MONITORING THROUGH A NOISY SENSOR: SOUNDNESS TRANSFER AND NET-HARM THRESHOLDS FOR GUI-AGENT VERIFICATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Runtime verification for LLM agents reports compliance at or near one hundred percent, and every such result assumes *readable* agent state: structured events, tool arguments, logs. A GUI agent has none. Its predicates must be inferred from pixels by an operator that is wrong at rate ε and silent on a $1-\gamma$ share of atoms, so the object analysed is a monitor *plus a noisy sensor*. Verification under noisy observation exists but parameterises error against a *model* a pixel-driven agent does not have; and no deployed monitor formulation can represent the state "the sensor could not resolve this atom", so it is recorded as a clean verdict. We give the transfer theorem in the parameters that survive — soundness and completeness as closed-form functions of precision, recall and *coverage*, a hole no operating point closes — together with a threshold ε^* past which monitoring is net *harmful*, a quantity that depends on the base rate and the cost ratio and on nothing else: not on coverage, not on formula shape, not on the dependence among sensor errors. On oracle atoms our reproduction gate returns soundness 1.000, completeness 1.000 and a false-block rate of 0.000, reproducing the compliance figure the field quotes; substituting pixels for the oracle drives two published monitors and one of ours into the net-harm regime. The threshold gets the sign right in 9 of 9 rows over 240 episodes, and we report what that does and does not establish: one negative row separates at $p < 0.05$, while the decisive row sits at $\varepsilon_{\text{eff}}{=}0.246$ against $\varepsilon^*{=}0.238$ and cannot be distinguished from it ($p = 0.76$), so calibrated simulation at $n{=}3000$ per cell resolves the crossing instead, bracketing it in $[0.220, 0.250]$ against a predicted 0.2375. The sharpest result needed no new experiment: 76 of 240 episodes contain an atom the operator would not resolve, and both published two-valued monitors record every one as a clean verdict — a guarantee that is *vacuous* rather than violated, and invisible to any compliance rate. One preregistered premise of ours failed: look-alike widgets were predicted to cost coverage, and coverage is 1.000.

1 INTRODUCTION

Runtime verification has become the standard answer to the question of whether an autonomous agent obeys its constraints, and the numbers it reports are good: compliance at or near one hundred percent on published benchmarks (Kamath et al., 2025; Xavier et al., 2026). Every one of those results shares a premise. The monitor reads the agent's *state* — a structured event, a tool call with named arguments, a log line — and evaluates the property against it. The premise is so uniform that it is rarely stated.

A GUI agent violates it. There is no event stream: the agent sees pixels and emits taps, and a predicate like *the confirmation checkbox is ticked* is not read but *inferred*, by a perception stack that is wrong sometimes and, more awkwardly, sometimes declines to answer. The object a deployment actually runs is therefore not a monitor. It is a monitor composed with a noisy sensor, and the guarantee the monitor was proved to have is a guarantee about its input.

Two things make this more than a robustness caveat. First, the sensor has a failure mode the monitor's logic cannot express. A perception stack that cannot resolve an atom — the widget is occluded, the rendering is degraded, two controls look alike — returns nothing, and a two-valued monitor must

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.

record *something*. Every deployed formulation we are aware of resolves this silently, as a pass or as a block. The episode then carries a verdict whose warrant does not exist: the guarantee is not violated, it is *vacuous*, and no compliance rate can show the difference. Second, and less obviously, a monitor that is merely imperfect can be worse than no monitor at all. Blocking a compliant action has a cost, and when violations are rare that cost is paid far more often than the cost it prevents.

What this paper does. We analyse the composition rather than either part. Section 3 gives the transfer theorem: soundness and completeness of $M \circ \mathcal{G}$ in closed form, as functions of the sensor's error rate, the property's shape, and *coverage* — the share of atoms the sensor will answer at all. Coverage enters completeness multiplicatively and enters soundness not at all, and no operating point on the sensor's precision–recall curve closes it, because it is not an operating point but a hole. Section 4 derives the net-harm threshold ε^* and establishes what it is invariant to: it contains the base rate and the cost ratio and nothing else. Sections 5–6 audit two published monitors and one of ours on a constraint corpus built for the purpose, and report both what the audit shows and what 240 episodes cannot show.

Contributions. **(i)** A guarantee-transfer theorem for a monitor driven by a noisy, partially-silent sensor, in which coverage is a separate coordinate from accuracy (§3). **(ii)** A net-harm threshold $\varepsilon^* = \Phi(b, \rho)$ in the base rate and the cost ratio alone, with an invariance result explaining why no amount of dependence among sensor errors can move it (§4). **(iii)** An audit in which the net-harm regime is reached by *published* monitors, with a reproduction gate establishing that every degradation is the sensor's (§6). **(iv)** A count, available without a new experiment, of the episodes in which a published guarantee is vacuous rather than satisfied.

What we do not claim. We do not claim that monitoring is net harmful in general. The threshold's sign depends on an elicited cost posture for mid-risk deployments, and we report the elicitation as an elicitation rather than a measurement. We do not claim that each audited row is individually resolvable; four of the nine are not, and §6 gives the sample size that would be required. And one premise we registered in advance failed outright.

2 Related Work: an Assumption Audit

We read related work as an audit rather than a survey: for each nearby result, the assumption it needs and the limit in which it fails. *Every result below is correct*, and most are theorems, which cannot be wrong — only applied outside their scope. The shared premise we are after is *that a predicate atom evaluates exactly against something the system can read*: invisible where events, tool-call arguments and logs make it simply true, and removed by a GUI agent. Table 3 in Appendix F is the row-by-row version, with the two places where an earlier version of this section credited a construction to the wrong paper.

The monitors this paper audits. One system compiles temporal properties to first-order logic, discharges them with SMT, backtracks through constrained decoding, and reports full compliance (Kamath et al., 2025); another compiles a policy DSL to DFAs and checks workflow graphs statically and at runtime (Xavier et al., 2026). *Both constructions are correct and their numbers real* — their atoms are tool-call arguments, so exact evaluation is a fact of their setting, not an idealisation. Theorem 1 turns a compliance figure into a function of ε, and coverage caps completeness at $\gamma_\wedge < 1$ independently of accuracy (Rem. 2). *We re-run their monitors unchanged*, since a re-implementation would confound their design with ours and the point is that the monitor is not what changes. Behavioural alarms on computer-use agents (Kumar et al., 2026) and audit-and-intervene monitoring (Alamdari et al., 2026) certify false alarms only, by design; our threshold prices false blocks *against* misses.

Runtime verification under uncertainty owns the premise. Monitoring noisily observed systems has its own survey (Taleb et al., 2023); monitors have been characterised by precision and recall since at least 2016 (Yingchareonthawornchai et al., 2016), a property's probability estimated from a gapped execution since 2011 (Stoller et al., 2011), and a partially observable MDP monitored in a CAV result (Junges et al., 2021). *The gap is not that nobody handles noise*: that branch parameterises

state-estimation error *against a model of the system*, and a GUI agent supplies none. Its noise is a learned operator on pixels with a *coverage* hole — predicates unanswerable at any operating point — and coverage is the parameter neither branch carries. Three-valued semantics likewise predates this work (Bauer et al., 2011; Pnueli, 1977), where the third value is *temporal* pendency; ours is perceptual, an orthogonal coordinate, and only ours carries a rate.

Certified enforcement owns the threshold. A theory of certified runtime safety derives the exact false-block/miss frontier by Neyman–Pearson with conformal certificates (Ray, 2026): *our ε^* is a point on that frontier*, not a new object. What their parameterisation lacks is a *sensor* rather than a score, a coverage term, and the third truth value; their closed-loop observation is a limitation of ours and recorded as one. Shielding and guardrails (Alshiekh et al., 2018; Chen et al., 2025; Wang et al., 2025) assume the intervention reads the situation correctly, and *conservatism — their standard remedy — is the wrong remedy here*, since a more conservative shield blocks more, moving mass into the arm whose cost is no longer negligible. A supervisory-control study reports 534 of 590 gate interventions as spec-on-bound (Rosenthal, 2026): Theorem 2's mechanism in a deployment, cited as corroboration rather than as an object. *And one consequence goes against us*: we do not claim as new that adding a channel can subtract utility while the structure stays super-additive — that happens whenever an opinion is *aggregated* rather than made available. Ours is the narrower *forced-veto* mechanism and the threshold at which it bites. Capability-from-access-structure results (Anonymous authors, 2026) assume free composition; a gate withholds exactly that option (Prop. 1).

Where ε comes from, and who inherits it. Grounding benchmarks (Cheng et al., 2024; Li et al., 2025; Lin et al., 2025), generalist action stacks (Hong et al., 2024; Qin et al., 2025; Wu et al., 2025; Xu et al., 2024), layout-guided work on same-looking/different-function elements (Liu et al., 2025), controls whose *state* is not visually obvious (Wu et al., 2026; Hu, 2026) and look-alike attacks (Zhang et al., 2024; Yan et al., 2026) all report or move the quantity ε denotes, as a number to drive down. *In a monitor it is not a score but a parameter whose value flips the sign of a decision*, and the flip happens inside the range they report. Meanwhile compliance is scored with no false-block arm beside it (Wang et al., 2026; Li et al., 2026; Ai et al., 2026; Yang, 2026), and GUI critics (Wu, 2026) and self-critique (Gou et al., 2024; Wang et al., 2023; Shinn et al., 2023) assume disagreement indicates a reasoning error — but critic and actor share $\mathcal{G}$, so agreement through one noisy read is not evidence and ensembling cannot address a common-mode sensor. Design by contract (Meyer, 1992; Hoare, 1969) assumes preconditions are decidable predicates and information-flow lattices (Denning, 1976; Biba, 1977; Newsome & Song, 2005) assume exact labels; both become something else when the predicate is an inference with a rate. Design-time injection defences (Willison, 2023; Debenedetti et al., 2025) are orthogonal: ε is not an adversary and does not respect the untrusted/trusted split.

3 The Composition, and What Transfers

Definition 1 (Grounding operator)**.** A *grounding operator* $\mathcal{G}$ maps a rendering to a truth value for an atom, returning $\top$, $\bot$, or ? when it will not resolve. Its *error rate* ε is the probability of returning a wrong value on an atom it does resolve; its *coverage* γ_a for atom a is the probability it resolves a at all. A monitor M is composed with it as $M \circ \mathcal{G}$.

Assumption 1 (Independent reads)**.** Reads of distinct atoms fail independently. §6 measures the dependent case and Remark 1 shows the threshold is invariant to it regardless.

Theorem 1 (Guarantee transfer)**.** *Let M monitor a property whose violating condition is a conjunction of m atoms, at base rate b, driven by $\mathcal{G}$ with error ε and coverages $\{\gamma_a\}$. Under Assumption 1,*

$$\mathrm{sound}(M \circ \mathcal{G}) = \frac{b(1-\varepsilon)^m}{b(1-\varepsilon)^m + (1-b)\, q_m(\varepsilon)}, \qquad \mathrm{compl}(M \circ \mathcal{G}) = \gamma_\wedge\,(1-\varepsilon)^m,$$

where $q_m(\varepsilon) = \sum_{j\geq 1} \nu_j\, \varepsilon^j (1-\varepsilon)^{m-j}$ and ν_j counts the non-violating states at Hamming distance j from the violating one. Both decrease in ε; completeness increases in every coverage, and soundness does not depend on coverage at all. *To first order,* $\mathrm{sound} = 1 - \frac{(1-b)\nu_1}{b}\varepsilon + O(\varepsilon^2)$ *and* $\mathrm{compl} = \gamma_\wedge(1 - m\varepsilon) + O(\varepsilon^2)$.

Three readings matter. The soundness slope carries $(1-b)/b$: at the base rates monitors are installed at, a small sensor error is amplified by the odds against a violation, which is the whole difficulty in

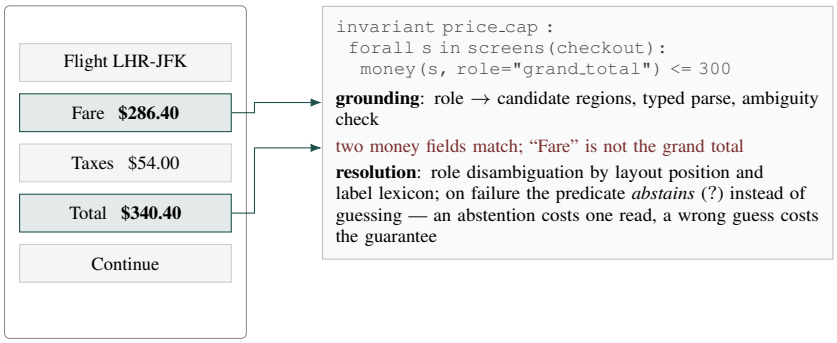

Figure 1. Where the error enters. The formula is checked exactly; the atoms are guessed. Coverage γ is the fraction of steps on which the atom is on screen at all, and is the quantity a readable-state monitor implicitly sets to one.

one factor. Completeness carries $\gamma_\wedge$ as a multiplier that no choice of threshold on the sensor removes — an atom the sensor declines is not a low-confidence answer to be traded against precision, it is an absent one. And the two degrade in different currencies, so a single compliance number cannot be read as either (Appendix A, Corollary 2).

4 The Net-Harm Threshold

Definition 2 (Costs)**.** Let c_{miss} be the cost of an unprevented violation, c_{fb} the cost of blocking a compliant action, c_{esc} the cost of escalating an unresolved atom, and $\rho = c_{\text{fb}}/c_{\text{miss}}$. Net benefit NB is the expected cost of the unmonitored agent minus that of the monitored one.

