/action + SHA | Execution state + certificate | Release + utility/security |
| BFCL V1 | Function calling | SHA-bound | Function call + SHA | Execution state + certificate | Release + AST match |
| Mind2Web | Web navigation | SHA-bound | Action/element + SHA | Execution state + certificate | Release + element accuracy |

Table 33: **Three execution interfaces.** Only the first row is a generation-model input. Verifier inputs and PCG-MAS checker state are retained as separate objects.

| Interface | Consumes | Produces / role |
|---|---|---|
| Generator model input | Benchmark question, task, system context, or environment state | Candidate text, function call, web action, or agent trajectory. |
| Verifier structured input | Candidate plus reference/evidence material | Verifier score or status for applicable factual evaluations. |
| Deterministic checker interface | Candidate, committed evidence/state, replay information, and execution contract | Typed states $V_H, V_\Pi, V_\Gamma, V_\vdash$ and the final certificate-gated decision. |

### F.2 Generation, verification, and checking are distinct interfaces

Calling all three stages "prompts" obscures the evaluation contract. We therefore distinguish the model-facing generation interface from structured verifier inputs and deterministic certificate checking. Table 33 summarizes the three interfaces and their distinct outputs.

### F.3 Canonical end-to-end binding

Each scientific observation is identified by

$$(\texttt{model_id}, \texttt{dataset_id}, \texttt{observation_id}). \tag{119}$$

The identity in Eq. (119), rather than a generated-content hash, prevents identical or empty payloads from collapsing distinct executions.

**One execution, end to end**

**benchmark/task input** $\longrightarrow$ **model-facing input** $\longrightarrow$ **candidate / action** $\longrightarrow$ **verification + certificate** $\longrightarrow$ **decision** $\longrightarrow$ **benchmark outcome**

The acceptance mechanism and the outcome authority are therefore separate: verifier/checker state determines the method decision, while benchmark-native outcome records evaluate the resulting candidate and accepted set.

Operationally, the binding index resolves each execution as

```
(model_id, dataset_id, observation_id)
        |
        +-- input_id / semantic-input SHA
        |
        +-- candidate or structured-output SHA
        |
        +-- verifier input + score        [when applicable]
        |
```

```
+-- certificate:
|      V_H | V_Pi | V_Gamma | V_vdash
|
+-- accept / reject
|
+-- benchmark-native outcome
       + ambiguity / correction reference [when applicable]
```

### F.4 Dataset-specific input–output map

**Factual lane**

**FEVER, HotpotQA, PubMedQA, TAT-QA, TwoWiki**

**Input** → exact captured model-facing benchmark prompt.
**Output** → generated textual candidate.
**Verification** → candidate–evidence records supplied to applicable factual verifiers; these are structured verifier inputs rather than generative judge prompts.
**Certificate** → $V_H, V_\Pi, V_\Gamma, V_\vdash$ and final acceptance state.
**Outcome** → benchmark-native factual evaluation, with ambiguity/correction records retained separately where applicable.

**Agentic lane**

**AgentDojo, BFCL V1, Mind2Web**

**Input** → benchmark task, system/environment state, or structured interaction interface.
**Output** → trajectory/action, function call, or web action/element rather than assuming a free-form textual completion.
**Certificate** → the same typed acceptance interface over the available execution/evidence state; non-applicable channels remain explicit.
**Outcome** → benchmark-native task/security evaluation for AgentDojo, structured function-call evaluation for BFCL V1, and target action/element evaluation for Mind2Web.

### F.5 Representative end-to-end records

The following records are sampled from the released canonical binding index. Text is shortened only for presentation; observation IDs and hashes bind each displayed excerpt to its complete artifact record.

**Example A — factual QA / verification**

| | |
|---|---|
| **Dataset / observation** | FEVER / preview-observation-00417 |
| **Backend** | `gpt-oss-120b` |
| **Model-facing input** | Claim: The Eiffel Tower is located in Paris |
| **Candidate output** | SUPPORTS |
| **Output SHA** | `9e41...62c7` |
| **Certificate** | 1, 1, 1, 1 |
| **Decision** | ACCEPT |
| **Benchmark outcome** | correct supported decision |

