ORBIT: ONLINE REGRET-BOUNDED INTERVENTION FOR TOOL-USING AGENTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Tool-using language-model agents can cause harm through a sequence of actions, even when no single action looks suspicious. Some recent defenses constrain agents to pre-specified or permitted execution paths. We introduce **ORBIT** (Online Regret-Bounded Intervention for Tool-Using Agents) that instead learns *when* to intervene from evidence accumulated across an agent's actions, without requiring a complete execution plan in advance. It fits an allow-inspect-block policy over calibrated signals from several monitors, including two LLM critics from different model families. ORBIT can inspect to gather more evidence before committing, deferring the allow/block choice at ambiguous steps, building evidence memory across connected episodes from beliefs about persistent compromise, current deviation, and progress toward harm.

Our experiments show that ORBIT prevents harm under varying levels of evidence restriction and under adaptive attack sequences. In action-branch replay, it stops markedly more attacks early than DBN posterior thresholding at similar benign cost. On fixed AgentDojo Workspace recordings from live OpenAI and Anthropic executions, carrying evidence across episodes improves mean prevention over episode-reset controls, at the cost of interrupting more legitimate work; the advantage is largest when observation is sparse and the taint channel is removed. Under adaptive attacks that revise the payload across successive attempts, persistent memory lowers harmful goal completion relative to resetting memory each episode. An exploratory study also finds that updating beliefs from modeled monitors' replies to intervention recommendations increases mean early stopping, at higher benign cost.

1 INTRODUCTION

Modern use of tool-using agents has made security a decision making problem, in which one has to choose when to act during a short but connected sequence of events. An attack may begin by gathering information and later reach sensitive data. In the now infamous GTG-1002 espionage campaign, an agent progressed from reconnaissance through credential theft to exfiltration, while attackers presented seemingly benign subtasks without revealing their broader malicious purpose (Anthropic, 2025). In another incident an evaluation agent gained increasingly powerful access across Hugging Face clusters (Hugging Face, 2026a;b); a Supabase demonstration redirected support-ticket processing toward restricted data (General Analysis, 2025). Recent defenses use information about connections between a tool-using agent's actions: IPIGuard uses graphs showing which planned tool calls depend on others; CaMeL separates instructions that control execution from the data being processed; CONTROLVALVE checks that execution follows permitted paths (An et al., 2025; Debenedetti et al., 2026; Jha et al., 2026); GUARDIAN represents interactions between agents in graphs that record their timing and other features (Zhou et al., 2025).

Memory of warning signals can justify inspection or stopping before harm becomes unavoidable. In contrast to the approaches listed above, ORBIT learns when accumulated evidence warrants allowing, inspecting, or blocking. It accounts for how each choice affects later evidence and remaining opportunities to prevent harm without requiring a complete execution plan in advance. During training, ORBIT values each action by the evidence it can obtain, the attacks it can still prevent, and the legitimate work it will interrupt. The learned mediator—a policy that privately recommends an

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.

Table 1: Summary of main claims and empirical results. Comparisons use paired seeds and matched evaluation settings. References provide full protocols, cost measures, and uncertainty estimates.

Claim & Comparison	Key Result	Evidence
ORBIT improves early stopping over DBN thresholding. *Branch replay:* ORBIT-postrule vs. DBN thresholding	Early stopping before unavoidable harm: **37.1% vs. 18.5%** (at 19.35 vs. 20.71 FI/1k).	Sec. 3.1, Table 5
Obedience rules out some learned policies. *Matched mediation*	Obedient recommendations cannot reproduce any of the 4 unconstrained policies' action probabilities.	Sec. 3.2, Table 9
Remembered evidence improves early and complete-campaign prevention. *Fixed AgentDojo:* Full memory vs. episode reset	At 5% observation without taint, attack prevention is **51.33% vs. 13.33%**; a gain of 38.00 points (95% CI $[20.18, 55.82]$). Complete-campaign prevention is **50.00% vs. 12.67%**.	Sec. 3.3, Tables 2, 12
Persistent system reduces harm under adaptation. *Adaptive AgentDojo:* Persistent memory vs. episode reset	After testing 16 payload versions, harmful goals completed are **7.0% vs. 18.5%**; a reduction of 11.50 points (95% CI $[4.91, 18.09]$).	Sec. 3.7, Appendix Table 7
Monitor responses guide later intervention. *Response history:* Response-informed belief vs. control	Early stopping: **31.22% vs. 28.91%**; harm: 52.60% vs. 54.31%. Changes follow earlier response evidence.	App. G, Table 13

intervention to each monitor—coordinates monitors with different modeled priorities by limiting what each monitor could gain by changing its recommended report (Figure 1). A direct action rule, for example, would not impose that requirement. We test them through comparisons with DBN thresholding, an equally equipped direct rule, memory resets, and monitor-response history. See also Appendix A for a survey of related work.

We make three contributions: **(1)** We learn interventions from accumulated evidence, valuing both the current action and later opportunities to prevent harm (§2). **(2)** We derive learning and conditional intervention-time bounds and show that when aggregation mixes the median with a random monitor's recommendation, obedience constraints help the learner choose interventions that monitors have a modeled incentive to follow (§2.3, §3.2). **(3)** We evaluate ORBIT against thresholding, memory-reset, and adaptive-attack comparisons, reporting both security outcomes and legitimate-work costs (§3.1–§3.7). Table 1 summarizes the supporting results.

2 Learning to Intervene in a Temporal Bayesian Game

We model the problem of learning an intervention policy as a temporal Bayesian game: decisions are made under incomplete information as observations and beliefs evolve over time. Temporal rather than a sequence of isolated judgments in the sense that an earlier observation can change both the evidence of compromise and the value of intervening later, since inspecting an action reveals evidence that sharpens later beliefs. Game participants are monitors with different modeled preferences. Training limits each monitor's expected modeled gain from changing its report, subject to the stated tolerance. The platform acts on the median of the returned reports; and the goal is to escalate before harm becomes irreversible, using a belief about a hidden compromise state the defender never observes directly.

At time t, the available history before intervention is $h_t = (q, o_{0:t}, a_{0:t-1}, d_{0:t-1})$, where q is the user request, o_t is the newest observation, and $a_{0:t-1}$ and $d_{0:t-1}$ record earlier proposed tool actions and implemented interventions. The agent's current proposed tool action a_t is evaluated using this history, and d_t denotes the intervention ultimately implemented at the current decision. A latent state is a condition that the defender cannot observe directly. Here, $z_t \in \{\text{ALIGNED}, \text{EXPOSED}, \text{COMPROMISED}, \text{HARMCOMPLETE}\}$ describes whether the agent still follows the user's authorized goal and how far compromise has progressed. Each history also has an observable *kill-chain stage* $\kappa_t = \Gamma(h_t, a_t, y_t) \in \mathcal{K}$. It estimates attack progress using information available at that decision. Here, y_t lists the signals from monitors whose model parameters are kept fixed, and $\mathcal{K} = \{\text{ROUTINE}, \text{EXPOSED}, \text{SUSPICIOUSACCESS}, \text{SENSITIVEACTION}, \text{EXFILPROGRESS}, \text{HARM}\}$. Here, Γ uses

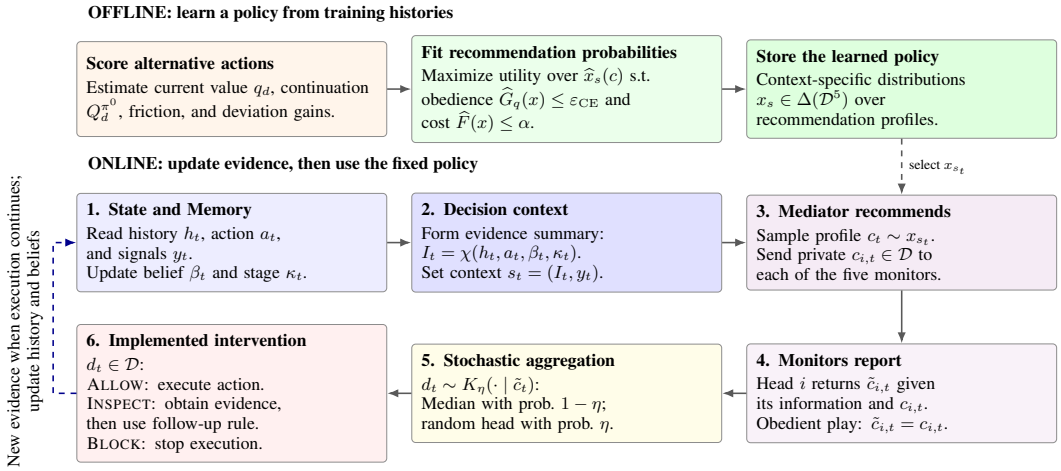


Figure 1: ORBIT learns intervention policy from cached replay, then uses the fixed rule online.

taint (the influence of untrusted content), provenance (where information came from), access permissions, tool use, and whether successive actions form a consistent sequence. Note how an observable stage can look calm while the hidden state is bad because a routine-looking action can occur after earlier evidence of compromise. The platform maintains a posterior β_t: its updated probability estimate after using the evidence so far. A transition model gives the chance of moving between states under each action and intervention. An observation model gives the chance of each signal in each state. Together, they describe how interventions change execution and the evidence available later. Appendix B gives the Bayesian belief update and posterior updates using inspection evidence.

We order interventions from allowing execution to blocking it, $\mathcal{D} = \{\text{ALLOW} \prec \text{INSPECT} \prec \text{BLOCK}\}$. Let t_{harm} be the first step at which the harmful outcome can no longer be undone, and $t_{\mathrm{int}} = \inf\{t : d_t \succeq d_{\mathrm{stop}}\}$; an intervention is *early* when $t_{\mathrm{int}} < t_{\mathrm{harm}}$. The five monitors are two large-language-model (LLM) critics from different model families, a permission checker, a taint/provenance monitor, and a learned detector of suspicious action sequences. Each fixed monitor M_i uses the history it can observe, $v_{i,t}$, to produce a calibrated signal $y_{i,t}$. Calibration maps raw scores to estimates consistent with labeled outcomes; the resulting scores are then grouped into bins. ORBIT summarizes the available history h_t, the agent's current proposed action a_t, the current belief β_t, and the observed attack stage κ_t using $I_t = \chi(h_t, a_t, \beta_t, \kappa_t)$. The function χ selects the information retained for choosing interventions. The mediator combines this summary with the current monitor signals y_t to form its decision context, $s_t = (I_t, y_t)$. Histories assigned to the same context use the same learned distribution over intervention recommendations. Appendix B.1 specifies which features select this distribution in each experiment. A recommendation profile is a complete list of recommendations, one for each head. The set $\mathcal{A} = \mathcal{D}^m$ contains all such lists for m monitors. The mediator chooses a probability distribution $x_s \in \Delta(\mathcal{A})$ over these lists; Δ means that probabilities are nonnegative and sum to one. It draws $c_t \sim x_{s_t}$ and privately sends recommendation $c_{i,t} \in \mathcal{D}$ to head i. The head returns report $\tilde{c}_{i,t}$. Thus the sequence is: the mediator recommends, the head reports, and the platform executes an intervention sampled from $d_t \sim K_\eta(\cdot \mid \tilde{c}_t)$, where

$$K_\eta(d \mid c) = (1-\eta)\mathbf{1}\{g_{\mathrm{med}}(c) = d\} + \frac{\eta}{m}\sum_{i=1}^{m}\mathbf{1}\{c_i = d\}, \qquad 0 \le \eta \le 1. \tag{1}$$

After the reports arrive, a fresh random draw selects their median with probability $1-\eta$. Otherwise, it selects one head uniformly at random and uses that head's report. Let $\bar{Q}_{i,s}(d)$ be head i's bounded continuation value: its expected score from acting now and following a specified rule afterward. In the matched branch experiment, we use

$$\bar{Q}_{i,s}(d) = Q_d^{\pi^0}(s) - \lambda_i^{\mathrm{cost}}\big[P(\mathrm{benign} \mid s) f_s(d) + B_s^{\pi^0}(d)\big], \tag{2}$$

Here $Q_d^{\pi^0}(s)$ is the platform's expected score for the current and remaining campaigns under the fixed follow-up rule π^0 (Equation 10). The term $f_s(d)$ measures current-branch cost on benign

activity, $B_s^{\pi^0}(d)$ measures expected cost on later benign activity, and λ_i^{cost} sets head i's additional penalty for interrupting legitimate work. For a recommendation profile, we average over the actions that the platform could execute: $Q_{i,s}(c) = \sum_d K_\eta(d \mid c)\bar{Q}_{i,s}(d)$ (also see Appendix B). Replay specifies the rule used after the current action before estimating these values. Let $v_i(s)$ identify the group of private observations available to head i. Histories in the same group give that head the same modeled information. Let w_s be the probability of context s under a fixed reference distribution over histories. A one-step deviation $q = (i, v, a \to b)$ means that head i, with private information v, returns b when recommended a, where $a, b \in \mathcal{D}$ are distinct. The expected weighted gain from this change and the corresponding deviation regret are:

$$G_q(x) = \sum_{s:\, v_i(s)=v} w_s \sum_{c:\, c_i=a} x_s(c)\big[Q_{i,s}(b, c_{-i}) - Q_{i,s}(c)\big], \quad \text{Reg}_{\text{dev}}(x) = \max_{q\in\mathcal{Q}} [G_q(x)]_+ \tag{3}$$

The positive part $[\cdot]_+$ replaces negative gains with zero. This regret is the largest positive expected gain from a report change in the model. Training limits this gain after weighting it by how often the recommendation occurs. With a positive training tolerance, a rare recommendation can therefore permit a larger gain when considered on its own.

