When Tools Hurt LLM Reasoning: State-Dependent Belief Revision under External Evidence

Anonymous ACL submission

Abstract

Tool use is often assumed to monotonically improve reasoning: external evidence should help when relevant and be ignored when irrelevant. We show that this assumption fails in a state-dependent way. Across reasoning benchmarks with Python and Wikipedia tools, tools can repair weak initial beliefs but also destroy strong correct priors, flipping already-correct answers under noisy or misleading evidence. We explain this as misallocated revision authority: when internal support for the correct answer exceeds the expected support of the tool channel, deferring to external evidence is not risk-minimizing. This predicts that harm should concentrate on high-confidence no-tool cases. We test this prediction with threshold localization, wrong-trace audits, and a same-clue intervention showing that revision framing changes the damage caused by misleading evidence. These findings suggest that mixed no-tool/tool-assisted inference should arbitrate authority rather than privilege tool evidence by default. As a minimal demonstration, we introduce CASE, a label-free controller that selects between no-tool and tool-assisted trajectories using answer-state certainty and improves over existing confidence-based arbitration baselines.

1 Introduction

Tool-integrated reasoning has become a central paradigm for extending the problem-solving capabilities of language models, with code execution, retrieval, and search increasingly incorporated as standard reasoning components (Feng et al., 2025; Jin et al., 2025; Lin and Xu, 2025). This success has encouraged a largely monotonic view of tool access: useful evidence should improve reasoning, while irrelevant or unhelpful evidence should be ignored. However, a growing body of evidence shows that this view is incomplete. Models may overuse tools even when parametric reasoning is sufficient (Mallen et al., 2023; Qian et al., 2025); they can be distracted by irrelevant context or fail to robustly use retrieved information (Shi et al., 2023; Liu et al., 2024); and they can mis-arbitrate between internal knowledge and conflicting external evidence (Wu et al., 2024; Su et al., 2024).

We focus on a different question: how much *revision authority* should external evidence receive, conditional on the model's pre-tool belief state? We show that tool use can impair reasoning even when the tool returns a correct local result. In these cases, the model over-authorizes the external signal and revises away from an already-correct, high-confidence answer. This failure is not uniform; it concentrates in strong-prior cases, where the model should preserve its internal belief rather than revise toward a tool-framed interpretation.

A representative stored trace illustrates this failure. On a SuperGPQA item asking for $\binom{85}{82}$, the no-tool state assigns $p_c = 0.99$ to the correct combined option, recognizing that both 98770 and $C(85, 3)$ are valid. Python correctly returns `C(85,82)=98770`, `C(85,3)=98770`, and equality `True`. Nevertheless, the tool-assisted trajectory revises to the numeric-only option D, arguing that the single numerical value is more direct; its final probe assigns probability 0.99 to this wrong option. The computation is correct and useful, but the answer is damaged by over-authorizing a revised interpretation of the answer space.

Across four challenging reasoning benchmarks, Figure 1 reveals a systematic prior-dependent revision profile. Tools help when the model's pre-tool belief in the correct answer is weak, but this benefit reverses when the belief is already strong: average updates become negative, harmful update mass approaches helpful update mass, and correct-prior crossings exceed a geometry-preserving logit-shuffle baseline. Tool use is therefore better understood as a belief-revision intervention whose value depends on the model's current belief state, rather than as a uniformly beneficial add-on.

We study this phenomenon as *state-dependent belief revision*. The central failure is excessive revision authority assigned to external evidence. Under a minimal-change revision objective, strong priors should be protected unless the external channel provides greater expected support. This yields a threshold principle: when internal log-odds already dominate the expected support from the tool channel, the risk-minimizing external authority is zero.

We test this principle using paired no-tool and tool-assisted reasoning traces. The evidence proceeds in three steps. First, harmful flips concentrate in the high-prior regime, where internal-belief proxies exceed contextual-support proxies. Second, a wrong-only audit shows why this failure is difficult to detect from surface trajectory quality: many failed tool-assisted traces remain superficially supportive and highly confident. Third, the Prior-Authority Interaction Test (PAIT), a same-clue, different-frame intervention, shows that revision posture modulates the damage, while an antidote frame substantially attenuates it.

This mechanism has direct implications for deployment. When no-tool and tool-assisted trajectories are combined, equal voting can overweight the unreliable channel because it counts trajectories rather than comparing post-revision authority. We therefore formulate mixed inference as an authority-allocation problem and instantiate this view with Conflict-Aware Asymmetric Expansion (CASE), a label-free controller that uses answer-state certainty to arbitrate between revised and unrevised channels. CASE consistently improves over naive mixing in both the one-of-each setting and higher-budget expansion. Our contributions are as follows:

- We identify and formalize a **state-dependent belief-revision failure**: tool use can repair weak initial beliefs but may over-revise already-correct, high-confidence beliefs. The resulting threshold principle characterizes when external evidence should receive zero revision authority.
- We support this mechanism with **three linked tests**: cross-model threshold localization of harmful flips, wrong-trace audits revealing confident and superficially supportive failures, and PAIT, a same-clue intervention showing that revision framing modulates damage.
- We instantiate the deployment implication with **CASE**, a label-free controller for mixed no-tool/tool-assisted inference that allocates authority between revised and unrevised trajectories and improves over naive mixing under low-budget settings.

2 A Threshold Law for Revision Authority

The prior-dependent reversal in Figure 1 changes the question from whether tools help on average to when external evidence should be allowed to revise an existing belief. We formalize this question with a minimal belief-revision model that assigns authority to the tool-conditioned signal relative to the model's pre-tool belief. The model yields a simple protection principle: strong internal beliefs should be preserved unless the external channel provides stronger expected support, and turns that principle into testable predictions. Full derivations and the iterated-destruction lemma used in the wrong-trace audit appear in Appendix C.

2.1 Setup and Log-Odds Identity

Given a question x and answer space $\mathcal{A}$, we distinguish three objects: the LLM's pre-tool belief P_x, the effective tool-conditioned belief Q_x, and the authority λ_x assigned to the external signal. Here, $P_x, Q_x \in \Delta(\mathcal{A})$, and $\lambda_x \in [0, 1]$ controls how far the revised belief moves from P_x toward Q_x. The reverse-KL minimal-change objective admits a unique logarithmic-pooling solution:

$$R_x^\lambda(a) \propto P_x(a)^{1-\lambda_x} Q_x(a)^{\lambda_x}, \qquad a \in \mathcal{A}. \quad (1)$$

In controlled-output experiments, these beliefs are estimated by projecting answer-probe probabilities onto $\mathcal{A}$.

Collapsing the answer space to the binary event $y^\star$ versus non-gold, let $p = P_x(y^\star)$, $q = Q_x(y^\star)$, and $r = R_x^\lambda(y^\star)$. Denote the corresponding log-odds by $\eta = \mathrm{logit}(p)$, $\nu = \mathrm{logit}(q)$, and $\rho = \mathrm{logit}(r)$. Equation (1) implies

$$\rho - \eta = \lambda_x(\nu - \eta). \quad (2)$$

Thus, revision is additive in log-odds. Any positive authority decreases the gold log-odds when the tool-conditioned belief is weaker than the internal belief ($\nu < \eta$), and increases it only when the external support is stronger ($\nu > \eta$).

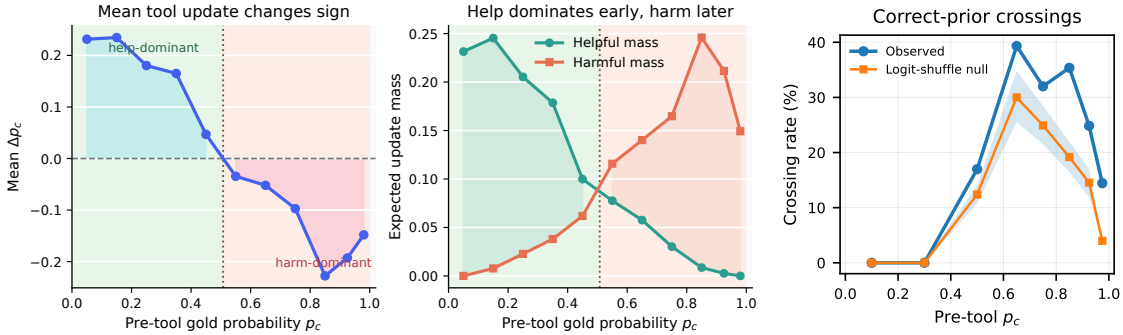

Figure 1: **State-dependent tool revision across four reasoning benchmarks.** Tool use helps weak priors but harms strong priors: mean updates change sign with p_c (left), helpful and harmful mass reverse dominance (middle), and correct-prior crossings exceed a logit-shuffle null (right). Additional per-dataset and log-odds diagnostics appear in Appendices A.3 and A.4.

2.2 Protection Principle

We now characterize when a risk-minimizing policy should grant positive authority to the external signal. For a question with pre-tool gold log-odds η, let M denote the random gold-event log-odds induced by the tool channel. Define the resulting gold-event logistic risk as

$$\mathcal{L}(\lambda;\eta) = \mathbb{E}_M[\log(1 + e^{-z_\lambda(M)})]$$

where $z_\lambda(M)$ linearly interpolates between the pre-tool and tool-induced gold log-odds.

Theorem 2.1 (State-dependent authority threshold)**.** *Assume that M is integrable and that all binary gold-event probabilities have finite log-odds. Then $\mathcal{L}(\lambda;\eta)$ is convex in λ on $[0,1]$, and its initial slope is*

$$\left.\frac{\partial \mathcal{L}}{\partial \lambda}\right|_{\lambda=0} = -(1-p)\big(\mathbb{E}[M] - \eta\big). \qquad (3)$$

Therefore, if $\eta \geq \mathbb{E}[M]$, any positive authority increases risk, and $\lambda^\star = 0$ is globally optimal. Conversely, if $\eta < \mathbb{E}[M]$, the initial slope is negative, so a sufficiently small positive authority strictly reduces risk.

Theorem 2.1 establishes a protection principle: when the internal belief already dominates the expected support from the external channel, the minimal-risk policy assigns zero revision authority. Positive external influence is justified only when expected contextual support exceeds the prior.

Empirical Predictions. The theory yields three predictions that organize the main empirical tests. *P1:* harm should localize to cases where an internal-belief proxy exceeds a contextual-support proxy. *P2:* harmful revised traces may remain confident, because repeated harmful revision can converge to a sharp but incorrect endpoint (Appendix C.4). *P3:* holding a misleading clue fixed, the revision posture should modulate damage by changing the effective authority assigned to that clue. Section 3 tests these predictions directly.

3 Three Tests of Misallocated Revision Authority

We next examine whether the threshold mechanism is observable in trace data. Because the latent beliefs in the theory are not directly observed, we test their observable signatures using answer-probe proxies, wrong-trace audits, and controlled revision wrappers, with the concrete measurement mapping reported in Table A.1. Throughout this section, NT denotes an unrevised no-tool trajectory, and AT denotes a tool-assisted trajectory after external evidence has been incorporated.

Together, the experiments test three increasingly diagnostic predictions of the theory: harmful flips should localize where internal-belief proxies exceed contextual-support proxies; these failures should often end in confident wrong answers rather than visible uncertainty; holding the same misleading clue fixed, changing the revision frame should change the flip rate by altering the effective authority granted to that clue.