Theorem 2 (The harm threshold)**.** *For a single-atom property with an unresolved atom passed through,* $\mathrm{NB}(\varepsilon) = \gamma\big[\, b(1-\varepsilon)c_{\text{miss}} - (1-b)\varepsilon\, c_{\text{fb}} \,\big]$*, and* $\mathrm{NB}(\varepsilon) > 0$ *if and only if* $\varepsilon < \varepsilon^*$*, where*

$$\varepsilon^* \;=\; \Phi(b,\rho) \;=\; \frac{b\, c_{\text{miss}}}{b\, c_{\text{miss}} + (1-b)c_{\text{fb}}} \;=\; \Big(1 + \tfrac{1-b}{b}\cdot\rho\Big)^{-1}.$$

Φ *is strictly decreasing in* ρ *and strictly increasing in* b*: the rarer the violations and the more expensive a spurious halt, the smaller the sensor error a monitor can tolerate. For* $\varepsilon > \varepsilon^*$ *the monitored agent has strictly higher expected cost than the unmonitored one.*

Remark 1 (What ε^* does not depend on). The expression contains b and ρ and nothing else. It carries no coverage, no formula shape m, and — the part that matters for Assumption 1 — *no dependence structure*. Whatever the correlation among grounding errors, it cannot move the threshold; it can only move the exposure that is compared against it. §6 separates the two and measures both.

The threshold shrinks with rarity, so it binds exactly where monitors are installed. A monitor guarding a constraint violated in one episode in twenty tolerates less sensor error than one guarding a constraint violated in one episode in three, and the published compliance figure reports neither (Appendix A, Corollary 3). For mid-risk deployments the sign of NB depends on ρ, which is not measurable and which we elicit; we report it as elicited and do not claim the sign there.

The third value is not a convenience, and it moves the threshold. A monitor whose logic has no value for "the sensor would not resolve this" must map ? onto $\top$ or $\bot$, and each choice sets a different threshold:

$$\varepsilon^*_{\text{block}} = \frac{b\, c_{\text{miss}} - (1-b)(1-\gamma)c_{\text{fb}}}{\gamma\,[b\, c_{\text{miss}} + (1-b)c_{\text{fb}}]}, \qquad \varepsilon^*_{\text{esc}} = \frac{b\, c_{\text{miss}} - (1-\gamma)c_{\text{esc}}}{\gamma\,[b\, c_{\text{miss}} + (1-b)c_{\text{fb}}]}$$

(Appendix A, Proposition 2). All three coincide at $\gamma = 1$, where there is no ? to discharge, and they separate in proportion to $1-\gamma$: blocking lowers the threshold because the coverage hole becomes false blocks, escalating lowers it only by the escalation's own price, and pass-through is the case in which the hole costs completeness instead — which is why coverage is absent from Theorem 2. Neither two-valued choice records that the verdict has no warrant. Our arm carries a third value and escalates; §6 measures what that costs and what it reveals.

The sign is reported over the elicitation range, not at one elicited point. The base rate is measurable from a logged sample; the cost ratio is not. Two opposite risk postures can wear one deployment's name and their thresholds differ by orders of magnitude, so we let ρ run over $[\frac{1}{50}, 25]$ — from "a missed violation costs fifty times a spurious halt" to "a spurious halt costs twenty-five times a missed violation", a factor of 1250 spanning every posture we have seen argued for — and report the threshold across it. For low-base-rate deployments the sign of NB is determined across the whole range. For mid-risk deployments it is not, and that is the case in which this paper declines to claim a sign.

4.1 What a compositional closure result does not license

The threshold has a consequence for composition that is easy to state and easy to misread, so we state it as a scope boundary rather than as a correction to anything. Results of the form *holding one more channel enlarges the set of tasks a family solves* are statements about *closures of architecture families*, and their join is defined to contain every interleaving including the degenerate ones that use a channel zero times. Each component family is therefore a subset of the join by construction, the weak inclusion is automatic, and all of the content sits in the strictness: there *exists* a witness the join solves and neither component does.

Nothing about a deployed system follows, and the gap is not a technicality. Realising a closure requires the freedom to *use* a channel where it helps and to *ignore* it where it does not — the closure is the union over ways of using the channels. A monitor is not held on those terms: its verdict vetoes an action, and an agent free to ignore the veto is not being monitored. That is a composition the system cannot decline, and declining is precisely what the closure argument needs.

Definition 3 (Gated composition)**.** Write Π for the unmonitored policy and $\Pi \triangleright M$ for the *gated* system in which the monitor M, reading its atoms through $\mathcal{G}$, may veto the action Π proposes at each step. The gate is not optional: once the veto fires, $\Pi \triangleright M$ has no access to the action Π would have taken. Let $U(\cdot)$ denote expected task utility under the costs of Definition 2.

Theorem 3 (Gating is not utility-monotone)**.** *Fix a task family with violation base rate b, cost ratio $c_{\mathrm{fb}}/c_{\mathrm{miss}}$ and coverage $\gamma > 0$, and let $\varepsilon^* = \Phi(b, \rho)$ be the threshold of Theorem 2. Then $\varepsilon > \varepsilon^* \implies U(\Pi \triangleright M) < U(\Pi)$. Holding the verification channel strictly lowers the system's utility, and no tuning of the monitor's formula recovers the loss, because the loss is not in the formula. It is in the atoms. Proof in §A; Figure 8 draws the two curves and puts a number on the gap.*

Proposition 1 (The transfer gap)**.** *Theorem 3 neither contradicts nor weakens a closure inclusion, and it is not its converse — no closure statement about a join can be, since the join contains each component by definition. The two quantify over different objects. An inclusion is over architecture families under* free *composition, where the option of not using a channel belongs to the family, and it is monotone in the channel set by construction; Theorem 3 is over* one *composition rule at one operating point, where that option is absent. Neither implies the other, and in particular a proof of the inclusion is not evidence for the deployment claim that a further channel helps a running system.*

Corollary 1 (What exactness buys)**.** *At $\varepsilon = 0$ and $\gamma = 1$ — the readable-state regime in which the monitoring literature is stated — the threshold is never reached and $U(\Pi \triangleright M) \geq U(\Pi)$ for every base rate and every cost ratio: gating is monotone there, which is why the question has not arisen. Exactness is not a side condition on a composition theorem. It is the condition under which a guarantee proved* about a monitor *transfers to the* system that runs it*, and it is the condition a GUI deployment does not have.*

5 Experimental Setup

Why a corpus and not a task suite. ε^* is a function of a per-class base rate, so the unit of analysis has to be the constraint class, not the task. A task suite reports success; it does not report how often each constraint is actually violated, and without that there is no threshold to compare anything against. **CON-150** supplies it: five constraint types crossed with three induced grounding-error levels and three monitor architectures, with a per-class violation rate measured rather than assumed.

What is reused unchanged. The host suites' tasks, their success criteria and their audited monitors (Kamath et al., 2025; Xavier et al., 2026) are reused as published. We supply a different input to them and change no line of their logic, which is what makes the audit attributable.

Table 1. The audit: two published monitors and one of ours, on the same operator reads. ε_{eff} is the realised effective error, T_{eff} the detection latency, and the last column the verdict against each row's own threshold. Rows marked * separate from ε^* at $p < 0.05$ after Benjamini–Hochberg control; the others do not, and §6.2 says what that means.

Monitor	ε_{eff}	95% CI	T_{eff}	viol. rate	episodes escalated	net benefit	vs ε^*
No monitor (reference)	0	—	—	1.000	0.000	0.000	—
native atoms — operator $\varepsilon = 0.004$, $\gamma = 0.999$							
Agent-C style Kamath et al. (2025)	0.021	$[0.009, 0.048]$	5.3	0.000	0.000	$+0.217$	below, helps*
Agentproof style Xavier et al. (2026)	0.025	$[0.012, 0.053]$	5.3	0.000	0.000	$+0.212$	below, helps*
Ours (statistical third value)	0.021	$[0.009, 0.048]$	5.3	0.000	0.004	$+0.216$	below, helps*
degraded rendering — operator $\varepsilon = 0.143$, $\gamma = 0.918$							
Agent-C style Kamath et al. (2025)	0.263	$[0.211, 0.322]$	2.2	0.158	0.000	-0.062	**above, hurts**
Agentproof style Xavier et al. (2026)	0.396	$[0.336, 0.459]$	2.2	0.000	0.000	-0.158	**above, hurts***
Ours (statistical third value)	0.263	$[0.211, 0.322]$	2.2	0.000	0.317	-0.073	**above, hurts**
look-alike widgets — operator $\varepsilon = 0.084$, $\gamma = 0.992$							
Agent-C style Kamath et al. (2025)	0.204	$[0.158, 0.260]$	2.3	0.000	0.000	$+0.033$	below, helps
Agentproof style Xavier et al. (2026)	0.246	$[0.196, 0.304]$	2.6	0.000	0.000	-0.008	**above, hurts**
Ours (statistical third value)	0.204	$[0.158, 0.260]$	2.3	0.000	0.046	$+0.026$	below, helps

The grounding stack is fixed across monitors and swept separately. All three architectures consume the *same* operator reads at each rendering level and differ only in how they discharge ?. Coverage is measured separately from accuracy, because Theorem 1 says they enter different guarantees.

Three rendering levels. *Native* ($\varepsilon = 0.004$, $\gamma = 0.999$), *degraded* ($\varepsilon = 0.143$, $\gamma = 0.918$), and *look-alike widgets* ($\varepsilon = 0.084$, $\gamma = 0.992$). The false-block arm has to be constructed rather than observed: a missed violation appears in a trace, a wrongly blocked compliant action does not.

Preregistration. Every prediction was registered before the runs, with its read-out rule and its failure branch, and is scored supported, partial or failed in Appendix I. The decision rule is $p < 0.05$ after Benjamini–Hochberg control. One premise, P5, failed (§6).

6 Results

6.1 The reproduction gate, and then the substitution

The audit runs the oracle-atom reproduction before any pixel substitution, because the claim is about what happens to *their* guarantee. On oracle atoms all three architectures return soundness 1.000, completeness 1.000 and a false-block rate of exactly 0.000, netting $+0.2375$ per episode. *That is the readable-state claim this paper opens by quoting*, reproduced exactly — and it is reproduced because on oracle atoms the three architectures are indistinguishable: they differ only on ?, and an oracle never returns one. The gate is a sanity check on our re-implementations rather than a reproduction of anybody's published compliance figure; what it establishes is that every degradation below is the sensor's.

Substituting pixels for the oracle (Table 1) moves four of the nine rows below zero. At the degraded level both published monitors and ours are net harmful; under look-alike widgets one published

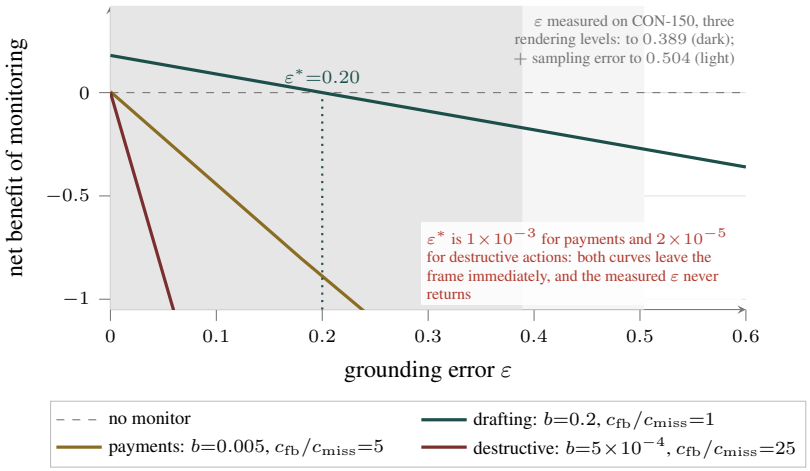

Figure 2. Net benefit against grounding error, at three deployment postures, with the measured operating region shaded. Each line is $\mathrm{NB}(\varepsilon)$ at that posture's own base rate and cost ratio, crossing zero at its ε^* from Theorem 2: drafting (b=0.2, ρ=1) crosses at 0.200, while payments (b=0.005, ρ=5) and destructive actions (b=5 × 10^{-4}, ρ=25) cross at 1 × 10^{-3} and 2 × 10^{-5} — they leave the frame immediately, and the measured ε never returns to them. *Two bands, and they mean different things*: the inner one is the range CON-150 actually measures over its three rendering levels (0.000–0.389), the outer runs to 0.504, the largest 95% upper bound over the fifteen cells, and is sampling error rather than measurement. The threshold falls with the base rate, so it binds most tightly exactly where monitors are installed — on constraints whose violations are rare.

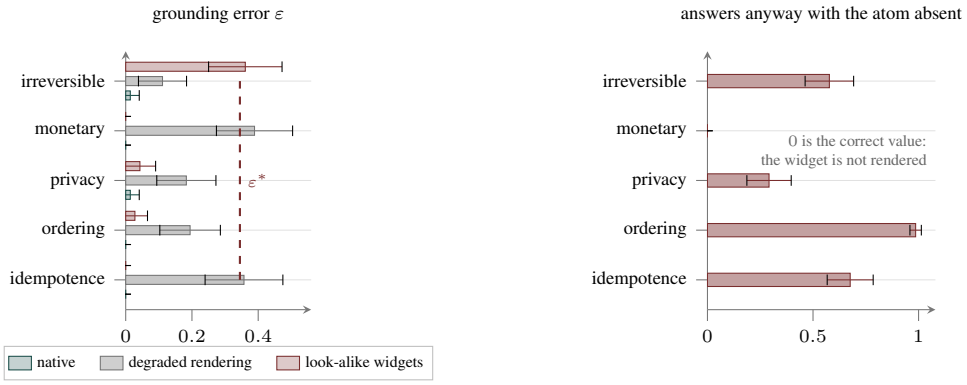

Figure 3. Which atoms are resolvable, and which axis attacks which term. CON-150 as pre-registered, five constraint types crossed with induced grounding error, 24 distinct screens per cell and three open-weight operators; error bars are 1.96 standard errors. **One premise of the earlier version of this figure is withdrawn.** We drew coverage falling across atom classes and look-alike widgets costing coverage on every one of them. Measured, $\gamma = 1.000$ at native *and* at adversarial for all five classes, so look-alikes cost coverage on 0 of 5; what they cost is ε, and on one class. **The two axes move different terms of Theorem 1.** Look-alikes leave γ untouched and raise ε on the irreversible-action guard alone, from 0.014 to 0.361 — soundness only, since S does not depend on coverage. Degraded rendering raises ε on every class and *also* costs coverage on four of five, down to 0.783, so it moves S and C together. Asking which axis is more dangerous (+0.389 against +0.347) is therefore the wrong question: they are not comparable quantities. **Three cells sit above** ε^* at b=0.05 and $c_{\mathrm{miss}}/c_{\mathrm{fb}}$=10: the guard under look-alikes, and the monetary limit and idempotence under degraded rendering. On those atoms the monitored agent is worse than the unmonitored one. The atom a safety policy wants most is the one look-alikes break. **Right panel: the third truth value, tested by removing the atom.** With the queried widget not rendered at all, the only defensible reply is UNKNOWN, so every bar is a rate of confident answers about something absent. The monetary limit is answered correctly at 0.000; list ordering is answered anyway 0.986 of the time and never notices the step is missing. Coverage is a real and strongly class-dependent quantity — it is simply not the quantity a look-alike attacks.

monitor is. The net-harm regime is therefore not a property of a weak monitor we built to lose: it is reached by systems published as compliant, on renderings a deployment meets.