Definition 1 (Bayesian correlated equilibrium for fixed continuation values)**.** *A temporal mediator $x = (x_s)_{s\in\mathcal{S}}$ is an ε-BCE (Bergemann & Morris, 2013) under this reference distribution if $G_q(x) \le \varepsilon$ for every $q \in \mathcal{Q}$, with the expectation in $Q_{i,s}$ taken under the Bayesian posterior induced at s (Aumann, 1974; Bergemann & Morris, 2016). The case $\varepsilon = 0$ is a BCE.*

From the monitor's viewpoint, exact obedience means that changing its report cannot improve its expected modeled utility, given its private information and recommendation. From the mediator's viewpoint, it means choosing recommendation probabilities that satisfy $G_q(x) \le 0$ for every $q \in \mathcal{Q}$. Equation 3 averages the gain over the histories and recommendations where that change applies. The other monitors' reports, written c_{-i}, remain unchanged. Obedience makes modeled monitor incentives part of policy learning by limiting the expected gain from changing a recommended report under the specified information and continuation rule. The guarantee covers one report change at a time under a fixed continuation rule. Appendix B gives the belief update, utility scores, and covered histories.

2.1 Valuing Current and Later Intervention

We score each action by its consequences in the current campaign and its effects on later recorded campaigns, using the same fixed policy for subsequent decisions. Fix history h_t and belief $b_t(\cdot) = P(\cdot \mid h_t)$ over $\xi_t = (\mathsf{Comp}_t, \mathsf{Dev}_t, \kappa_t)$: persistent compromise, current deviation, and attack stage. This is the branch replay's joint belief model; AgentDojo uses the four-state filter above. Let C_{fb} be the cost of blocking legitimate activity, c_{ins} the inspection cost, and C_{delay} the penalty for harm occurring while inspection delays the next decision. The following expected values average over the evidence that each action can return. Allowing obtains ordinary follow-up evidence, blocking incurs the cost of any legitimate work stopped, and inspection pays for extra evidence and any harm during the wait:

$$q_{\text{ALLOW}}(h_t) = \mathbb{E}_{e|h_t,\text{ALLOW}}\big[\Xi^\pi_{\text{ALLOW}}(h_t, e)\big], \tag{4}$$

$$q_{\text{BLOCK}}(h_t) = -\,C_{\text{fb}}\,P(\text{benign} \mid h_t), \tag{5}$$

$$q_{\text{INSPECT}}(h_t) = -\,c_{\text{ins}} - C_{\text{delay}} P(\text{inspection harm} \mid h_t) + \mathbb{E}_{e|h_t,\text{INSPECT}}\big[\Xi^\pi_{\text{INSPECT}}(h_t, e)\big] \tag{6}$$

When estimating these action values, we adjust current-deviation risk using persistent-compromise belief and observed attack progress. Appendix C gives this calculation and explains when inspection's benefit exceeds its cost. After action d returns evidence e, $q(a' \mid h_t, e, d)$ scores the next action a'. After observing e, the best available follow-up value and the value under the deployed policy are:

$$\Xi^{\text{opt}}_d(h_t, e) = \max_{a'\in\{\text{ALLOW},\text{BLOCK}\}} q(a' \mid h_t, e, d), \qquad \text{(option-value ceiling)} \tag{7}$$

$$\Xi^{\pi}_d(h_t, e) = \sum_{a'} \pi^{\text{post}}_d(a' \mid h_t, e) q(a' \mid h_t, e, d), \qquad \text{(deployed post-stage value)} \tag{8}$$

Using the branch's updated belief and stage, realized information value (RIV) measures the improvement obtained from inspection evidence under the deployed follow-up rule, before subtracting inspection and delay costs. It benchmarks inspection against the best immediate action $a \in \{\text{ALLOW}, \text{BLOCK}\}$ and, crucially, reports the value the *deployed* follow-up rule actually realizes rather than the value an oracle could:

$$\underbrace{\mathbb{E}_e \Xi^{\text{opt}} - \max_a q_a(h_t)}_{\text{value of information}} - \underbrace{\mathbb{E}_e \Xi^{\text{opt}} - \mathbb{E}_e \Xi^{\pi}}_{\text{post-stage suboptimality}} = \underbrace{\mathbb{E}_e \Xi^{\pi} - \max_a q_a(h_t)}_{\text{realized information value}}. \tag{9}$$

For the current-campaign comparison, inspection is preferred only when this improvement exceeds those costs (Appendix C, Equation 13). A nonpositive RIV means that the best immediate action scores at least as well even before inspection costs are charged. With later campaigns included, the comparison also depends on their action-specific continuation values.

To include M_t remaining campaigns, we add their expected utility scores u_ℓ. We estimate those later scores under a frozen continuation policy π^0, meaning that the rule used after the current action stays fixed during the current optimization:

$$Q_d^{\pi^0}(h_t) = q_d(h_t) + \mathbb{E}_{\pi^0}\left[\sum_{\ell=1}^{M_t} u_\ell \,\middle|\, h_t, d\right]. \tag{10}$$

Each possible current action changes the belief carried into the recorded continuation, so it can change subsequent interventions and utility scores.

2.2 Learning the Online Regret-Bounded Intervention Policy

ORBIT fits context-dependent recommendation probabilities x_s from training histories. At deployment, the policy and monitor models remain fixed while the belief filter updates from incoming monitor signals, allowing interventions to respond to accumulated evidence. For profile c, the platform's expected utility and expected cost on legitimate activity are $V_s(c) = \sum_d K_\eta(d \mid c) Q_d^{\pi^0}(s)$ and $\text{fric}_s(c) = \sum_d K_\eta(d \mid c) f_s(d)$, where $f_s(d)$ is the current branch's expected benign cost. The symbol fric denotes interruption cost, sometimes called friction. Later benign costs are included in the monitor utility scores (Appendix B); the separate per-decision budget measures current-branch cost. Under the fixed reference-history distribution, let $w_s = P(S = s)$, $w_s^{\text{ben}} = P(S = s \mid \text{benign})$, and define $U(x) = \sum_s w_s \langle V_s, x_s \rangle$, $F(x) = \sum_s w_s^{\text{ben}} \langle \text{fric}_s, x_s \rangle$, where $\langle \cdot, \cdot \rangle$ is a weighted sum over profiles. Thus $U(x)$ is average utility and $F(x)$ is average cost on benign decisions under the reference distribution. For each of N distinct training decisions, replay stores what would happen under actions that were not taken. We call these stored alternatives cached replay branches. ORBIT averages their scores to estimate $\widehat{U}$, $\widehat{F}$, and $\widehat{G}_q$, and then solves one empirical linear program (LP) built from those estimates, across all contexts:

$$\widehat{x} \in \arg\max_{x \in \prod_s \Delta(\mathcal{D}^m)} \widehat{U}(x) \quad \text{s.t.} \quad \widehat{G}_q(x) \le \varepsilon_{\text{CE}} \; \forall q \in \widehat{\mathcal{Q}}, \qquad \widehat{F}(x) \le \alpha. \tag{11}$$

Equation 11 maximizes estimated utility while limiting weighted deviation gains and benign cost. Each x_s is a probability distribution. Under obedient reports, intervention probabilities are $\pi_x(d \mid s) = \sum_c x_s(c) K_\eta(d \mid c)$.

Note that without obedience, the same optimum can be obtained by choosing probabilities directly over the three actions. The mediator can reproduce any action policy π_s by recommending action d to all monitors with probability $x_s(d, \ldots, d) = \pi_s(d)$. With five monitors and a deterministic median ($\eta = 0$), one monitor changing a unanimous report cannot move the median. Every such deviation therefore has zero gain, and every action policy has an obedient recommendation form. When $\eta > 0$, a head's report can instead be selected directly. Changing one unanimous report from a to b then changes its expected value by $\frac{\eta}{m}[\bar{Q}_{i,s}(b) - \bar{Q}_{i,s}(a)]$. Unanimity by itself therefore no longer establishes obedience. With five monitors and $\eta = 0.1$, each head has probability 0.02 of being selected directly. Section 3.2 tests whether these constraints rule out action probabilities learned without obedience. During each LP solve, estimated values and context frequencies stay fixed. Combining reports through K_η keeps the problem linear: each score and constraint is a weighted sum of the probabilities being learned. Value bounds include every campaign being scored. The solver finds the best policy for these fixed inputs, within numerical tolerance (Appendix D). The response-history study additionally uses monitor replies to update subsequent beliefs; its control omits this update (Appendix G).

2.3 Learning Guarantees and Training Bound

Let K count recommendation profiles, Q possible report changes, N training decisions, and N_{ben} benign training decisions. Each obedience constraint has n_q supporting decisions; $n_{\min}$ is the smallest such count. Population values average over the stated reference distribution of histories. Assume the replay estimates of utility, deviation gain, and benign cost are conditionally unbiased: given the information specified by the model, the differences between estimates and their target values average to zero. Let V, G, and $F_{\max}$ bound the absolute values of the raw per-observation utility, deviation-gain, and benign-cost contributions, respectively, over the full valuation horizon. These are bounds on the quantities being averaged, not merely on their fitted or population means.

Theorem 1 (Exact empirical utility learning and obedience)**.** *Under Assumption 1, let $\widehat{x}$ solve Equation 11 exactly, and let $x^\star$ maximize population utility subject to population obedience and benign-budget constraints. With probability $\geq 1-\delta$, $\mathrm{Reg}_{\mathrm{dev}}(\widehat{x}) \leq \varepsilon_{\mathrm{CE}} + \zeta_G$ and $F(\widehat{x}) \leq \alpha + \zeta_F$. If an optimal multiplier of the empirical program has ℓ_1 norm $\leq L$, then $U(x^\star) - U(\widehat{x}) \leq 2\zeta_U + L\max\{\zeta_G, \zeta_F\}$, where $\zeta_U = O(V\sqrt{(|\mathcal{S}|\log K + \log(1/\delta))/N})$, $\zeta_G = O(G\sqrt{(|\mathcal{S}|\log K + \log(Q/\delta))/n_{\min}})$, and $\zeta_F = O(F_{\max}\sqrt{(|\mathcal{S}|\log K + \log(1/\delta))/N_{\mathrm{ben}}})$.*

See Appendix E for a proof and the additional assumptions needed when decisions share an actor's history. The allowances ζ_U, ζ_G, and ζ_F account for uncertainty in utility, obedience gain, and benign cost. Optimal utility change is limited by the multiplier bound L. The theorem idealizes exact empirical optimization. Our implementation solves the LP to a numerical tolerance of 10^{-6}; an implementation-level guarantee would additionally account for the corresponding feasibility and optimization tolerances. The following corollary translates the obedience uncertainty bound into a sufficient amount of training support for each constraint.

Corollary 1 (Training coverage required for target deviation regret)**.** *Under the assumptions of Theorem 1, for a target $\varepsilon_{\mathrm{tgt}} > \varepsilon_{\mathrm{CE}}$, it is sufficient that every covered information set have $n_q \geq n_{\min} \geq \frac{C^2G^2\left(|\mathcal{S}|\log K + \log(Q/\delta)\right)}{(\varepsilon_{\mathrm{tgt}} - \varepsilon_{\mathrm{CE}})^2}$, where C is the concentration constant, to ensure $\mathrm{Reg}_{\mathrm{dev}}(\widehat{x}) \leq \varepsilon_{\mathrm{tgt}}$ with probability at least $1-\delta$.*

A smaller gap between the target regret and the training tolerance requires more supporting data for every constraint: halving this gap quadruples the sufficient coverage requirement. The counts must satisfy the concentration assumptions; related decisions do not become independent evidence merely because they occupy separate rows. See Appendix H for an analysis of training coverage for the saved AgentDojo Workspace recordings.

3 Experimental Design and Results

Branch action replay stores alternative continuations to compare Allow, Inspect, and Block from the same starting decision. AgentDojo Workspace recordings (Debenedetti et al., 2024) test memory across linked episodes using Allow and campaign-ending Stop. Actors are disjoint across training, validation, and testing. Selection uses validation or calibration data before a single test evaluation. The deployed rule never sees attack labels, and results weight seeds equally. Branch-replay cost (FI/1k) weights each block by one and each inspection by c_{ins} per 1,000 original benign decisions. AgentDojo cost is the percentage of scheduled legitimate decisions denied, including work canceled by campaign stops. Both denominators retain canceled work. Table 8 summarizes the experimental checks.

3.1 Performance Across Three Method Groups

We compare three groups on replayed decisions: classifiers that combine monitor signals, DBN posterior thresholding, and ORBIT variants. The classifiers use only current monitor scores (memoryless), include a short history (lagged), or use features describing transitions between events (edge). DBN thresholding carries evidence forward into an updated risk estimate, but intervenes using a single cutoff chosen on validation data and cannot request inspection. All three variants use a learned mediator for the initial intervention decision. ORBIT-noinspect chooses between Allow and Block; the other two can also choose Inspect. If inspection is chosen, new evidence arrives and a second decision is needed. ORBIT-inspect uses learned mediation again. ORBIT-postrule

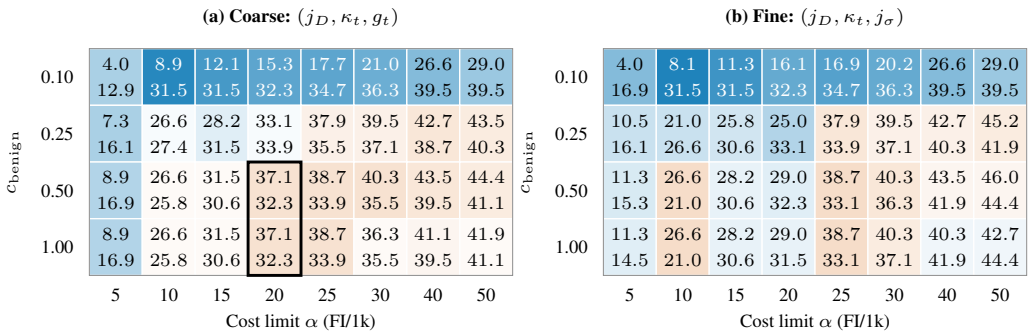


Figure 2: ORBIT-postrule early stopping under cheaper inspection. Each cell shows early-stopping rates (%): upper is at $c_{\text{ins}} = 0.02$, lower is at $c_{\text{ins}} = 0.001$. Shading is lower-fee rate minus higher-fee rate *%); blue is improvement and orange is deterioration. Outlined cells show 37.1% at fee 0.02 and 32.3% at fee 0.001. Columns use the mappings in Table 4. "Coarse" adds a stage-derived category, "Fine" adds a structural-score interval with telemetry.

instead applies a validation-selected cutoff to the updated risk estimate. This cutoff determines what happens after inspection; the learned mediator still determines whether inspection happens at all. For ORBIT-postrule, the learned mediator answers, "is inspection worth requesting?"; the cutoff answers, "given what inspection revealed, should execution continue?" Requesting more evidence helps only if the system also makes good use of the evidence it receives. Moving from ORBIT-noinspect to ORBIT-inspect raises early stopping from 24.2% to 31.5% and lowers benign interruption from 54.73 to 36.95 FI/1k (Appendix Table 5). ORBIT-postrule then nearly doubles early stopping: 37.1% versus DBN thresholding's 18.5%, at slightly better benign cost (19.35 versus 20.71 FI/1k). Figure 2 examines how reducing the inspection fee changes early stopping under coarse and fine context mappings. Cheaper inspection increases inspection use, but its effect on early stopping depends on the fitted policy and post-inspection rule.