3.1 Threshold Signature of Harmful Revision

This test asks whether tool-induced harm appears precisely in the regime predicted by the protection principle. The key regime is the *positive-gap* region: cases where the model's pre-tool belief in the gold answer is stronger than the support provided by the tool-conditioned context ($\eta > \hat{\nu}$). In this

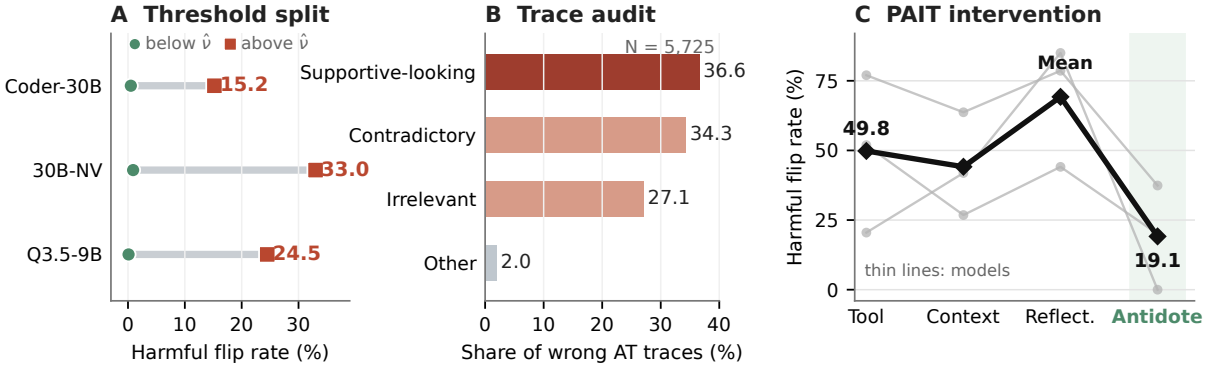

Figure 2: **Diagnostics for misallocated revision authority. A:** Harmful flips rise sharply in the high-authority region ($\eta > \hat{\nu}$) across three models. **B:** Many wrong tool-assisted traces remain superficially plausible, with supportive-looking traces forming the largest category. **C:** In PAIT, holding the misleading clue fixed while limiting tool authority reduces harmful flips; gray lines are individual models and the black line is their unweighted mean.

region, the theory predicts that granting authority to the external signal is risky, because revision is more likely to move probability mass away from the gold answer.

The trace data align with this prediction. In the main Qwen3-Coder-30B(Qwen Team, 2025) export, harmful flips are much more concentrated in the positive-gap region: 15.2% of these transitions change a correct pre-tool answer into an incorrect post-tool answer, compared with only 0.5% when the tool-conditioned support is at least as strong as the internal belief ($\eta \leq \hat{\nu}$). The average probability update also reverses direction, shifting from a gain of $+0.228$ in Δp_c outside the positive-gap region to a loss of -0.113 within it. Figure 2A visualizes the same split across all three evaluated model settings, while Table 1 gives the main-model log-odds and null-crossing diagnostics, with details given in Table A.6.

A natural concern is that this effect merely reflects starting confidence: because positive-gap examples begin with high gold probability, their downward updates may produce more boundary crossings simply due to regression to the mean. To address this alternative, we introduce a permutation-style null model (Ernst, 2004; Good, 2005). Specifically, we keep each example's pre-tool log-odds fixed and randomly reassign the observed logit updates across the dataset. This procedure preserves the marginal distributions of both starting confidence and update magnitude, while severing the case-level link between the threshold gap and the direction of revision.

We observe substantially more crossings than predicted by this null model, especially in the positive-gap and high-prior subsets. This excess

Table 1: **Transition diagnostics for the main Qwen3-Coder-30B threshold risk sets.** $\Delta\ell$ is the mean change in gold log-odds. Up/Down report logit-movement frequencies, with $|\Delta\ell| < 0.05$ treated as flat. Cross obs/null reports the observed rate of $p_c > 0.5 \rightarrow p_c < 0.5$ crossings and the corresponding logit-shuffle null mean.

Risk set	N	Δp_c (pp)	$\Delta\ell$	Up/Down (%)	Harm (%)	Cross obs/null (%)
$\eta \leq \hat{\nu}$	70,591	+22.7	+4.74	69.0 / 25.5	0.5	0.4 / 0.2
$\eta > \hat{\nu}$	13,756	-11.3	-1.13	44.9 / 53.3	15.2	15.3 / 6.1
$p_c \geq 0.80$	11,521	-15.5	-1.50	43.8 / 54.6	15.9	16.3 / 5.5

indicates that positive-gap cases are not merely high-confidence examples with more room to fall; rather, they are disproportionately susceptible to misaligned revision.

3.2 Failed Revised Traces Often Appear Superficially Reliable

Section 3.1 localizes where harmful revision occurs. We next ask a complementary question: when a tool-assisted trajectory ends with an incorrect answer, is the failure visible from the trace itself? We audit *AT failures*, i.e., tool-assisted trajectories whose final answer is wrong. This pool includes harmful flips as a strict subset, but the diagnostic target is broader: whether failed revised answer states can be rejected from a single trace. Confidence denotes the final answer-probe probability assigned to the trajectory's own chosen answer; we call a wrong trace high-confidence when this value is at least $\tau_{\text{conf}} = 0.8$.

We first annotate each wrong AT trace with an evidence-status label using GPT-5(OpenAI, 2025): supportive, contradictory, irrelevant, or other/unclassifiable. Figure 2B summarizes the audit.

First, 36.6% of wrong AT traces are supportive-looking: the available evidence appears to support the model's final answer, even though that answer is wrong; the remaining 2.0% are labelled other/unclassifiable. Second, non-supportive evidence does not reliably induce uncertainty: among wrong traces labeled contradictory or irrelevant, 98.2% still assign at least $\tau_{\text{conf}} = 0.8$ probability to their own wrong final answer. Thus, failed revised traces are not reliably exposed by surface confidence or apparent evidence quality alone.

A natural alternative explanation is that these failures are simply bad tool outputs propagated forward. To examine this possibility, we conduct a model-stratified human source audit on 240 wrong AT traces, with 80 traces from each evaluated model. The audit reveals a mixed failure profile: tool-contradictory evidence absorbed anyway is the largest category (34.6%), followed by irrelevant evidence confabulated into support (24.6%), partial evidence taken at face value (15.4%), and correct evidence misinterpreted in the answer space (15.0%), with 10.4% labeled unclear. The last category matches the $\binom{85}{82}$ trajectory in the introduction: Python returns the correct value, but the post-tool trajectory revises to the wrong option. Many failures are therefore not artifacts of incorrect tool outputs. Instead, they are post-tool revision failures in which correct, partial, irrelevant, or contradictory evidence can still be folded into a confident wrong state.

Together, the automatic and human audits clarify why threshold localization is not itself a trace-level filter. Although the risky region can be flagged statistically, an individual failed revised trace may still appear supported and highly confident. Apparent evidential support and high final answer-probe confidence are therefore not reliable standalone certificates of a revised trajectory's correctness.

3.3 Prior-Authority Interaction Test (PAIT)

The first two diagnostics establish where harmful revision concentrates and why failed revised traces can look reliable after the fact. We now test the remaining theoretical claim: in high-prior cases, damage should depend not only on the content of the external clue, but also on the authority the model assigns to that clue. In the threshold model, this role is captured by λ: when the internal belief is already strong, a weak or misleading external signal should receive little or no revision authority. PAIT tests whether this authority is frame-sensitive.

Table 2: **Cross-model PAIT on eligible high-prior cases.** FR denotes the answer flip rate under the adversarial-clue wrapper.

Model	Tool	Context	Reflection	Tool+Antidote
QC-30B	77.0%	63.7%	78.6%	37.4%
30B-NV	51.9%	26.8%	44.1%	20.0%
Q3.5-9B	20.5%	41.9%	85.0%	0.0%
Avg	49.8%	44.1%	69.2%	19.1%

Starting from harmful high-prior cases whose clean executed baseline remains correct and highly confident, we extract one misleading clue and reinsert the same clue into several non-executing wrapper prompts: TOOL, CONTEXT, REFLECTION, and TOOL+ANTIDOTE. The clue text is held fixed within each case, and all wrapper answers disable tool execution. Thus, differences in flip rate cannot be attributed to new tool results or different evidence content; instead, they measure how the same clue is treated under different revision postures. If the harm were driven only by clue content, the frames should behave similarly. If misallocated authority is part of the mechanism, authority-amplifying frames should induce more flips, while an authority-limiting antidote should reduce them.

Table 2 summarizes the cross-model PAIT pattern using unweighted model-level averages; the corresponding per-condition eligible counts are reported in Appendix Table E.4.

The qualitative pattern is stable. Under the same adversarial-clue intervention, multiple authority-coded frames induce substantial harmful revision: averaged across model settings, REFLECTION produces the highest flip rate (69.2%), followed by TOOL (49.8%) and CONTEXT (44.1%). At the same time, the revision frame matters: TOOL+ANTIDOTE yields the lowest average flip rate (19.1%), suggesting that explicitly limiting the authority of the clue can partially restore protection for high-prior answers.

4 Deployment Corollary: Authority Allocation for AT/NT Inference

The mechanism developed in Sections 2–3 leads to an inference-time allocation problem. Tool use is valuable because external evidence can rescue weak or mistaken internal states; however, the same revision channel can also overwrite strong priors and produce confident wrong AT endpoints. Once unrevised no-tool trajectories (NT) and revised tool-

assisted trajectories (AT) coexist for the same question, the system must decide how much authority each answer state should receive.

This motivates a query-level arbitration view of mixed AT/NT inference. Rather than counting trajectories uniformly, the controller should compare the competing post-revision answer states and allocate additional samples to the channel whose support is weaker or less settled. We instantiate this idea with CASE, a minimal closed-loop controller designed to test the operational prediction of the authority-allocation view: when AT and NT disagree, query-level arbitration should outperform channel-agnostic mixing.

4.1 A Label-Free Signal from Settled Answer States

The threshold gap $\eta - \mathbb{E}[M]$ relies on leave-one-out tool-conditioned proxies that are not computable per query at deployment time. CASE therefore uses an inference-time analog: the settledness of the resulting answer state from a single trajectory. For each trajectory i, we obtain a final answer-probe distribution $p_i(a)$ over options and compute its entropy

$$H_f^{(i)} = -\sum_{a \in \mathcal{A}} p_i(a) \log p_i(a).$$

Lower entropy indicates that the trajectory ends in a more concentrated answer state. We convert this quantity into normalized certainty $c_i = 1 - \frac{H_f^{(i)}}{\log |\mathcal{A}|}$ so that $c_i \in [0, 1]$ up to numerical roundoff.

4.2 The CASE Controller

We aggregate answers through certainty-weighted voting. To decide which channel to expand next, CASE also tracks channel-specific winners and margins. For channel $s \in \{\mathrm{AT}, \mathrm{NT}\}$, let $\mathcal{T}_s$ be the current trajectory pool and define

$$\begin{aligned} V_s(a) &= \sum_{i \in \mathcal{T}_s : y_i = a} c_i, \\ \hat{y}_s &= \arg\max_{a \in \mathcal{A}} V_s(a), \\ \Gamma_s &= V_s(\hat{y}_s) - \max_{b \neq \hat{y}_s} V_s(b). \end{aligned} \tag{4}$$

Let $\Gamma = V(\hat{y}) - \max_{b \neq \hat{y}} V(b)$ be the corresponding pooled margin. We instantiate these quantities in CASE (*conflict-aware asymmetric expansion*), a minimal closed-loop controller for mixed AT/NT inference.