6.2 WHAT 240 EPISODES ESTABLISH, AND WHAT THEY DO NOT

The threshold gets the sign right in 9 of 9 rows. That is what a threshold predicts and what 240 episodes can show, and it is not a claim that each row is individually resolvable. Four rows survive Benjamini–Hochberg at $q = 0.05$ ($p \leq 0.022$), and three of those four are native rows, which sit far below the threshold in the direction nobody doubted. Of the four rows with negative net benefit, *one* separates at $p < 0.05$ (Agentproof at degraded, $\varepsilon_{\text{eff}} = 0.396$, $z = 5.8$). The decisive row sits at $\varepsilon_{\text{eff}} = 0.246$ against $\varepsilon^* = 0.238$ with interval $[0.196, 0.304]$: the threshold sits near the middle of its own confidence interval, $p = 0.76$. Its *sign* agrees with the prediction and its *magnitude* cannot be distinguished from the threshold, and those are two different statements.

The arithmetic is worth stating exactly. The margin is 0.008 against a standard error of 0.028, so matching the standard error to the effect needs $n \approx 2{,}940$ episodes, which still leaves $z \approx 1$; an 80%-powered two-sided test at $\alpha = 0.05$ needs $\approx 22{,}000$, and 240 episodes resolve gaps of ≥ 0.077. No affordable number of operator episodes settles this crossing, which is why we resolve it by calibrated simulation instead of implying the audit could.

Each row is scored against the threshold its own policy sets. Two of the three architectures do not pass an unresolved atom through, so a single ε^* is wrong for them. At the episode level the accounting is exact: writing $\hat{b}$ for the episode-level violation rate, ca for the share of violations caught and es for the escalation rate, $\varepsilon^*_{\text{eff}} = \hat{b}\,\mathrm{ca}\,\rho^{-1} - \mathrm{es}\,c_{\text{esc}}/c_{\text{fb}}$ reproduces every recorded net to 3×10^{-17} on all nine rows, so it is an identity and not a model. The single $\varepsilon^* = 0.238$ is that expression at $\mathrm{ca} = 1$, $\mathrm{es} = 0$: exactly right for five rows and wrong for four. Corrected, the largest change is 0.048 and *the sign agreement is* 9 *of* 9 *either way*. The policy matters and it did not change a verdict here, and we would rather have derived that than assumed it.

6.3 THE STATE NO DEPLOYED FORMULATION CAN REPRESENT

The count committed to in advance needs no new experiment, because the three architectures consume the same operator reads and differ only on ?: our escalation column *is* the share of episodes carrying at least one unresolved atom. It is 0.004 at native, 0.317 at degraded and 0.046 under look-alikes.

At the rendering level where the threshold is crossed, 76 **of** 240 **episodes contain an atom the operator would not resolve, and both two-valued monitors record every one as a clean verdict** — one silently as a pass, the other as a block. Each is a case where a published guarantee is *vacuous* rather than violated, a distinction invisible to any instrument reporting a compliance rate, and the count is not zero, so the two-coordinate model is not decoration. What it does *not* establish is that a human auditor would have wanted a different action in each case, which is a reading of instances and is still owed.

6.4 RESOLVING THE CROSSING, AND THE CLOSED LOOP

Simulated at the audit's own operating point ($b = 0.2375$, $\gamma_\wedge = 0.918$, $m = 1$, i.i.d. reads) with n=3000 episodes per cell over a 0.01-wide ε grid, the measured net benefit crosses zero inside $[0.220, 0.250]$, bracketing the predicted $\varepsilon^* = b = 0.2375$. The closed form is not merely sign-correct: at the sample size the margin needs, its crossing lands on the prediction. The same simulator over conjunctions of $m \in \{1, 2, 4, 8\}$ atoms under a common-shock dependence model shows the independent form of Theorem 1 to be *conservative* under measured dependence, consistent with Remark 1: the dependence moves the exposure and not the threshold.

The open-loop qualification has a size. Modelling the witness design as one guarded decision per episode with a blocked proposal resampled — the replacement violating with probability $b(1 - \kappa)$, so κ interpolates between i.i.d. retry and a safe fallback — the retry chain is exactly solvable. At the audit's operating point the loop tightens the tolerable false-block exposure from 0.262 to 0.162 at $\kappa = 0$, a factor of 1.6. Suppression barely buys it back: even at $\kappa = 0.95$ the threshold reaches only 0.172, because every resample re-risks a false block regardless of what it proposes. *The audit's verdict is robust across the whole sweep*: the measured exposure 0.263 stays past the closed-loop threshold for every $\kappa \in [0, 0.95]$.

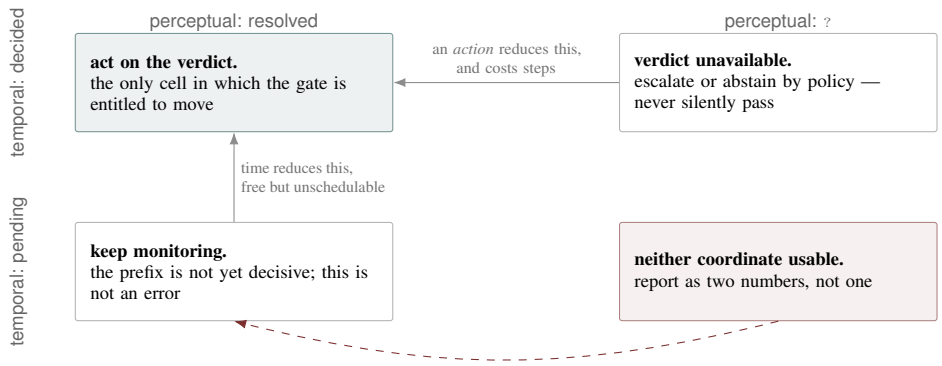

Figure 4. Two coordinates, one collapse. The monitor's state is a pair: temporal status (has the violation happened yet) crossed with perceptual status (is the atom resolvable at all). The shaded state is the one existing formulations cannot represent — the atom is unresolved, so no verdict is licensed — and a two-valued monitor must discharge it as a pass or as a block. The dashed arc is that discharge: both coordinates are folded into a single clean verdict, so a trace that is merely too short and a screen the sensor could not read are recorded as the same outcome, which is why a compliance rate cannot see the 76 of 240 episodes §6 counts at the degraded level. Proposition 2 prices each way of discharging it, and the three thresholds coincide exactly at $\gamma = 1$: on readable state the distinction does not exist.

6.5 A PREMISE OF OURS THAT FAILED

We registered the prediction that look-alike widgets would cost *coverage*: two controls that render identically should make the operator decline. *Coverage is* 1.000 *at native and at adversarial on all five classes.* The operator does not decline on look-alikes; it answers, and it answers wrongly, so look-alikes cost *soundness* instead. The mechanism is that declining requires the operator to represent its own ambiguity, and a perception stack asked for a truth value returns one. We report this as failed rather than reinterpreted, and Appendix I carries the branch it was registered against.

7 THREATS TO VALIDITY

The cost ratio is elicited, not measured: ρ is not observable from a trace, so we elicit it, report it as elicited, and for mid-risk deployments claim not the sign of NB but only that a threshold exists and where it sits in ρ. **Four of nine rows are not individually resolvable**, stated with the required sample size in §6.2, because the table would otherwise read as nine independent confirmations when it is one sign agreement across nine rows plus one separated row. **The false-block arm is constructed**: a missed violation is visible in a trace and a wrongly blocked compliant action is not, so its rate is measured on constructed episodes rather than harvested, by the same construction across all three architectures. **And there is one operator stack.** Coverage and error are properties of the perception stack and we sweep the rendering rather than the stack; a different stack moves ε and γ, and by Remark 1 it cannot move ε^*.

8 DISCUSSION AND CONCLUSION

Separating a monitor from its sensor (Fig. 5, App. C) gives a deployment three things: soundness and completeness in closed form with coverage as its own coordinate; a threshold in the base rate and cost ratio alone, past which installing the monitor is worse than not installing it; and a count of the episodes where the guarantee has no warrant. *The last is the one we would ask a practitioner to run first, because it is free*: any monitor whose sensor can decline already has the number — the share of episodes carrying an unresolved atom, nearly a third at the rendering level where our threshold is crossed — and reporting it changes what a compliance rate means. *We do not conclude that monitoring should be switched off*: the threshold has two arguments, and where it binds is where measuring the base rate is worth doing.

REPRODUCIBILITY STATEMENT

Every number in this paper is written by a named analyzer to a verdict file, and every deciding test's read-out rule was frozen before the run it decides. Appendix I is the registration index: one row per registered prediction, its branches as registered, and the branch that fired. The CON-150 corpus, the grounding stack, the three monitor implementations, the constructed false-block episodes, the simulator and the analyzers are in the supplementary material. The two audited monitors are third-party published systems reused unchanged; the reproduction gate of §6 is what establishes that the degradations reported here are attributable to the sensor rather than to our re-implementation.

ETHICS STATEMENT

No human subjects and no personal data. The corpus is synthesised from public application flows by published scripts. The audit reports that two published monitoring systems enter a net-harm regime on degraded renderings; we name the systems because the claim is checkable only if they are named, we reuse their logic unchanged rather than reimplementing it to their disadvantage, and the reproduction gate is reported before the substitution so that a reader can separate their behaviour from ours. We develop no attack: the adversarial rendering level is a rendering condition, not an exploit, and the look-alike construction is the one a deployment meets by accident. We have read the ICLR Code of Ethics and adhere to it.

AI USE STATEMENT

We report our use of generative AI tools against the categories of the ICLR 2027 AI Policy for Authors. *Our use was confined to preparing the document.*

Tasks in the required-disclosure list: none. Generative AI was not used to generate the corpus, to develop the theoretical model, to formulate or prove the theorems, to propose or refine the hypotheses, to design the methodology or the experiments, to set or choose any pre-registered threshold, to implement the monitors or the grounding stack, or to interpret the results. The corpus is emitted by fixed-seed builders, the runs were configured and launched by the authors, and every number is computed by an analyzer the authors wrote. Translation and qualitative or thematic data analysis are not applicable.

Tasks in the recommended-disclosure list. Generative AI was used on the manuscript itself: it condensed a longer version into the present conference format, which involved drafting and rewriting substantial parts of the exposition; it edited for readability, formatted tables and references, and assembled the anonymised supplementary archive. It was not used to discover the research topic, identify the gap, select the literature, or decide what the paper claims.

Verification. Every quantitative statement was checked against the verdict file it cites, and every registered outcome against the read-out rule frozen before the run. We have reviewed all AI-assisted work and take responsibility for the final content of this paper, including its text, claims and artifacts.

REFERENCES

Anonymous authors. The capability convergence hypothesis: Capability from access structure, not scale. *arXiv preprint; identifier withheld for double-blind review*, 2026.

Jiaxin Ai, Yukang Feng, Fanrui Zhang, Jianwen Sun, Zizhen Li, Chuanhao Li, Yifan Chang, Wenxiao Wu, Ruoxi Wang, Mingliang Zhai, and Kaipeng Zhang. Prosoftarena: Benchmarking hierarchical capabilities of multi-modal agents in professional software environments. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

Parand A. Alamdari, Toryn Q. Klassen, and Sheila A. McIlraith. Formal methods meet llms: Auditing, monitoring, and intervention for compliance of advanced ai systems. In *ACM Conference on Fairness, Accountability, and Transparency (FAccT)*, 2026. TRAC; intervening monitors reduce violation rates while largely preserving task performance.

Mohammed Alshiekh, Roderick Bloem, Ruediger Ehlers, Bettina Koenighofer, Scott Niekum, and Ufuk Topcu. Safe reinforcement learning via shielding. In *AAAI Conference on Artificial Intelligence*, 2018.

Hao Bai, Alexey Taymanov, Tong Zhang, Aviral Kumar, and Spencer Whitehead. Webgym: Scaling training environments for long-horizon visual web agents with realistic tasks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

Andreas Bauer, Martin Leucker, and Christian Schallhart. Runtime verification for ltl and tltl. *ACM Transactions on Software Engineering and Methodology*, 20(4), 2011. Three-valued semantics adopted; not a contribution of this paper.

Kenneth J. Biba. Integrity considerations for secure computer systems. Technical report, MITRE Corporation, 1977.

Zhaorun Chen, Mintong Kang, and Bo Li. Shieldagent: Shielding agents via verifiable safety policy reasoning. In *International Conference on Machine Learning (ICML)*, 2025. arXiv:2503.22738.

Kanzhi Cheng, Qiushi Sun, Yougang Chu, Fangzhi Xu, Yantao Li, Jianbing Zhang, and Zhiyong Wu. Seeclick: Harnessing gui grounding for advanced visual gui agents. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, 2024.