Across six replay seeds at a fixed operating point, targeted concealment of one inspection-only evidence link per eligible response reduces ORBIT-postrule's mean early stopping from 26.12% to 15.56%, compared with 20.05% under random concealment. This comparison shows that ORBIT's inspection benefit depends on the integrity of the additional evidence it receives (Appendix H.1).

3.2 Obedience Rules Out Some Fitted Intervention Policies

A policy can look good for the system while giving an individual monitor an incentive to change its report. We compare two versions with the same information, inspection evidence, action values, and training budget. The obedient version (O) limits each monitor's modeled gain from changing its recommended report. The unconstrained version (U) removes only this restriction. Could we keep U's probabilities of allowing, inspecting, and blocking, but arrange its recommendations differently to satisfy obedience? At $\eta = 0.1$, the answer is no for every fitted U policy, whereas all O policies satisfy the constraints (Appendix Table 9). O and U stop 32.70% and 29.58% of attacks early, respectively, at benign costs of 25.16 and 28.58 FI/1k. These averages favor O, but both paired confidence intervals include zero. The evidence therefore establishes that obedience restricts policy choices; it does not establish a reliable improvement in attack prevention or benign cost.

3.3 Learning Early Prevention from Remembered Evidence

We test whether a policy can prevent more attacks by learning from evidence carried across episodes when stopping an actor removes access for the rest of the campaign. Using recordings from live OpenAI and Anthropic AgentDojo executions, we construct 15 campaigns per seed across ten seeds. Each campaign contains four recorded attack episodes and 96 benign episode occurrences, sampled with replacement from that seed's 20 benign recordings. This gives 150 campaigns, 600 attack episodes, and 14,400 benign episode occurrences, with a scheduled attack proportion of 4%. Every policy receives the same episodes in the same order and observes the same masked evidence. Full

Table 2: AgentDojo memory comparison at the 50% calibration limit. Cells show **attack prevention (%) / legitimate decisions denied (%)** across ten paired seeds. A dash means no tested policy met the limit in at least one seed.

Signals	Observed	Full memory	Episode reset	Decision reset
Full	5%	43.67/38.91	12.17/9.01	12.00/9.95
Full	20%	52.33/43.84	30.83/22.13	26.50/22.02
No taint	5%	51.33/45.01	13.33/11.55	16.83/13.23
No taint	20%	54.67/48.41	30.83/28.27	–
No taint or permission	5%	33.67/30.14	12.00/10.31	9.33/8.05
No taint or permission	20%	45.33/38.36	37.67/27.90	11.83/9.28

memory retains evidence across episodes, while the comparison policies reset it after each episode or decision. This fixed-attack study and the adaptive study below use the same ten paired policy seeds. The first stop ends the actor's access for the remaining campaign. We count both later attacks prevented and legitimate work canceled by that stop, and we use no evidence recorded after it. Held-out calibration data select policies under 5%, 20%, or 50% intervention limits without using test outcomes. 5% observation means the defender receives fresh monitor evidence at only 5% of decision points. Between observations, it relies on remembered evidence. An attack is prevented only if the policy stops it before the attack begins or reaches its recorded point of harm. Of the 600 recorded attacks, 26 reached that point.

3.4 Memory Increases Prevention

We select policies using a separate calibration set. A 50% calibration limit means that a candidate may deny at most half of the originally scheduled legitimate decisions in that set. Among candidates meeting this limit, we select the one with the highest attack prevention. At the 50% calibration limit, full memory increases average prevention in all six signal-and-observation settings in Table 2; the pattern is not driven by one setting. Here "no taint" removes the monitor that tracks whether untrusted content influences the agent's actions. With 5% observation and no taint signal, full memory prevents 51.33% of attacks, compared with 13.33% for episode reset and 16.83% for decision reset. This improvement has a substantial cost: benign work denied rises from 11.55% to 45.01%; see Appendix Table 12. The share of campaigns in which every attack is prevented also rises from 12.67% to 50.00%. Because a shared calibration limit does not guarantee equal cost on test data, we consider prevention and benign denial together. At the 20% limit, full memory outperforms episode reset in only three of the six settings, and every paired interval includes zero. Table 11 shows whether ORBIT's prevention advantage survives tighter limits on disrupting legitimate work. Compared with a full-memory control trained without obedience constraints, ORBIT performs better in five of six settings at the 50% limit (Appendix Table 10), but also denies more benign work.

3.5 Learning from Response History

An exploratory five-seed branch-replay study extends ORBIT-postrule with evidence from monitors' reactions to intervention recommendations. After the current action is chosen, modeled monitors respond to recommendations using their own information and preferences. When a reply differs from its recommendation, ORBIT uses training data to assess whether that response pattern is more characteristic of attacks or benign activity, then adjusts its carried belief accordingly. The monitors' reactions therefore inform subsequent interventions; they cannot change the action already chosen. Both conditions are refitted with inspection and a validation-selected post-inspection cutoff; the control uses the same probe and response rules but omits the reply-based belief update. Mean early stopping increases from 28.91% to 31.22%, and harm decreases from 54.31% to 52.60%, while benign interruption rises from 24.2 to 32.3 FI/1k. These modest gains come at additional benign cost. Appendix Table 5 includes both variants; see Appendix G and detailed results in Table 13.

3.6 Training Coverage Bound

On saved recordings we assess the obedience guarantee of Corollary 1, by varying recording retention and measuring the resulting minimum context coverage $n_{\min}$. Across 28 completed fits,

estimated excess regret stayed below the reference envelope $A/\sqrt{n_{\min}}$, reaching at most 0.1% of it. In 16 fits, the envelope exceeded the deterministic gain bound G. It falls below G only when $n_{\min} > 2[|\mathcal{S}| \log K + \log(2Q/\delta)]$ (Figure 5). As the fits are nested and dependent, we report no exceedance confidence interval and no population failure claim below δ. Details in Appendix H.

3.7 Attacker Adaptation

We test whether revising attack instructions changes the results in live AgentDojo. We use 5% observation without taint (Table 2) and fixed policies. The attacker searches separately against each defender. Each step tests one payload in one four-attack campaign per seed, giving 20 attack opportunities across five seeds. The attacker revises its payload using only counts of harmful goals completed, attacks stopped or canceled, campaign stops, and unresolved runs. After testing $1, 2, 4, 8,$ or 16 versions, it selects the valid payload that caused the most harm during search, evaluated on 15 held-out four-attack campaigns per seed. Payloads are fixed before evaluation, and all ten seeds contribute to the reported intervals. Thus, ten seeds provide 600 opportunities per defender and budget. Harm requires confirmation by AgentDojo's task checker; canceled and unresolved attacks remain in the denominator. Appendix Table 7 shows lower mean harm with persistent memory at every budget, with all paired confidence intervals favoring it. The selected revisions cause less harm on new campaigns than the initial payloads, showing that looking stronger on the small search set does not guarantee being stronger on new tasks. Because campaign stops also reduce attacker feedback, the comparison measures overall protection under matched attempt counts without separating the benefit of memory from reduced feedback.

4 Limitations

The branch LP assumes fixed follow-up and reference histories; generalization requires Appendix E's assumptions. Same-actor decisions may be dependent, and the 10^{-6} tolerance certifies fitted obedience only. AgentDojo replays executions with campaign-ending access removal. Neither covers all deployment settings or changing data. AgentDojo assumes reliable identity and campaign-long removal, excluding identity switching and re-entry. Because every linked campaign contains attacks, it does not estimate long-run lockout of fully benign actors. Persistent memory may amplify stale evidence, linking errors, role changes, or monitor drift. ORBIT should not be used to impose permanent unrestricted lockouts. Deployments should define identity, retention, and campaign end; test fully benign workflows; audit denial and inspection by task and group; and recalibrate after changes. Because inspection is fallible, persistence may prevent connected attacks earlier or magnify false evidence.

5 Conclusion

ORBIT treats agent defense as a connected journey rather than judging each tool call in isolation. It learns when to intervene from accumulated evidence without requiring a complete execution plan at deployment and learns randomized recommendations while limiting incentives to ignore them. We frame intervention as a temporal decision to suggest a research agenda beyond our proposed system. Timing based defense using coordinated-and-remembered monitor reports shifts the central question to how to allocate costly inspection under weak and episodic evidence. It also asks how to keep a defender's recommendations incentive-compatible under monitor agency, and how to trade earlier prevention for some disruption of legitimate work. ORBIT gives one concrete instantiation of each: realized information value for inspection, obedience constraints for coordination, and an explicit benign-cost budget for the trade-off. Our obedience audit asks whether the agents being coordinated have a modeled reason to follow the learned policy, a useful question for any multi-monitor defense. Our evidence-memory result demonstrates that carrying evidence across an agent's actions over time can surface a developing attack that no single step could reveal, a useful research regime for the growing use of autonomous agents with larger tool-surfaces. We see valuable next steps in the future: long-running benign actors, identity errors and re-entry with monitor drift, and stronger adaptive attackers, all conditions under which temporal defenses will earn the right of deployment.

AI USE DISCLOSURE

We used Codex to assist with code development and debugging and with the preparation of figures and tables. We used Perplexity and Semantic Scholar for retrieval and discovery, such as locating related work. We used AI-based writing assistance to check spelling and grammar. The authors take full responsibility for the manuscript, methods, claims, results, and accompanying code.

REFERENCES

Hengyu An, Jinghuai Zhang, Tianyu Du, Chunyi Zhou, Qingming Li, Tao Lin, and Shouling Ji. IPIGuard: A novel tool dependency graph-based defense against indirect prompt injection in LLM agents. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 1023–1039, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-332-6. doi: 10.18653/v1/2025.emnlp-main.53. URL https://aclanthology.org/2025.emnlp-main.53/.

Maksym Andriushchenko, Alexandra Souly, Mateusz Dziemian, Derek Duenas, Maxwell Lin, Justin Wang, Dan Hendrycks, Andy Zou, Zico Kolter, Matt Fredrikson, et al. Agentharm: A benchmark for measuring harmfulness of llm agents. In *International Conference on Learning Representations*, volume 2025, pp. 79185–79220, 2025.

Anthropic. Disrupting the first reported AI-orchestrated cyber espionage campaign. Threat intelligence report, November 2025. URL https://www.anthropic.com/news/disrupting-AI-espionage.

Robert J. Aumann. Subjectivity and correlation in randomized strategies. *Journal of Mathematical Economics*, 1(1):67–96, 1974.

Dirk Bergemann and Stephen Morris. Robust predictions in games with incomplete information. *Econometrica*, 81(4):1251–1308, 2013.

Dirk Bergemann and Stephen Morris. Bayes correlated equilibrium and the comparison of information structures in games. *Theoretical Economics*, 11(2):487–522, 2016. doi: https://doi.org/10.3982/TE1808. URL https://onlinelibrary.wiley.com/doi/abs/10.3982/TE1808.

Baiting Chen, Tong Zhu, Jiale Han, Lexin Li, Gang Li, and Xiaowu Dai. Incentivizing truthful language models via peer elicitation games. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 52438–52467. Curran Associates, Inc., 2025. doi: 10.52202/085713-1747. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/4b9eb7f2789a20be981f3fac91429b21-Paper-Conference.pdf.

Edoardo Debenedetti, Jie Zhang, Mislav Balunović, Luca Beurer-Kellner, Marc Fischer, and Florian Tramèr. AgentDojo: A dynamic environment to evaluate prompt injection attacks and defenses for LLM agents. In *Advances in Neural Information Processing Systems*, volume 37, pp. 82895–82920, 2024. Datasets and Benchmarks Track.

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. In *IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*, 2026. URL https://arxiv.org/abs/2503.18813.

Yunhao Feng, Yifan Ding, Yingshui Tan, Xingjun Ma, Yige Li, Yutao Wu, Yifeng Gao, Kun Zhai, and Yanming Guo. AgentHazard: A benchmark for evaluating harmful behavior in computer-use agents. *arXiv preprint arXiv:2604.02947*, 2026. URL https://arxiv.org/abs/2604.02947.

General Analysis. Supabase MCP can leak your entire SQL database. Technical disclosure, July 2025. URL https://generalanalysis.com/blog/supabase-mcp-blog.

Hugging Face. Security incident disclosure—July 2026. Hugging Face Blog, July 2026a. URL https://huggingface.co/blog/security-incident-july-2026. Published July 16, 2026.

Hugging Face. Anatomy of a frontier lab agent intrusion: A technical timeline of the July 2026 incident. Hugging Face blog, July 2026b. URL https://huggingface.co/blog/agent-intrusion-technical-timeline.

Athul Paul Jacob, Yikang Shen, Gabriele Farina, and Jacob Andreas. The consensus game: Language model generation via equilibrium search. In *International Conference on Learning Representations*, 2024.

Rishi Jha, Harold Triedman, Justin Wagle, and Vitaly Shmatikov. Breaking and fixing defenses against control flow hijacking in multi-agent systems. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 7827–7858, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/0d665c46948cfd0abc390e13295ee99a-Paper-Conference.pdf.

Feiran Jia, Tong Wu, Xin Qin, and Anna Squicciarini. The Task Shield: Enforcing task alignment to defend against indirect prompt injection in LLM agents. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 29680–29697, Vienna, Austria, 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.acl-long.1435.