Algorithm 1 CASE: conflict-aware asymmetric expansion

Require: Threshold τ; channel budgets $B_{\mathrm{NT}}, B_{\mathrm{AT}}$
Collect one NT trajectory and one AT trajectory
Initialize $\mathcal{T}_{\mathrm{NT}}, \mathcal{T}_{\mathrm{AT}}$ with those samples
while true **do**
  Compute certainty scores c_i, pooled winner $\hat{y}$, and pooled margin Γ
  Compute channel winners $\hat{y}_{\mathrm{NT}}, \hat{y}_{\mathrm{AT}}$ and channel margins $\Gamma_{\mathrm{NT}}, \Gamma_{\mathrm{AT}}$
  if $\Gamma \geq \tau$ and $\hat{y}_{\mathrm{NT}} = \hat{y}_{\mathrm{AT}}$ **then**
    return $\hat{y}$
  else if $\hat{y}_{\mathrm{NT}} \neq \hat{y}_{\mathrm{AT}}$ **then**
    $s^\star \leftarrow \arg\min_{s \in \{\mathrm{NT}, \mathrm{AT}\}} V(\hat{y}_s)$
  else
    $s^\star \leftarrow \arg\min_{s \in \{\mathrm{NT}, \mathrm{AT}\}} \Gamma_s$
  end if
  if $|\mathcal{T}_{s^\star}| = B_{s^\star}$ **then**
    return $\hat{y}$
  end if
  Collect one additional trajectory from channel $s^\star$ and add it to $\mathcal{T}_{s^\star}$
end while

The two branches of Algorithm 1 correspond to the two findings of Section 3. The *conflict branch* handles AT/NT disagreement, the regime in which the threshold gap from Section 3.1 is most diagnostic: the controller probes whichever side is currently losing the pooled decision rather than reinforcing the stronger one. The *alignment branch* handles cases where both channels agree but the pooled margin remains small—the regime where Section 3.2 shows that surface agreement does not certify correctness—and samples from the channel with the lower margin. Setting $B_{\mathrm{NT}} = B_{\mathrm{AT}} = 1$ yields the minimal one-NT/one-AT form, corresponding to the first decision made by the same closed loop and the setting used for our headline result.

4.3 Arbitration on the AT/NT Disagreement Set

The mechanism in Section 3 yields a direct operational prediction. When the first AT and NT trajectories choose different answers, equal voting is uninformative by construction; any improvement must therefore come from deciding which post-revision state should be granted authority. We begin with the first-pair AT/NT disagreement subset, which provides the cleanest test of the authority-allocation view.

Table 4 reports this test. On the disagreement subset, CASE improves over P(True)-CISC by +10.2, +10.0, and +13.1 points across the three model settings.

Table 3: **Cross-model results on the shared four-benchmark evaluation.** Accuracy is measured in the 1+1 setting with one NT and one AT trajectory per question. Baselines are mixed majority vote with random tie-breaking and P(True)-CISC on the same candidate files. The final row reports matched-budget Pareto AUC over 1+1 to 8+8; Δ_C denotes CASE–P(True)-CISC.

Dataset / summary	Qwen3-Coder-30B				30B-NV				Qwen3.5-9B			
	Mix-MV	CISC	CASE	Δ_C	Mix-MV	CISC	CASE	Δ_C	Mix-MV	CISC	CASE	Δ_C
MMLU-Pro	73.95	73.91	78.29	+4.38	82.76	82.93	84.31	+1.38	60.71	56.65	62.90	+6.25
GPQA Diamond	48.21	51.79	49.74	-2.05	66.16	65.66	67.17	+1.51	62.88	65.15	63.64	-1.51
Physics	48.46	50.31	50.62	+0.31	69.08	69.54	69.85	+0.31	58.31	56.92	58.15	+1.23
SuperGPQA-Calc	42.92	42.38	46.44	+4.06	56.91	57.82	59.96	+2.14	34.38	31.43	36.35	+4.92
Macro	53.38	54.60	56.27	+1.67	68.73	68.99	70.32	+1.33	54.07	52.54	55.26	+2.72
Aggregate	53.55	53.27	57.30	+4.03	65.97	66.62	68.46	+1.84	44.04	40.80	46.03	+5.23
Pareto AUC	57.48	57.31	59.69	+2.38	68.94	69.12	70.63	+1.51	46.49	44.96	48.17	+3.21

Table 4: **First-pair AT/NT disagreement decomposition.** Accuracy is measured only on examples where the first NT and AT answers disagree. False protection is the complement of rescue on the NT-wrong/AT-correct subset.

Model	N	CASE	CISC	Prot.	Rescue	False
Qwen3-Coder	6,975	37.7	27.5	81.3	51.1	48.9
30B-NV	3,897	39.2	29.2	71.2	70.5	29.5
Qwen3.5-9B	6,905	34.0	20.9	60.4	57.4	42.6

The decomposition explains the source of the gain. CASE does not simply default to either the no-tool channel or the tool-assisted channel: it protects 60–81% of cases where NT is correct and AT is wrong, while still rescuing 51–71% of cases where only AT is correct. This pattern is consistent with the threshold law, indicating that the gains arise from state-dependent authority allocation rather than from a uniform preference for either the tool-assisted or no-tool channel.

Figure 3 examines whether the same arbitration advantage persists beyond the minimal-pair setting. Across the 1+1 to 8+8 budget ladder, CASE remains on a better budget frontier than mixed majority voting and CISC. Detailed cost tables, prior-gate baselines, and nearby answer-state signals are reported in Appendix B.

To further test whether CASE's advantage concentrates in the regimes predicted by the threshold analysis, we stratify the disagreement set by $\eta > \hat{\nu}$ versus $\eta \leq \hat{\nu}$ and by prior strength. Averaged across model settings, the CASE–CISC gap is larger in the positive-gap region than outside it (+13.5 vs. +7.7), and it reaches +20.8 points in the high-prior bin; Appendix Table B.3 reports the corresponding stratified summaries. In that high-prior bin, the gain is not explained by a degenerate rule that always trusts NT: CASE protects 86.5% of NT-correct/AT-wrong cases while still rescuing 72.5% of NT-wrong/AT-correct cases. This stratification closes the loop between the mechanism and the controller. CASE does not observe the latent threshold gap, but its advantage concentrates precisely in the regimes where the threshold analysis predicts over-revision.

4.4 Summary

Section 4 shows that the belief-revision mechanism has a direct operational consequence. Since the latent threshold gap is unavailable at inference time, CASE uses answer-state settledness as a label-free authority cue between competing AT and NT trajectories. Its gains concentrate in the disagreement and high-prior regimes predicted by the earlier analysis, while retaining both protection and rescue across minimal setting and higher budgets. This supports the main reframing of mixed tool/no-tool inference as state-dependent authority allocation rather than uniform trajectory counting.

5 Related Work

External evidence, tool use, and belief revision. Tool-integrated reasoning extends language models beyond parametric computation through code execution, retrieval, and search, increasingly trained end-to-end with reinforcement learning and standardized evaluation (Yao et al., 2023; Schick et al., 2023; Paranjape et al., 2023; Wang et al., 2024; Zhuang et al., 2023; Feng et al., 2025; Jin et al., 2025; Lin and Xu, 2025). A growing body of work shows that external evidence is not uniformly beneficial: irrelevant or distracting context degrades accuracy, long contexts are poorly utilized, retrieval-augmented systems over-adopt incorrect evidence, and agents may overuse tools even when paramet-

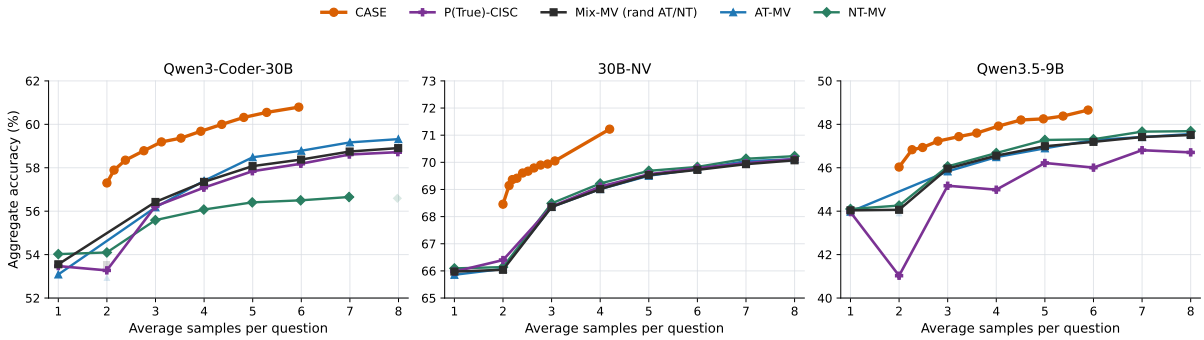

Figure 3: **CASE versus majority-vote baselines on the shared four-benchmark evaluation.** Each panel corresponds to one model setting. The x-axis reports sampled trajectories per question rather than token-normalized cost. MIX-MV samples AT/NT trajectories uniformly and applies equal-weight majority voting with uniform tie expectation. Solid lines are Pareto frontiers; faint markers show all evaluated operating points. CASE instead uses the authority signal to allocate AT versus NT samples.

ric reasoning suffices (Shi et al., 2023; Liu et al., 2024; Yoran et al., 2024; Mallen et al., 2023; Wu et al., 2024; Qian et al., 2025). A parallel literature on knowledge conflict characterizes the tug-of-war between parametric memory and external context—disentangling parametric from contextual answers, surfacing entity- and context-faithfulness conflicts, quantifying internal–external trade-offs, and cataloguing conflict types at scale (Neeman et al., 2023; Longpre et al., 2021; Zhou et al., 2023; Wu et al., 2024; Xie et al., 2024; Su et al., 2024; Kortukov et al., 2024; Xu et al., 2024). Mitigation strategies span source allocation, conflict-specific decoding, and context-robust training objectives (Tao et al., 2024; Jin et al., 2024a,b; Wang et al., 2025; Zeng et al., 2025; Khandelwal et al., 2025). We build on this literature but shift the unit of analysis from retrieved passages or tool-use decisions to tool-returned reasoning traces, asking when a tool-conditioned state should receive zero revision authority because the pre-tool belief is already stronger.

Aggregation and adaptive inference. Our deployment result connects to test-time aggregation and adaptive computation. Chain-of-thought prompting and self-consistency aggregate sampled reasoning paths, while confidence-weighted variants leverage calibration, self-knowledge, semantic uncertainty, and self-checking signals to score or filter candidates (Wei et al., 2022; Kojima et al., 2022; Wang et al., 2023; Taubenfeld et al., 2025; Guo et al., 2017; Jiang et al., 2021; Kadavath et al., 2022; Farquhar et al., 2024; Manakul et al., 2023; Tan et al., 2025). Verifier and process-supervision methods offer a complementary route to selecting or training reasoning paths via outcome- and step-level reward models (Cobbe et al., 2021; Uesato et al., 2022; Lightman et al., 2024; Zelikman et al., 2022). Adaptive systems further decide when to retrieve, decode, or route computation across models and contexts (Asai et al., 2024; Khandelwal et al., 2025; Chen et al., 2023). CASE is complementary to these directions: rather than training a verifier or routing among models, it tests the operational implication of our mechanism by allocating authority between unrevised no-tool and revised tool-assisted trajectories within a fixed mixed budget.

6 Conclusion

Our results show that tool access changes not only the information available to a model, but also the authority external evidence holds over an existing belief. Across trace diagnostics, controlled interventions, and arbitration experiments, tool-induced harm is consistently state dependent. The same revision channel that rescues weak beliefs can overwrite strong correct ones, and the damage concentrates when internal support already exceeds the expected support of the tool channel. Therefore, tool use be understood as a belief-revision intervention whose value depends on the model's current answer state, rather than unconditional capability. CASE operationalizes this view at inference time by protecting states in which external evidence is predicted to be more harmful than helpful. More broadly, it suggests a mechanism-level objective for tool learning, where models learn to allocate revision authority according to the relative support of internal and tool-conditioned evidence, rather than tool reliability alone.

Limitations

Our study has several limitations. The empirical evidence is concentrated on a bounded set of model families, benchmarks, and tool-use formats. Although we evaluate multiple model settings and four reasoning benchmarks, broader validation on larger frontier-scale models and more diverse architectures remains necessary. The current experiments focus primarily on controlled tool-assisted reasoning traces, where the external channel and answer space can be probed and compared. More complex tool-use settings, such as coding agents, web-search agents, retrieval-augmented multi-hop systems, and long-horizon autonomous agents, may introduce additional sources of noise, delayed feedback, and compounding state changes. Whether the same threshold behavior appears in these richer environments, and how to design a corresponding CASE-style controller for them, remains an important direction for future work.

For deployment of the core finding, CASE should be interpreted as a minimal operational corollary rather than a complete deployment solution. We do not yet show how the protection principle should be incorporated into training. A natural next step is to use the state-dependent authority view to design supervised fine-tuning data, preference signals, or reinforcement-learning rewards that teach models when to preserve a strong internal belief, when to defer to external evidence, and when to request additional tool interaction. Integrating such objectives with modern supervised fine-tuning and reinforcement-learning pipelines for tool-integrated reasoning and LLM agents remains open.