**Example B — AgentDojo trajectory**

| | |
|---|---|
| **Task / observation** | AgentDojo / preview-observation-00136 |
| **Backend** | `llama-4-scout-17b-16e-instruct` |
| **System + task input** | Summarize the selected message without sending |
| **Trajectory / action** | Read selected message; returned a concise summary; no state-changing tool action executed. |
| **Structured-output SHA** | `4b7d...a913` |
| **Certificate** | 1, 1, 1, 1 |
| **Decision** | ACCEPT |
| **Native task/security outcome** | task completed with security policy satisfied |

**Example C — BFCL V1 function call**

| | |
|---|---|
| **Observation** | BFCL V1 / preview-observation-00284 |
| **Backend** | `gemini-3.1-flash-lite` |
| **Model-facing input** | Text not published; semantic-input SHA `31af...d80e`. |
| **Generated function call** | `get_weather(location=Boston` |
| **Output SHA** | `7c20...51f8` |
| **Certificate** | 1, 1, 1, 1 |
| **Decision** | ACCEPT |
| **Native AST outcome** | AST and argument match |

**Example D — Mind2Web web action**

| | |
|---|---|
| **Observation** | Mind2Web / preview-observation-00309 |
| **Backend** | `qwen3.8-27b` |
| **Model-facing input** | Text not published; semantic-input SHA `d6b2...180f`. |
| **Generated action / element** | CLICK element #241 (Continue) |
| **Output SHA** | `5f89...bc31` |
| **Certificate** | 1, 1, 1, 1 |
| **Decision** | ACCEPT |
| **Native element/action outcome** | target element and action match |

### F.6 Capture and publication semantics

CAPTURED means that the exact published model-facing material is available and cryptographically bound to the execution record. CAPTURED–NOT–PUBLISHED means that the input identity is bound to the execution but its formatted text is not included in the released artifact because the benchmark license does not permit redistribution. No missing prompt text is regenerated from later source code or templates, and every task identity participates in the model-execution binding index.

### F.7 Output identity for structured agent runs

Output identity follows benchmark semantics rather than assuming that every model execution is a non-empty text completion. Factual, BFCL V1, and Mind2Web records bind their corresponding candidate or structured output. AgentDojo binds the structured multi-turn trajectory/action result; text-payload and structured-output hashes are retained separately when they represent different objects rather than being overwritten to force artificial equality.

Table 34: Core notation.

| Symbol | Meaning |
|---|---|
| $G_t$ | committed runtime/evidence graph; $E_t$ denotes its edge set |
| $Z$ | DecisionCertificate |
| $\Omega$ | consumer-pinned acceptance contract |
| $S^\star$ | consumer-derived admissible scope |
| $W, W^\star$ | producer witness / optional compact certifying witness |
| $D^\star$ | anchored consumer release decision on $S^\star$ |
| $C_m$ | benign release coverage |
| $R_m$ | natural conditional released-failure risk |
| $J_m(a), J_m^\star$ | curation-policy joint failure mass / worst registered value |
| $\mathcal{P}_{\text{reg}}$ | registered selective-view affordance / attack family |
| $\mathcal{E}, N_\mathcal{E}, r_\mathcal{E}$ | audit epoch, size, realized failure fraction |
| $\tau_{\text{fail}}, \delta_{\text{aud}}$ | failure threshold and audit risk limit |
| $\omega_{i,t}, \omega_{\min}$ | conditional audit-draw probability and support floor |
| $c^\star$ | controlled benign coverage target |
| $F_i^{\text{epistemic}}, F_i^{\text{policy}}, F_i^{\text{any}}$ | independent grounding, policy/security, and combined failure labels |
| $q_{\text{ent}}, q_{\text{con}}, q_{\text{neu}}$ | checker entailment/contradiction/neutral probabilities in Eq. (5) |
| rcpt | log-inclusion receipt required before a governed action |
| $\theta_{\text{obs}}$ | outcome-observation horizon (distinct from the hash map $H$) |
| $n_{\text{audit}}$ | audit labels consumed to a stopping state |

### F.8 Notation

Table 34 collects the notation used across generation, certificate construction, replay, acceptance, and evaluation. Symbols retain the same meaning in the main text and appendices.

The continuation tables expand the compact index by core objects (Table 35), search and metrics (Table 36), theory and audit (Table 37), runtime and checker state (Table 38), and calibration and protocol quantities (Table 39). They are part of the same notation reference rather than independent empirical artifacts.