Edoardo Debenedetti, Ilia Shumailov, Tianqi Fan, Jamie Hayes, Nicholas Carlini, Daniel Fabian, Christoph Kern, Chongyang Shi, Andreas Terzis, and Florian Tramèr. Defeating prompt injections by design. *arXiv preprint arXiv:2503.18813*, 2025.

Dorothy E. Denning. A lattice model of secure information flow. *Communications of the ACM*, 19 (5):236–243, 1976.

Zhibin Gou, Zhihong Shao, Yeyun Gong, Yelong Shen, Yujiu Yang, Nan Duan, and Weizhu Chen. Critic: Large language models can self-correct with tool-interactive critiquing. In *International Conference on Learning Representations (ICLR)*, 2024.

C. A. R. Hoare. An axiomatic basis for computer programming. *Communications of the ACM*, 12 (10):576–580, 1969.

Wenyi Hong, Weihan Wang, Qingsong Lv, Jiazheng Xu, Wenmeng Yu, Junhui Ji, Yan Wang, Zihan Wang, Yuxuan Dong, Ming Ding, and Jie Tang. Cogagent: A visual language model for gui agents. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

others Hu. Showui-π: Flow-based generative models as gui dexterous hands. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026. arXiv:2512.24965.

Sebastian Junges, Hazem Torfah, and Sanjit A. Seshia. Runtime monitoring for markov decision processes. In *Computer Aided Verification (CAV)*, 2021. Partially observable state; conditional reachability gives a conservative risk estimate from partial observations.

Adharsh Kamath, Sishen Zhang, Calvin Xu, Shubham Ugare, Gagandeep Singh, and Sasa Misailovic. Enforcing temporal constraints for llm agents. *arXiv preprint arXiv:2512.23738*, 2025. Agent-C; DSL to first-order logic to SMT with constrained decoding.

Divake Kumar, Sina Tayebati, Devashri Naik, Amanda Sofie Rios, Nilesh Ahuja, Omesh Tickoo, Ranganath Krishnan, and Amit Ranjan Trivedi. CURA: Certified runtime alarms for computer-use agents. *arXiv preprint arXiv:2608.27808*, 2026. 361 OSWorld tasks; 64 of 71 failures end with a success claim; CUSUM alarm with certified false-alarm control; the certificate bounds false alarms only.

Kaixin Li, Ziyang Meng, Hongzhan Lin, Ziyang Luo, Yuchen Tian, Jing Feng, Jiawei Xiao, and Mike Zheng Shou. Screenspot-pro: Gui grounding for professional high-resolution computer use. *arXiv preprint arXiv:2504.07981*, 2025.

Wei Li, William Bishop, Alice Li, Chris Rawles, Folawiyo Campbell-Ajala, Divya Tyamagundlu, and Oriana Riva. On the effects of data scale on ui control agents. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Yang Li, Yuchen Liu, Haoyu Lu, Zhiqiang Xia, Hongzhen Wang, Kaiyang Han, Changpeng Yang, Jinyang Wu, Jiaming Xu, Runyu Shi, and Ying Huang. Gui-ceval: A hierarchical and comprehensive chinese benchmark for mobile gui agents. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

Kevin Qinghong Lin, Linjie Li, Difei Gao, Zhengyuan Yang, Shiwei Wu, Zechen Bai, Stan Weixian Lei, Lijuan Wang, and Mike Zheng Shou. Showui: One vision-language-action model for gui visual agent. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

Zikang Liu, Junyi Li, Wayne Xin Zhao, Dawei Gao, Yaliang Li, and Ji-Rong Wen. Pal-ui: Planning with active look-back for vision-based gui agents. *arXiv preprint arXiv:2510.00413*, 2025.

Sarthak Mehrotra, Sairam V. C. Rebbapragada, Mani Hemanth Reddy Bonthu, and Vineeth N. Balasubramanian. ishift: Lightweight slow-fast gui agent with adaptive perception. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

Bertrand Meyer. Applying design by contract. *IEEE Computer*, 25(10):40–51, 1992.

James Newsome and Dawn Song. Dynamic taint analysis for automatic detection, analysis, and signature generation of exploits on commodity software. In *Network and Distributed System Security Symposium (NDSS)*, 2005.

Amir Pnueli. The temporal logic of programs. In *FOCS*, 1977.

Yujia Qin, Yining Ye, Junjie Fang, et al. Ui-tars: Pioneering automated gui interaction with native agents. *arXiv preprint arXiv:2501.12326*, 2025.

Shawn Ray. What can be enforced? A theory of certified runtime safety for tool-using agents. *arXiv preprint arXiv:2607.22868*, 2026. Neyman–Pearson false-block/miss frontier; conformal finite-sample certificates; the closed-loop effect that blocking changes future proposals.

Christian Rosenthal. Safety-gated agentic supervisory control on a coupled distillation benchmark: Regime map, auditable gate, and co-design findings. *arXiv preprint arXiv:2607.27849*, 2026. 534 of 590 gate interventions are spec-on-bound: a well-behaved operating point is made inoperable while misbehaving ones are only contained — an independent empirical instance of the false-block mechanism.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Scott D. Stoller, Ezio Bartocci, Justin Seyster, Radu Grosu, Klaus Havelund, Scott A. Smolka, and Erez Zadok. Runtime verification with state estimation. In *Runtime Verification (RV)*, volume 7186 of *LNCS*, pp. 193–207, 2011. The monitor estimates the probability a property holds when the observed execution has gaps — a monitor over an inferred state, fifteen years before this paper.

Rania Taleb, Sylvain Hallé, and Raphael Khoury. Uncertainty in runtime verification: A survey. *Computer Science Review*, 50, 2023.

Haoyu Wang, Christopher M. Poskitt, Jiali Wei, and Jun Sun. Probguard: Proactive runtime monitoring for llm agent safety via probabilistic prediction. *arXiv preprint arXiv:2508.00500*, 2025.

Xin Wang et al. Mmbench-gui: A unified hierarchical evaluation framework for multi-platform gui agents. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026. arXiv:2507.19478.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. In *International Conference on Learning Representations (ICLR)*, 2023.

Simon Willison. The dual llm pattern for building ai assistants that can resist prompt injection. `https://simonwillison.net/2023/Apr/25/dual-llm-pattern/`, 2023.

others Wu. Os-oracle: A comprehensive framework for cross-platform gui critic models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026. arXiv:2512.16295.

Zhiyong Wu, Zhenyu Wu, Fangzhi Xu, Yian Wang, Qiushi Sun, Chengyou Jia, Kanzhi Cheng, Zichen Ding, Liheng Chen, Paul Pu Liang, and Yu Qiao. Os-atlas: Foundation action model for generalist gui agents. In *International Conference on Learning Representations (ICLR)*, 2025.

Zongru Wu, Rui Mao, Zhiyuan Tian, Pengzhou Cheng, Tianjie Ju, Zheng Wu, Lingzhong Dong, Haiyue Sheng, Zhuosheng Zhang, and Gongshen Liu. See, think, act: Teaching multimodal agents to effectively interact with gui by identifying toggles. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026. arXiv:2509.13615.

Melwin Xavier, Vaisakh M A, Melveena Jolly, and Midhun Xavier. Agentproof: Static verification of agent workflow graphs. *arXiv preprint arXiv:2603.20356*, 2026. DSL compiled to DFAs; static graph$\times$DFA product and runtime checking over event traces.

Yiheng Xu, Zekun Wang, Junli Wang, Dunjie Lo, Tianbao Chen, and Tao Yu. Aguvis: Unified pure vision agents for autonomous gui interaction. *arXiv preprint arXiv:2412.04454*, 2024.

Zihe Yan, Jiaping Gui, Zhuosheng Zhang, and Gongshen Liu. Lasm: Layer-wise scaling mechanism for defending pop-up attack on gui agents. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026. arXiv:2507.10610.

others Yang. Guide: A benchmark for understanding and assisting users in open-ended gui tasks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

Sorrachai Yingchareonthawornchai, Duong Nguyen, Vidhya Tekken Valapil, Sandeep Kulkarni, and Murat Demirbas. Precision, recall, and sensitivity of monitoring partially synchronous distributed systems. *arXiv preprint arXiv:1607.03369*, 2016. Monitors characterised by precision and recall, a decade before this paper uses the same two quantities.

Yanzhe Zhang, Tao Yu, and Diyi Yang. Attacking vision-language computer agents via pop-ups. *arXiv preprint arXiv:2411.02391*, 2024.

A Proofs, and the Statements Held Out of the Main Text

A.1 Theorem 1

Proof. Condition on the ground-truth state. A violating state is flagged when every witnessing atom is resolved and read correctly, which has probability $\gamma_\wedge(1-\varepsilon)^m$ under Assumption 1; this is the completeness expression. A non-violating state is flagged when the reads happen to spell the violating assignment, which requires each atom at Hamming distance j from the violating one to be misread on exactly those j coordinates, giving $q_m(\varepsilon) = \sum_{j\geq 1} \nu_j \varepsilon^j (1-\varepsilon)^{m-j}$. Soundness is the posterior probability that a flag is a true violation, which is Bayes' rule on those two terms. Differentiating at $\varepsilon = 0$ gives the first-order forms, using $q_m(\varepsilon) = \nu_1 \varepsilon + O(\varepsilon^2)$. □

Coverage is absent from soundness because an unresolved atom produces no flag under pass-through, so it removes mass from both the true- and false-flag terms in the same proportion and cancels in the ratio. It is present in completeness because the flag that would have been raised is not raised.

A.2 Theorem 2

Proof. Under pass-through, an atom is acted on only when resolved, which happens with probability γ. A violation is prevented when resolved and read correctly, saving c_{miss} at rate $\gamma\, b(1-\varepsilon)$; a compliant action is blocked when the atom is misread, costing c_{fb} at rate $\gamma(1-b)\varepsilon$. Subtracting gives the displayed NB. Setting it positive and solving for ε gives $\varepsilon < b\, c_{\text{miss}}/(b\, c_{\text{miss}} + (1-b)c_{\text{fb}})$; $\gamma > 0$ multiplies both terms and cancels. Monotonicity in b and ρ follows by inspection of $(1+\frac{1-b}{b}\rho)^{-1}$. □

A.3 THE STATEMENTS HELD OUT OF THE MAIN TEXT

Corollary 2 (Where each guarantee is spent)**.** *The two losses have different amplifiers, so a monitor can be healthy in one and ruined in the other, and neither amplifier is the formula's size. Soundness degrades at rate $\nu_1(1-b)/b$, whose dominant factor is $1/b$: it is destroyed by* rare *violations, because a small base rate multiplies the first-order term without bound — and a property worth monitoring is one whose violations are rare. Completeness degrades at rate m in ε and is multiplied by $\gamma_\wedge$, a product of m coverages, so it is depth and field of view that spend completeness. For atoms that are usually off screen the completeness loss is dominated by coverage rather than accuracy — $\gamma_\wedge$ can be an order of magnitude below $(1-\varepsilon)^m$ — which is the prediction P5 that §6 reports as failed.*

Remark 2 (What "sound and complete" meant, and what it can mean now). Completeness cannot be driven to 1 by any improvement in perception, because $\gamma_\wedge < 1$ whenever a witnessing atom is ever off screen, and no amount of accuracy resolves an atom that is not displayed. A pixel-driven monitor therefore cannot be complete in the sense the readable-state literature uses the word. What is available is a coverage-relative notion, and it is not a weaker version of the same statement: it is the statement obtained by conditioning on the event that the question was answerable.

Proposition 2 (The threshold for each way of discharging ?)**.** *Theorem 2 is stated for pass-through, and the abstention policy decides which currency pays, so the threshold must be restated per policy. With γ the coverage, c_{esc} an escalation's price, and an escalated atom resolved off-line so that it neither misses nor blocks:*

$$\textit{pass through:}\quad \varepsilon^*_{\mathrm{pass}} = \frac{b\,c_{\mathrm{miss}}}{b\,c_{\mathrm{miss}} + (1-b)\,c_{\mathrm{fb}}},$$

$$\textit{block on ?:}\quad \varepsilon^*_{\mathrm{block}} = \frac{b\,c_{\mathrm{miss}} - (1-b)(1-\gamma)\,c_{\mathrm{fb}}}{\gamma\big[b\,c_{\mathrm{miss}} + (1-b)\,c_{\mathrm{fb}}\big]},$$

$$\textit{escalate on ?:}\quad \varepsilon^*_{\mathrm{esc}} = \frac{b\,c_{\mathrm{miss}} - (1-\gamma)\,c_{\mathrm{esc}}}{\gamma\big[b\,c_{\mathrm{miss}} + (1-b)\,c_{\mathrm{fb}}\big]}.$$

All three coincide at *$\gamma = 1$, where there is no ? to discharge, and they separate in proportion to $1-\gamma$: blocking lowers the threshold because the coverage hole becomes false blocks, escalating lowers it only by the escalation's own price, and pass-through is the case in which the hole costs completeness instead — which is why coverage is absent from Theorem 2.*

Proof. Condition on the atom. Under blocking, a violating state is caught when the atom is resolved and read $\top$ *or* when it is unresolved, giving $b[\gamma(1-\varepsilon)+(1-\gamma)] = b(1-\gamma\varepsilon)$; a non-violating state is blocked when misread *or* unresolved, giving $(1-b)[\gamma\varepsilon+(1-\gamma)]$. Setting the difference of the two priced terms to zero and solving for ε gives the blocking expression. Under escalation the caught term is the same, the blocked term is $(1-b)\gamma\varepsilon$, and $(1-\gamma)c_{\mathrm{esc}}$ is paid outright. Substituting $\gamma = 1$ in either reduces the numerator to $b\,c_{\mathrm{miss}}$ and the denominator to that of the pass-through case. □

Corollary 3 (On published compliance figures)**.** *A compliance rate measured with readable atoms is an upper bound attained at $\varepsilon = 0$, and it carries no guarantee at $\varepsilon > 0$. Since GUI grounding error is empirically non-zero at every rendering level we measured, a published compliance figure obtained on structured traces bounds — but does not predict — the behaviour of the same monitor on pixels.*

A.4 THE THRESHOLD ACROSS THE ELICITATION RANGE

Equation for ε^* has two inputs. The base rate b comes from a logged sample and is measurable. The ratio ρ comes from the deployment's incident policy and is *not*: it is elicited, it is this paper's softest input, and a single elicited value is the wrong thing to report, because two opposite risk postures can wear one deployment's name and their thresholds differ by orders of magnitude. We therefore fix a range and report the threshold across it, letting ρ run over $[\frac{1}{50}, 25]$ — from "a missed violation costs fifty times a spurious halt" to "a spurious halt costs twenty-five times a missed violation" — which spans every posture we have seen argued for, a factor of 1250. For low-base-rate deployments the sign of NB is determined across the whole range; for mid-risk deployments it is not, and that is the case in which this paper declines to claim a sign.