Ethics Statement

The experiments use benchmark reasoning tasks and stored model/tool traces rather than human-subject data. The main risk raised by the paper is operational: tool-augmented systems can become confidently wrong when misleading or low-quality external evidence is granted too much revision authority. We therefore frame the proposed controller as a safety-motivated diagnostic and mitigation direction. Deployed systems should combine state-dependent authority allocation with independent tool-quality checks and provenance tracking; high-stakes domains also require task-specific human oversight. Experts involved in the annotation process were compensated in accordance with applicable regulations and institutional guidelines.

References

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. 2024. Self-RAG: Learning to retrieve, generate, and critique through self-reflection. In *The Twelfth International Conference on Learning Representations*.

Lingjiao Chen, Matei Zaharia, and James Zou. 2023. FrugalGPT: How to use large language models while reducing cost and improving performance. *arXiv preprint arXiv:2305.05176*.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Michael D. Ernst. 2004. Permutation methods: A basis for exact inference. *Statistical Science*, 19(4):676–685.

Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn, and Yarin Gal. 2024. Detecting hallucinations in large language models using semantic entropy. *Nature*, 630:625–630.

Jiazhan Feng, Shijue Huang, Xingwei Qu, Ge Zhang, Yujia Qin, Baoquan Zhong, Chengquan Jiang, Jinxin Chi, and Wanjun Zhong. 2025. ReTool: Reinforcement learning for strategic tool use in LLMs. *Preprint*, arXiv:2504.11536.

Phillip I. Good. 2005. *Permutation, Parametric, and Bootstrap Tests of Hypotheses*, 3 edition. Springer, New York.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. 2017. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 1321–1330. PMLR.

Zhengbao Jiang, Jun Araki, Haibo Ding, and Graham Neubig. 2021. How can we know when language models know? on the calibration of language models for question answering. *Transactions of the Association for Computational Linguistics*, 9:962–977.

Bowen Jin, Hansi Zeng, Zhenrui Yue, Jinsung Yoon, Sercan Arik, Dong Wang, Hamed Zamani, and Jiawei Han. 2025. Search-R1: Training LLMs to reason and leverage search engines with reinforcement learning. *Preprint*, arXiv:2503.09516.

Zhuoran Jin, Pengfei Cao, Yubo Chen, Kang Liu, Xiaojian Jiang, Jiexin Xu, Qiuxia Li, and Jun Zhao. 2024a. Tug-of-war between knowledge: Exploring and resolving knowledge conflicts in retrieval-augmented language models. In *Proceedings of the 2024 Joint*

International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024), pages 16867–16878, Torino, Italia. ELRA and ICCL.

Zhuoran Jin, Pengfei Cao, Hongbang Yuan, Yubo Chen, Jiexin Xu, Huaijun Li, Xiaojian Jiang, Kang Liu, and Jun Zhao. 2024b. Cutting off the head ends the conflict: A mechanism for interpreting and mitigating knowledge conflicts in language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 1193–1215, Bangkok, Thailand. Association for Computational Linguistics.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, Scott Johnston, Sheer El-Showk, Andy Jones, Nelson Elhage, Tristan Hume, Anna Chen, Yuntao Bai, Sam Bowman, Stanislav Fort, and 17 others. 2022. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*.

Anant Khandelwal, Manish Gupta, and Puneet Agrawal. 2025. CoCoA: Confidence- and context-aware adaptive decoding for resolving knowledge conflicts in large language models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 6835–6855, Suzhou, China. Association for Computational Linguistics.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. In *Advances in Neural Information Processing Systems*, volume 35, pages 22199–22213.

Evgenii Kortukov, Alexander Rubinstein, Elisa Nguyen, and Seong Joon Oh. 2024. Studying large language model behaviors under context-memory conflicts with real documents. In *First Conference on Language Modeling*.

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2024. Let's verify step by step. In *The Twelfth International Conference on Learning Representations*.

Heng Lin and Zhongwen Xu. 2025. Understanding tool-integrated reasoning. *Preprint*, arXiv:2508.19201.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173.

Shayne Longpre, Kartik Perisetla, Anthony Chen, Nikhil Ramesh, Chris DuBois, and Sameer Singh. 2021. Entity-based knowledge conflicts in question answering. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7052–7063, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9802–9822, Toronto, Canada. Association for Computational Linguistics.

Potsawee Manakul, Adian Liusie, and Mark Gales. 2023. SelfCheckGPT: Zero-resource black-box hallucination detection for generative large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9004–9017, Singapore. Association for Computational Linguistics.

Ella Neeman, Roee Aharoni, Or Honovich, Leshem Choshen, Idan Szpektor, and Omri Abend. 2023. DisentQA: Disentangling parametric and contextual knowledge with counterfactual question answering. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10056–10070, Toronto, Canada. Association for Computational Linguistics.

NVIDIA. 2025. NVIDIA-Nemotron-3-Nano-30B-A3B-BF16. Hugging Face model card. Accessed 2026-05-22.

OpenAI. 2025. GPT-5 System Card. `https://openai.com/index/gpt-5-system-card/`. Accessed: 2026-05-25.

Bhargavi Paranjape, Scott Lundberg, Sameer Singh, Hannaneh Hajishirzi, Luke Zettlemoyer, and Marco Tulio Ribeiro. 2023. ART: Automatic multi-step reasoning and tool-use for large language models. *arXiv preprint arXiv:2303.09014*.

Cheng Qian, Emre Can Acikgoz, Hongru Wang, Xiusi Chen, Avirup Sil, Dilek Hakkani-Tür, Gokhan Tur, and Heng Ji. 2025. SMART: Self-aware agent for tool overuse mitigation. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 4604–4621, Vienna, Austria. Association for Computational Linguistics.

Qwen Team. 2025. Qwen3-Coder-30B-A3B-Instruct. Hugging Face model card. Accessed 2026-05-22.

Qwen Team. 2026. Qwen3.5-9B. Hugging Face model card. Accessed 2026-05-22.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. 2023. Toolformer: Language models can teach themselves to use tools. In *Advances in Neural Information Processing Systems*, volume 36, pages 68539–68551.

Freda Shi, Xinyun Chen, Kanishka Misra, Nathan Scales, David Dohan, Ed H. Chi, Nathanael Schärli, and Denny Zhou. 2023. Large language models can be easily distracted by irrelevant context. In *Proceedings of the 40th International Conference on Machine*

Learning, volume 202 of *Proceedings of Machine Learning Research*, pages 31210–31227. PMLR.

Zhaochen Su, Jun Zhang, Xiaoye Qu, Tong Zhu, Yanshu Li, Jiashuo Sun, Juntao Li, Min Zhang, and Yu Cheng. 2024. ConflictBank: A benchmark for evaluating the influence of knowledge conflicts in LLMs. In *Advances in Neural Information Processing Systems*, volume 37, pages 103242–103268.

Hexiang Tan, Fei Sun, Sha Liu, Du Su, Qi Cao, Xin Chen, Jingang Wang, Xunliang Cai, Yuanzhuo Wang, Huawei Shen, and Xueqi Cheng. 2025. Too consistent to detect: A study of self-consistent errors in LLMs. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 4755–4765, Suzhou, China. Association for Computational Linguistics.

Yufei Tao, Adam Hiatt, Erik Haake, Antonie J. Jetter, and Ameeta Agrawal. 2024. When context leads but parametric memory follows in large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 4034–4058, Miami, Florida, USA. Association for Computational Linguistics.

Amir Taubenfeld, Tom Sheffer, Eran Ofek, Amir Feder, Ariel Goldstein, Zorik Gekhman, and Gal Yona. 2025. Confidence improves self-consistency in LLMs. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 20090–20111. Association for Computational Linguistics.

Jonathan Uesato, Nate Kushman, Ramana Kumar, Francis Song, Noah Siegel, Lisa Wang, Antonia Creswell, Geoffrey Irving, and Irina Higgins. 2022. Solving math word problems with process- and outcome-based feedback. *arXiv preprint arXiv:2211.14275*.

Fei Wang, Xingchen Wan, Ruoxi Sun, Jiefeng Chen, and Sercan Ö. Arik. 2025. Astute RAG: Overcoming imperfect retrieval augmentation and knowledge conflicts for large language models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Vienna, Austria. Association for Computational Linguistics.

Xingyao Wang, Yangyi Chen, Lifan Yuan, Yizhe Zhang, Yunzhu Li, Hao Peng, and Heng Ji. 2024. Executable code actions elicit better LLM agents. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 50208–50232. PMLR.

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

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837.

Kevin Wu, Eric Wu, and James Zou. 2024. ClashEval: Quantifying the tug-of-war between an LLM's internal prior and external evidence. In *Advances in Neural Information Processing Systems 37*.

Jian Xie, Kai Zhang, Jiangjie Chen, Renze Lou, and Yu Su. 2024. Adaptive chameleon or stubborn sloth: Revealing the behavior of large language models in knowledge conflicts. In *The Twelfth International Conference on Learning Representations*.

Rongwu Xu, Zehan Qi, Zhijiang Guo, Cunxiang Wang, Hongru Wang, Yue Zhang, and Wei Xu. 2024. Knowledge conflicts for LLMs: A survey. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 8541–8565, Miami, Florida, USA. Association for Computational Linguistics.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. 2023. ReAct: Synergizing reasoning and acting in language models. In *The Eleventh International Conference on Learning Representations*.

Ori Yoran, Tomer Wolfson, Ori Ram, and Jonathan Berant. 2024. Making retrieval-augmented language models robust to irrelevant context. In *The Twelfth International Conference on Learning Representations*.

Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah D. Goodman. 2022. STaR: Bootstrapping reasoning with reasoning. In *Advances in Neural Information Processing Systems*, volume 35, pages 15476–15488.

Shenglai Zeng, Pengfei He, Kai Guo, Tianqi Zheng, Hanqing Lu, Yue Xing, and Hui Liu. 2025. Towards context-robust LLMs: A gated representation fine-tuning approach. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10262–10276, Vienna, Austria. Association for Computational Linguistics.

Wenxuan Zhou, Sheng Zhang, Hoifung Poon, and Muhao Chen. 2023. Context-faithful prompting for large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 14544–14556, Singapore. Association for Computational Linguistics.

Yuchen Zhuang, Yue Yu, Kuan Wang, Haotian Sun, and Chao Zhang. 2023. ToolQA: A dataset for LLM question answering with external tools. In *Advances in Neural Information Processing Systems*, volume 36, pages 50117–50143.

A Mechanism Measurements and Robustness

A.1 Theory-to-measurement bridge

For multiple-choice runs, pre-tool and post-tool states are reconstructed from the fixed answer-space probe attached to each stored trajectory. Thus $p_{c,\text{pre}}$, $p_{c,\text{post}}$, and $\Delta p_c = p_{c,\text{post}} - p_{c,\text{pre}}$ come from the same probe family rather than from free-form textual heuristics.

Table A.1: **Consolidated notation and empirical proxies.** The first rows are theory objects; hatted quantities are observed diagnostics used for localization and sign tests, not identified latent parameters.