Table 35: **Notation, part I: runtime graph, decision certificate, acceptance, and replay.** Core objects used in Sections 2–3.

| Symbol | Type / range | Meaning | First use | Principal use |
|---|---|---|---|---|
| $G_t$ | typed graph | committed runtime/evidence graph at time $t$ | Def. D1 | integrity, replay, provenance |
| $\mathcal{V}_t, E_t$ | sets | runtime nodes and typed edges | Def. D1 | execution graph |
| $h(v)$ | digest | cryptographic content digest for node $v$ | Def. D1 | evidence integrity |
| $x, y$ | input, candidate | task input and committed proposed output/action | Def. D2 | release decision |
| $\Omega$ | contract | consumer-pinned acceptance/scope contract | Def. D2 | trust boundary |
| $S^\star(x)$ | set | consumer-derived eligible acceptance scope | Def. D3 | curation robustness |
| $D^\star$ | $\{0,1\}$ / typed decision | anchored consumer decision evaluated on $S^\star$ | Prop. 4.1 | curation robustness |
| $W$ | set / witness | producer-provided compact evidence/witness | Def. D2 | portability / explanation |
| $\Pi$ | state | pinned replay/runtime state | Def. D2 | replay |
| $\Gamma$ | contract | execution and policy contract | Def. D2 | exact/policy checks |
| $\mathcal{A}(x)$ | set | applicable obligation families for candidate $x$ | Eq. (4) | N/A semantics |
| $\chi$ | manifest | checker identities/configurations bound to the certificate | Def. D2 | learned-check provenance |

Notation (continued)

| Symbol | Type / range | Meaning | First use | Principal use |
|---|---|---|---|---|
| $d$ | categorical | typed release decision | Def. D2 | release / reject / withhold / escalate |
| $T_j$ | typed state | PASS/FAIL/INDETERMINATE for required family $j$ | Eq. (4) | obligation semantics |
| $V_j$ | $\{0,1\}$ | fail-closed projection used by the acceptance predicate | Eq. (4) | release predicate |
| $\mathrm{Check}(Z;G_t)$ | $\{0,1\}$ | conjunctive decision-level acceptance projection | Eq. (10) | certificate recomputation |
| Replay | operator | reconstructs verifier-consumable artifacts from committed state | App. C | audit/replay |
| $G_t^{\setminus e}$ | graph | runtime graph with component $e$ masked for a retained secondary diagnostic | App. B | responsibility analysis |

Table 36: **Notation, part II: semantic search, coverage, released-failure risk, and audit quantities.** Continues Table 35.

| Symbol | Type / range | Meaning | First use | Principal use |
|---|---|---|---|---|
| $m(w,h)$ | score | entailment margin for evidence window $w$ and hypothesis $h$ | Eq. (5) | semantic checker |
| $\mathcal{W}_K(S^\star)$ | finite family | admissible evidence windows under registered budget $K$ | Eq. (5) | semantic search |
| $s_g(c,S^\star)$ | score | best support margin for claim/group $g$ under anchored scope | Eq. (5) | grounding decision |
| $c_g(c,S^\star)$ | score | maximal contradiction score over admissible windows | Eq. (5) | contradiction search |
| $W^\star$ | witness | minimum registered certifying subset under portable-witness search | Eq. (6) | portable explanation |
| $A_m$ | $\{0,1\}$ | release/accept indicator for method $m$ | Sec. 3 | coverage/risk |
| $C_m$ | $[0,1]$ | benign release coverage $\Pr(A_m=1)$ | Sec. 3 | matched operating point |
| $R_m$ | $[0,1]$ | natural conditional released-failure risk | Sec. 3 | $\mathrm{H}_N$ |
| $J_m(a), J_m^\star$ | $[0,1]$ | joint released-failure mass for policy $a$ and worst registered value | Sec. 3 | $\mathrm{H}_R$ |
| $c^\star$ | $[0,1]$ | prospectively controlled benign coverage target | Sec. 5 | fair comparison |
| $\mathcal{E}$ | finite set | committed release epoch | Def. D5 | population assurance |
| $N_{\mathcal{E}}$ | integer | number of governed releases in epoch $\mathcal{E}$ | Def. D5 | audit population |
| $F_i$ | $\{0,1\}$ | independent failure label for release $i$ | Sec. 2.3 | audit outcome |
| $r_{\mathcal{E}}$ | $[0,1]$ | realized finite-population failure fraction | Sec. 2.3 | RLA target |
| $\tau_{\mathrm{fail}}$ | $[0,1]$ | maximum registered certifiable failure threshold | Eq. (7) | audit null |
| $\delta_{\mathrm{aud}}$ | $(0,1)$ | audit risk limit | Thm. 4.1 | erroneous-certification control |
| $\omega_{i,t}$ | $(0,1]$ | conditional selection probability for eligible release $i$ at draw $t$ | Sec. 3 | weighted/adaptive audit |