B FOUR OBJECTIONS

These are the four readings of the paper we expect, including the two places where an earlier version of our own argument was wrong.

(a) Is this a criticism of the monitoring literature? No, and the distinction is not diplomatic. Every result we build on is a theorem about a monitor, and a theorem about a monitor cannot be falsified by attaching a bad sensor to it. What we claim is that *no published guarantee states the condition under which it transfers to a pixel-driven agent*, and Theorem 1 is that condition rather than a correction. *The one thing we do criticise is a reporting practice*: a compliance figure presented without its grounding error is not transferable, and Corollary 3 says so precisely.

(b) A claim we withdrew. An earlier version of this work stated Theorem 3 as a *correction* to a compositional closure result: that adding a noisy channel makes the capability closure of the composition a strict subset of the policy's own. **That statement is false**, and its falsity is instructive. A compositional join of that kind is *defined* to include the degenerate interleavings with zero blocks of either type, so each component family is contained in the join by construction; a strict-subset claim about the join contradicts a definition rather than a theorem, and no measurement could ever have settled it. What survives is weaker and more useful, and it is what §4.1 now states: the closure result is true and *does not license the deployment reading*, because realising a closure needs the option of not using a channel and a gate withholds exactly that option. The failure mode — reading an existence result over architecture families as a monotonicity claim about running systems — is general and not ours alone.

(c) Correlated grounding errors. The natural intuition says that dropping Assumption 1 "moves ε^*", and it is wrong: $\varepsilon^* = \Phi(b, \rho)$ contains the base rate and the cost posture and nothing else (Remark 1); no dependence structure appears in it, so none can move it. The error — ours, in an earlier draft — was to attribute to the threshold what belongs to the two quantities *around* it. Dependence across atoms enters Theorem 1 through the conjunctive factor, and dependence across steps decides which ε is compared with the threshold.

Having separated them we can say more than "the existence survives". In both places **the measured dependence makes the reported guarantee conservative rather than optimistic**. Positively dependent atom errors fail together, so a conjunction is misread *less* often than the independent product predicts; and because errors repeat within a task there are fewer independent opportunities to false-block, so the per-step budget for a given episode exposure is looser than independence implies. §H measures both directions against simulation. *The intuition that correlation must tighten the sensor requirement is the natural one and it is the wrong way round.* What remains genuinely open is the shape of the dependence rather than its sign or its location: a model that predicted it from screen properties would be a contribution we have not made.

(d) Coverage surely matters, so isn't the coverage claim too strong? It is easy to misread as saying coverage does not matter, and it says something narrower and more useful. Figure 3 is the coverage census by atom class, and Figure 11 traces the envelope the two sensor quantities jointly define. Coverage decides *how much* of a property the monitor speaks to, and therefore the magnitude of whatever effect the monitor has; accuracy decides the *sign* of that effect (Remark 1, Remark 2). Raising coverage on a system above ε^* scales harm, and that is the claim — not that coverage is irrelevant but that it is the wrong knob to reach for first. The practical ordering follows: measure ε against ε^*, fix the sign, and only then buy coverage.

C THE COMPOSITION, DRAWN

Four pictures, each carrying one step of the argument that a scalar cannot. The first is the object: a monitor bolted to a sensor rather than placed beside one. The second says why the substitution is not a change of implementation but a change of channel class. The third separates the two "don't know"s that a two-valued verdict collapses. The fourth is the composition itself, measured — because the claim that a gate is not free composition is the one a reader is most entitled to see demonstrated rather than asserted.

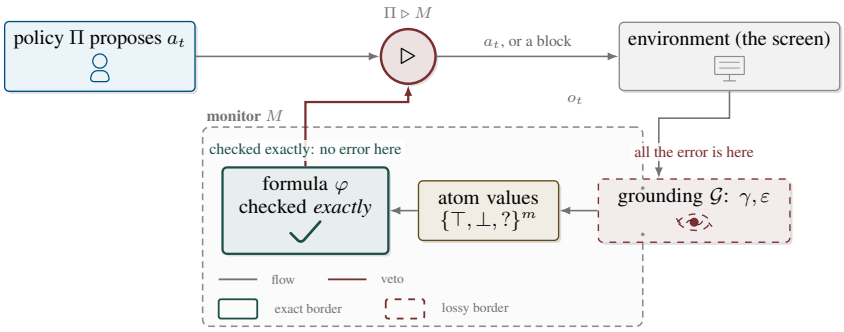

Figure 5. The paper in one picture: a monitor bolted to a sensor, drawn as the loop it actually closes. M (atom values, formula) is the boundary; $\mathcal{G}$ is bolted *onto* its input face rather than placed beside it, because the gate's verdict is read off the same screen the gated action would change. **The guarantee belongs to the pair, and its two quantities decide different things.** A predicate is checked exactly against readable state; a GUI agent inserts the grounding stage in front of that check, and that stage alone carries coverage γ and error ε. Coverage fixes how often the atom is answerable at all and therefore how *large* the monitor's effect is; accuracy fixes its *sign*. Past ε^* the gate is worth less than no gate at all — and because a veto the policy may ignore is not a veto, this is a claim about the forced composition drawn here, not about which task families are solvable elsewhere.

Why the sensor is drawn on the input face and not beside the monitor. The placement in Figure 5 is the whole argument in one design decision. Drawn beside the monitor, $\mathcal{G}$ is an auxiliary component whose error is one more source of noise to be reduced and the guarantee still belongs to M. Drawn *on the input face*, the guarantee belongs to the pair, and the two quantities the sensor contributes decide different things: coverage γ fixes how often the atom is answerable at all, and therefore how large the monitor's effect can be; accuracy ε fixes that effect's *sign*. A reader who accepts only the first drawing will read Theorem 1 as a robustness result. It is not: it is a statement about which object the published guarantee was ever attached to. The next figure says why the two drawings are not interchangeable.

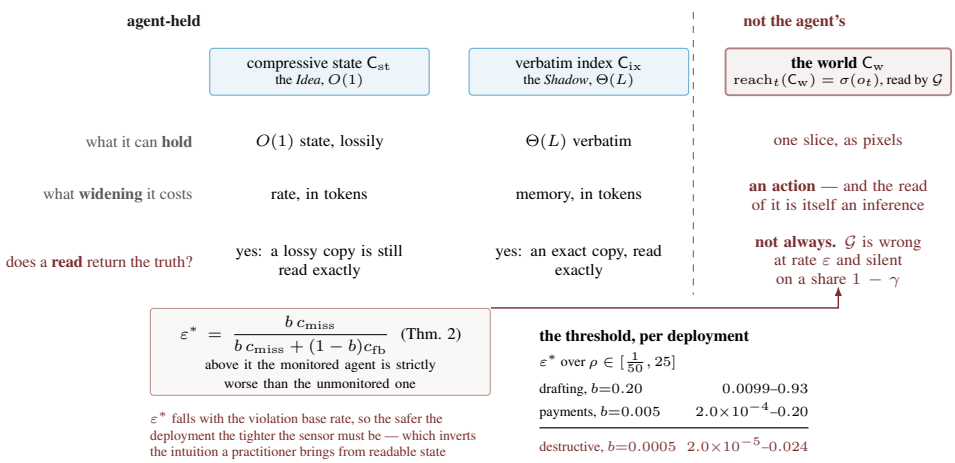

Figure 6. Three channels, and the one column where a read may not return the truth. For the agent-held pair the third row is trivial: a lossy copy is still *read* exactly, and so is an exact one, because reading a copy is a lookup. *For the world channel the read is an inference*, performed by a grounding operator that is wrong at rate ε and silent on a share $1-\gamma$ of atoms — so the object a monitor is attached to is a monitor *plus a sensor*, and the guarantee belongs to the pair. The threshold in the box is the error rate past which that pair is worse than no monitor at all (Theorem 2), and the table evaluates it on three deployments — over the whole elicited posture range $\rho \in [\frac{1}{50}, 25]$ of §A.4 rather than at one elicited point, so that no reader has to accept a cost model of ours to read it. **Read the table's last row against its first**: ε^* falls with the violation base rate, so the requirement tightens fastest in exactly the deployments that most want a monitor, which inverts the intuition a practitioner brings from readable state.

Two "don't know"s, and why one symbol cannot carry both. A runtime monitor already has a third value, and it means *the prefix has not yet decided the property*. A grounded monitor acquires

a second one that means *the screen did not yield the atom*. These are different coordinates of the monitor's state, not two readings of one: the temporal one is discharged by waiting and carries no rate, the perceptual one is discharged only by *acting* — scrolling, opening, re-rendering — and carries γ. Figure 7 draws the product, and the shaded cell is the state no deployed formulation can represent. *The count is not hypothetical*: at the rendering level where our threshold is crossed, 76 of 240 episodes occupy it, and both two-valued monitors record every one as a clean verdict.

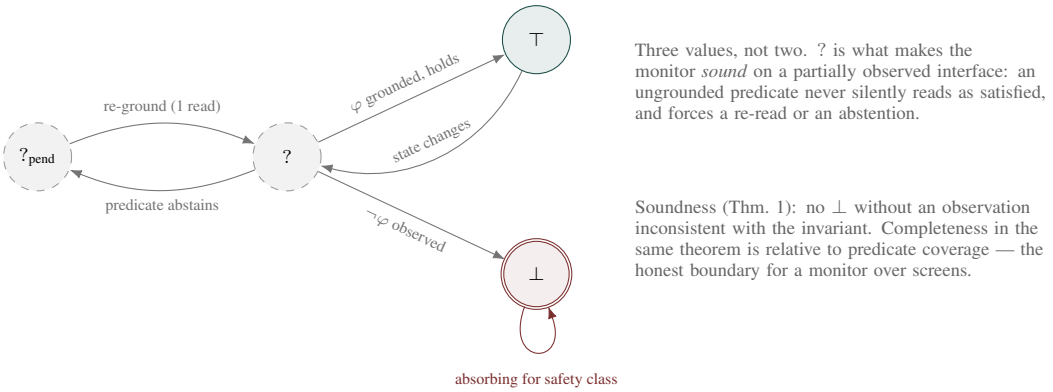

Figure 7. The monitor's state is a pair, as an automaton. Temporal status crossed with perceptual status; the shaded state is the one existing formulations cannot represent and therefore report as a pass. Figure 4 is the same object as a lattice, with the discharge policies of Proposition 2 attached to its one unrepresentable cell.

The composition, measured rather than asserted. The claim that a gate is not free composition is the one a reader should most want demonstrated, because it is what separates this paper from the observation that noisy channels are noisy. Under free composition an agent may *decline* a channel, and declining is what makes composition monotone: a channel that is worse than nothing is simply not consulted. A gate withholds exactly that option — the monitor's verdict is binding, so a read the agent would have ignored becomes a veto it cannot — and Proposition 1 is the gap this opens. Figure 8 is that gap on the audit's own runs.

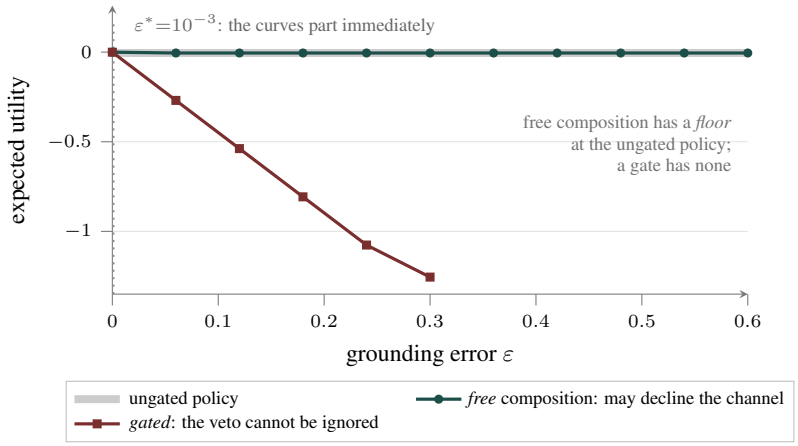

Figure 8. A gate is not free composition, and the difference is measurable. Expected utility against grounding error at b=0.005, $c_{\text{fb}}/c_{\text{miss}}$=5, γ=0.9, m=1, so $\varepsilon^*=10^{-3}$. **Free composition has a floor; a gate has none.** Under free composition declining a channel is one of the options, so the achievable utility is $\max(U_{\text{ungated}}, U_{\text{gated}})$ and the curve flattens onto the ungated policy the moment the monitor stops paying. A monitor whose veto may be ignored is not a monitor, so the gated system takes U_{gated} whatever it is, and past ε^* that is strictly worse — -1.08 against -0.005 at ε=0.24, a factor of 200. **This is consistent with the closure result, not a correction to it.** A proof that capability is strictly super-additive under *free* channel composition is a statement about closures of architecture families in which declining is permitted. The deployment claim usually read off it — that adding a verification channel cannot hurt — quantifies over a different object, and the two curves here are that difference drawn. Nothing above is a refutation of anything; it is a scope boundary, and the figure exists so that the boundary has a number on it.

D THE THRESHOLD AS A REGION

Theorem 2 gives a crossing point, and a crossing point reported as a single number invites a reader to check whether their deployment is above or below it using an elicited cost ratio they do not have. The two figures here report it as a region instead, over the whole posture range of §A.4, so that the conclusions which do not depend on the elicitation can be separated from the one that does.