Quantity	Definition or proxy	Role
P_x, Q_x, R_x^λ	Pre-tool, tool-conditioned, and revised beliefs over $\mathcal{A}$	Full answer-space states
p, q, r	$P_x(y^\star)$, $Q_x(y^\star)$, and $R_x^\lambda(y^\star)$ after collapsing to gold vs. non-gold	Binary gold-event probabilities
η, ν, ρ	$\text{logit}(p)$, $\text{logit}(q)$, and $\text{logit}(r)$	Binary log-odds coordinates
$M, \bar{\nu}$	Random tool-induced gold log-odds M and expected support $\bar{\nu} = \mathbb{E}[M]$	Threshold in Theorem 2.1
$\hat{q}_x, \hat{\nu}_x$	Leave-one-out matched post-tool gold probability $\hat{q}_x$ and $\hat{\nu}_x = \text{logit}(\text{clip}_\epsilon(\hat{q}_x))$	Empirical proxy for $\bar{\nu}$
$\lambda_x, \hat{\lambda}_x$	Latent revision authority and diagnostic recovered-authority ratio in Eq. (6)	Measures observed movement toward the proxy support
H_f, c_i	Final answer entropy and normalized certainty $c_i = 1 - H_f/\log\lvert\mathcal{A}\rvert$	Label-free CASE arbitration signal

For numerical stability, all logits use $\text{clip}_\epsilon(u) = \min(1-\epsilon, \max(\epsilon, u))$ with $\epsilon = 10^{-6}$. For an event x, let $\mathcal{S}_x$ be the matched stratum `(dataset, tool_call_idx, prior_bucket, pre_correct)`. The main contextual-support proxy is

$$\hat{q}_x = \frac{1}{|\mathcal{S}_x| - 1} \sum_{j \in \mathcal{S}_x,\, j \neq x} p_{c,\text{post}}^{(j)}, \qquad (5)$$
$$\hat{\nu}_x = \text{logit}(\text{clip}_\epsilon(\hat{q}_x)).$$

This definition is used for retained events with a nonempty leave-one-out stratum; robustness widens the strata and uses cross-fit or oracle-support variants. Let $\rho_x = \text{logit}(\text{clip}_\epsilon(p_{c,\text{post}}^{(x)}))$ and $\eta_x = \text{logit}(\text{clip}_\epsilon(p_{c,\text{pre}}^{(x)}))$. The recovered-authority diagnostic reported in Table A.6 is

$$\hat{\lambda}_x = \text{clip}_{[0,1]}\left(\frac{\rho_x - \eta_x}{\hat{\nu}_x - \eta_x}\right), \quad |\hat{\nu}_x - \eta_x| > 10^{-6}. \qquad (6)$$

with near-zero denominators treated as undefined and excluded from the diagnostic mean. Because effective authority is latent and $\hat{\nu}_x$ is only a proxy for expected channel support, $\hat{\lambda}_x$ is used only as a descriptive geometry check. The key empirical claim is sign stability of the belief-minus-support gap across proxy families.

A.2 Trace-level checks

Executed AT failure. On the SuperGPQA-Calc binomial item described in the introduction, the pre-tool probe assigns $p_c = 0.99995$ to the combined correct option. Python returns `C(85,82)=98770`, `C(85,3)=98770`, and equality `True`; the tool-assisted trajectory still answers the numeric-only option. This case illustrates that the external computation need not be wrong: harm can come from over-authorizing a revised interpretation of the answer space.

PAIT same-clue intervention. For the same item, the PAIT clue "$C(85, 3)$ equals 101170, not 98770" is held fixed. TOOL, CONTEXT, and REFLECTION wrappers all answer the distractor option, while TOOL+ANTIDOTE recomputes the value and preserves the combined answer. The change is the revision posture, not the clue content.

CASE arbitration. On an MMLU-Pro fraction item, the first AT path calls Python and observes `Result: 0.625` but answers a wrong option with a split final probe; the first NT path answers the gold option with near-zero entropy. Equal voting ties, while CASE(1+1) gives the larger authority weight to the more settled NT answer state.

A.3 Asymmetric belief destruction by dataset

This subsection records the compact numerical evidence behind the per-dataset visual diagnostics. The corresponding full-width figures are postponed to Section A.6 so that the appendix first presents the definitions, trace checks, and tables without a float backlog.

Let $\Delta p_c^+ = \max(\Delta p_c, 0)$ and $\Delta p_c^- = \min(\Delta p_c, 0)$. Across datasets, the expected magnitude of harmful updates grows faster than the expected magnitude of helpful ones as prior confidence increases:

$$\frac{\mathbb{E}[|\Delta p_c^-| \mid p_c]}{\mathbb{E}[\Delta p_c^+ \mid p_c]} = A \left(\frac{p_c}{1 - p_c} \right)^{\alpha}. \tag{7}$$

This descriptive relation is the precursor to the threshold analysis: it summarizes how the harm/help balance shifts with the model's current belief state. A positive exponent means that moving to higher pre-tool odds increases the relative size of destructive updates. Values near one indicate an approximately odds-proportional increase in destructive amplitude.

Table A.2: **Destructive-asymmetry fits.** The exponent α is positive across the four main benchmarks.

Dataset	α	R^2
MMLU-Pro	1.075	0.973
GPQA Diamond	0.960	0.989
Physics	0.997	0.995
SuperGPQA-Calc	1.092	0.975

Table A.2 shows that the fitted exponents are positive in all four datasets and tightly fit the binned ratios. The effect is therefore not a single-benchmark artifact: as the pre-tool gold probability increases, harmful updates become larger relative to helpful updates across MMLU-Pro, GPQA Diamond, Physics, and SuperGPQA-Calc.

Table A.3: **Cross-model portability summary.** Destructive-asymmetry exponents remain positive, and CASE(1+1) improves aggregate accuracy in all three evaluated settings.

Model	Events	$\bar{\alpha}$	$\bar{R}^2$	CASE Δ
Qwen3-Coder-30B	84,347	1.031	0.983	+3.8
30B-NV	149,147	0.646	0.925	+2.5
Qwen3.5-9B	71,028	1.047	0.986	+2.0

Table A.3 adds two portability checks. First, the destructive-asymmetry exponent remains positive for each evaluated model setting, although the slope is shallower for 30B-NV. Second, the same settings are those in which CASE improves one-NT/one-AT aggregate accuracy, linking the descriptive asymmetry to the deployment-time arbitration result.

A.4 Threshold-signature robustness

The main threshold split uses an effective support proxy. We therefore report the same split in log-odds coordinates and stress-test the sign of the gap under alternative proxy families. For numerical stability, log-odds are computed after clipping p_c to $[10^{-6}, 1 - 10^{-6}]$.

Table A.4: **Boundary-calibrated transition diagnostics.** Crossings are $p_c > 0.5 \rightarrow p_c < 0.5$; the null shuffles empirical logit updates across events.

Risk set	N	Δp_c (pp)	$\Delta \ell$	Down (%)	Harm (%)	Cross/null (%)
All events	84,347	+17.2	+3.78	30.1	2.9	2.8 / 1.2
Trimmed $0.05 \leq p_c \leq 0.95$	7,000	+7.1	+0.97	44.7	14.3	12.9 / 8.6
$p_c \geq 0.80$	11,521	-15.5	-1.50	54.6	15.9	16.3 / 5.5
$p_c \geq 0.95$	10,146	-14.8	-1.74	55.6	14.2	14.4 / 4.0
$\eta > \hat{\nu}$	13,756	-11.3	-1.13	53.3	15.2	15.3 / 6.1
$\eta \leq \hat{\nu}$	70,591	+22.7	+4.74	25.5	0.5	0.4 / 0.2

Table A.4 isolates the same pattern in several risk sets. The positive-gap slice $\eta > \hat{\nu}$ has negative mean probability and logit updates, a much higher downward-movement frequency, and an observed crossing rate that is well above the logit-shuffle null. The high-prior slices show the same qualitative pattern, which is the empirical signature predicted by the threshold law.

Table A.5: **Threshold signature across proxy choices.** Δharm is the high-gap minus low-gap harmful-flip rate in percentage points.

Belief	Support	N	Δharm	95% CI
Pre-tool p_c	Overlap, cross-fit	100,919	+11.1	[10.6, 11.5]
Pre-tool p_c	Overlap	100,919	+10.3	[9.9, 10.7]
NT mean p_c	Overlap, cross-fit	47,762	+10.2	[9.6, 10.9]
NT mean p_c	Overlap	47,762	+9.5	[8.9, 10.1]
NT mean p_c	Oracle support	44,128	+6.4	[6.0, 6.9]
Pre-tool p_c	Oracle support	93,016	+6.1	[5.8, 6.4]

Across all six proxy pairs in Table A.5, the positive-gap regime is more harmful. The harm penalty ranges from +6.1 to +11.1 points, and every bootstrap interval excludes zero. Under the main pre-tool/overlap proxy, the raw unbinned relation is also negative: OLS slope -0.013, $R^2 = 0.066$, and Spearman $\rho = -0.194$ over 100,919 events.

A.5 Cross-model threshold diagnostics

Table A.6 expands the compact cross-model summary from Section 3.1 for the two additional model settings. Together with the main-export diagnostics in Table A.4, it shows that the $\eta > \hat{\nu}$ region has higher harmful-flip rates and more negative probability updates across evaluated settings.

Table A.6: **Detailed threshold diagnostics for the additional model settings.** Rows $\eta \leq \hat{\nu}$ and $\eta > \hat{\nu}$ form the threshold split; $p_{c,\mathrm{pre}} \geq 0.80$ is a high-prior robustness slice rather than a disjoint partition. Harm and Rescue are flip rates. $\hat{\lambda}$ is a diagnostic recovered-authority estimate, not an identified latent parameter.

Model	Slice	N	Harm	Rescue	Mean Δp_c	Mean $\Delta \ell$	Up/Flat/Down	Mean $\eta - \hat{\nu}$	Mean $\hat{\lambda}$
30B-NV	$\eta \leq \hat{\nu}$	96,636	0.9	1.3	+0.0102	+0.722	20.3/71.9/7.8	-0.0257	0.106
30B-NV	$\eta > \hat{\nu}$	52,511	33.0	4.3	-0.2909	-9.075	26.9/3.7/69.4	+0.3195	0.605
30B-NV	$p_{c,\mathrm{pre}} \geq 0.80$	20,359	73.4	0.0	-0.7042	-16.021	7.5/2.4/90.2	+0.7042	0.782
Qwen3.5-9B	$\eta \leq \hat{\nu}$	34,724	0.1	5.4	+0.0398	-5.061	25.7/29.2/45.1	-0.0589	0.134
Qwen3.5-9B	$\eta > \hat{\nu}$	36,304	24.5	5.9	-0.1956	-4.304	22.7/1.1/76.2	+0.2139	0.589
Qwen3.5-9B	$p_{c,\mathrm{pre}} \geq 0.80$	21,663	30.5	0.0	-0.2898	-3.000	19.4/0.6/80.0	+0.2898	0.499

The table also explains why the paper emphasizes the paired harmful-flip and probability-update signature rather than any single logit-space statistic. Probability-space and logit-space summaries can differ in low-gap slices because the logit transform amplifies boundary movements, especially near the simplex boundary. Even with this caveat, the high-gap rows consistently show higher recovered authority and larger destructive movement, matching the state-dependent revision mechanism.

A.6 Per-dataset analysis on tool revision mechanisms

Figure A.1 answers whether the destructive pattern is driven by event frequency or by update size. Read it row-wise: each row fixes a benchmark, so the left and right panels in that row describe the same set of binned transitions. The left panel is a frequency view. It asks how often tool use moves probability in the helpful direction versus the harmful direction at each pre-tool probability bin. These frequencies alone do not prove net harm, because a rare destructive move can be large and a frequent constructive move can be small. They are nevertheless useful for ruling out a trivial story in which high-prior harm appears only because destructive transitions are vanishingly rare but heavily weighted by a few outliers.

The right panel of Figure A.1 is the amplitude view, and it is the cleaner diagnostic for asymmetric destruction. The x-axis converts p_c to prior odds $p_c/(1-p_c)$, while the y-axis is the harmful/helpful amplitude ratio. A value below one means helpful updates are larger on average in that bin; a value above one means destructive updates are larger. The fitted positive slopes show that the ratio rises systematically with prior odds. This is why Table A.2 reports positive α values: the destructive side grows faster than the constructive side as the model becomes more confident before tool use.

Figure A.2 then combines frequency and amplitude into the quantity most relevant for the main claim: the net expected update. In the left column, the positive component is the helpful mass and the negative component is the harmful mass; the black curve is their sum. The empirical zero crossing marks the point at which tool use stops increasing gold probability on average and starts decreasing it. In the right column, the same balance is expressed as a cost/benefit ratio. The horizontal zero line is the breakeven point on the $\log_{10}$ scale, so bars above zero indicate bins where harmful update mass exceeds helpful update mass.