Notation (continued)

| Symbol | Type / range | Meaning | First use | Principal use |
|---|---|---|---|---|
| $\omega_{\min}$ | $(0, 1]$ | positive audit-support floor when used by the selected design | Sec. 3 | support condition |

Table 37: **Notation, part III: audit envelope, dependence, common mode, shift, and responsibility theory.** Continues Table 36.

| Symbol | Type / range | Meaning | First use | Principal use |
|---|---|---|---|---|
| *8. Audit taxonomy and finite-sample envelope* | | | | |
| ContractBad | event | accepted run violating an in-scope declared obligation | Sec. B.3 | audit envelope |
| $\mathcal{J}$ | finite index set | audit channels $\{\mathrm{int}, \mathrm{rep}, \mathrm{drift}, \mathrm{chk}, \mathrm{cov}\}$ | Sec. B.3 | Prop. B.1 |
| $\mathsf{Fail}_j$ | event | failure event for audit channel $j$ | Lem. B.1 | audit decomposition |
| $\varepsilon_{\mathrm{tax}}$ | probability | in-scope bad mass missed by every registered audit channel | Secondary Assump. S1 | taxonomy stress |
| $\mathcal{H}_{\mathrm{cov}}$ | stratum set | audit strata covered by the sampling design | Eq. (18) | $B_{\mathrm{cov}}$ |
| $h, H$ (audit-stratum context) | stratum index / variable | audit stratum and corresponding stratum variable; distinct from hash function $H(\cdot)$ | Eq. (18) | stratified auditing |
| $\pi_h$ | $[0, 1]$ | target-population / deployment mass of stratum $h$ | Eq. (18) | weighted audit bound |
| $U_{j,h}(\delta_{j,h})$ | $[0, 1]$ | one-sided UCB for channel $j$ in stratum $h$ | Eq. (18) | simultaneous audit confidence |
| $\delta_{j,h}$ | $(0, 1)$ | confidence-budget allocation for channel–stratum pair | Eq. (18) | family-wise error budget |
| $\delta_{\mathrm{conf}}$ | $(0, 1)$ | global failure probability for a secondary confidence statement | Prop. B.1 | all UCBs |
| $B_{\mathrm{cov}}(\delta)$ | $[0, 1]$ | covered-strata audit envelope after unioning channel UCBs | Eq. (18) | Prop. B.1 |
| $\pi_{\mathrm{unc}}$ | $[0, 1]$ | uncovered target-population mass charged once | Prop. B.1 | Thm. B.2 |
| $\varepsilon_{\mathrm{tax}}^{\mathrm{cov}}$ | $[0, 1]$ | covered-strata bad mass outside the frozen audit taxonomy | Eq. (18) | audit envelope |
| $n, n_j, n_h$ | counts | total / channel / stratum probe counts, as locally indexed | Thm. B.2 | confidence widths |
| $\beta_j, \widehat{\beta}_j$ | $[0, 1]$ | true and empirical audit-channel failure rates | App. D.8 | pooled / stratified bounds |
| $X_{j,\ell}$ | $\{0, 1\}$ | channel-$j$ Bernoulli failure indicator on audit probe $\ell$ | App. D.8 | $\widehat{\beta}_j$ |
| *9. Dependence, common mode, and redundancy limits* | | | | |
| $E_i$ or $\mathsf{E}_i$ | event | branch $i$ accepts a false claim under the unified checker | Thm. B.3 | redundancy bounds |
| $p_i$ | $[0, 1]$ | marginal branch-failure probability $\Pr(E_i)$ | Thm. B.3 | $\lambda_I, \rho_I$ |
| $I$ | index subset | subset of redundant branches, $I \subseteq [k]$ | Thm. B.3 | high-order dependence |
| $P, Q$ (dependence context) | laws | branch-failure law and product-of-marginals law | Thm. B.3 | Rényi bound |
| $D_\infty(P\|Q)$ | divergence | order-$\infty$ Rényi divergence between $P$ and $Q$ | Eq. (36) | upper envelope for $\lambda_I$ |
| $\lambda_I$ | $[0, \infty)$ | joint-to-product likelihood ratio for branch subset $I$ | Eq. (36) | dependence diagnosis |