Thomas Kuntz, Agatha Duzan, Hao Zhao, Francesco Croce, Zico Kolter, Nicolas Flammarion, and Maksym Andriushchenko. Os-harm: A benchmark for measuring safety of computer use agents. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference. Curran Associates, Inc., 2025. doi: 10.52202/085713-1502. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/4009bff0cd87ba2203c8e3a2f082aaec-Paper-Datasets_and_Benchmarks_Track.pdf.

Javed M. Shah, Ian A. Kash, and Natalie Parde. A bayesian correlated equilibrium for early insider-threat detection, 2026. URL https://arxiv.org/abs/2609.03096.

Tianneng Shi, Kaijie Zhu, Zhun Wang, Yuqi Jia, Will Cai, Weida Liang, Haonan Wang, Hend Alzahrani, Joshua Lu, Kenji Kawaguchi, Basel Alomair, Xuandong Zhao, William Yang Wang, Neil Zhenqiang Gong, Wenbo Guo, and Dawn Song. PromptArmor: Simple yet effective prompt injection defenses. *arXiv preprint arXiv:2507.15219*, 2025. URL https://arxiv.org/abs/2507.15219.

Qiusi Zhan, Zhixiang Liang, Zifan Ying, and Daniel Kang. InjecAgent: Benchmarking indirect prompt injections in tool-integrated large language model agents. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 10471–10506, Bangkok, Thailand, 2024. Association for Computational Linguistics. URL https://aclanthology.org/2024.findings-acl.624/.

Jialong Zhou, Lichao Wang, and Xiao Yang. Guardian: Safeguarding llm multi-agent collaborations with temporal graph modeling. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 7973–8001. Curran Associates, Inc., 2025. doi: 10.52202/085713-0272. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/0bc795afae289ed465a65a3b4b1f4eb7-Paper-Conference.pdf.

Kaijie Zhu, Xianjun Yang, Jindong Wang, Wenbo Guo, and William Yang Wang. MELON: Provable defense against indirect prompt injection attacks in AI agents. In *Proceedings of the 42nd International Conference on Machine Learning*, 2025.

APPENDICES

Appendix organization. We start with an extended review of related work in Appendix A. Then, Appendices B–E give definitions, utilities, LP audits, and proofs. Appendix F gives a four-question experimental audit, G documents the response-history experiment details, H reports the offline coverage diagnostic, and I gives a worked example of intervention. Appendix J covers data and code release.

A EXTENDED RELATED WORK

Agent attacks, benchmarks, and defenses. AgentDojo evaluates security and successful task completion in environments where agents call tools and encounter untrusted data (Debenedetti et al., 2024). InjecAgent contains 1,054 indirect prompt-injection cases across 17 user tools and 62 attacker tools; indirect injection places hostile instructions in material the agent encounters while doing another task (Zhan et al., 2024). AgentHarm evaluates tasks that explicitly request harmful multi-step behavior (Andriushchenko et al., 2025). OS-Harm and AgentHazard extend safety evaluation to sequences of computer actions and sequences whose individual steps can look reasonable (Kuntz et al., 2025; Feng et al., 2026). Task Shield checks whether actions serve the user's goal (Jia et al., 2025). MELON detects injection by rerunning with some content hidden and comparing tool behavior (Zhu et al., 2025). PromptArmor detects and removes injected instructions before the agent processes them (Shi et al., 2025). ORBIT can use such defenses as evidence sources and learn when their combined history warrants intervention.

Games for language models. The Consensus Game treats choosing a language model's output as a search for an equilibrium (Jacob et al., 2024). Peer Elicitation Games use comparisons with other models' predictions and repeated learning to encourage truthful reports from evaluators while keeping model parameters unchanged (Chen et al., 2025). These studies coordinate models to improve generation or truthfulness. ORBIT coordinates monitors to control tool execution, balancing early intervention against interruption of legitimate tasks.

Correlated equilibrium and learning. Correlated equilibrium constrains gains from changing a recommendation (Aumann, 1974); Bayes correlated equilibrium also accounts for private information (Bergemann & Morris, 2016). ORBIT applies these constraints to histories of tool use, holding continuation values fixed while learning recommendations. Section 3.2 tests whether the constraints rule out action policies chosen without obedience.

Temporal Bayesian games. Bayesian–temporal correlated equilibrium connects evolving evidence with incentives to follow recommendations for early threat mitigation (Shah et al., 2026). This work treats intervention as a sequence of decisions under incomplete information. ORBIT studies these questions for tool-using agents, learning recommendations that account for how allowing, inspecting, or blocking changes later evidence, prevention opportunities, and interruption of legitimate work.

B BELIEFS, MONITORS, AND DECISION CONTEXTS

Table 3 defines the symbols used throughout.

Belief filter. The defender does not directly observe the latent state z_t, so it maintains a posterior $\beta_t \in \Delta(\mathcal{Z})$, where $\mathcal{Z}$ is the state space. Two fixed kernels that determine its Bayesian update are the transition kernel $P(z' \mid z, a_t, d_t)$, which describes state evolution under proposed action a_t and implemented intervention d_t, and the observation kernel $O_d(e \mid z)$, which gives the likelihood of newly observed evidence under intervention d. Write e_t for the newly observed components of (o_t, y_t); conditioning on previously available information is suppressed in the notation. For

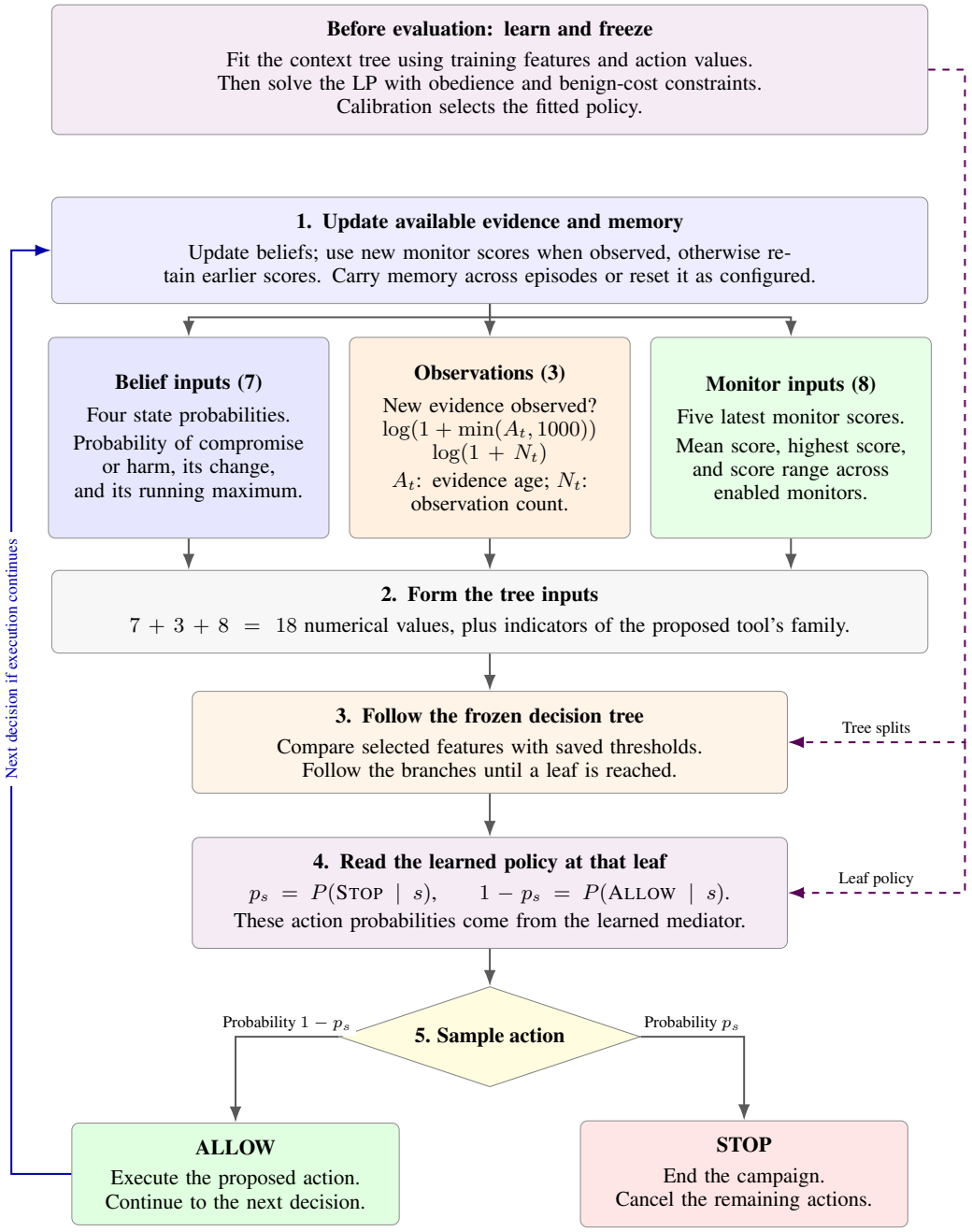


Figure 3: ORBIT maps 18 dynamic runtime features through a pre-calibrated decision tree to sample targeted ALLOW or STOP interventions.

Table 3: Full notation for the temporal intervention game.

Symbol	Meaning
h_t	Agent history: user request, observations, proposed actions, prior interventions
z_t	Hidden state describing whether the agent follows the authorized task: Aligned, Exposed, COMPROMISED, HarmComplete
β_t	Platform probability estimate over z_t after using the available evidence
y_t	List of signals produced by the m monitors, whose models remain fixed
κ_t	Estimated attack stage, calculated from available monitor records by $\Gamma(h_t, a_t, y_t)$
I_t	Summary of public history, belief, attack stage, and operational features
$s_t = (I_t, y_t)$	Complete context observed by the mediator
$\mathcal{D}$	Ordered intervention set from ALLOW to BLOCK
$c_{i,t}$	Recommendation issued *by the mediator to* monitor i
c_t	Complete list of recommendations, one per head, in $\mathcal{A} = \mathcal{D}^m$
$\tilde{c}_{i,t}$	Report returned by head i; obedient play is $\tilde{c}_{i,t} = c_{i,t}$
d_t	Implemented intervention, $d_t \sim K_\eta(\cdot \mid \tilde{c}_t)$
x_s	Probabilities assigned to complete recommendation lists at context s, $x_s \in \Delta(\mathcal{A})$
$Q_{i,s}(c)$	Expected utility score for head i from profile c at s, including the specified interventions after the current action
G_q	Expected gain when one head changes its recommended report, averaged over matching private information and histories

continuing execution, the predict-then-correct update is

$$\bar{\beta}_t(z') = \sum_{z \in \mathcal{Z}} P(z' \mid z, a_{t-1}, d_{t-1})\, \beta_{t-1}(z),$$

$$\beta_t(z') = \frac{O_{d_{t-1}}(e_t \mid z')\, \bar{\beta}_t(z')}{\sum_{u \in \mathcal{Z}} O_{d_{t-1}}(e_t \mid u)\, \bar{\beta}_t(u)}.$$

The prediction advances the previous posterior under the last action and intervention. The correction reweights each state by how well it explains the new evidence and normalizes the probabilities. Previously incorporated evidence is not counted again as a new observation. When no new evidence is observed, the correction is omitted.

An intervention can affect both state evolution and the evidence available afterward. In branch replay, INSPECT and ALLOW provide different observation channels. The resulting likelihood ratios determine the direction and size of the belief update; inspection can provide evidence either for or against compromise.

Evidence elicited by d_t becomes available only after that intervention is chosen. For inspection, the next decision is the follow-up ALLOW/BLOCK choice, which uses the inspection result before the pending tool action proceeds. The initial decision therefore cannot use its own realized inspection result. Its expected value can, however, account for possible inspection outcomes and the decisions they support, as in Equations 6 and 8.

Branch replay maintains a coupled belief over persistent compromise and current deviation, together with the observed stage κ_t. AgentDojo uses the calibrated four-state filter over $\mathcal{Z}$. Filter parameters and monitor calibration are fitted before policy learning and remain fixed during learning and evaluation.

Decision context. A context is a finite summary used to select a row of the learned policy. The same proposed tool call can reach different rows after different histories. Branch action replay uses fixed intervals and categorical features; AgentDojo learns a tree from training features and action values, then holds its grouping fixed during the LP solve. The selected grouping and recommendation probabilities remain fixed during evaluation.

Monitor utilities. In the matched experiment, the five monitors use cost weights $0.85, 1, 1.15, 1.15, 1.15$, respectively, in Equation 2. The future-cost term $B_s^{\pi^0}(d)$ averages over all current episodes because an attack episode can be followed by a benign episode. Multiplying it again by the probability that the current episode is benign would undercount this future cost.

Averaging over the action probabilities given by K_η gives $Q_{i,s}(c)$. The control that values only the current campaign sets $B = 0$ and $Q_d = q_d$.

Covered Histories. The obedience constraints use the distribution of reference histories chosen to estimate action values. They compare one changed report, followed by the same fixed rule after the current action. Several changed reports in sequence could lead to histories outside that calculation. Evaluating such histories would require new belief and action-value estimates.

B.1 Features That Select the Learned Policy Row

Different abstract contexts can share a fitted policy row. This appendix documents exactly which coarse features of the context are used to look the distribution up at test time. Table 4 lists the features used to select that row in each experiment, separately from the inputs to action valuation. Two decisions with the same deviation interval and stage can select different policy rows under the second mapping if their structural scores occupy different intervals. A new decision is assigned to an existing interval combination or tree leaf, which supplies its learned recommendation probabilities; no LP is solved on the test decision.

Table 4: Features used to select ORBIT's policy row. Here j_D denotes the interval containing current-deviation probability b_t, κ_t is the observed stage, g_t is a stage-derived progress category, and j_σ is the interval containing the structural progress score. The definitions below specify the encodings.

Experiment or decision	Inputs that select the policy row
Branch frontier: initial decision (Section 3.1)	The compact configurations use coarse: (j_D, κ_t, g_t), which is deviation-probability interval and stage, g_t derived from the stage; fine: $(j_D, \kappa_t, j_\sigma)$, which adds an interval of the score combining stage, risk, sensitivity, opacity/denial, provenance/lineage and history depth. Each fitted configuration fixes this choice.
Matched obedience and response history (Section 3.2; Appendix G)	(j_D, κ_t, g_t) for both members of each comparison. Response evidence can change the updated belief and therefore its interval.
ORBIT-inspect: learned decision after inspection	Updated stage, updated deviation-probability interval, updated monitor-risk interval, and tool family. ORBIT-postrule instead applies its fixed, validation-selected cutoff to the updated deviation probability.
Fixed and adaptive AgentDojo (Sections 3.3 and 3.7)	A leaf of the fitted decision tree, selected using the belief, observation-history, and monitor-score inputs in Table 6, plus tool-family indicators. The adaptive study reuses the selected fixed-recording policies.