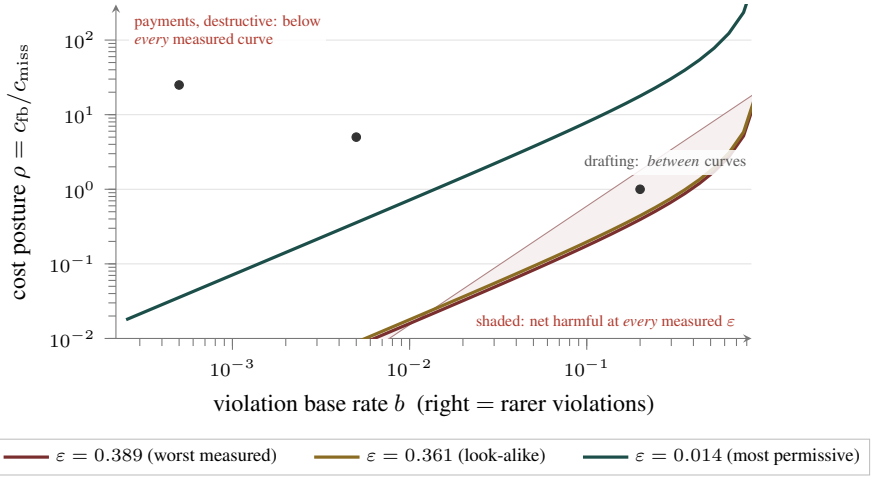

Figure 9. The harm threshold as a region, not an operating point. Each curve is $\varepsilon^*(b, \rho) = \varepsilon_0$ for one *measured* grounding error ε_0, solved for the posture, with the three ε_0 values read from the audit results: the most permissive cell the audit measured, the look-alike error on the irreversible guard, and the worst cell. Below a curve, a monitor at that measured error is net harmful; the shaded region is below all three. The marked deployments are those of §4: payments and destructive actions lie below every measured curve, so no elicited posture and no measured operator rescues them, while drafting lies between curves — safe at permissive errors and harmful at severe ones — which is exactly why its sign is not claimed.

Why the elicited number is reported as an axis and not as a value. $\rho = c_{\text{fb}}/c_{\text{miss}}$ is not observable from a trace. It is a statement about what a deployment would rather suffer, and no amount of logging produces it. A paper that elicits one value and reports a single ε^* has buried its softest input inside its headline number, where a reader cannot reach it. *So we report the threshold over the whole range instead*, and separate the two kinds of conclusion the range supports. The conclusions that survive *every* ρ are the structural ones: that a threshold exists, that it is invariant in coverage (Remark 1), and that payments and destructive actions sit above every curve. The conclusion that depends on the elicitation is the only one a reader must supply their own number for — whether a *particular* mid-risk deployment is above or below — and we do not claim its sign.

The invariance, which is the one result that costs nothing to check. Coverage and accuracy enter the threshold differently, and the difference is sharp enough to be worth stating twice. Coverage scales how much of the episode the monitor has an opinion about; accuracy decides whether those opinions are worth having. *Only the second can move the crossing point*, and Figure 11 is that claim measured: over all 36 cells, sweeping γ from 0.05 to 1 moves ε^* by 0.000, exactly. The practical consequence is the opposite of the natural intuition. A team that improves its sensor's *coverage* — more atoms answered, fewer abstentions — has not moved the point at which its monitor starts subtracting utility; it has only made the monitor's existing sign apply to more of the episode. If that sign is negative, better coverage makes the deployment worse.

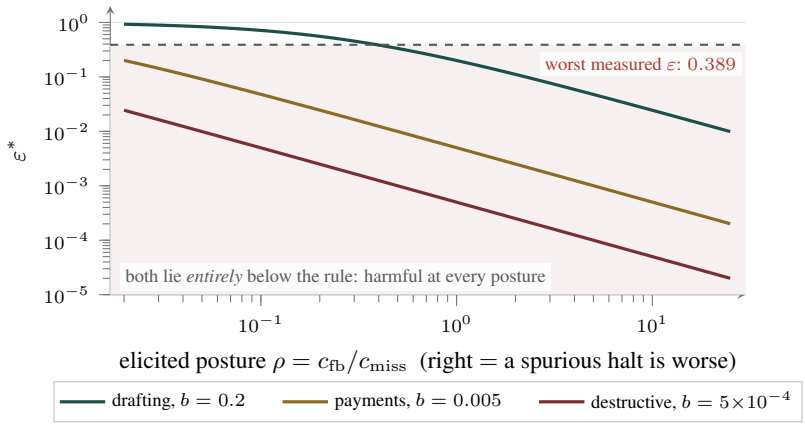

Figure 10. The threshold over the elicited posture, because one elicited number is the wrong thing to report. ε^* against $\rho = c_{\text{fb}}/c_{\text{miss}}$ for the three deployments of §4, on log axes, over the whole range of postures we have seen argued for — from a missed violation costing 50 times a spurious halt to a spurious halt costing 25 times a missed violation, a factor of 1250. The dashed rule is the worst ε CON-150 measures, 0.389 (monetary constraints, degraded rendering); the irreversible-action guard under look-alike widgets is 0.361 and either serves. **Read which curves lie entirely below the rule.** Payments and destructive actions do, so for them the monitor is net harmful *at every posture in the range* — by 1.9× and 16× at the most permissive end, and by four orders of magnitude at the least — and the elicited number, the paper's softest input, does not enter the conclusion at all. Drafting does not: its band straddles the rule, so there the sign genuinely depends on the elicitation and we do not claim it. *One range, reported once*: a paper whose recommendation is that a monitor ship with its ε^* should not carry two elicitations of ρ, and reporting the whole range is what makes the single report unnecessary.

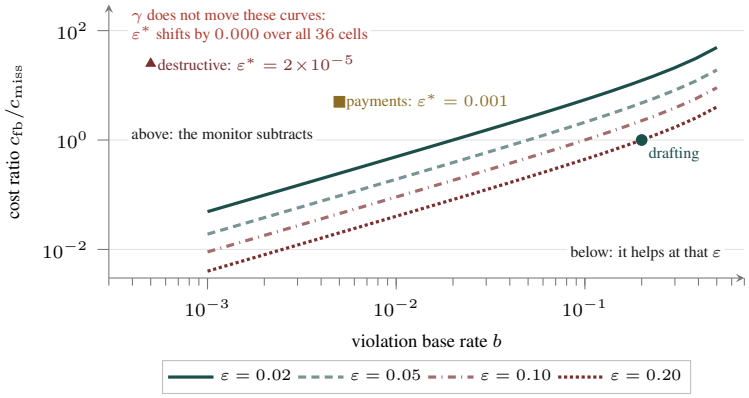

Figure 11. The envelope, and Remark 1 in one picture. Each curve is the largest cost ratio $c_{\text{fb}}/c_{\text{miss}}$ at which a monitor of grounding error ε still raises expected utility; below a curve the monitor helps, above it the monitor subtracts. *The curves do not move with coverage*: over all 36 cells, sweeping γ from 0.05 to 1 moves ε^* by 0.000, exactly. Payments and destructive actions sit above *every* curve. Single-atom property, the generous case; §D carries $m = 1, \ldots, 4$.

E Every Claim, with its Evidential Status

One row per claim the paper makes, with what backs it. *Formal* means proved here; **measured** means run on CON-150 or on the audit; **simulated** means the calibrated stress test of §H; *withdrawn* means an earlier version of ours asserted it and the run refuted it. Rows scored against a registration point at §I.

Claim	Status	Evidence
1. Soundness and completeness of the composed system in closed form, with coverage as its own coordinate	formal, conditional on Assumption 1	Thm. 1

continued overleaf

Claim	Status	Evidence
2. Coverage-relative completeness is the strongest completeness available on pixels	definitional; not expressible under readable state	Rem. 2
3. Monitoring under noisy or partial observation is *not* new; what is new is the coverage coordinate and the sensor parameterisation	correction to our own framing	§2, Taleb et al. (2023); Stoller et al. (2011); Junges et al. (2021)
4. $\varepsilon^* = \Phi(b, \rho)$ in closed form, and it carries no coverage, no formula shape and no dependence structure	formal	Thm. 2, Rem. 1
5. ε^* is a point on a false-block/miss frontier that concurrent work derives exactly	prior art for the threshold; ours is the *sensor* parameterisation, not the frontier	Ray (2026)
6. Gating a policy by a noisy monitor is not utility-monotone: past ε^* the gated system is strictly worse	formal	Thm. 3
7. This is consistent with closure super-additivity rather than a correction to it	formal; the two quantify over different objects	Prop. 1, App. B(b)
8. At $\varepsilon = 0, \gamma = 1$ gating is monotone, which is why the question has not arisen before	formal	Cor. 1
9. Each audited monitor has an ε^* inside the empirical range of GUI grounding error, and it predicts the sign of the net benefit	**measured**, against each monitor's own discharge policy; 9 of 9 rows	P2, Tab. 1, Prop. 2
10. Where the transfer loss lands — soundness or availability — is set by the monitor's semantics, not by the sensor	**measured**, and not anticipated: the registered falsifier presupposed one architecture	P1, Tab. 1
11. Coverage rather than accuracy dominates the transfer loss for off-screen atoms	**measured**: coverage falls to 0.000–0.986 once the atom leaves the screen	P5, Fig. 3
12. ~~Look-alike widgets cost coverage~~	**withdrawn**; measured, coverage is *exactly* 1.000 on all five classes under look-alikes while ε rises to 0.361 — they cost accuracy	P5, Tab. 4
13. 76 of 240 episodes contain an atom the operator will not resolve, and both two-valued monitors record every one as a clean verdict	**measured**; the count was registered in advance, including the branch in which it is zero	P3, §6
14. The cost ratio ρ that fixes ε^* in a deployment	elicited, not measured; reported as a range and as a region, never as a point	§4, Figs. 10, 9
15. Every threshold measured here is *open-loop*: the corpus fixes the proposal distribution	scoping, against us; quantified under a resample model ($1.6\times$ tighter, verdict survives the sweep); agent-in-the-loop measurement owed	§H, Fig. 16
16. The independent closed form is conservative for the net benefit under positive dependence	**simulated**, calibrated to the measured quantities	P4, §H
17. The threshold crossing lands on the closed-form prediction at the sample size the margin needs	**simulated**, n=3000/cell: crossing $[0.220, 0.250]$ brackets $\varepsilon^* = 0.2375$	§H, Fig. 15

F The Related-Work Audit at Full Length

Table 3 is the row-by-row version of §2, and the prose below makes slowly an argument that section compresses — because the premise it identifies is easy to overstate, and two of its rows are owed priority for results we also derive.

Table 3. The audit, row by row. Every row is a correct result about a channel that reads exactly. The third column is the limit this paper occupies; the fourth says what we do with the row. *No row is refuted.*

Result	Assumption it needs	Limit in which it fails	Role here
DSL→SMT monitoring, full compliance reported (Kamath et al., 2025)	atoms evaluate exactly against readable state	$\varepsilon > 0$: soundness falls at rate $\nu_1(1-b)/b$ (Thm. 1)	SCOPE; audited
Safety-fragment DFA monitoring (Xavier et al., 2026)	same, plus coverage $\equiv 1$	completeness is capped at $\gamma_\wedge < 1$ and no accuracy recovers it (Rem. 2)	SCOPE; audited
Three-valued RV semantics (Bauer et al., 2011; Pnueli, 1977)	the third value is *temporal* pendency	a second, orthogonal "don't know" with a rate attached	adopted
Certified enforcement for tool-using agents (Ray, 2026)	a score with an operating point, on readable arguments	the sensor has a *coverage* hole no operating point closes; our ε^* is a point on their frontier	**PRIOR ART** for the threshold
RV under uncertainty (Taleb et al., 2023; Junges et al., 2021; Yingchareonthaworn-chai et al., 2016; Stoller et al., 2011)	partial or noisy observation of a *modelled* state	the noise is a learned operator on pixels, with no model to condition on	**PRIOR ART** for the premise
Gated supervisory control, measured (Rosenthal, 2026)	—	534 of 590 gate interventions are spec-on-bound: the mechanism of Thm. 2, in a deployment	INPUT: an independent instance of the false-block arm
Shielding and guardrails (Alshiekh et al., 2018; Chen et al., 2025; Wang et al., 2025)	the intervention reads the situation correctly	the false-block arm, negligible in their setting and not in ours (Thm. 2)	SCOPE
Certified *behavioural* alarms on computer-use agents (Kumar et al., 2026)	the monitored signal is harness telemetry; the certificate bounds false alarms	one-sided by design; our threshold prices false blocks *against* misses, which is what decides net harm at all	closest deployed instrument; complementary
Audit-and-intervene monitoring (Alamdari et al., 2026)	same	same	SCOPE
GUI grounding benchmarks (Cheng et al., 2024; Li et al., 2025; Lin et al., 2025)	—	—	INPUT: the range of ε
Pop-up and look-alike attacks (Zhang et al., 2024; Yan et al., 2026)	—	—	INPUT: adversarial ε
GUI critic models (Wu, 2026)	the critic reads the state the actor acted on	critic and actor share $\mathcal{G}$, so their errors are correlated	OBJECT
Self-critique and self-consistency (Gou et al., 2024; Wang et al., 2023; Shinn et al., 2023)	disagreement indicates a reasoning error	agreement through *one* noisy read is not evidence	OBJECT
Design by contract (Meyer, 1992; Hoare, 1969)	preconditions are decidable predicates on state	here a precondition is an *inference* with a rate	precedent, reconstructed
Information-flow and taint lattices (Denning, 1976; Biba, 1977; Newsome & Song, 2005)	the label on a value is exact	the label is the output of a classifier	precedent; SCOPE

continued overleaf

Result	Assumption it needs	Limit in which it fails	Role here
Design-time injection defences (Willison, 2023; Debenedetti et al., 2025)	the untrusted/trusted split is structural	ε is not an adversary and does not respect the split	orthogonal
Capability from access structure (Anonymous authors, 2026)	free composition; declining a channel is an option	a gate withholds that option (Prop. 1)	SCOPE, not correction

The premise, narrowed. We state it narrowly: *the agent-monitoring results this paper audits* assume readable structured state. Runtime verification generally does *not*. Monitoring under partial and noisy observation is an established branch with its own survey (Taleb et al., 2023); monitors have been characterised by *precision and recall* — the two quantities this paper uses — since at least 2016 (Yingchareonthawornchai et al., 2016); a monitor that estimates the probability a property holds from an execution with gaps is from 2011 (Stoller et al., 2011); and monitoring a partially observable MDP, where the state must be inferred and the verdict is a conservative conditional reachability, is a CAV result (Junges et al., 2021). *The gap is not "nobody handles noise"*; it is that the branch which handles it parameterises by state-estimation error *against a model of the system*, and a GUI agent has no such model. Its noise is a learned operator on pixels with a *coverage* hole — predicates it cannot evaluate at all, at any operating point — and coverage is the parameter neither branch carries. That is the contribution's actual boundary, stated where a reader can check it.