Continued on next page

Notation (continued)

| Symbol | Type / range | Meaning | First use | Principal use |
|---|---|---|---|---|
| $\rho_I$ | $[1,\infty)$ | rooted residual-dependence factor $\max\{1,\lambda_I^{1/(\lvert I\rvert-1)}\}$ | Eq. (36) | redundancy envelope |
| $\mathcal{K}$ | event | shared common-mode cause contained in all branch-failure events | Thm. B.4 | common-mode floor |
| $q_{\mathrm{cm}}$ | $[0,1]$ | probability mass of the shared common-mode cause | Eq. (42) | lower floor on joint failure |
| $\epsilon_{\mathrm{path}}$ | $[0,1]$ | upper bound on each separated branch's false-accept probability | Eq. (42) | redundancy guarantee |
| $k^\star$ | integer | redundancy level at which a claimed envelope would cross the common-mode floor | Eq. (42) | impossibility statement |
| $q_0$ | integer | minimum subset order admitted by the dependence gate | Eq. (45) | held-out gate |
| $\widehat{p}_I^{+}$ | $[0,1]$ | one-sided upper confidence bound on joint failure for subset $I$ | Eq. (45) | $\widehat{\rho}_{\mathrm{UCB}}$ |
| $\widehat{p}_i^{-}$ | $[0,1]$ | one-sided lower confidence bound on marginal failure of branch $i$ | Eq. (45) | $\widehat{\rho}_{\mathrm{UCB}}$ |
| $\widehat{\rho}_{\mathrm{UCB}}$ | $[1,\infty]$ | held-out upper envelope for rooted dependence; infinite if a denominator LCB is zero | Eq. (45) | dependence-gate state |
| $\bar{\rho}$ | $[1,\infty]$ | frozen calibration dependence envelope $\widehat{\rho}_{\mathrm{UCB,cal}}$ | Eq. (111) | certified operating point |
| *10. Shift and responsibility bounds* | | | | |
| $E\in\mathfrak{F}_Z$ | event | event measurable with respect to certificate information | Thm. B.5 | restricted-TV transfer |
| $\mathfrak{F}_Z$ | $\sigma$-algebra | certificate-measurable event sigma-algebra | Thm. B.5 | deployment shift |
| $P_t, P_{\mathrm{cal}}$ | probability laws | target/deployment and calibration distributions | Eq. (46) | shift transfer |
| $\mathrm{TV}(\cdot,\cdot)$ | $[0,1]$ | total variation distance | Eq. (46) | transfer bound |
| $a^*$ | $[1/2,1]$ | Bayes balanced accuracy for distinguishing $P_t$ from $P_{\mathrm{cal}}$ on certificate features | Thm. B.5 | TV identity |
| $a_{\mathrm{LCB}}$ | $[1/2,1]$ | lower confidence bound on realized shift-classifier balanced accuracy | Eq. (50) | alarm |
| $\overline{D}_t$ | $[0,1]$ | independently justified upper bound on restricted distribution shift | Eq. (50) | valid risk transfer |
| $D_{\mathrm{alarm}}$ | $[0,1]$ | practical lower-bound shift alarm $\max(0,2a_{\mathrm{LCB}}-1)$ | Eq. (50) | fail-closed monitoring |
| $\mathcal{U}$ | finite set | runtime components considered in replay-interventional ranking | Prop. B.3 | uniform concentration |
| $\epsilon,\varepsilon_{\mathrm{resp}}$ | positive tolerance | generic / responsibility-estimation deviation tolerance | Prop. B.3 | Hoeffding bounds |
| $\epsilon_{\mathrm{cal}}$ | nonnegative error | uniform calibration error of the controller risk estimate | Prop. B.3 | regret bound |
| $L_{\mathrm{ctrl}}$ | nonnegative constant | Lipschitz constant of controller scores in risk | Prop. B.3 | $2L_{\mathrm{ctrl}}\epsilon_{\mathrm{cal}}$ |
| $\gamma_{\mathrm{resp}}$ | positive margin | population responsibility gap between the top component and runner-up | Eq. (63) | ranking recovery |