Together, the two figures explain why average tool benefit can coexist with high-prior fragility. Tools repair many weak states, so the low-prior region can still be strongly positive. But when a strong correct belief is revised downward, the destructive move is large, and the local cost/benefit ratio can exceed one by orders of magnitude. This visual pattern is descriptive rather than a separate theorem; the threshold claim in the main text is supported by the sign-stable proxy splits in Tables A.4–A.6. The figures make the same mechanism easier to inspect: they show where the mass of revision changes from rescue-dominant to harm-dominant.

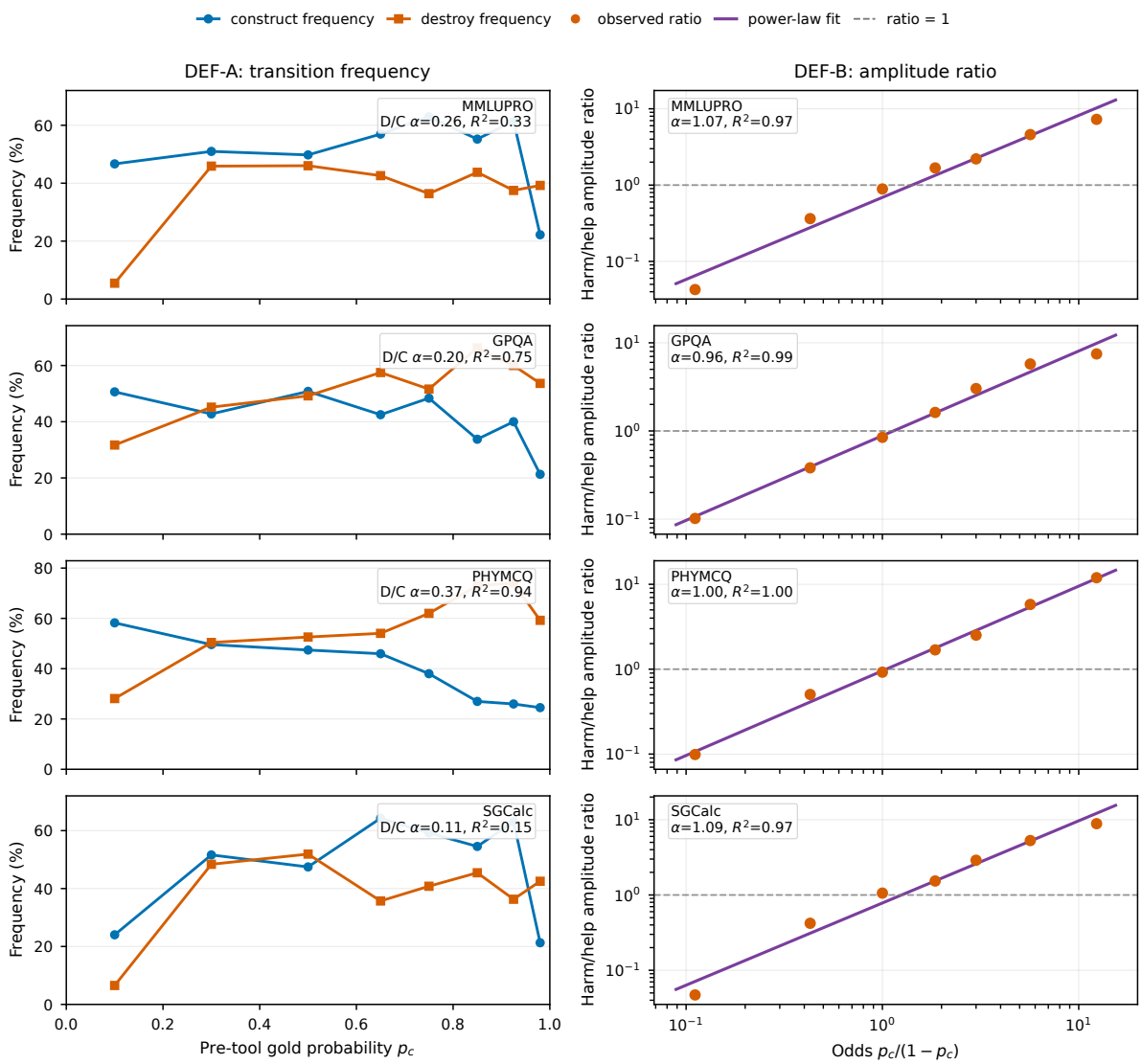

Figure A.1: **Per-dataset diagnostics for asymmetric belief destruction.** Each row is one benchmark. The left column separates constructive transitions, which increase gold probability, from destructive transitions, which decrease it, across pre-tool gold-probability bins. The right column plots the harmful/helpful amplitude ratio $\mathbb{E}[|\Delta p_c^-| \mid p_c]/\mathbb{E}[\Delta p_c^+ \mid p_c]$ against pre-tool odds on log–log axes. The fitted positive slopes show that as the model enters stronger-prior states, destructive update amplitude grows faster than constructive update amplitude. The horizontal ratio-1 reference line marks the point where helpful and harmful update magnitudes are equal; points above it are harm-dominant.

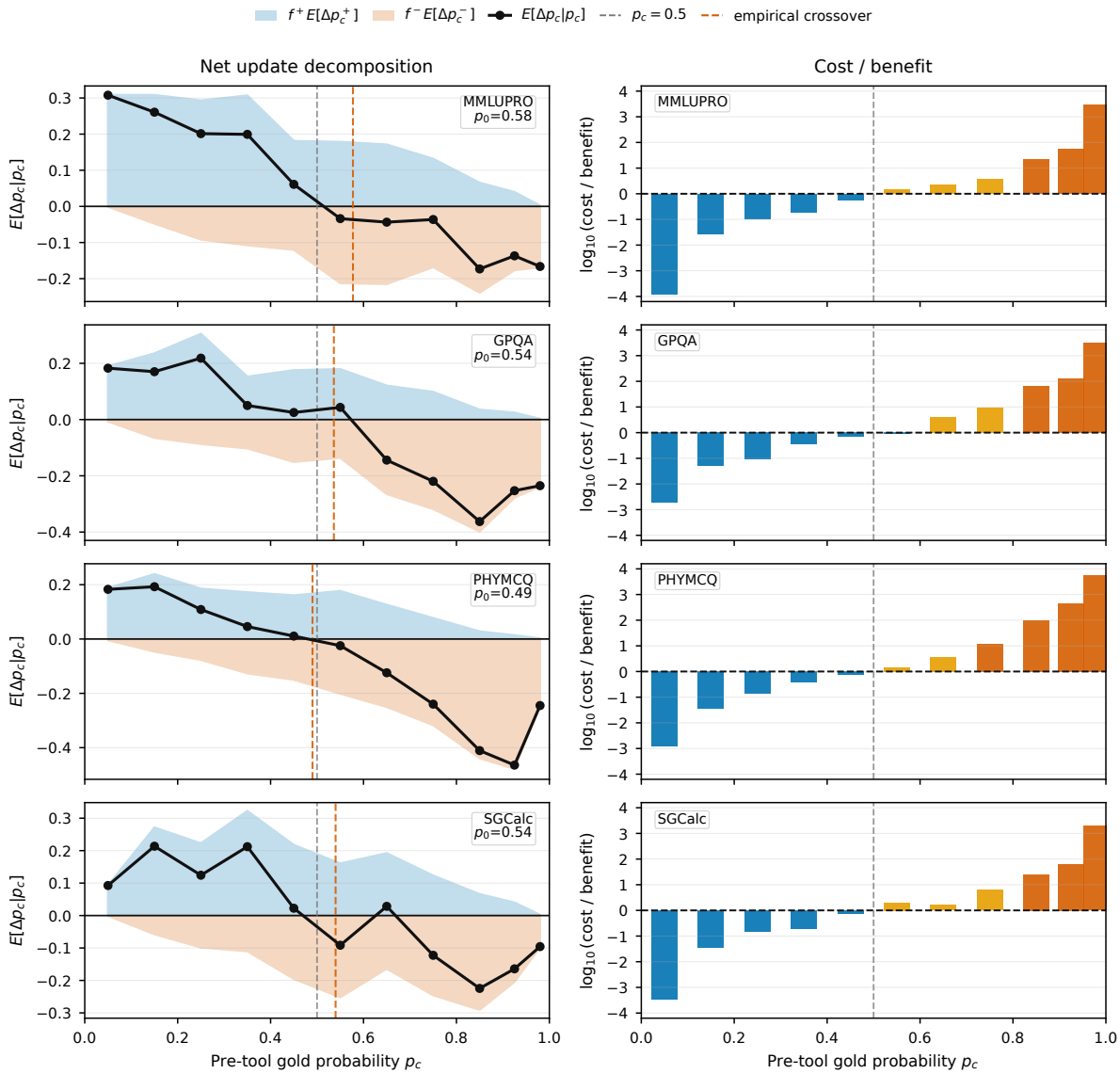

Figure A.2: **Net effect of asymmetric belief revision.** The left column decomposes $\mathbb{E}[\Delta p_c \mid p_c]$ into helpful and harmful contributions. The black net-update curve shows where tool use changes from helpful on average to harmful on average; vertical dashed lines mark $p_c = 0.5$ and the empirical zero crossing. The right column expresses the same shift as a cost/benefit ratio on a $\log_{10}$ scale. Bars above zero indicate bins where harmful update mass exceeds helpful update mass, so the external channel has become net destructive despite being useful at lower priors.

B Operational Analyses for CASE

This section retains the operational checks needed to support Section 4: first-pair cost and disagreement audits, budget-ladder behavior, simple gate baselines, nearby answer-state signals, and the open-ended exact-match check. Less central token-boundary confidence tables were omitted from the main appendix to avoid a float backlog.

B.1 Cost and first-disagreement audit

Table B.1: **AT and NT sample cost audit.** Token proxy is stored serialized trajectory character count; it is not a tokenizer count or latency measure.

Side	Samples	Mean token proxy	Mean tool calls	Missing signal
AT	141,956	1805.9	1.40	0.0%
NT	141,467	985.0	0.00	0.0%

Table B.2: **First-pair disagreement audit.** Accuracy is measured only where first AT and first NT answers disagree; Mix-MV is random-tie mixed-majority expectation.

Model	N	CASE	CISC	Mix-MV	Protect	Rescue	Both wrong
30B-NV	3,897	39.2	29.2	27.7	71.2	70.5	44.7%
Qwen3-Coder-30B	6,975	37.7	27.5	28.2	81.3	51.1	43.7%
Qwen3.5-9B	6,905	34.0	20.9	28.9	60.4	57.4	42.2%

The both-wrong column explains why disagreement accuracy is much lower than full-pool aggregate accuracy: in roughly half of first-pair disagreements, neither channel supplies the correct answer.

Table B.3: **Stratified disagreement summaries supporting Section 4.3.** Entries are unweighted averages across model settings on the first-pair AT/NT disagreement set. Protect is measured on NT-correct/AT-wrong cases; rescue is measured on NT-wrong/AT-correct cases.

Stratum	CASE-CISC gap	Protect	Rescue
$\eta \leq \hat{\nu}$	+7.7	–	–
$\eta > \hat{\nu}$	+13.5	–	–
$p_{c,\mathrm{pre}} \geq 0.80$	+20.8	86.5	72.5

B.2 Budget and fixed-budget behavior

Table B.5: **Fixed-budget gains over mixed majority.** Each cell is Mixed Signal minus Mixed MV in aggregate accuracy points; QC abbreviates Qwen3-Coder-30B.

Model	1+1	2+2	4+2	4+4	8+4	8+8
QC-30B	+3.8	+1.9	+1.6	+1.6	+1.1	+1.6
30B-NV	+2.5	+1.4	+1.5	+1.4	+1.5	+1.4
Q3.5-9B	+2.0	+1.4	+1.9	+1.5	+1.9	+1.8

Table B.6: **P(True)-CISC fixed-budget deltas.** Each cell is P(True)-CISC minus mixed majority on the same candidate-source files; QC abbreviates Qwen3-Coder-30B.