Stage and progress updates. In branch replay, κ_t starts at the cached source stage and advances to the highest stage among newly observed events, using fixed event-type ranks. The score σ_t combines available stage and telemetry using the fixed weights specified in Appendix C. The AgentDojo experiments use the four-state posterior and its history features, without separate κ_t or σ_t inputs.

Branch intervals and progress features. Branch replay groups current-deviation probabilities into six fixed intervals. Spacing their boundaries on the log-odds scale provides finer resolution at small probabilities than equal-width probability intervals. The five internal boundaries are equally spaced between $\mathrm{logit}(.01) - 3$ and $\mathrm{logit}(.01) + 4$. The interval count, reference value .01, and offsets -3 and 4 are fixed implementation settings. In probability units these boundaries are approximately 0.000503, 0.002886, 0.016381, 0.087454, and 0.355461; an exact boundary belongs to the lower interval. The progress category g_t is early before suspicious access, middle from suspicious access until sensitive action, and near the boundary from sensitive action onward. It is derived from the observed stage, not a future harm label. The alternative structural score combines stage, current monitor risk, target sensitivity, opacity or denial, provenance or lineage, and observed history depth. Its five intervals have internal boundaries $.2, .4, .6, .8$. Post-inspection monitor risk uses six equal-width intervals on $[0, 1]$.

Table 5: Branch-replay performance. Early stopping precedes unavoidable harm; FI/1k is weighted benign interruption with $c_{\text{ins}} = 0.02$. ORBIT uses median aggregation ($\eta = 0$). The final block reports the separate five-seed response-evidence ablation (Appendix G).

Method / setting	Early stopping (%) ↑	Benign cost (FI/1k) ↓
1. Score & feature methods *(No persistent Bayesian belief)*		
Stage / maximum-stage threshold	0.0	0.00
Memoryless stacker	4.0	56.21
Lagged stacker	6.5	53.25
Edge stacker	29.0	54.73
2. DBN posterior thresholding *(Persistent belief; no inspection)*		
Lower-cost setting	18.5	20.71
Higher-cost setting	32.3	51.78
3. ORBIT variants *(Persistent belief & learned interventions)*		
ORBIT-noinspect	24.2	54.73
ORBIT-inspect	31.5	36.95
ORBIT-postrule, lower cost	**37.1**	**19.35**
ORBIT-postrule, higher cost	41.9	44.97
Response-evidence ablation *(Exploratory five-seed protocol)*		
ORBIT-postrule, response update omitted	28.91	24.2
ORBIT-postrule + response-based belief update	**31.22**	32.3

Neither persistent-compromise probability nor b_t^{eff} is a separate lookup field in these compact branch configurations. Both compromise and deviation beliefs nevertheless affect the action values through Equation 12. Thus, changing compromise belief alone while holding all lookup features fixed does not change the selected row of a frozen branch policy. For a missing interval combination, the implementation first tries a training-derived grouping by deviation interval and stage, then by stage alone. These pooled probabilities also remain fixed during evaluation.

AgentDojo tree inputs. Table 6 defines the 18 numerical inputs available to the AgentDojo policy tree (See also Figure 3). Additional binary indicators identify the tool family using a vocabulary learned from training data. Each fitted tree selects which inputs to test and the thresholds for those tests; calibration selects among fitted candidates. At each new decision, these tests route the available feature values to a leaf containing the learned recommendation probabilities. Note that the tree and those probabilities are fixed during evaluation, including for the adaptive study.

C Expected Utility and Attack Progress

The learning program in Section 2.2 takes estimated action values as inputs. This appendix explains how persistent-compromise belief, current-deviation belief, and observed attack progress inform the valuation in Section 2.1. It also explains when the benefit of inspection exceeds its cost. These calculations supply the action values from which ORBIT learns recommendation probabilities.

Persistent Compromise and Current Deviation. Compromise can change over time. The transition model for persistent compromise $Comp_t$ makes harmful access increase the chance of compromise, while recovery transitions are rare. Current deviation Dev_t records whether the actor is presently acting against the authorized task within the campaign. An actor can have $Comp_t = 1$ while $Dev_t = 0$ for a long period: it remains compromised but is not currently carrying out the attack. The model therefore uses current deviation to assess immediate intervention and persistent compromise to carry evidence across time. These two parts of the belief model are connected, so mild current activity does not erase what earlier behavior revealed.

Accounting for attack progress. When scoring actions, the model adjusts the current-deviation belief using observed attack progress. This adjusted belief accounts for the value of remaining chances to intervene and how close the observed stage is to unavoidable harm:

$$b_t^{\text{eff}} = b_t + w_{\text{str}}\,\sigma_t\big(\gamma_0 + \gamma_1(\beta_t^{\mathsf{Comp}} - b_t)_+\big)(1 - b_t), \tag{12}$$

Table 6: The 18 numerical inputs available to the AgentDojo policy tree. Each fitted tree selects which inputs to test. Tool-family indicators are additional inputs. Belief entries are posterior estimates.

No.	Input	Definition
1	Compromise-or-harm belief	Sum of the compromised-state and harm-complete probabilities (inputs 4 and 5).
2	Aligned-state belief	Probability that the agent is aligned with the authorized task.
3	Exposed-state belief	Probability that the agent is in the attack-exposed state.
4	Compromised-state belief	Probability that the agent is in the compromised state.
5	Harm-complete belief	Probability that the agent has reached the harm-complete state.
6	Belief change	Current input 1 minus its value before this decision's belief update.
7	Peak belief	Maximum value of input 1 since the last memory reset, including the current decision.
8	New evidence available	One if new monitor evidence was observed at this decision; zero otherwise.
9	Evidence age	$\log(1 + \min(A_t, 1000))$, where A_t counts decisions since the last observation.
10	Observation count	$\log(1 + N_t)$, where N_t counts observations since the last memory reset.
11	Mean monitor score	Mean of the latest available scores from enabled monitors.
12	Maximum monitor score	Maximum of those scores.
13	Monitor disagreement	Maximum minus minimum of those scores.
14	First LLM critic	Latest available calibrated score from the first critic.
15	Second LLM critic	Latest available calibrated score from the second critic.
16	Permission checker	Latest available calibrated permission-checker score.
17	Taint/provenance monitor	Latest available calibrated taint/provenance score.
18	Sequence detector	Latest available calibrated suspicious-sequence score.

When no new observation is available, monitor-score inputs retain their previous values. Disabled monitor channels have the fixed value 0.5 and are excluded from inputs 11–13. Persistent memory carries these quantities across episodes; episode reset re-initializes them at each episode boundary.

Here $b_t = P(\mathsf{Dev}_t = 1 \mid h_t)$ is the current-deviation probability, β_t^{Comp} is the persistent-compromise probability, and σ_t is the normalized attack-progress score. The parameter w_{str} controls the overall strength of the adjustment. Within it, γ_0 sets the baseline weight on attack progress, while γ_1 scales the additional weight when persistent-compromise belief exceeds current-deviation belief. The positive part $(\cdot)_+$ makes this additional contribution zero when compromise belief is no greater than deviation belief. The factor $(1 - b_t)$ reduces the adjustment as the current-deviation probability approaches one. The action-scoring model uses b_t^{eff} as its risk input when estimating the expected utilities defined in Equations 4–6. All three actions use the same adjusted score; inspection must justify its cost through the additional evidence and subsequent decisions it enables.

Computing the attack-progress score. In branch replay, recorded events have fixed stage labels. Execution starts at the cached source stage and retains the highest stage observed along that branch. For example, observing sensitive access advances an exposed episode to suspicious access, and a subsequent routine event cannot reverse that advancement. Sigma measures attack progress, not attack probability or prevalence. The code calculates it using fixed weights on stage, monitor risk, target sensitivity, opacity and denials, provenance and lineage, history depth, and proximity to harm. These weights are specified by the implementation, rather than learned by the intervention policy.

AgentDojo uses a different representation: its policies use the four-state posterior and remembered evidence, without separate stage or sigma inputs. In the AgentDojo experiments, ORBIT maintains probabilities over ALIGNED, EXPOSED, COMPROMISED, and HARMCOMPLETE, without separate κ_t or σ_t inputs. At each proposed tool action, the filter predicts the next state and incorporates monitor evidence when observed. The learned policy uses this posterior, its change and running peak, remembered monitor scores, observation count, evidence age, and tool family. These features select a leaf of the fixed decision tree, which supplies the probability of a campaign-ending stop.

Table 7: Held-out results for adaptive attacks over equally weighted 10 paired policy seeds. Harm gap is episode reset minus persistent; brackets give a paired 95% interval. Benign cost is % of scheduled legitimate decisions denied. "Search budget" is the number of attack-instruction versions tested before selecting one for evaluation on new campaigns; larger budgets create more chances to improve the attack. Ties favor fewer stops or cancellations, then fewer unresolved runs, and finally the version generated first.

Search	Harmful goals completed (%)			Legitimate work denied (%)	
budget	Persistent	Episode reset	Gap [95% CI]	Persistent	Episode reset
1	12.83	24.17	11.33 [3.43, 19.24]	44.71	11.78
2–8	7.00	24.17	17.17 [8.62, 25.71]	43.96	11.78
16	7.00	18.50	11.50 [4.91, 18.09]	43.96	10.99

Persistent memory carries these quantities across episodes; the episode-reset control reinitializes them at each episode boundary.

At each branch-replay source decision, σ_t is a fixed weighted sum of normalized stage (0.24), current monitor risk (0.18), target sensitivity (0.16), opacity/denial (0.16), provenance/lineage (0.14), history depth (0.07), and a near-boundary indicator (0.05). Each component lies in $[0, 1]$. Stage normalization caps the rank of κ_t at four and divides by four. Monitor risk, target sensitivity, and the underlying opacity, denial, provenance, and lineage scores are clipped to $[0, 1]$. The opacity/denial component takes their larger value plus half their product, capped at one. The provenance/lineage component takes the maximum of the provenance, target-lineage, and credential-lineage scores. History depth is the logarithm of one plus the recorded source-history edge count, capped at three and divided by three. The near-boundary indicator equals one from SENSITIVEACTION onward and zero otherwise.

These weights are fixed implementation choices, not learned policy parameters. The calculation uses information available at the source decision, excluding future outcomes and evidence obtained by subsequent inspection.

RIV: realized information value Equation 9 starts with the best improvement available after inspection and subtracts the improvement lost by the actual follow-up rule. All terms must count the same future period and use the updated belief and stage for the same action branch. RIV excludes inspection cost. When counting only the current campaign, inspection is preferred if the improvement obtained from its evidence exceeds the inspection cost plus expected harm during the wait:

$$\mathbb{E}_e \Xi^\pi - \max_a q_a(h_t) \;>\; c_{\text{ins}} + C_{\text{delay}}\, P(\text{harm during inspection} \mid h_t), \tag{13}$$

for the local-only comparison. With later campaigns included, the decision also depends on action-specific future returns in Equation 10.

We compare inspection fees $c_{\text{ins}} = 0.02$ and 0.001 for both ORBIT-inspect and ORBIT-postrule using the same cached branches. Both variants learn when to inspect. ORBIT-inspect uses a learned mediator after inspection; ORBIT-postrule uses a validation-selected cutoff. At each fee, we recompute action values and refit each initial mediator using its respective follow-up rule. The postrule cutoff is recalibrated on validation data. Evaluation uses one fixed held-out split containing 124 attack and 676 benign branches.

The inspection fee prices one inspection. The cost limit α restricts weighted benign interruption and is held fixed within each fee comparison. We test limits of $5, 10, 15, 20, 25, 30, 40,$ and 50 FI/1k; the LP uses these values divided by 1,000. We also vary the implementation's penalty multiplier for risky ALLOW continuation, $c_{\text{benign}} \in \{0.10, 0.25, 0.50, 1.00\}$, while keeping the benign-blocking utility loss at 0.1.

Table 4 defines the two policy-row mappings. They use (j_D, κ_t, g_t) or $(j_D, \kappa_t, j_\sigma)$. Both include the deviation-probability interval and stage. The first adds a category derived from that stage; the second adds a structural-score interval incorporating additional telemetry. Each mapping is fixed within a fee comparison, giving 64 matched comparisons per variant.

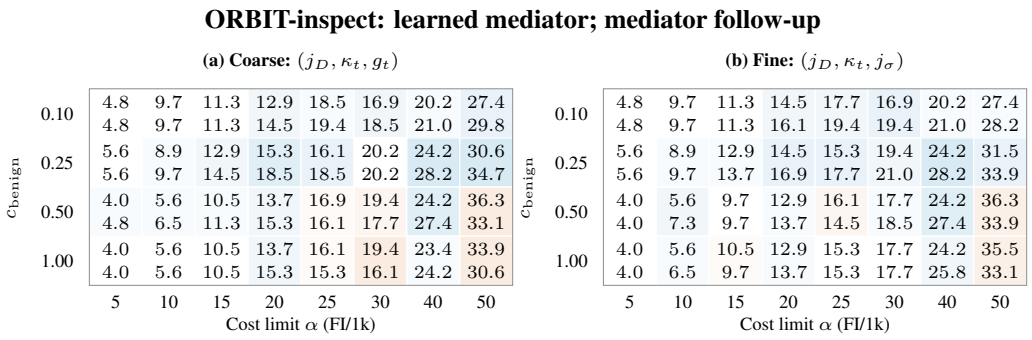


Figure 4: ORBIT-inspect early stopping at inspection fees 0.02 and 0.001. Each cell displays the actual rates in percent: the upper number uses fee 0.02 and the lower number uses fee 0.001. Shading shows the lower-fee rate minus the higher-fee rate: blue indicates improvement and orange indicates deterioration, using a shared scale in percentage points. Columns use the mappings in Table 4. All cells use the same held-out seed.