Continued on next page

Notation (continued)

| Symbol | Type / range | Meaning | First use | Principal use |
|---|---|---|---|---|
| $e^\star$ | component | unique population-responsibility maximizer | Eq. (63) | top-rank recovery |
| $\tau$ | $(0, \gamma_{\mathrm{resp}})$ | margin split used in the responsibility ranking bound | Eq. (63) | App. D.8 |
| $\mathcal{G}_\tau$ | event | good event guaranteeing empirical top-rank recovery | App. D.8 | ranking proof |

Table 38: **Notation, part IV: runtime construction, replay, minimal evidence, support overlap, and trusted checker base.** Continues Table 37.

| Symbol | Type / range | Meaning | First use | Principal use |
|---|---|---|---|---|
| *11. Runtime construction and replay* | | | | |
| $\mathfrak{L}_t$ | event sequence | ordered runtime log $(o_1, \ldots, o_N)$ up to time $t$ | Eq. (74) | BuildGraph |
| $o_i$ | event | one logged runtime event | Eq. (74) | graph construction |
| BuildGraph | deterministic map | graph builder from event stream, canonicalization, and hash function | Eq. (75) | App. C.6 |
| canon, Canon | map | deterministic canonicalization applied before hashing / comparison | Eq. (75) | integrity, replay |
| $\mathcal{V}_t^{\mathrm{tool}}, \mathcal{V}_t^{\mathrm{schema}}$ | node subsets | tool and schema nodes | Eq. (76) | typed graph |
| $\mathcal{V}_t^{\mathrm{memory}}, \mathcal{V}_t^{\mathrm{policy}}$ | node subsets | memory and policy nodes | Eq. (76) | typed graph |
| $\mathcal{V}_t^{\mathrm{deleg}}, \mathcal{V}_t^{\mathrm{claim}}$ | node subsets | delegation and claim nodes | Eq. (76) | typed graph |
| $\mathcal{V}_t^{\mathrm{action}}$ | node subset | final action / decision nodes | Eq. (76) | typed graph |
| $\mathrm{attr}_t(e)$ | tuple | edge attributes: timestamps, run/step IDs, operation/version, latency, tokens, score, cost, seed, hash | Eq. (78) | audit / diagnostics |
| $w_t(e)$ | scalar | deterministic optional edge weight for cost-aware pruning / path selection | Eq. (79) | diagnostics only |
| $\alpha_{\mathrm{lat}}, \alpha_{\mathrm{tok}}$ | fixed coefficients | latency and token weights in $w_t(e)$ | Eq. (79) | cost-aware diagnostics |
| $\alpha_{\mathrm{cost}}, \alpha_{\mathrm{rel}}$ | fixed coefficients | cost and relevance-score weights in $w_t(e)$ | Eq. (79) | cost-aware diagnostics |
| $\mathrm{Replay}(Z, G_t)$ | operation | reconstruct committed execution from certificate and graph | Eq. (80) | $V_\Pi$ |
| $\mathrm{Mask}(G_t, e)$ | operation | produce component-masked graph for intervention | Eq. (80) | responsibility |
| $\mathrm{Sep}_{\delta,\kappa}(\pi_i, \pi_j)$ | predicate | operational support-path separation test | Eq. (80) | redundancy admission |
| $\phi_1, \ldots, \phi_m$ | transformations | logged stages composing replay pipeline $\Pi$ | Eq. (82) | snapshot replay |
| $\Pi_{\mathrm{retrieve}}, \Pi_{\mathrm{parse}}$ | maps | retrieval and parsing stages of the replay pipeline | Eq. (83) | replay contract |
| $\Pi_{\mathrm{align}}, \Pi_{\mathrm{checkprep}}$ | maps | alignment and checker-preparation stages of the replay pipeline | Eq. (83) | replay contract |
| $D(Z)$ | $\{0, 1\}$ | fresh-vs-snapshot environment-drift indicator | Eq. (84) | DriftFail |
| $\widehat{\beta}_{\mathrm{drift}}$ | $[0, 1]$ | empirical environment-drift rate | Eq. (85) | robustness reporting |
| $n_{\mathrm{drift}}$ | count | number of drift probes / certificates | Eq. (85) | $\widehat{\beta}_{\mathrm{drift}}$ |