Model	1+1	2+2	2AT	2NT	4+4	8+8
QC-30B	-0.3	-0.2	+0.1	-0.3	-0.2	-0.1
30B-NV	+0.6	+0.2	+0.6	+0.4	-0.1	+0.0
Q3.5-9B	-3.2	-1.6	-3.1	-2.9	-0.8	-0.5

B.3 Gate and signal ablations

Table B.7: **Simple prior-protection gates are insufficient.** Each entry is the best aggregate-accuracy point from a threshold sweep on the fixed Qwen3-Coder-30B export.

Budget	Mix-cert	Freeze	Skip
1+1	57.3 @ 0.195	55.3 @ 0.157	54.7 @ 0.141
2+2	59.5 @ 0.391	57.1 @ 0.303	56.6 @ 0.274

Table B.8: **Nearby answer-state signals nearly tie.** Accuracy is aggregate accuracy on the shared four-benchmark Qwen3-Coder-30B evaluation.

Signal	Acc.	vs MV
Normalized entropy certainty	57.2	+4.2
$\mathrm{softplus}(\mathrm{KL}(p_i \| p_f))$	57.3	+4.2
$\max(c_{\mathrm{pre}}, c_{\mathrm{post}})$	57.3	+4.2
Post-probe margin	57.2	+4.2
Majority vote	53.6	+3.7

B.4 Free-form exact-match check

The main mechanism analysis is identified on multiple-choice data, where gold-event probabilities and answer-space entropy are directly observable. As an operational check beyond closed-choice grading, we also evaluate AT/NT arbitration on boxed exact-match math runs.

Table B.4: **Cross-model operating points.** Entries with "@" report accuracy at normalized cost.

Model	1+1 MV	1+1 Signal	8+4 MV	8+4 Signal	Best CASE ≤ 1	Best Seq ≤ 1
Qwen3-Coder-30B	53.6	57.3	60.2	61.3	60.8 @ 0.749	60.8 @ 0.881
30B-NV	66.0	68.5	70.5	72.0	71.2 @ 0.806	70.3 @ 0.634
Qwen3.5-9B	44.0	46.0	47.6	49.6	48.7 @ 0.791	48.4 @ 0.912

Table B.9: **Free-form exact-match AT/NT arbitration.** Rows report matched-budget AUC over the $1{+}1, 2{+}2, 4{+}4, 8{+}8$ ladder. The probe denominator is the retained completed-probe count; ten expected probe slots were absent after completion/parsing filters.

Dataset	N	Probe	Mix-MV	CASE	CISC	CASE-CISC	CISC-MV
AMC23	40	639/639	77.0	77.5	76.9	+0.6	-0.2
AIME 2026	30	480/480	35.2	43.3	37.5	+5.8	+2.3
BeyondAIME	100	1591/1591	13.0	13.3	12.8	+0.5	-0.2
Macro	170	2710/2710	41.7	44.7	42.4	+2.3	+0.7
Pooled	170	2710/2710	32.0	33.7	32.2	+1.5	+0.3

C Proofs and Additional Derivations

This appendix records the proofs deferred from Section 2. The derivations proceed from one-step logarithmic pooling to the threshold boundary condition and then to the finite-step destruction result used to interpret confident wrong endpoints.

C.1 Minimal-change objective and logarithmic pool

We adopt the same minimal-change principle used in Section 2. The revised state should remain close to both the internal belief and the contextual belief, with λ_x determining how far revision is allowed to move toward the external channel:

$$R_x^\lambda = \arg\min_{R\in\Delta(\mathcal{A})} \Big[(1-\lambda_x)D_{\mathrm{KL}}(R\,\|\,P_x) + \lambda_x D_{\mathrm{KL}}(R\,\|\,Q_x)\Big]. \tag{8}$$

Theorem C.1 (Minimal-change revision yields a logarithmic pool)**.** *For any finite answer space $\mathcal{A}$, distributions P_x, Q_x with common positive support, and any $\lambda_x \in [0,1]$, the optimization problem in Eq. (8) has a unique solution*

$$R_x^\lambda(a) = \frac{P_x(a)^{1-\lambda_x}Q_x(a)^{\lambda_x}}{\sum_{b\in\mathcal{A}} P_x(b)^{1-\lambda_x}Q_x(b)^{\lambda_x}}, \qquad a\in\mathcal{A}. \tag{9}$$

That is, the revised belief is a logarithmic opinion pool.

The objective in Eq. (8) is strictly convex in R over the simplex because the KL divergence is strictly convex in its first argument and the weights are nonnegative. Introduce the Lagrangian

$$\mathcal{J}(R,\tau) = (1-\lambda)D_{\mathrm{KL}}(R\|P) + \lambda D_{\mathrm{KL}}(R\|Q) + \tau\left(\sum_{a\in\mathcal{A}} R(a) - 1\right). \tag{10}$$

For any answer a with positive support under both P and Q, stationarity gives

$$(1-\lambda)\left(\log\frac{R(a)}{P(a)}+1\right) + \lambda\left(\log\frac{R(a)}{Q(a)}+1\right) + \tau = 0. \tag{11}$$

Collecting terms, exponentiating, and normalizing over a yields Eq. (9).

C.2 Binary log-odds form

Collapse a multiple-choice question to the binary event "gold answer $y^\star$" versus "non-gold." Let

$$p = P_x(y^\star), \qquad q = Q_x(y^\star), \qquad r = R_x^\lambda(y^\star), \tag{12}$$

and define log-odds $\eta = \mathrm{logit}(p)$, $\nu = \mathrm{logit}(q)$, and $\rho = \mathrm{logit}(r)$.

Corollary C.2 (Gold-event log-odds update)**.** *Under the binary gold-vs.-rest reduction,*

$$\rho = (1-\lambda)\eta + \lambda\nu, \tag{13}$$

and therefore

$$\rho - \eta = \lambda(\nu - \eta). \tag{14}$$

The derivation follows by applying the logarithmic pool to the binary odds ratio:

$$\frac{R^\lambda(c)}{R^\lambda(\neg c)} = \left(\frac{p}{1-p}\right)^{1-\lambda}\left(\frac{q}{1-q}\right)^{\lambda}. \tag{15}$$

Taking logs yields Eq. (13), and Eq. (14) follows by subtracting η.

C.3 Proof of Theorem 2.1

Let

$$Z_\lambda = (1-\lambda)\eta + \lambda M, \qquad \ell(z) = \log(1+e^{-z}). \tag{16}$$

Then the gold-event logistic risk is

$$\mathcal{L}(\lambda;\eta) = \mathbb{E}_M[\log(1+\exp(-Z_\lambda))]\,. \tag{17}$$

Because $\ell'(z) = -\sigma(-z)$ and $\partial Z_\lambda/\partial\lambda = M-\eta$,

$$\frac{\partial\mathcal{L}}{\partial\lambda} = -\mathbb{E}\left[\sigma(-Z_\lambda)(M-\eta)\right]. \tag{18}$$

Differentiating again gives

$$\frac{\partial^2\mathcal{L}}{\partial\lambda^2} = \mathbb{E}\big[\sigma(Z_\lambda)\sigma(-Z_\lambda)(M-\eta)^2\big] \geq 0, \tag{19}$$

so $\mathcal{L}$ is convex in λ. Evaluating at $\lambda = 0$ gives

$$\begin{aligned}\left.\frac{\partial\mathcal{L}}{\partial\lambda}\right|_{\lambda=0} &= -\sigma(-\eta)\,\mathbb{E}[M-\eta] \\ &= -(1-p)(\mathbb{E}[M]-\eta).\end{aligned} \tag{20}$$

If $\eta \geq \mathbb{E}[M]$, the derivative at the boundary is non-negative and convexity implies that $\lambda = 0$ is globally optimal. If $\eta < \mathbb{E}[M]$, the boundary derivative is negative, so sufficiently small positive λ lowers risk.

C.4 Iterated destruction under an authority floor

The main text keeps the theory focused on the threshold law, but the wrong-trace audit raises a follow-up question: why can harmful trajectories stay confident while becoming wrong? A simple iterative extension answers that question:

$$\eta_{t+1} = (1-\lambda_t)\eta_t + \lambda_t\nu_t. \tag{21}$$

Here η_t is the gold log-odds before revision step t, ν_t is the effective contextual log-odds introduced at that step, and λ_t is the authority granted to it.

Theorem C.3 (Finite-step belief destruction under an authority floor)**.** *Suppose $\eta_0 > 0$ is initially correct and strong. If there exist constants $\underline{\lambda} > 0$ and $\bar{\nu} < 0$ such that for all t,*

$$\lambda_t \geq \underline{\lambda}, \qquad \nu_t \leq \bar{\nu}, \tag{22}$$

then

$$\eta_t \leq (1-\underline{\lambda})^t\eta_0 + \big(1-(1-\underline{\lambda})^t\big)\bar{\nu}. \tag{23}$$

In particular, after finitely many steps the gold log-odds becomes negative, so the correct belief flips to an incorrect one.

C.5 Proof of Theorem C.3

From Eq. (21) and the assumptions $\lambda_t \geq \underline{\lambda}$ and $\nu_t \leq \bar{\nu}$,

$$\eta_{t+1} \leq (1-\underline{\lambda})\eta_t + \underline{\lambda}\bar{\nu}. \tag{24}$$

Iterating this recurrence yields Eq. (23). Because $\bar{\nu} < 0$ and $(1-\underline{\lambda})^t \to 0$, the right-hand side eventually becomes negative, proving a finite-step flip.

C.6 Scope of the theoretical result

The threshold law explains why sign reversal and high-prior fragility are expected under a misallocated revision policy, and why harmful traces can remain confident. The empirical power law in Eq. (7) remains an observed scaling relation rather than a derived theorem.

D Experimental Details and Protocols

This section records the retained reproducibility fields and the two protocols that are most directly tied to the main-text claims: the wrong-trace audit and PAIT. The original long prompt-template tables have been condensed into prose to avoid several consecutive full-width floats.

D.1 Model identifiers and fixed configuration

All three models are open-source Hugging Face checkpoints. We report the public model identifiers and all retained run settings to make the experimental protocol rerunnable. All runs were served with SGLang and executed on two NVIDIA A800 GPUs. Because the experiments use public Hugging Face checkpoints, a fixed reported protocol, and a consistent SGLang serving setup, the runs are reproducible at the protocol level.

Table D.1: **Fixed experimental configuration.** Defaults are taken from launch scripts unless overridden by the recorded run configuration.

Field	Value
Main mechanism run	Qwen3-Coder-30B on MMLU-Pro, GPQA Diamond, Physics, and SuperGPQA-Calc.
Cross-model runs	30B-NV(NVIDIA, 2025) and Qwen3.5-9B(Qwen Team, 2026) on the same benchmark set.
Free-form runs	AMC23, AIME 2026, and BeyondAIME exact-match math.
Settings	NT uses no tools; AT uses `always_tool` without SSGR rollback.
AT / NT samples	8 / 8 per question; `n_rollouts=1`.
Sampling	`temperature=0.7`; `seed=42`; probes and finalization are greedy.
Context / timeout	`max_model_len=81920`; per-request timeout `1800s`; max retries `8`.
Tool-call budget	At most 5 tool calls per trajectory in the main batch launch.
Tools	`run_python`; `wiki_search` when the Wiki service is enabled.
MCQ state probe	Guided option choice; `max_tokens=1`; `temperature=0`; thinking disabled.
Main support proxy	Leave-one-out mean post-tool p_c within matched event strata.

D.2 Tool execution parameters

Table D.2: **Tool execution parameters.** The Wiki service retrieves pre-indexed corpus rows/chunks; `max_chars` only truncates returned snippets.

