WHEN EVIDENCE CHANGES, DOES REASONING FOLLOW? REVISING LANGUAGE MODEL CONCLUSIONS UNDER CHANGING EVIDENCE

Anonymous authors
Paper under double-blind review

ABSTRACT

Reasoning establishes relationships between premises and interpretations, which may be reconsidered when the premises change. We study how language models respond when relationships change while maintaining earlier reasoning in context. We change a fact that affects the correct answer while leaving the earlier reasoning in the context, then test whether models use the updated evidence or repeat an outdated conclusion. The experiments cover controlled arithmetic, logic, and navigation tasks, GSM8K, FinQA, CRUXEval, and Wikidata problems, and the STALEFLOW tool-action test, which examines whether updated information governs a subsequent action. We compare retaining the full history with dependency-based pruning: removing reasoning steps that depend on a changed premise, including their downstream conclusions, while preserving unaffected reasoning. This comparison tests whether stale reasoning obstructs revision and whether selectively removing it improves accuracy. Across these studies, we evaluate Qwen3-1.7B, 4B, 8B, 14B, and 32B; Qwen2.5-3B, 7B, and 72B; Llama-3.1-8B and 70B; Phi-4; Mistral-7B-Instruct-v0.3; and OLMo-2-7B, including base checkpoints, together with five OpenAI models. On the controlled problems, removing only the invalidated steps lowers Qwen3-4B-Thinking's rate of repeating the outdated answer from 0.969 to 0.070, and under immediate answering the effect appears in all 17 models tested and in all 40 model–dataset cells on the benchmark problems. A correction notice fails when models answer immediately but works when they reason first, so the better remedy depends on how the model answers. Across $1{,}800$ multi-turn sessions, dependency-based pruning improves accuracy by 27.3 percentage points compared with retaining the full history. In STALEFLOW, Qwen2.5-7B uses an outdated value in 62.5% of immediate-action cases. These findings frame revision as maintaining relationships between evidence and conclusions: checking what has lost support, preserving what remains valid, and computing a new answer where needed.

Code: https://anonymous.4open.science/r/stale-reasoning

1 INTRODUCTION

An answer can become outdated even when the reasoning that produced it was correct: a tool corrects a price, a user changes an assumption, or a document replaces an earlier figure. The model then has the new evidence in its context alongside conclusions drawn from the old evidence, and it has to work out which of those conclusions still hold. Consider a model that computes an invoice total and decides that a purchase fits the budget: if the unit price changes, the total has to be recomputed and the budget decision rechecked, while the delivery address is unaffected. Whether a statement is still usable depends on what it was derived from, since a recent summary can repeat an outdated total and an older statement can remain valid through many updates.

We study how language models handle this situation, and find that when a fact is edited and the earlier reasoning stays in context, models tend to give the answer that reasoning supports. We report three main findings, the first of which is that the stale reasoning itself produces the error. On 750 controlled problems with exact answers before and after each edit, removing only the invalidated