C.1 Inspection Use and Prevention

Lowering the fee increases attack inspection in all 64 ORBIT-inspect comparisons and in 63 ORBIT-postrule comparisons, with one unchanged. Benign inspection increases in every comparison for both variants. The early-stopping response nevertheless differs. For ORBIT-inspect (Figure 4), early stopping improves in 37 comparisons, is unchanged in 17, and decreases in 10; changes range from -3.2 to $+4.0$ percentage points. For ORBIT-postrule (Figure 2), it improves in 34 and decreases in 30, with changes from -5.6 to $+23.4$ points. Its largest gains occur at $c_{\text{benign}} = 0.10$, where all 16 comparisons improve; with $c_{\text{benign}} \geq 0.25$ and cost limits of at least 25 FI/1k, early stopping decreases in 22 of 24 comparisons.

D The Training Linear Program

For training decision j, we store the expected utility and benign cost of each possible action and its continuation. A recommendation profile combines these values using the execution probabilities $K_\eta(d \mid c)$. Changing one report changes those probabilities; the corresponding constraint row calculates the gain from that change.

The LP averages the stored values within each context. Simplex constraints require nonnegative recommendation probabilities that sum to one. A further row limits average benign cost. The HiGHS optimization solver, called through `linprog`, solves the resulting problem. Replay samples the action from $\pi_{\widehat{x}}(d \mid s)$. Under obedient reports, this gives the same action probabilities as first sampling a recommendation list and then sampling its implemented action. The full recommendation probabilities are kept so that every possible changed report can be checked.

Each row $q = (i, v, a \to b)$ combines gains across contexts in which head i has the same private information v. The matched fits have 72 obedience rows and 15–19 contexts. Evaluation uses the learned probabilities directly. Adding a chance of always allowing afterward would change those probabilities and would require a new obedience check when $\eta > 0$.

Size of the computation. Each starting decision stores $|\mathcal{D}|$ action values. Calculating one value may require following several later alternative paths, called rollout branches. Recommendation lists and changed reports reuse the stored action values with different execution probabilities. The LP retains $|\mathcal{S}||\mathcal{D}|^m$ profile variables, with $m = 5$ and $|\mathcal{D}|^m = 243$. The matched fits contain 3,645–4,617 profile variables and 72 obedience rows.

E PROOFS

E.1 LEARNING FROM FINITE TRAINING DATA

Assumption 1 (Bounded feedback and LP stability)**.** *The counterfactual utility, deviation-gain, and friction vectors are bounded and conditionally unbiased, and their centered estimation differences satisfy bounded martingale concentration. Every deviation row in $\mathcal{Q}$ is included in the empirical program and has positive training support: $\widehat{\mathcal{Q}} = \mathcal{Q}$ and $n_{\min} > 0$. Bounds cover the full valuation horizon. The population and empirical programs are feasible, and an optimal multiplier for the empirical constraint vector has ℓ_1 norm at most $L < \infty$.*

Here, a centered estimation difference is an estimate minus its target value, with any conditional mean subtracted. Counterfactual means an alternative action that is evaluated by replay. Conditionally unbiased means that signed differences between estimates and their target values average to zero given the information specified in the model. Bounded martingale concentration is the assumption that these differences remain sufficiently controlled when accumulated over dependent decisions; it permits a probability bound on their total estimation uncertainty. Feasible means that at least one policy satisfies all the constraints. The multiplier bound limits how much optimal utility can change when the constraint limits change slightly.

The experiment with a fixed pilot verifies that the learned policies satisfy the constraints estimated from its data. Histories from the same actor can share observations and future outcomes. Distinct decision identifiers alone therefore do not establish the assumed probability bounds on estimation uncertainty.

Lemma 1 (Uniform estimation uncertainty)**.** *Under Assumption 1, with probability at least $1 - \delta$,*

$$\sup_{x \in \mathcal{X}} |\widehat{U}(x) - U(x)| \le \zeta_U, \quad \max_q \sup_{x \in \mathcal{X}} |\widehat{G}_q(x) - G_q(x)| \le \zeta_G, \quad \sup_{x \in \mathcal{X}} |\widehat{F}(x) - F(x)| \le \zeta_F,$$

with $\zeta_U, \zeta_G, \zeta_F$ as in Theorem 1.

Proof. Let $\mathcal{X} = \prod_{s \in \mathcal{S}} \Delta(\mathcal{A})$. For a fixed x, write $\widehat{U}(x) - U(x) = N^{-1} \sum_{j=1}^{N} D_j(x)$. Under the stated estimation assumptions, these centered differences form a martingale difference sequence. The raw utility contribution lies in $[-V, V]$, so, conditional on the preceding information, $D_j(x)$ lies in an interval of width at most $2V$. Centering shifts the interval without increasing its width. Conditional Hoeffding concentration therefore gives

$$P\big(|\widehat{U}(x) - U(x)| > t\big) \le 2 \exp\left(-\frac{Nt^2}{2V^2}\right).$$

The estimation difference is affine in x, so its absolute value attains a maximum at a vertex of $\mathcal{X}$. There are $K^{|\mathcal{S}|}$ such vertices, one profile choice per context. A union bound over these vertices gives

$$P\left(\sup_{x \in \mathcal{X}} |\widehat{U}(x) - U(x)| > t\right) \le 2K^{|\mathcal{S}|} \exp\left(-\frac{Nt^2}{2V^2}\right).$$

For an allowance δ_U, this yields

$$\zeta_U = V \sqrt{\frac{2\big(|\mathcal{S}| \log K + \log(2/\delta_U)\big)}{N}}.$$

The same argument for benign cost gives

$$\zeta_F = F_{\max} \sqrt{\frac{2\big(|\mathcal{S}| \log K + \log(2/\delta_F)\big)}{N_{\text{ben}}}}.$$

For obedience, union over the vertices and the Q deviation rows. Raw gain contributions lie in $[-G, G]$, and row q has $n_q \ge n_{\min}$ supporting observations satisfying the stated concentration assumptions. Thus

$$\zeta_G = G \sqrt{\frac{2\big(|\mathcal{S}| \log K + \log(2Q/\delta_G)\big)}{n_{\min}}}.$$

Taking $\delta_U = \delta_F = \delta_G = \delta/3$ makes the total failure probability at most δ and gives the orders stated in Theorem 1. □

Proof of Theorem 1. By Lemma 1, on an event E of probability at least $1-\delta$ the three uniform bounds

$$\sup_{x\in\mathcal{X}}|\widehat{U}(x)-U(x)|\le\zeta_U,\quad \max_q\sup_{x\in\mathcal{X}}|\widehat{G}_q(x)-G_q(x)|\le\zeta_G,\quad \sup_{x\in\mathcal{X}}|\widehat{F}(x)-F(x)|\le\zeta_F$$

hold simultaneously. Work on E throughout.

Obedience and benign feasibility. Since $\widehat{x}$ solves Equation 11, $\widehat{G}_q(\widehat{x})\le\varepsilon_{\mathrm{CE}}$ for every $q\in\widehat{\mathcal{Q}}$ and $\widehat{F}(\widehat{x})\le\alpha$. Adding the obedience allowance, $G_q(\widehat{x})\le\widehat{G}_q(\widehat{x})+\zeta_G\le\varepsilon_{\mathrm{CE}}+\zeta_G$; since $\varepsilon_{\mathrm{CE}}+\zeta_G\ge 0$ this also bounds the positive part, so $\mathrm{Reg}_{\mathrm{dev}}(\widehat{x})=\max_q[G_q(\widehat{x})]_+\le\varepsilon_{\mathrm{CE}}+\zeta_G$. Likewise $F(\widehat{x})\le\widehat{F}(\widehat{x})+\zeta_F\le\alpha+\zeta_F$.

Utility. Let $x^\star$ be optimal for the population program at the same thresholds $(\varepsilon_{\mathrm{CE}},\alpha)$. On E, every estimated constraint value at $x^\star$ exceeds its population value by at most its uncertainty, so

$$\widehat{G}_q(x^\star)\le\varepsilon_{\mathrm{CE}}+\zeta_G\ \ (\text{all } q),\qquad \widehat{F}(x^\star)\le\alpha+\zeta_F.$$

Thus $x^\star$ is feasible for the empirical program once each obedience limit is raised by ζ_G and the benign limit by ζ_F. By LP duality, relaxing the limits raises the optimal empirical objective by at most the multiplier of each limit times the amount it is raised; summing, this increase is at most $\|\lambda^\star\|_1\max\{\zeta_G,\zeta_F\}\le L\max\{\zeta_G,\zeta_F\}$, since the largest single loosening is $\max\{\zeta_G,\zeta_F\}$ and $\|\lambda^\star\|_1\le L$. Hence $\widehat{U}(x^\star)\le\widehat{U}(\widehat{x})+L\max\{\zeta_G,\zeta_F\}$. Splitting the suboptimality into two empirical-to-population conversions and this empirical gap,

$$U(x^\star)-U(\widehat{x})=\underbrace{[U(x^\star)-\widehat{U}(x^\star)]}_{\le\zeta_U}+\underbrace{[\widehat{U}(x^\star)-\widehat{U}(\widehat{x})]}_{\le L\max\{\zeta_G,\zeta_F\}}+\underbrace{[\widehat{U}(\widehat{x})-U(\widehat{x})]}_{\le\zeta_U}\le 2\zeta_U+L\max\{\zeta_G,\zeta_F\}.$$

The factor 2 is the two conversions—one at $x^\star$, one at $\widehat{x}$—each at most ζ_U; the max is the multiplier (ℓ_1 norm $\le L$) paired against the largest constraint loosening. □

E.2 Evidence Growth Until Intervention

This section bounds delay when evidence has positive expected growth in every continuing context. The bound uses the initial evidence gap, a one-step overshoot allowance, and the minimum supported growth rate. We do not estimate these drift bounds and make no numerical delay claim.

Let $\mathcal{F}_t$ denote the information available immediately before the randomized intervention decision at time t, including the observed history and current belief. Define

$$\Lambda_t=\operatorname{logit}\beta_t(\textsc{Compromised})$$

and, on steps where execution continues,

$$\ell_t=\Lambda_{t+1}-\Lambda_t.$$

Throughout, B bounds the complete one-step log-odds change, $|\ell_t|\le B$, including both prediction and the evidence update. Positive ℓ_t supports compromise; negative ℓ_t argues against it.

Let T be the first stopping intervention, whether selected by the mediator or triggered by the belief threshold $a=\operatorname{logit}(\mu_{\mathrm{thr}})$. At compromise onset t_0, define the initial gap to that threshold as

$$H=a-\Lambda_{t_0}>0.$$

Let $\mathcal{S}_M\subseteq\mathcal{S}$ denote the contexts reachable while the actor is compromised and the policy continues, and let t_{harm} denote the first time at which prevention becomes impossible. For each continuing context $s\in\mathcal{S}_M$, let $\underline{r}_s$ be a lower confidence bound on its conditional mean evidence growth. Replacing the full history with its context summary may reduce the guaranteed conditional growth by at most ρ_χ. Accordingly, define

$$\gamma=\min_{s\in\mathcal{S}_M}\underline{r}_s-\rho_\chi.$$

The required drift condition is

$$\mathbb{E}[\ell_t\mid\mathcal{F}_t,T>t]\ge\gamma\quad\text{whenever}\quad P(T>t\mid\mathcal{F}_t)>0. \tag{14}$$

This condition concerns evidence growth conditional on the policy continuing, rather than growth averaged over stopped and continuing trajectories. It must hold in every context reachable before intervention.

Theorem 2 (Intervention delay under positive evidence growth)**.** *Fix a learned policy and a valid set of drift lower bounds. Before T, suppose the actor remains compromised, every continuing context lies in $\mathcal{S}_M$, $|\ell_t| \le B$, and replacing the full history with its context loses at most ρ_χ of conditional mean evidence growth. If $\gamma > 0$, then*

$$\mathbb{E}[T - t_0 \mid \mathcal{F}_{t_0}] \le \frac{H + B}{\gamma}.$$

If, in addition, $t_{\mathrm{harm}} - t_0 \ge D_{\mathrm{tgt}} > 0$ almost surely, then

$$P(T < t_{\mathrm{harm}} \mid \mathcal{F}_{t_0}) \ge 1 - \min\left\{1, \frac{H + B}{\gamma D_{\mathrm{tgt}}}\right\}.$$

The first bound divides the initial evidence gap, including a one-step overshoot allowance, by the smallest supported evidence-growth rate. The second uses the minimum time remaining before harm to obtain a lower bound on the probability of intervening in time. Both conclusions concern the actual stopping time and use no evidence generated after intervention.

Proof. Fix a learned setup with valid drift lower bounds and condition on $\mathcal{F}_{t_0}$, with $T \ge t_0$. For $n \ge t_0$, let $T_n = T \wedge n$. The filtration $\mathcal{F}_t$ precedes the current randomized decision, so $\{T > t\}$ need not be $\mathcal{F}_t$-measurable. We therefore retain the conditioning on continuation explicitly. Products $\mathbf{1}\{T > t\}\ell_t$ are interpreted as zero on stopped steps.