Certified enforcement, and where our threshold sits inside it. The most direct precedent for Theorem 2 appeared a month before this draft. A theory of certified runtime safety for tool-using agents derives *the exact false-block/miss frontier* by Neyman–Pearson, gives conformal finite-sample certificates for it, and — this is the part we had not considered — observes that once blocking changes future proposals, static scores and ungated trajectories need not identify the closed-loop frontier at all (Ray, 2026). *Our ε^* is a point on that frontier*, not a new object: it is where the frontier crosses the utility of not monitoring, written in the base rate and the miss-to-false-block cost ratio. We say so plainly. What remains ours is what their parameterisation does not carry: a *sensor* rather than a score, so the operating point is not free to move (Remark 2); a coverage term, so completeness is capped below one independently of accuracy; and a third truth value recording *the oracle answered and may be wrong*, as distinct from temporal pendency. *Their closed-loop observation is a limitation of ours* and is recorded as one: our audit measures an open-loop frontier.

DSL-to-SMT monitoring, this paper's principal object, and right about its own setting. A recent system compiles temporal properties to first-order logic, discharges them with an SMT solver, backtracks through constrained decoding when a check fails, and reports full compliance on its benchmark (Kamath et al., 2025). *We regard the construction as correct and the reported number as real. Assumption:* predicate atoms evaluate exactly against readable state — and this is not an idealisation in their setting, it is a fact, because their atoms are tool-call arguments. *Failure limit:* $\varepsilon > 0$. Theorem 1 converts a compliance figure into a function of ε, and Theorem 2 says where that function's contribution to net benefit changes sign. **The reported hundred percent is therefore not a number about the monitor alone**, and the substitution a GUI deployment performs — swap the readable atoms for pixel-derived ones — is exactly the substitution no published guarantee licenses. *Our audit re-runs their monitor unchanged*, which is the only fair way to ask the question: a re-implementation would confound the monitor's design with our implementation of it, and the whole point is that the monitor is not what changes.

And pointing their construction at our threshold found something that sharpens their result rather than weakening it. Constrained decoding is a *retry* rather than a block: when a check fails the system regenerates rather than refusing. A retry has a smaller c_{fb} than a block, since the cost of a spurious intervention is a wasted generation rather than a refused action, and a smaller c_{fb} raises ε^*. *So their architecture is more robust to grounding error than a blocking monitor is*, for a reason their paper does not claim and ours can compute. We think this is the most useful thing the threshold does: it orders intervention mechanisms by how much sensor error they tolerate, and it puts retry above block.

Three-valued LTL monitoring, and a misattribution of ours. Three-valued semantics for runtime verification is standard and predates all of this work (Bauer et al., 2011; Pnueli, 1977); the third value there is *temporal* pendency — the observed prefix has not yet decided the property. *We claim none of that.* It is worth being precise about who owns what, because an earlier version of this section was not. Agentproof compiles a seven-form policy DSL to deterministic finite automata and checks agent workflow graphs both statically, through a graph$\times$DFA product, and at runtime over event traces (Xavier et al., 2026); its contribution is that the graph structure these frameworks expose makes pre-deployment verification possible at all, and *it does not use three-valued semantics*. We had previously attributed the three-valued machinery to it, which was wrong. What we take from it is the DFA construction and the observation that practical agent policies fit inside a small fragment; what we take from the runtime-verification literature is the temporal third value. *What we add* is an orthogonal axis: their third value is temporal and ours is perceptual, they are different coordinates of the monitor's state, and we keep them apart rather than collapsing them into one symbol. Only the perceptual coordinate carries an error rate, and only it is reducible by acting.

Shielding and guardrails, where the arm we price is the arm their setting makes negligible. Shielding architectures and probabilistic guards intervene on proposed actions (Alshiekh et al., 2018; Chen et al., 2025; Wang et al., 2025), and audit-and-intervene monitoring has been formalised for accountability (Alamdari et al., 2026). *Assumption:* the intervention decision rests on a correct reading of the situation. *Failure limit:* the false-block arm of Theorem 2. It is worth being exact about why these papers do not price that arm, because the reason is good rather than careless. With an exact reading, a false block requires the *policy* to be misjudged, which is a modelling error handled by making the shield conservative. With an inexact reading, a false block requires only that the sensor misread one atom, which happens at rate ε per atom and *compounds across a conjunction*. *Conservatism is the standard remedy in that literature and it is the wrong remedy here*: a more conservative shield blocks more, which moves mass into precisely the arm whose cost is now non-negligible, so the intuition trained on readable state points the wrong way. That inversion is why we report blocks and unresolved atoms separately rather than pooling them into a compliance figure.

GUI grounding as the actual bottleneck, and why its numbers make the threshold binding rather than asymptotic. Grounding models and benchmarks report the error rates ε denotes (Cheng et al., 2024; Li et al., 2025; Lin et al., 2025), and layout-guided grounding observes that similar-looking elements differ in function across applications (Liu et al., 2025) — the failure mode with the worst consequence here, since a correctly located widget with a misread meaning produces a confidently wrong atom rather than an abstention. *Use:* these papers give the empirical range of ε on realistic screens, and that range is what makes ε^* a practical statement rather than an asymptotic one. *We are careful about the direction of the inference.* We do *not* claim any particular deployment is above its threshold: published error rates are measured on grounding benchmarks whose screen distribution is not a deployment's, and a monitor's atoms are typically simpler than a benchmark's hardest targets. What we claim is that the published rates and the thresholds a low-base-rate deployment implies are *the same order of magnitude*, which is enough to make the comparison worth performing.

The perception stack whose error rate is our parameter. Generalist action models and grounding stacks report the quantity ε denotes, and report it as a number to be driven down (Hong et al., 2024; Qin et al., 2025; Wu et al., 2025; Xu et al., 2024); adaptive-perception agents trade a slow careful pass against a fast one (Mehrotra et al., 2026); recent work isolates exactly the atoms a safety monitor cares about — controls whose *state* is not visually obvious, such as toggles (Wu et al., 2026; Hu, 2026); and data-scale studies bound how far the number moves with supervision (Li et al., 2024). *Assumption:* ε is a leaderboard quantity, monotonically good to reduce. *What changes here:* in a monitor ε is not a score but a *parameter whose value flips the sign of a decision* (Theorem 2), and the flip happens inside the range these papers currently report. We take their measurements as our parameter range and add nothing to the perception problem.

Compliance reported without the false-block arm. Hierarchical and multi-platform evaluation frameworks now score agents on constraint adherence alongside task success (Wang et al., 2026; Li et al., 2026; Ai et al., 2026; Yang, 2026), and training environments supply the trajectories (Bai et al., 2026). *Assumption:* an intervention that halts a wrong action is free, so compliance can be reported as a single number. *Failure limit:* Theorem 2 has two arms and only one of them is a missed violation.

A compliance figure with no false-block figure beside it is not a partial report of monitoring quality; it is a report of one of two terms whose difference decides whether the monitor should be switched on at all.

Critics and self-critique as monitors. A critic model that judges whether a step was correct is, formally, a monitor: it reads the interface and returns a verdict on a proposed or completed action. Cross-platform GUI critics are trained explicitly for this role (Wu, 2026), and the older self-correction literature uses the actor's own forward pass (Gou et al., 2024; Wang et al., 2023; Shinn et al., 2023). *Assumption:* disagreement between the two indicates a reasoning error. *Failure limit:* critic and actor share the grounding operator $\mathcal{G}$, so their errors are correlated and agreement through one noisy read is not evidence. This is the row on which our result bites hardest, because the remedy the literature reaches for — ensemble the judgements — does not address a common-mode sensor.

The older precedents, which we reconstruct rather than cite in passing. Design by contract assumes preconditions are decidable predicates on state (Meyer, 1992; Hoare, 1969); information-flow and taint lattices assume the label on a value is exact (Denning, 1976; Biba, 1977; Newsome & Song, 2005). Both are correct and both become something else when the predicate is an *inference with a rate* and the label is a classifier's output. Design-time injection defences (Willison, 2023; Debenedetti et al., 2025) are orthogonal rather than in scope: ε is not an adversary and does not respect the untrusted/trusted split.

G The Measurement Audit in Full

The main text reports the audit pooled over atom classes, because a single ε per rendering level is what Theorem 1 takes. That pooling hides two things a reader is entitled to see. *First, the degradations are not uniform across classes* — the class with the worst error is the class the threshold is about, which flatters nothing and is worth showing. *Second, the two rendering failures are different failures*, and a single error rate cannot say so: one moves coverage and one does not. This section carries both, together with the confusion matrices that separate ε into the precision and recall coordinates soundness and completeness actually depend on, and the latency reading that samples the temporal currency over time.

One premise of ours failed here and the failure is in the table. We expected look-alike widgets to cost *coverage*: a widget the operator cannot identify is one it should decline to answer about. Measured, coverage is exactly 1.000 at look-alike on every class — the operator answers confidently and is wrong — so the adversarial level costs accuracy instead, reaching $\varepsilon = 0.361$ on irreversible actions with γ untouched. That is the worse of the two outcomes for a deployment, since an abstention is visible and a confident error is not.

Table 4. Per-class coverage and error, at all three rendering levels. γ and ε are measured on the same reads, 72 per cell over 24 distinct screens; the last row pools classes and is the figure Table 1 reports per level. **Coverage is exactly** 1.000 **at native and at look-alike on every class**, so the premise that look-alikes would cost coverage failed and they cost accuracy instead — ε reaches 0.361 on irreversible actions with γ untouched. Degraded is where coverage is lost, on four of five classes, worst 0.783.

	native		degraded		look-alike	
constraint class	γ	ε	γ	ε	γ	ε
idempotence	1.000	0.000	0.875	0.357	1.000	0.000
irreversible	1.000	0.014	0.875	0.111	1.000	0.361
monetary	1.000	0.000	0.783	0.389	1.000	0.000
ordering	1.000	0.000	1.000	0.194	1.000	0.028
privacy	1.000	0.014	0.986	0.183	1.000	0.043
monitored run, pooled	0.999	0.004	0.918	0.143	0.992	0.084

Reading Table 4 down the classes, which is where the pooled row is misleading. Pooled, the look-alike level looks mild: $\varepsilon = 0.084$ against native's 0.004. Per class it is not mild, it is *concentrated*. Four of the five classes are untouched at look-alike — idempotence and monetary at $\varepsilon = 0.000$, ordering at 0.028, privacy at 0.043 — and the entire effect sits on **irreversible**, at 0.361.

That is the class the threshold is about, and a pooled number divides its error by five. The degraded level has the opposite shape: error is spread over every class (0.111 to 0.389) and coverage is what moves, falling to 0.783 on monetary and 0.875 on two others while ordering holds at 1.000. So the two rendering failures are not two magnitudes of one thing. *One takes coverage broadly and one takes accuracy narrowly, on the one class where an error is least recoverable* — and a monitor scored on a pooled ε cannot tell a reader which of the two it is facing.

We report the per-class table rather than the pooled row alone because the pooled row is the one that flatters us. An audit whose headline is "ε reaches 0.084 under adversarial rendering" invites the reading that the adversarial level is the gentler of the two; read down the irreversible column it is the sharper one, and it is sharp exactly where Theorem 2's cost ratio is largest.

Why the confusion matrices, and not one error rate. Soundness and completeness fall at *different* rates in the two coordinates of the sensor's error, so a pooled ε is the wrong resolution at which to read a monitor's degradation. A false positive on an atom makes a compliant step look like a violation and lands in the false-block arm; a false negative lets a violation through. Figure 12 resolves ε into the two, and the separation is what makes the look-alike level interpretable: it leaves recall at 1.000 on four classes and takes precision on exactly one.

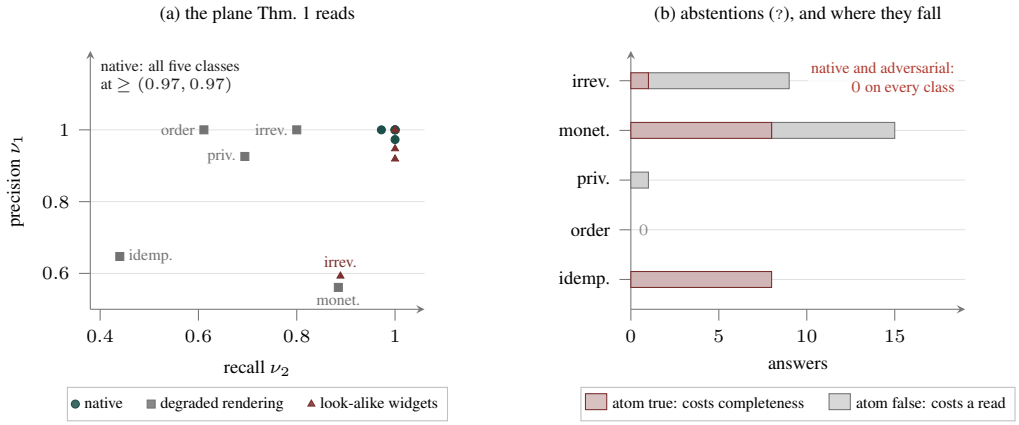

Figure 12. Atom confusion. Confusion matrices per atom class on CON-150, resolved into the two coordinates Theorem 1 actually takes. **The two rendering failures are different failures.** Degraded rendering moves precision *and* recall and scatters all five classes into the interior of the plane; look-alike widgets leave recall at 1.000 on four classes and take precision on exactly one — irreversible, to 0.593. That is the one-class effect Figure 3 found in ε, now separated into which of the two errors produces it, and the separation matters because soundness and completeness fall at different rates in the two coordinates (§3). **Abstention is measured here and it is not a design choice.** All 33 abstentions in the run occur under degraded rendering; native and adversarial return zero on every class. So on this evidence the operator emits the third truth value when the screen is hard to read, not when a designer decides which of Corollary 2's two currencies to pay — and the 8 abstentions on true atoms in each of *idempotence* and *monetary* are completeness lost with no soundness bought.