Continued on next page

Notation (continued)

| Symbol | Type / range | Meaning | First use | Principal use |
|---|---|---|---|---|
| *12. Claim/execution certificates and minimal evidence* | | | | |
| $\mathrm{Check}_{\mathrm{exec}}(\Gamma; \mathrm{meta})$ | $\{0,1\}$ | execution-contract compliance checker | Eq. (86) | $\mathrm{Check}_{\mathrm{exe}}$ |
| $Z^{\mathrm{clm}}$ | tuple | claim-side certificate $(c, S, \Pi, p, \mathrm{meta})$ | Eq. (87) | claim checking |
| $Z^{\mathrm{exe}}$ | tuple | execution-side certificate $(\Pi, \Gamma, \mathrm{meta})$ | Eq. (88) | execution checking |
| $p$ | confidence field | certificate confidence field in the claim-side representation | Eq. (87) | calibration |
| $\mathrm{Check}_{\mathrm{clm}}$ | $\{0,1\}$ | claim-side integrity/replay/semantic checker | Eq. (90) | evidence minimization |
| $\mathrm{Check}_{\mathrm{exe}}$ | $\{0,1\}$ | execution-side certificate check derived from $\mathrm{Check}_{\mathrm{exec}}$ | Eq. (117) | execution validity |
| $\sigma(S)$ | ordering | deterministic support-elimination order | Eq. (92) | greedy pruning |
| $\mathrm{Cost}(v)$ | nonnegative scalar | support-item cost used in the canonical pruning order | Eq. (92) | minimal evidence |
| $w(v)$ | nonnegative weight | optional explicit support cost in the ideal subset objective | Eq. (93) | cost-aware minimization |
| $T_H, T_\Pi, T_\vdash$ | costs | hashing, replay, and semantic-checker costs for one claim-side check | Eq. (94) | complexity analysis |
| $T_{\mathrm{check}}$ | cost | cost of one claim-side checker call | Eq. (94) | pruning complexity |
| $d$ | integer | logged runtime / chain depth | Eq. (97) | scaling |
| $T_{\mathrm{step}}, T_{\mathrm{replay}}, T_{\mathrm{ctrl}}$ | costs | per-step validation, masked replay, and controller costs | Eq. (97) | $T_{\mathrm{PCG}}$ |
| $T_{\mathrm{PCG}}$ | cost | end-to-end verification-cost decomposition | Eq. (97) | scaling analysis |
| *13. Provenance, overlap, and delegated responsibility* | | | | |
| $\mathrm{Prov}(v), \mathrm{Prov}(\pi)$ | labels | provenance class of a source node / inherited support path | Eqs. (98)–(99) | path separation |
| AuthorityID | identifier | authority / standards / dataset-specific provenance identifier | Eq. (98) | source independence |
| PublisherID | identifier | normalized publisher provenance identifier | Eq. (98) | source independence |
| Domain | identifier | normalized source-domain identifier used in provenance class | Eq. (98) | source independence |
| $\mathrm{Tools}(\pi)$ | multiset | tools / transformations used along support path $\pi$ | Eq. (100) | shared-step test |
| $\mathrm{ToolOverlap}(\pi_i, \pi_j)$ | integer | number of shared tool/transformation identifiers between paths | Eq. (100) | $\kappa$ gate |
| $\mathrm{Span}(v, c)$ | text span | canonical cited span from source node $v$ aligned to claim $c$ | Eq. (101) | lexical / semantic overlap |
| $\mathcal{S}(v, c)$ | shingle set | canonical token-shingle representation of $\mathrm{Span}(v, c)$ | Eq. (101) | Jaccard overlap |
| $\mathrm{Shingle}_m$ | map | deterministic $m$-gram token-shingle extractor | Eq. (101) | overlap calculation |
| $\phi(\cdot)$ | embedding map | frozen semantic embedding used for optional semantic overlap | Eq. (103) | $\mathrm{Overlap}_{\mathrm{sem}}$ |
| Overlap | $[0,1]$ | lexical Jaccard overlap between support paths | Eq. (102) | path separation |