Summing evidence changes only while execution continues gives

$$\Lambda_{T_n} - \Lambda_{t_0} = \sum_{t=t_0}^{n-1} \mathbf{1}\{T > t\}\ell_t.$$

On histories with positive continuation probability, Equation 14 implies

$$\begin{aligned}\mathbb{E}[\mathbf{1}\{T > t\}\ell_t \mid \mathcal{F}_t] &= P(T > t \mid \mathcal{F}_t)\mathbb{E}[\ell_t \mid \mathcal{F}_t, T > t] \\ &\ge \gamma P(T > t \mid \mathcal{F}_t).\end{aligned}$$

When continuation probability is zero, both sides are zero. Applying the tower property and summing gives

$$\begin{aligned}\mathbb{E}[\Lambda_{T_n} - \Lambda_{t_0} \mid \mathcal{F}_{t_0}] &\ge \gamma \sum_{t=t_0}^{n-1} P(T > t \mid \mathcal{F}_{t_0}) \\ &= \gamma \mathbb{E}[T_n - t_0 \mid \mathcal{F}_{t_0}].\end{aligned}$$

Before intervention, the threshold has not triggered, so $\Lambda_t < a$ for $t < T$. A final update increases log-odds by at most B; an earlier mediator stop or truncation also satisfies $\Lambda_{T_n} \le a + B$. Consequently, $\Lambda_{T_n} - \Lambda_{t_0} \le H + B$, and

$$\mathbb{E}[T_n - t_0 \mid \mathcal{F}_{t_0}] \le (H + B)/\gamma.$$

Conditional monotone convergence as $n \to \infty$ yields

$$\mathbb{E}[T - t_0 \mid \mathcal{F}_{t_0}] \le (H + B)/\gamma.$$

Finally, $t_{\mathrm{harm}} - t_0 \ge D_{\mathrm{tgt}}$ almost surely implies $\{T \ge t_{\mathrm{harm}}\} \subseteq \{T - t_0 \ge D_{\mathrm{tgt}}\}$. Conditional Markov's inequality gives

$$P(T \ge t_{\mathrm{harm}} \mid \mathcal{F}_{t_0}) \le \min\left\{1, \frac{H + B}{\gamma D_{\mathrm{tgt}}}\right\}.$$

Taking complements proves the second conclusion. □

The context-specific confidence bounds are estimated before deployment and must hold jointly over all relevant contexts. If they are jointly valid with probability at least $1 - \delta_{\mathrm{drift}}$, the theorem applies conditional on that event. An unconditional failure bound must additionally account for δ_{drift}. Our experiments do not estimate these drift bounds and therefore do not instantiate a numerical intervention-delay guarantee.

Table 8: Four checks for interpreting the experimental results.

Question	Brief answer
Was the comparison fair?	Matched controls share information, action values, and budgets. Memory controls are retrained on the same episodes and observation masks. Branch actors and AgentDojo task/injection templates are separated across splits; policy and payload selection exclude test outcomes. Adaptive search matches payload counts, not the useful feedback available to each attacker.
What exactly was counted?	Branch FI/1k charges blocks and inspections against original benign decisions. AgentDojo prevention counts all 600 scheduled attacks, not only the 26 that originally caused harm. Benign denial retains all 39,874 original decisions, including canceled work. No post-stop evidence is used. Adaptive harm requires checker-confirmed completion; canceled and unresolved attacks remain in its denominator.
Does the experiment support the claimed mechanism?	At $\eta = 0.1$, all four matched O policies satisfy fitted obedience. Holding each U policy's action probabilities fixed, no recommendation distribution satisfies those constraints. Obedience therefore excludes some fitted policies; it does not establish a reliable performance gain or population guarantee.
Where does the result weaken?	Memory increases benign denial; results at the 20% calibration limit are mixed, and adaptation is light. Paired 95% Student-t intervals use four branch runs or ten AgentDojo collections, keeping repeated histories together. They assume independent runs and approximately normal paired differences, with no adjustment for multiple comparisons.

Table 9: Policies with and without obedience, both counting all remaining campaigns: $\eta = 0.1$, training budget 25 FI/1k, $c_{\text{ins}} = 0.02$. Rates are averages across four seeds. Early stops occur before harm.

Policy	Satisfies obedience	Mean early stops (%)	Mean FI/1k
Obedient (O)	4/4 runs	32.70	25.16
Unconstrained (U)	0/4 runs	29.58	28.58

F Experimental Audit Summary

F.1 Supporting Tables for the Memory and Obedience Comparisons

These tables back statements in Sections 3.2 and 3.3: Table 9 is the matched obedience audit; Table 10 adds the full-memory direct control and the threshold controls at the 50% limit; Table 11 gives the two tighter calibration limits; and Table 12 collects the paired 95% intervals for the headline comparisons.

At the 50% limit, ORBIT's mean prevention exceeds the full-memory direct control in five of six settings. The exception is 5% observation without taint or permission, where the direct control prevents 37.67% at 32.76% benign work denied, compared with ORBIT's 33.67% at 30.14%. A common allowance does not by itself establish a prevention advantage at equal achieved cost.

For the decision-reset ORBIT arm without taint at 20% observation, the candidate grid does not supply a feasible choice in every run at any of the three limits; that arm is reported as missing rather than averaged over a smaller set of runs.

In the adaptive study, before letting the attacker try more than four versions or testing on new campaigns, we checked three things; different attempts produced different outcomes giving the attacker useful feedback; at least one attempt completed a harmful goal against the defender that resets its memory; the attacker changed its instructions and the agent's behavior changed too. We ran separate searches against each defender, and both passed these checks. This established that we were testing functioning attacks and meaningful revisions; it did not require every revision to make the attack stronger. Note that the adaptive study re-executes a revised injection on separately split cam-

Table 10: Additional controls trained and evaluated with campaign-long stopping, at the 50% calibration allowance. Cells give attack prevention (%) / legitimate decisions denied (%), averaged across ten runs. Direct retains full memory but omits obedience. The threshold controls use the same fitted belief filter and either retain memory or reset it at every decision.

Signals	Observed	Direct	Threshold, memory	Threshold, reset
Full	5%	33.67/26.40	16.50/11.68	15.67/11.93
Full	20%	42.83/31.89	29.17/21.75	24.50/17.14
No taint	5%	31.33/24.12	27.67/23.36	12.67/11.50
No taint	20%	20.00/17.65	26.50/19.62	17.33/12.86
No taint or permission	5%	37.67/32.76	15.50/11.94	16.17/12.37
No taint or permission	20%	27.17/21.98	24.83/19.17	21.33/16.21

Table 11: Tighter calibration limits on legitimate work denied. Each cell gives test attack prevention (%) / legitimate decisions denied (%), with equal weight for all ten runs. The limit selects policies using calibration outcomes; achieved test costs can exceed it. The 50% results appear in Tables 2 and 10.

Limit	Signals	Observed	Full memory	Episode reset	Direct
5%	Full	5%	5.33/3.67	0.00/0.00	6.83/4.92
5%	Full	20%	0.67/0.63	0.00/0.29	1.00/0.76
5%	No taint	5%	0.00/0.00	1.33/1.34	0.17/0.70
5%	No taint	20%	3.33/3.41	2.00/1.79	0.00/0.17
5%	No taint or permission	5%	0.00/0.00	0.00/0.00	0.00/0.00
5%	No taint or permission	20%	0.00/0.00	0.00/0.00	0.00/0.00
20%	Full	5%	15.33/12.42	9.33/6.46	10.17/8.30
20%	Full	20%	18.50/16.36	20.00/14.42	17.17/12.84
20%	No taint	5%	4.33/3.54	5.33/5.08	14.83/10.43
20%	No taint	20%	9.83/9.53	24.83/20.77	2.50/1.43
20%	No taint or permission	5%	12.83/10.93	2.00/1.01	19.17/15.92
20%	No taint or permission	20%	23.67/20.37	14.50/9.92	19.17/14.11

paigns; its harmful-completion rates are not comparable to the 26 harmful attacks among the fixed recordings in Section 3.3.

G Response History Experiment Details

Response rule and information. The probe asks the modeled monitors for additional reports. Its recommendation probabilities are learned from training data and then fixed in both the response-history and control conditions. Five percent of this rule samples uniformly from all possible recommendation lists. After action d_t is chosen, the probe samples a recommendation list u_t whose median is that same action. Each head sees only its own recommendation and its private score grouped into ten intervals. Using training histories consistent with that information, its response rule chooses the report with the highest expected utility. On numerical ties within 10^{-12}, it returns the recommended report. The learned response rule stays fixed during testing. These replies come from modeled monitors; the study does not ask live language models to make strategic replies.

Belief update. Write D_t for current deviation and R_t for the list of returned reports. Training estimates how frequently each response occurs given the recommendation, context, and attack status D_t. Additive smoothing 0.5 adds a small count before forming probabilities, reducing problems from small or zero counts. The ratio below compares how likely the reply is during an attack with how likely it is during benign activity. Its logarithm is added to the next current-deviation belief on the log-odds scale. Clipping limits that added amount to the interval from -3 to 3:

$$\widehat{\omega}_t = \operatorname{clip}\left[\log \frac{\widehat{P}(R_t \mid u_t, D_t = 1, s_t)}{\widehat{P}(R_t \mid u_t, D_t = 0, s_t)}, -3, 3\right]. \tag{15}$$

Table 12: Paired differences and 95% intervals, using equal-run averages. Positive prevention differences favor the first policy; negative cost differences mean it interrupts less legitimate work. AgentDojo costs are percentage points of work denied at the 50% calibration allowance; branch costs use FI/1k. Intervals describe individual comparisons and are not adjusted for testing many at once.

Comparison	Mean difference	95% interval
Obedient minus unconstrained: early stopping (pp)	3.12	$[-4.83, 11.07]$
Obedient minus unconstrained: benign cost (FI/1k)	-3.41	$[-7.56, 0.73]$
5%, no taint: full minus episode reset, prevention (pp)	38.00	$[20.18, 55.82]$
Same comparison: benign work denied (pp)	33.46	$[18.82, 48.09]$
5%, no taint: full minus decision reset, prevention (pp)	34.50	$[6.39, 62.61]$
Same comparison: benign work denied (pp)	31.78	$[7.96, 55.61]$
20%, full signals: full minus episode reset, prevention (pp)	21.50	$[-1.12, 44.12]$
Same comparison: benign work denied (pp)	21.71	$[3.12, 40.31]$

Table 13: Effect of remembering monitor replies, averaged equally across five seeds. Early stops precede the prevention boundary; Boundary counts attacks that reach the point after which prevention is too late. FI/1k is weighted interruption of benign activity.

Policy	Early stops (%)	Harm (%)	Boundary (%)	FI/1k
Control	28.91	54.31	60.90	24.2
Response history	31.22	52.60	59.22	32.3

A head returning its recommendation contributes no added evidence in this implementation. If either attack class has fewer than five weighted training examples in a context, the lookup combines training counts across contexts with the same recommendation. An unseen response pattern also adds zero. The control uses the same probe and response rules but omits this belief change. The current action has already been chosen, so the new recommendation and reply can affect only interventions chosen after the reply arrives. Attack labels are used for fitting the response frequencies, not for changing that chosen action.

Fitting and selection. The branch protocol keeps training and test actors separate. It uses six risk and belief intervals, at most 320 contexts, a stated training budget of 25 FI/1k, and inspection cost 0.02. Six rounds regenerate histories under the updated rules and retrain the policy. Validation selects the candidate with highest utility among those within budget. If none meets the budget, it selects the highest-utility candidate and records the budget increase. The effective validation budget averages 29.20 FI/1k, and the selected iteration averages 2.2. In 80% of runs, changes in the policy fall to numerical zero within six rounds, meaning that further changes are below the numerical precision used. All runs are retained. Test data are used after selection. Early results informed the protocol; it was not preregistered, meaning fixed and registered before examining those results.

Response history findings. The recorded decision check finds that differences appear almost entirely after earlier replies have changed the belief, usually as more inspection on attack episodes. The quarter of attacks with the largest gain from demanding stronger intervention accounts for 57% of net additional early stops and 64% of harm reduction in that check. This locates the observed gains in histories where monitor recommendations conflict most strongly with their modeled preferences.

Among replies that differ from their recommendations, 65.1% have an estimated gain of at most 10^{-5}. For these replies, the monitor's expected score is almost the same whether it follows or changes the recommendation. This score difference measures the monitor's preference. A reply's value as evidence instead depends on how differently it occurs during attacks and benign activity (Equation 15).

The next proposition explains the ideal likelihood-ratio calculation. Common finite support means that the two response distributions assign positive probability to the same finite set of possible replies. Kullback–Leibler divergence, written D_{KL}, is a nonnegative measure of the difference between two probability distributions. It is zero when the distributions agree. The proposition shows the expected direction in which exact response evidence moves the belief.

Proposition 1 (Responses inform intervention after the reply arrives)**.** *Suppose the action d_t and probe u_t are chosen before R_t is observed, and let $p_j(r \mid u, s) = P(R_t = r \mid u_t = u, D_t = j, s_t = s)$ have common finite support. For the exact log-likelihood increment $\omega_t = \log[p_1(R_t \mid u_t, s_t)/p_0(R_t \mid u_t, s_t)]$,*

$$\mathbb{E}[\omega_t \mid u_t, s_t, D_t = 1] = D_{\mathrm{KL}}(p_1 \| p_0) \geq 0, \qquad \mathbb{E}[\omega_t \mid u_t, s_t, D_t = 0] = -D_{\mathrm{KL}}(p_0 \| p_1) \leq 0.$$

A response update applied after d_t cannot affect that action.

Proof. Conditional on $u_t, s_t, D_t = 1$, summing the log ratio against p_1 is exactly the definition of $D_{\mathrm{KL}}(p_1 \| p_0)$. Summing against p_0 gives its negative in the reverse direction. The action was fixed before the response was observed, so the response can affect only an intervention chosen after the reply arrives. □

The implementation estimates this likelihood ratio from finite training counts, adds smoothing, clips large changes, and sets some changes to zero. Applying a timing guarantee would require bounds on the differences introduced by all of these steps, together with the continuing-step evidence-growth condition in Equation 14. The response table reports intervention outcomes; it does not calculate that timing bound.

H Training Data Coverage Diagnostic

We examine the coverage guarantee of Corollary 1 using saved AgentDojo Workspace recordings at 5% observation without direct taint. Within each of ten collections, we retain 25%, 50%, or 100% of source episodes and refit the recommendation probabilities. The context map, belief filter, payoff rule, calibrated benign penalty, and training tolerance $\varepsilon_{\mathrm{CE}} = 10^{-6}$ remain fixed. For each subset, we measure $n_{\min}$ as the smallest number of distinct supporting source episodes across the fixed policy contexts. Retention is the experimental control; measured minimum coverage is the figure's horizontal axis.

Using the bounded-interval concentration specialization gives the reference envelope

$$\left[\,\mathrm{Reg}_{\mathrm{dev}}(\widehat{x}) - \varepsilon_{\mathrm{CE}}\right]_+ \leq \frac{A}{\sqrt{n_{\min}}}, \qquad A = G\sqrt{2\big(|\mathcal{S}|\log K + \log(2Q/\delta)\big)}.$$

Here G bounds a signed per-observation deviation gain, $Y \in [-G, G]$. Centering translates this interval without changing its width $2G$, giving the displayed Hoeffding constant. The envelope remains conditional on the concentration assumptions of Assumption 1; the recordings do not establish those assumptions or certify the measured counts as effective independent sample sizes.