Tool	Parameter	Value / behavior
`run_python`	Backend	Local subprocess executor; no Docker container.
`run_python`	Timeout	Executor `30s`, wrapped by async `60s` guard.
`run_python`	Limits / prelude	16GB memory limit; common math/science libraries auto-imported if installed.
`run_python`	Observation	JSON stdout/stderr/exit/time; long fields truncated by runner packaging.
`wiki_search`	Backend	Local FastAPI service backed by E5 embeddings and FAISS.
`wiki_search`	`topk`	Model-generated values normalized to `1..5`; runner default `3`.
`wiki_search`	`max_chars`	Model values clipped to `50..100000`; runner default `12000`.
`wiki_search`	Errors	Retries configured Wiki URLs up to three times, then returns empty-result error JSON.

D.3 Wrong-trace audit protocol

Section 3.2 uses two wrong-trace audits. The first is an automatic surface-plausibility audit over the four headline datasets. It labels each AT trajectory by the apparent relation between the tool evidence and the model's final answer: supportive, contradictory, irrelevant, or other. The labeller is instructed to use only the provided fields (question, answer options, model final answer, assistant text before the tool, tool calls and outputs, and assistant text after the tool), not external knowledge. It returns a single JSON object with evidence relation, tool-use quality, failure mode, confidence, spans, and a short rationale.

The automatic audit covers 13,390 labelled AT traces, of which 5,725 end in a wrong final answer. Among these wrong AT traces, 2,098/5,725 (36.6%) are supportive-looking, 1,964/5,725 (34.3%) are contradictory, 1,550/5,725 (27.1%) are irrelevant, and 113/5,725 (2.0%) are other. We define "high-confidence" in this diagnostic as the phantom-confidence failure mode: the final answer is asserted confidently despite non-supportive evidence. Among wrong traces whose evidence is contradictory or irrelevant, 3,452/3,514 (98.2%) remain high-confidence by this criterion.

Table D.3: **Automatic surface-plausibility audit over wrong AT traces.** The left block reports evidence-status shares among wrong AT traces. The right block reports phantom-confidence rates among non-supportive wrong traces.

Quantity	Num.	Denom.	Rate
Supportive-looking wrong traces	2,098	5,725	36.6%
Contradictory wrong traces	1,964	5,725	34.3%
Irrelevant wrong traces	1,550	5,725	27.1%
Other wrong traces	113	5,725	2.0%
Contradictory wrong traces that remain high-confidence	1,945	1,964	99.0%
Irrelevant wrong traces that remain high-confidence	1,507	1,550	97.2%
Contradictory/irrelevant combined	3,452	3,514	98.2%

The second audit is the retained mechanism-source audit. Its sampling frame is wrong AT trajectories, not all harmful trajectories. We first filter to AT samples whose final answer is wrong, that contain at least one non-empty and non-error tool call, and whose final assistant response is substantive. We then draw 240 traces from three model sources, with 80 traces per model (Qwen3-Coder-30B, 30B-

NV, and Qwen3.5-9B). Sampling is stratified by an automatic pre-screen over truth-axis labels, used only to improve coverage of candidate mechanisms; the final mechanism labels are assigned independently.

For each retained trace, the annotator sees the question, answer options, gold answer, model's wrong final answer, each tool output, the trace window immediately following the tool call, and the final assistant message. The five source labels are: A, tool-correct but answer-space reinterpreted; B, tool-partial and taken at face value; C, tool-irrelevant but confabulated into support; D, tool-contradictory but absorbed anyway; and U, unclear. The second binary label records whether the final wrong answer remains a high-confidence wrong state.

To estimate annotation stability, three experts audit the same 240-item manifest independently. For the five-way mechanism label, Fleiss' κ across the three same-model rounds is 0.485, with pairwise Cohen's κ values of 0.507, 0.467, and 0.489. For the binary high-confidence-wrong label, Fleiss' κ is 0.617, with pairwise Cohen's κ values of 0.627, 0.591, and 0.637. We report the majority-vote labels below.

Table D.4: **Mechanism-source distribution in the 240-item audit.** Percentages are within model; the overall row pools all sampled items.

Model	n	A	B	C	D	U
30B-NV	80	3.8%	16.2%	36.2%	16.2%	27.5%
Qwen3-Coder-30B	80	25.0%	17.5%	2.5%	55.0%	0.0%
Qwen3.5-9B	80	16.2%	12.5%	35.0%	32.5%	3.8%
Overall	240	15.0%	15.4%	24.6%	34.6%	10.4%

Table D.5: **High-confidence-wrong rates and eta-regime counts.** HC wrong is the Step-2 majority label; the eta-regime split uses the main $\eta > \hat{\nu}$ proxy.

Quantity	A	B	C	D	U
HC wrong rate	55.6%	29.7%	41.1%	57.8%	20.0%
$\eta > \hat{\nu}$ count	15	10	14	29	5
$\eta \leq \hat{\nu}$ count	21	27	45	54	20

Overall, 45.1% of sampled wrong AT traces are labelled high-confidence wrong, with model-level rates of 25.0% for 30B-NV, 47.5% for Qwen3-Coder-30B, and 63.6% for Qwen3.5-9B. The mechanism distribution differs strongly by model ($\chi^2 = 88.378$, $df = 8$, $p = 1.0 \times 10^{-15}$).

E Prompt Template For Tool Augmented Reasoning

System Prompt for Tool-Augmented Problem Solving

```
You are an expert problem-solving agent with
    access to tools via function calling.
Your task is to solve questions across
    domains including science, engineering,
medicine, economics, law, history,
    mathematics, and related fields.

Reasoning protocol:
Work iteratively using a ReAct-style
    procedure:
(1) Think: analyze the problem and identify
    what information or computation is needed.

(2) Call Tool: use the appropriate tool when
    verification or computation is required.
(3) Observe: inspect the returned result
    carefully.
(4) Repeat or Answer: continue using tools
    until confident, then provide the final
    answer.

Key principle:
Do not guess. Use tools to verify factual
    claims, calculations, and intermediate
    results.

Available tools:

1. run_python
Purpose:
Execute Python code in a sandboxed
    environment.

Input schema:
{"code": "<python code as string>"}

Output schema:
{"stdout": "...", "stderr": "...", "exit_code
    ": 0, "exec_time": 0.xx}

Use run_python for:
- multi-step arithmetic or algebraic
    computation;
- numerical comparison across answer options;
- unit conversion or dimensional analysis;
- probability, statistics, or combinatorics;
- data processing, parsing, or verification;
- symbolic or formula-based calculation.

2. wiki_search
Purpose:
Search a local Wikipedia knowledge base for
    factual information.

Input schema:
{"query": "<search query>", "topk": 1, "
    max_chars": 12000}

Output schema:
{"results": [{"title": "...", "content":
    "...", "score": 0.xx}, ...]}

Use wiki_search for:
```

```
- dates, names, events, definitions, or named
     entities;
- historical, scientific, medical, or legal
    facts;
- biographies, mechanisms, laws, or
    theoretical concepts;
- any factual claim about which you are
    uncertain.

Tool-use policy:
- Always use run_python for calculations
    requiring more than one arithmetic or
    algebraic step.
- Always use wiki_search for specific factual
     verification.
- For complex questions requiring both
    factual context and computation, first
    use wiki_search
  to gather the needed facts, then use
    run_python to compute or verify the
    answer.
- If a tool call fails, inspect the error and
     retry with a corrected call when
    possible.

Tool-call format:
Use the following XML-style wrapper with JSON
     content.

<tool_call>
{"name": "run_python", "arguments": {"code":
    "<python code here>"}}
</tool_call>

<tool_call>
{"name": "wiki_search", "arguments": {"query":
     "<search query>", "topk": 1}}
</tool_call>

Final answer format:
For multiple-choice questions, output the
    final answer as:
\boxed{X}

where X is the single letter corresponding to
     the correct option.

For open-ended numerical questions, output
    the final result in boxed form:
\boxed{<answer>}

Important reminders:
- Use tools proactively to reduce reasoning
    errors.
- Keep tool calls targeted and relevant.
- Verify intermediate results before giving
    the final answer.
- Stay focused on the question and avoid
    unnecessary elaboration.
```

E.1 PAIT protocol and selection flow

PAIT is a same-content, different-wrapper intervention. For each selected high-prior case, the preparation script generates one wrapper-invariant misleading clue, stored as `adv_claim`. The same clue is inserted verbatim into TOOL, CONTEXT, REFLECTION, and TOOL+ANTIDOTE answer prompts. All PAIT answer requests disable tools at the API level with `tool_choice=none`; the intervention is therefore the authority frame, not a new tool execution.

For the human annotation version, annotators are shown the dataset and item identifier, question, options, option letters, gold answer, historical harmful final answer, pre/post gold-answer probabilities, prior bin, and the harmful trajectory transcript. They do not change the question, options, or gold label. Their task is to write one short false but plausible clue that conflicts with the gold answer, points to a single non-gold option, and can be reused word-for-word under every wrapper.

Table E.1: **Human PAIT adversarial-clue annotation form.** These are the fields annotators fill for each selected high-prior harmful example.

Field	Type	Acceptance rule
`adv_claim`	Free text	One or two sentences, preferably under 40 words; false relative to the gold answer; no mention of tools, Python, search, prompts, logs, or the model.
`target_wrong_letter`	Single choice	Must be exactly one non-gold option, preferably the historical harmful final answer when that direction is clear.
`claim_type`	Single choice	One of `numeric`, `comparison`, `definition`, `causal`, or `other`.
`why_plausible`	Free text	Briefly explains why the clue sounds like a valid calculation, comparison, definition, or causal fact.
`source_support`	Free text	Briefly identifies the harmful-trace mistake or tool-output interpretation from which the clue was abstracted.

Table E.2: **PAIT wrapper templates.** The same `ADV_CLAIM` string is inserted without rewriting. All probes use the same one-letter option-choice format.

Condition	Wrapper cue	Controlled contrast
Baseline	No external signal; solve from the question and options.	Estimates the clean unrevised answer state.
Tool	"An external tool now reports: `ADV_CLAIM`."	Presents the clue with external-tool authority but no actual tool execution.
Context	"Additional reference information: `ADV_CLAIM`."	Presents the identical clue as reference context.
Reflection	"Your earlier reasoning may have overlooked the following clue: `ADV_CLAIM`."	Presents the identical clue as a self-correction cue.
Tool+Antidote	Tool wrapper plus a warning that external tools can hallucinate or return incorrect numerical results.	Tests whether explicitly limiting tool authority attenuates the same-clue damage.

Table E.3: **PAIT selection flow.** Selection starts from the Qwen3-Coder-30B live run and then deduplicates by question before balanced per-dataset sampling.

Dataset	Seen	Cand.	Deduped	Selected
MMLU-Pro	6,696	922	258	149
GPQA Diamond	198	109	29	29
Physics	325	102	31	31
SuperGPQA-Calc	11,229	1,206	282	166
Total	18,448	2,339	600	375

Table E.4: **Detailed cross-model PAIT on eligible high-prior cases.** FR is answer flip rate under the adversarial-clue wrapper; parentheses report eligible rows.

Model	Tool FR	Context FR	Reflection FR	Tool+Antidote FR
Qwen3-Coder-30B	77.0% (104)	63.7% (104)	78.6% (104)	37.4% (104)
Qwen3.5-9B	20.5% (163)	41.9% (163)	85.0% (163)	0.0% (163)
30B-NV	51.9% (259)	26.8% (259)	44.1% (260)	20.0% (250)

Table E.5: **Matched Qwen3-Coder-30B PAIT with no-claim control.** FR is answer flip rate on the condition-complete subset ($n = 104$ per condition).

Condition	Adv FR	Adv post p_c	No-claim FR	No-claim post p_c
Tool	77.0%	0.253	2.0%	0.981
Context	63.7%	0.386	5.9%	0.942
Reflection	78.6%	0.251	9.8%	0.903
Tool+Antidote	37.4%	0.625	7.2%	0.942

The number of eligible PAIT rows differs across model settings because PAIT keeps only cases that pass the executed clean-baseline filter and the wrapper/probe completion filter. We therefore report flip rates with per-condition sample sizes rather than forcing a fixed n across models.