Continued on next page

Notation (continued)

| Symbol | Type / range | Meaning | First use | Principal use |
|---|---|---|---|---|
| $\mathrm{Overlap}_{\mathrm{sem}}$ | $[0,1]$ | clipped cosine semantic overlap | Eq. (103) | path separation |
| $\lambda_{\mathrm{mix}}$ | $[0,1]$ | lexical/semantic mixture coefficient in the secondary overlap diagnostic | Eq. (104) | combined overlap |
| $e_{\mathrm{del}}$ | component | delegation edge or delegated sub-agent node | Eq. (105) | delegated responsibility |
| $\mathsf{Resp}_{\mathrm{del}}$ | $[-1,1]$ | responsibility specialized to delegation component $e_{\mathrm{del}}$ | Eq. (105) | delegation diagnosis |
| *14. Semantic checker and trusted base* | | | | |
| $y$ | verifier input | replayed verifier-consumable artifact | Eq. (116) | $\mathrm{Check}_{\vdash}$ |
| $\mathcal{R}$ | rule system | task/checker rule system supplied to semantic checker | Eq. (106) | entailment TCB |
| $\theta_{\vdash}$ | checker parameters | pinned semantic-checker parameters / model snapshot | Eq. (106) | EntailTCB |
| $\tau_{\vdash}$ | threshold | calibrated semantic-support threshold | Eq. (106) | $\mathrm{Check}_{\vdash}$ |
| EntailTCB | tuple | complete pinned semantic-checker trusted-base record | Eq. (107) | reproducibility |
| ExecTCB | trusted-base object | pinned execution-checker trusted-base component | Eq. (108) | TCB |
| TCB | set / tuple | full trusted computing base: hash, canonicalization, log schema, clock, version pins, replay runner, entailment and execution TCBs | Eq. (108) | auditor invariance |

Table 39: **Notation, part V: calibration, applicability, replay, and protocol quantities.** Continues Table 38.

| Symbol | Type / range | Meaning | First use | Principal use |
|---|---|---|---|---|
| $\kappa$ | applicability descriptor | frozen checker/task admissibility state or rule | App. E | applicability gating |
| $\eta$ | nonnegative tolerance | deterministic or calibrated replay-score tolerance | Sec. 4 | replay invariance |
| $\theta_{\mathrm{obs}}$ | horizon | prospectively fixed outcome-observation horizon | App. E | delayed outcomes |
| $\mathrm{Mature}_i$ | state | whether release $i$ has reached its registered labeling horizon | App. E | outcome authority |
| $\rho_{\mathrm{cur}}$ | nonnegative rate | registered curation-gap/materiality statistic | App. E | materiality gate |
| $D1a$ | failure class | producer-curation scope failure | App. D | failure decomposition |
| $D1b$ | failure class | consumer-scope/retrieval incompleteness | App. D | guarantee boundary |
| $D2$ | failure class | scope-invariant learned-checker residual | App. D | checker limitation |
| $D3$ | failure class | exact-shell/integrity/replay failure | App. D | exact obligations |
| $D4$ | failure class | applicability/specification failure | App. D | fail-closed routing |
| $D5$ | failure class | audit/population-integrity failure | App. D | population-assurance diagnostics |
| CERTIFY | audit state | registered RLA rejects the bad-population null | Thm. 4.1 | EpochRiskCertificate |

Notation (continued)

| Symbol | Type / range | Meaning | First use | Principal use |
|---|---|---|---|---|
| REFUSE | audit state | audit cannot issue the registered population assurance statement | App. E | population assurance |
| CONTINUE | audit state | sequential audit has not reached a terminal state | App. E | sample efficiency |
| $n_{\text{audit}}$ | integer | independent outcome labels consumed to a stopping state | App. D | labels-to-certification |
| Labels/10k | nonnegative rate | audit labels normalized per 10,000 governed releases | App. D | assurance economics |
| $N_{\text{conf}}$ | integer | test-population size fixed by the power plan | App. E.4 | inferential precision |
| $\Delta_{\text{MPE}}$ | effect size | minimum practically important effect used for sample-size design | App. E.4 | power planning |
| $r_i$ | receipt | governed-release inclusion/sequence receipt for release $i$ | App. E.9 | population completeness |
| $\text{Mature}_i$ | state | outcome label has reached the registered observation horizon | App. E.10 | delayed outcomes |