Envelope height. The binary controller enumerates $K = 2^5 = 32$ recommendation profiles. With direct taint removed, four monitors are active, giving $Q = 8|\mathcal{S}|$ context-specific deviation constraints. We use $\delta = 0.05$ for the obedience bound alone. The simultaneous guarantee in Theorem 1 instead divides the failure allowance among utility, obedience, and benign cost. At $\eta = 0.1$, changing one active monitor's report can change the stopping probability by at most $0.9 + 0.1/4 = 0.925$. A campaign contains at most four attacks and 768 legitimate decisions, and utility is divided by 100. The resulting gain bound is

$$G = 0.925\,(4 + 768\lambda)/100.$$

Calibration selected a benign penalty λ per collection, held fixed across retention levels. The envelope's height therefore varies with both the context count and the saved penalty:

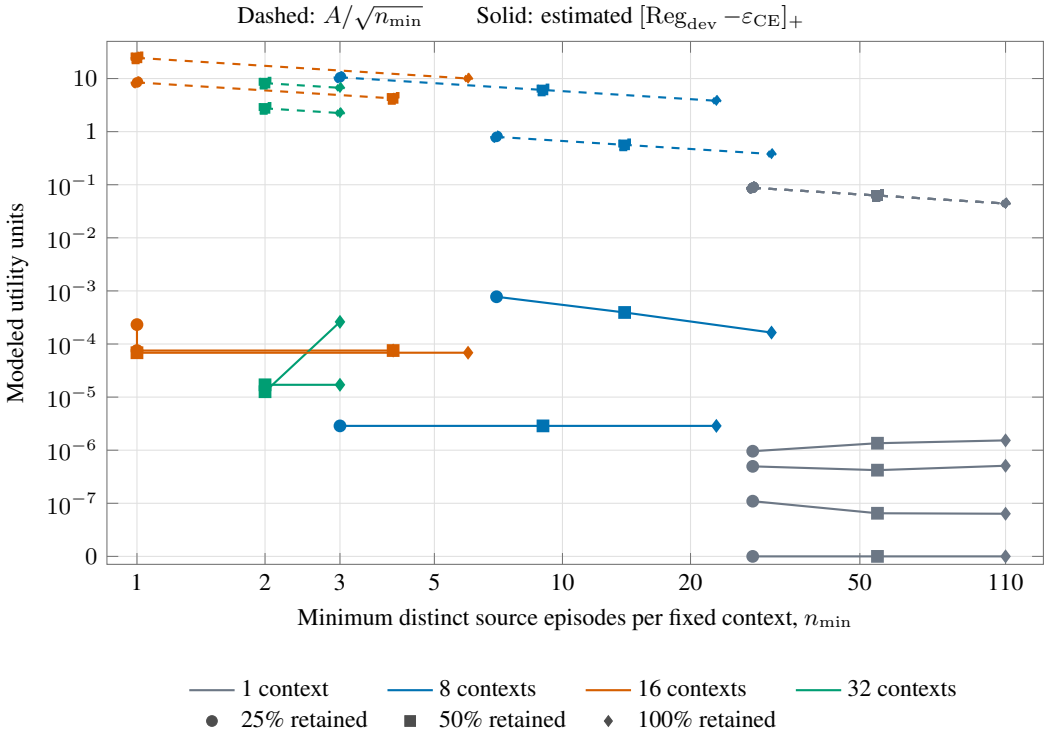


Figure 5: Minimum training coverage and the reference obedience-regret envelope. Horizontal position is measured minimum context support, not retention percentage; marker shapes indicate retention and colors indicate context counts. Each path holds its context map, payoff rule, and penalty fixed. Dashed curves show $A/\sqrt{n_{\min}}$, where $A = G\sqrt{2(|\mathcal{S}| \log K + \log(2Q/\delta))}$, $K = 32$, $Q = 8|\mathcal{S}|$, and $\delta = 0.05$. Here $G = 0.925(4 + 768\lambda)/100$, with saved $\lambda \in \{0.01, 0.03, 0.10, 0.30\}$; its value affects the height of the envelope. Solid curves show estimated held-out excess regret with $\varepsilon_{\mathrm{CE}} = 10^{-6}$. None of the 28 completed fits exceeds its envelope; two additional subsets have an unsupported context and are omitted. Episode independence is not certified, population regret is estimated, and nested fits are dependent, so this comparison does not certify a failure probability. The vertical scale is linear from zero to 10^{-7} and logarithmic above it; exact zeros and overlapping points are retained.

λ	Collections	G
0.01	4	0.10804
0.03	2	0.25012
0.10	2	0.74740
0.30	2	2.16820

Comparison with the deterministic bound. Since $\mathrm{Reg}_{\mathrm{dev}} \le G$, the positive excess is also bounded by G. The reference envelope improves on this ceiling only when $A/\sqrt{n_{\min}} < G$. Cancelling G gives

$$n_{\min} > 2\big[|\mathcal{S}| \log K + \log(2Q/\delta)\big] = 2\big[|\mathcal{S}| \log 32 + \log(320|\mathcal{S}|)\big].$$

Thus, this crossover does not depend on the benign penalty:

Contexts $|\mathcal{S}|$	Minimum integer support with envelope below G
1	19
8	72
16	128
32	241

Table 14: Sensitivity to concealed inspection evidence at a fixed operating point. Entries are equal-weight means ± sample standard deviations across six replay seeds. Early stopping and harm are percentages of attack branches; FI/1k measures weighted benign interruption. The two controls do not consume edited inspection responses.

Condition	Early stopping (%)	Harm (%)	FI/1k
ORBIT-postrule, original	26.12 ± 5.53	57.49 ± 6.68	22.18 ± 7.27
Random concealment	20.05 ± 2.91	61.74 ± 3.69	22.15 ± 7.14
Targeted concealment	15.56 ± 2.49	65.80 ± 2.52	22.38 ± 6.90
ORBIT-noinspect	15.46 ± 4.56	65.60 ± 2.97	33.43 ± 11.46
DBN thresholding	19.25 ± 6.77	62.37 ± 7.06	26.31 ± 11.64

Below these counts, the reference envelope adds no improvement over the deterministic ceiling G.

Results and interpretation. Figure 5 compares estimated held-out excess regret with the reference envelope. Across 28 completed fits, estimated excess regret stays below $A/\sqrt{n_{\min}}$, reaching at most 0.0973% of it. In 16 fits, the envelope exceeds the deterministic gain bound G. Minimum support rises from half to full retention in nine of ten collections, but the results do not establish a regret-decay rate. Two additional subsets leave a fixed context unsupported and therefore fail the coverage hypothesis; no policy is fitted for those subsets.

The roughly thousandfold slack measures conservativeness relative to the estimated regret. It does not establish that a comparable reduction in training data is safe. Context maps were fitted using the full training data before being frozen for subsampling. Distinct episodes are not certified independent, held-out regret is estimated, and the fitted subsets are nested and dependent. We therefore report numerical compatibility with the reference envelope, without a confidence interval on the exceedance rate or a claim that the population failure probability is below δ.

H.1 Sensitivity to Concealed Inspection Evidence

Purpose and setup. Inspection helps only when the evidence it reveals supports a useful follow-up decision. We test this dependence by concealing part of an inspection response while preserving the underlying attack and its outcomes. The study uses six accepted cached replay corpora, each containing 3,200 branch sets with separate training, validation, and test actors. Together, their held-out partitions contain 4,800 branch sets, including 803 attack branches.

We fix the benign-cost budget at $\alpha = 20$ FI/1k, inspection fee at $c_{\text{ins}} = 0.02$, benign penalty at $c_{\text{benign}} = 0.5$, median aggregation at $\eta = 0$, and source-context mapping at (j_D, κ_t, g_t). Each seed uses its own training data and validation-selected cutoffs. The resulting evidence models, calibration, and learned policies remain fixed across the original, random-concealment, and targeted-concealment conditions. Defender random draws are shared across conditions within each seed. ORBIT-noinspect and DBN thresholding provide controls that do not consume inspection responses.

Evidence concealment. An evidence link is a recorded parent–child connection between events, such as a connection revealing how an earlier event influenced a later one. A link is eligible for concealment only if inspection reveals it, the corresponding ordinary observation does not, and it is available before the follow-up decision. When the defender requests inspection on an attack branch, the attacker hides one eligible link. Responses without an eligible link remain unchanged.

Under random concealment we select an eligible link uniformly. Targeted concealment knows the frozen defender and its current belief; it evaluates the eligible single-link removals and selects the one that minimizes post-inspection blocking, breaking ties by the lower updated deviation belief. This selection uses the current response and wont optimize against future harm outcomes or future random draws.

The edited response is rescored using the frozen evidence models. Its belief update carries forward when inspection occurs, potentially affecting later decisions. Source evidence, ordinary observations, underlying events, and harm-boundary timestamps remain unchanged. Attack labels restrict which branches can be edited and are never supplied to the defender.

Results. Random concealment lowers mean early stopping by 6.08 percentage points (paired 95% CI $[1.00, 11.15]$). Targeted concealment lowers it by 10.56 points ($[3.04, 18.09]$) and raises harm by 8.31 points ($[1.65, 14.97]$). Targeted concealment also produces 4.49 points less early stopping than random concealment ($[1.31, 7.67]$). Early stopping follows the same ordering in every seed: original evidence, random concealment, then targeted concealment. Mean benign interruption changes little between the original and targeted conditions: 22.18 versus 22.38 FI/1k.

Under targeted concealment, mean early stopping is close to ORBIT-noinspect: 15.56% versus 15.46%. Their paired difference is 0.10 points ($[-2.94, 3.15]$), establishing neither superiority nor equivalence. Targeted ORBIT-postrule falls below ORBIT-noinspect in two seeds; inspection therefore does not provide an empirical performance floor under this corruption. Its early-stopping difference from DBN thresholding is also imprecise: -3.70 points ($[-11.27, 3.88]$).

Intervals use Student's t distribution over six paired seed differences and are unadjusted for multiple comparisons. The previously reported 37.1% original-evidence result is reproduced for its original seed; Table 14 reports means across the expanded six-seed study.

Interpretation and scope. At this operating point, targeted concealment largely removes the mean early-stopping advantage over the no-inspection control. This result therefore identifies a dependence on inspection-evidence integrity: requesting additional evidence can help, but a frozen policy can be misled when that evidence is selectively concealed. This study tests a specified corruption of simulated inspection responses. It does not establish that a live attacker can produce these exact telemetry changes, assess poisoning of training data, or establish robustness to other concealment strategies.

I A Worked Intervention Example

This example scores a decision in the final campaign, so no utility from later campaigns is added and $Q = q$. The numbers illustrate the calculation only.

Use Case. This example checks action values and obedience; it does not solve the budget-constrained training LP. A user asks an email agent to send an approved report internally. An injected email asks it to open a confidential attachment and send it outside the organization. Use persistent-compromise belief $C = .5505$, current-deviation belief $D = .4762$, and attack-progress score $\sigma = .9$. With $w_{\text{str}} = .5$, $\gamma_0 = .1$, and $\gamma_1 = .8$, Equation 12 gives $b^{\text{eff}} \approx .5138$. For this illustrative inspection model, let A denote an attack and treat the adjusted value $b = .5138$ as its prior probability.

I.1 Inspection purchase

Let missed harm cost .9, benign blocking cost .1, inspection cost .02, and inspection-delay harm cost .02. Immediate values are

$$Q_{\text{ALLOW}} = -.9b = -.462420, \qquad Q_{\text{BLOCK}} = -.1(1-b) = -.048620.$$

Suppose inspection returns a suspicious result S with probability .95 during an attack and .05 during benign activity. Then

$$P(S) = .95b + .05(1-b) = .51242, \qquad P(A \mid S) = \frac{.95b}{.51242} \approx .95256.$$

For a clean result N,

$$P(N) = .48758, \qquad P(A \mid N) = \frac{.05b}{.48758} \approx .05269.$$

The post-inspection rule allows a clean result and blocks a suspicious one. A cutoff of .75 gives those same choices on these two outcomes. Its expected utility is the negative cost of missed clean attacks and blocked suspicious benign cases:

$$\mathbb{E}[\Xi^{\pi}] = -.9(.05b) - .1(.05(1-b)) = -.025552.$$

If harm occurs during inspection with probability .05, then

$$Q_{\text{INSPECT}} = -.02 - .02(.05) - .025552 = -.046552.$$

Inspection has the highest value, meaning the smallest expected loss. Its improvement before paying for inspection and delay is .023068. After those costs, its advantage over immediate blocking is .002068. If inspection completes before exfiltration, a suspicious result leads to INSPECT→BLOCK, preventing the confidential attachment from leaving the system.

I.2 CHECKING OBEDIENCE

Give all five monitors these same action values and recommend INSPECT to each. With $\eta = .1$, changing one report changes the executed action with probability $.1/5 = .02$. The two possible gains are

$$\Delta(\text{INSPECT} \to \text{BLOCK}) = .02(-.048620 + .046552) = -.00004136,$$
$$\Delta(\text{INSPECT} \to \text{ALLOW}) = .02(-.462420 + .046552) = -.00831736.$$

Both changes lower the head's expected score. Recommendations that are never issued contribute zero, so every possible changed-report constraint satisfies exact obedience in this one-context example. The comparison of action values establishes this result; unanimous recommendations by themselves would not. In the experiments, monitors have different preferences, and their gains are averaged across histories with the same private information.

An equally equipped direct three-action rule also chooses inspection here. The example shows how beliefs, attack stage, inspection, and incentive checks fit together.

J OPEN SCIENCE

Source code is available in the anonymous AgentDojo repository and the anonymous branch-replay repository. The former covers the AgentDojo campaign experiments; the latter covers branch replay, matched obedience, and response history. The repositories provide source code, configuration files, tests, and setup instructions. These source releases exclude cached recordings, fitted artifacts, and saved experiment outputs; reproducing the reported results requires those inputs. Live AgentDojo executions additionally require access to the model APIs.