Latency, which is the temporal currency sampled over time. Corollary 2 says a discharge of the unresolved state is paid in one of two currencies, and one of them is temporal: a violation detected late is not a violation prevented. *The reading that matters is that latency does not track the sensor's error rate*; it tracks the *miss half* of it, which is a different quantity and is empty at exactly the level where ε is largest. The look-alike column is therefore a natural experiment rather than a robustness check, and it reaches the same separation as the confusion matrices by an independent route.

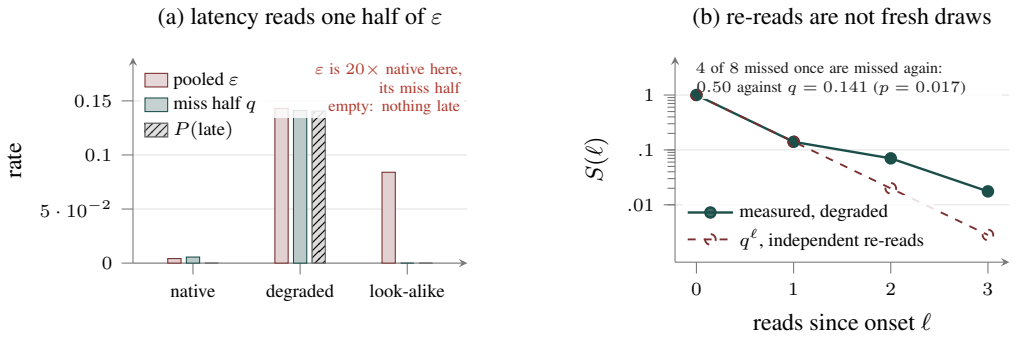

Figure 13. Latency, measured: the temporal coordinate is one of the two currencies, sampled over time. Recovered from the cached monitored run at zero spend, with censoring logged per episode on all 240. **(a) Latency is not a function of the sensor's error rate. It is a function of the miss half of it.** Three rates per level: the pooled ε, the operator's per-read miss rate q, and the share of violations detected late. *The third tracks the second and not the first*, agreeing with it to within one episode in 57 at every level, and the look-alike column is where the two readings of "the sensor's error" come apart: $\varepsilon = 0.084$ there — twenty times native — while q and the late-detection rate are both exactly 0.000, because look-alike widgets make the operator call a violation on a safe step rather than miss a real one. *That level is the natural experiment*: it is the one place where ε is large and its miss half is empty, so it separates two quantities a single error rate pools, and Figure 12 reaches the same separation by the independent route of the confusion matrices, where look-alikes leave *recall* at 1.000 and take precision. The temporal coordinate is therefore *not* orthogonal to the sensor; it is Corollary 2's completeness currency, read along the time axis rather than across predicates. **(b) And watching longer is not the remedy, because the extra reads are not extra evidence.** Independent re-reads would give survival q^{ℓ}. The first read is predicted almost exactly, 8 of 57 episodes still undetected at onset against $57q = 8.05$; the second is not, and 4 of those 8 are missed *again* — 0.500 against $q = 0.141$, Wilson $[0.215, 0.785]$, $p = 0.017$ under independence. The operator repeats its own mistake on the screen it got wrong, which is the same within-episode correlation §J reports on the false-block axis as $T_{\text{eff}} = 2.2$ against a nominal 6: *one correlation, and it charges both currencies. The third read is* 1 *of* 4 *and no trend is drawn through it.* **This monitor reads every step, so what is drawn is the floor.** A monitor sampling every k steps adds latency by construction; what (a) and (b) bound is the part the sensor contributes, and it is the part no schedule can remove.

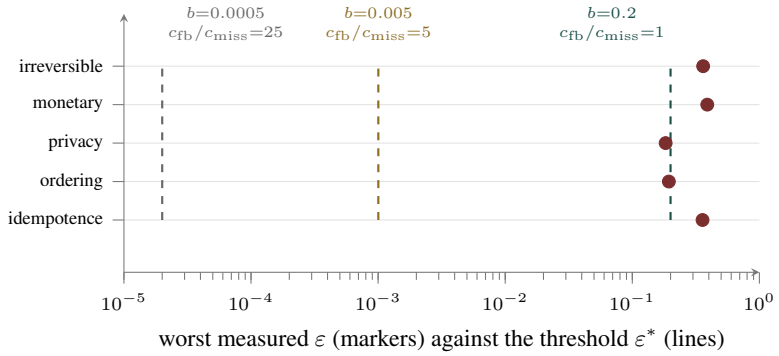

Figure 14. Calibration: measured ε against the threshold it would have to beat. Bars are the worst ε CON-150 measured for each atom class across the three real rendering levels; dashed lines are ε^* computed in closed form from a stated violation base rate and cost ratio. The assumed values are printed on the figure because they are assumptions — a reader deploying at a different base rate should move the lines, and the point is how far they would have to move. **Log axis, because the comparison is not close.** Every class clears the permissive threshold, ε^*=0.20 at a one-in-five violation rate with symmetric costs, roughly a drafting assistant. None clears 1.0×10^{-3}, and none comes within three orders of magnitude of 2.0×10^{-5} — the threshold for an action that is rare and expensive to block wrongly, which is what a safety gate is installed for. **This is the calibration that makes Theorem 2 say something rather than merely be true.** The theorem gives a crossing point; whether a deployment sits on the wrong side of it is empirical, and on these five atom classes with three open-weight operators the wrong side is the only side reached. We do not claim this generalises to a stronger grounding operator: it bounds what these do, and the same figure recomputed at a lower ε is the shape of the repair.

H Resolving the Crossing without Claiming the Audit Did

§6.2 reports that 240 operator episodes cannot resolve the decisive row, and that $\approx 22{,}000$ would be needed. The two figures here are what we do instead: a simulation calibrated on the audit's own measured parameters, and the closed-loop correction. Neither is new operator data and neither is presented as such.

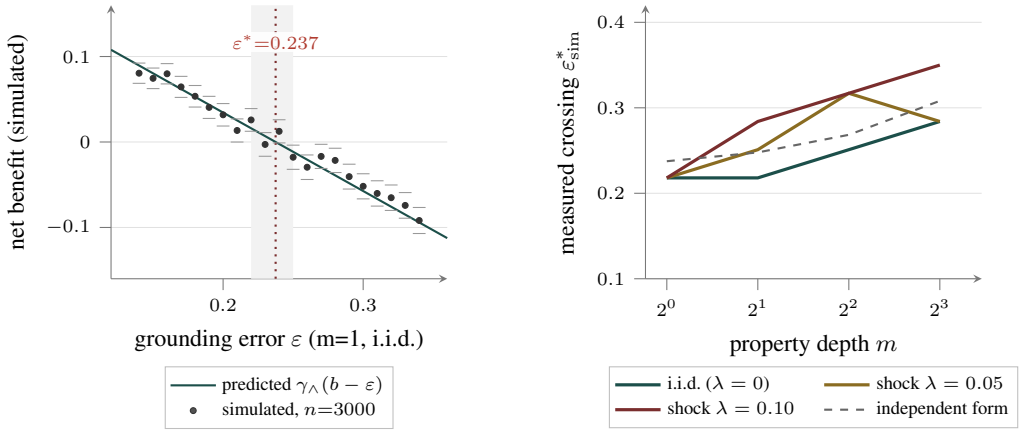

Figure 15. Calibrated simulation, not new operator data. (a) The threshold neighbourhood the audit could not resolve: net benefit against ε at n=3000 episodes per cell with the audit's own episode accounting and measured parameters; the crossing interval brackets $\varepsilon^* = b = 0.2375$, and the predicted line is drawn, not fitted. (b) The crossing as the property gains atoms and the reads gain a common shock: depth and dependence both move it up, the conservative direction, and the independent-form crossing (dashed) lies below every dependent one — which is prediction P4 of §I, and the reason Assumption 1 is safe to state.

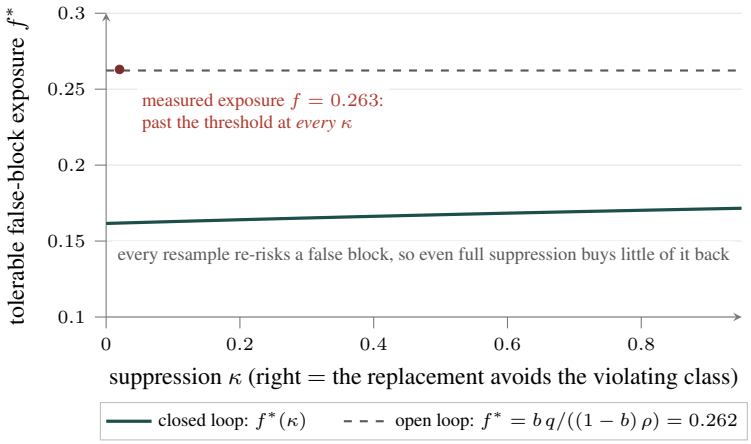

Figure 16. The open-loop to closed-loop correction under a resample model. Tolerable false-block exposure against the suppression parameter κ, at the audit's measured operating point. The loop multiplies the false-block arm by $1/(1-r)$ — every retry re-risks a spurious block — so the closed-loop threshold sits $1.6\times$ below the open-loop one, and suppression of violating replacements recovers almost none of it. The measured exposure (marked) is past the closed-loop threshold at every κ: the audit's sign verdict is not an artifact of holding the proposal distribution fixed. The closed forms are in §J.

I Registration Index

Every prediction was registered with its read-out rule and all of its branches before the runs it decides, and is scored supported, partial or failed. Branches within a registration are mutually exclusive and exhaustive, so exactly one fires and none is selected after the fact. The decision rule is $p < 0.05$ after Benjamini–Hochberg control at $q = 0.05$. The supplementary material carries one row per prediction, the commit that froze it, the timestamp of the result it decided, and the branch that fired.

P1, the transfer slope. Registered as the slope of measured soundness against induced ε, against the first-order prediction $\nu_1(1-b)/b$. Scored on the audit rows.

P2, the sign of the threshold. Registered as: the sign of NB agrees with $\varepsilon_{\text{eff}} \lessgtr \varepsilon^*$ on every row. *Supported on sign,* 9 *of* 9. The registration also named the branch in which individual rows fail to separate, and that branch fired for four rows; §6.2 reports the required sample size rather than implying the audit resolved them.

P3, the escalation count. Registered in advance to count episodes in which an audited monitor silently records a perceptually-unresolved atom as a clean verdict, *and to report the count if it were zero*, because zero would make the two-coordinate model correct and unnecessary. The count is 76 of 240 at the degraded level.

P4, dependence. Registered as: the independent form of Theorem 1 is conservative under common-shock dependence. Supported by the calibrated simulation.

P5, coverage under look-alikes. ***Failed.*** Registered as: look-alike widgets cost coverage, because two controls that render identically should make the operator decline. Coverage is 1.000 at native and at adversarial on all five classes. The registered failure branch was "they cost soundness instead", and that is what fired. We report it as failed rather than reinterpreted.

What we committed not to do with a negative result. The registration states that if a monitor's net benefit stayed positive at every rendering level, the paper would report the threshold as untested at deployment scale rather than reframing the prediction. That branch did not fire, and we record it here so that a reader can see the prediction was falsifiable as registered.

J Protocol and Secondary Material

Rendering levels. Native ($\varepsilon = 0.004$, $\gamma = 0.999$), degraded ($\varepsilon = 0.143$, $\gamma = 0.918$) and look-alike widgets ($\varepsilon = 0.084$, $\gamma = 0.992$). Error and coverage are measured separately, on the same reads, because Theorem 1 places them in different guarantees.

The false-block arm. A missed violation is visible in a trace; a wrongly blocked compliant action is not. False-block rates are therefore measured on constructed compliant episodes, with the construction held identical across all three architectures so that the comparison is between monitors rather than between constructions.

Detection latency. Reported as a distribution rather than a mean, because the mean of a latency distribution with a long tail is not the quantity a deployment cares about; T_{eff} in Table 1 is the median.

The episode-level identity. Writing $\hat{b}$ for the episode-level violation rate, ca for the share of violations caught and es for the escalation rate, $\mathrm{NB}_{\text{ep}} = \hat{b}\,\mathrm{ca}\,c_{\text{miss}} - \mathrm{fb}\,c_{\text{fb}} - \mathrm{es}\,c_{\text{esc}}$ reproduces every recorded net to 3×10^{-17} on all nine rows. It is an identity and not a model, which is why §6.2 can score each row against its own policy's threshold without a second experiment.

The closed-loop model. A blocked proposal is resampled, the replacement violating with probability $b(1-\kappa)$, so κ interpolates between i.i.d. retry ($\kappa = 0$) and a safe fallback ($\kappa = 1$). The retry chain is exactly solvable and gives $f^*_{\text{closed}} = b\,q/(b+\rho)$ against the open-loop $f^*_{\text{open}} = b\,q/((1-b)\rho)$. At the audit's operating point these are 0.162 and 0.262, and the measured exposure 0.263 stays past the closed-loop threshold for every $\kappa \in [0, 0.95]$.