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

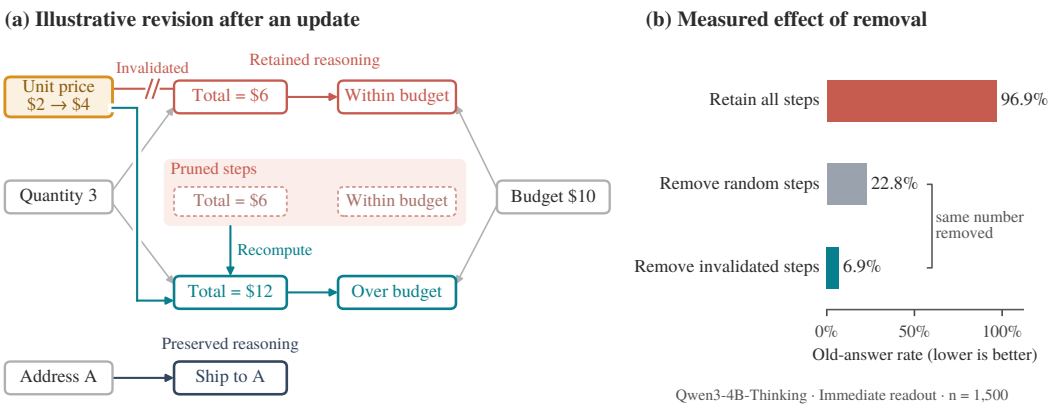


Figure 1: (a) An edited price invalidates the previously computed total and budget decision, while the quantity, the budget, and the shipping instruction remain supported. Retained reasoning carries the old conclusion forward despite the update, whereas dependency-aware revision prunes the unsupported descendants, recomputes the total from the updated price, and then reassesses the budget; the recomputed path is illustrative. (b) On the controlled problems, removing the invalidated steps lowers Qwen3-4B-Thinking's old-answer rate from 96.9% to 6.9% under immediate readout, against 22.8% for removing the same number of randomly chosen steps (deletion-control run, Table 1). An old-answer rate of 6.9% does not imply 93.1% accuracy, since revision accuracy after removal is 30.6% and the model must still recompute the answer itself.

steps lowers Qwen3-4B-Thinking's old-answer rate from 0.969 to 0.070. The rate rises as more of the stale derivation is present, and the median effect across the 12 open models we test is +0.653. Deleting the same number of randomly chosen steps leaves three times as much repetition, and a correct derivation of the same kind and length improves accuracy. The effect therefore depends on which steps stay in context, and context length alone does not explain it.

Second, the better remedy depends on how the model answers: on GSM8K, deleting the stale conclusion works better when Llama-3.1-8B answers immediately, and an explicit correction notice works better when it generates a solution first. A remedy validated under one response procedure can therefore fail under the other, and both procedures are common in deployed systems. Third, we locate the effect in a small set of attention heads, reduce it with training, and measure it in tool calls. Ablating twenty heads that attend to the stated conclusion lowers the old-answer rate in Qwen2.5-7B from 0.880 to 0.090, compared with 0.780 for twenty random heads. Training on edited problems raises revision accuracy from 0.152 to 0.513 while keeping answers stable when an edit does not change them. In a tool-calling task, the outdated value reaches the committed call in 62.5% of Qwen2.5-7B's immediate-action episodes, and pruning reasoning that depends on a changed fact raises accuracy in 1,800 multi-turn sessions by 27.3 points.

Contributions. (i) A benchmark of edited derivations with exact answers before and after each edit and matched irrelevant edits, extended to GSM8K, FinQA, CRUXEval, Wikidata, and STALEFLOW, a tool-calling task in which an update arrives before the call is made. (ii) Controls that separate stale reasoning from context length, from the edited line, and from the presence of reasoning, with the effect replicated in 17 of 17 models, in 40 of 40 benchmark cells, and under full chain-of-thought generation. (iii) A comparison of remedies showing that correction notices and context-aware decoding do not remove the effect, and that the better context edit changes with the response procedure. (iv) A mechanistic analysis in attention heads and a training objective that teaches models to change their answer only when an edit requires it. (v) INERTIA-WILD, 473 naturally occurring revision episodes mined from WildChat, released with a datasheet.

2 METHODS

2.1 REASONING DEPENDENCIES

Let $\mathcal{F}_t$ denote the evidence at update t, and note that a stored reasoning line v has content and a justification, namely the premises, earlier lines, or rules it uses. These relationships form a directed graph $G_t = (V_t, \mathcal{L}_t)$ with an edge $u \to v$ when v uses u, following the justification graphs of truth maintenance systems (Doyle, 1979; de Kleer, 1986). An edit touches a set $\mathcal{C}_t$ of lines directly, and the lines that need revalidation are

$$\mathcal{A}_t = \mathcal{C}_t \cup \mathrm{Desc}_{G_t}(\mathcal{C}_t). \tag{1}$$

Membership in $\mathcal{A}_t$ marks a line for rechecking, and a marked conclusion can still hold if it has an independent valid justification. For $x = 2$, the derivation $y = x + 3 = 5$ followed by $2y = 10$ is valid, and updating x to four invalidates both y and the total while leaving unrelated statements intact.

Selective revision therefore has three parts: *detection* identifies which statements lose support, *propagation* follows that loss to the reasoning built on them, and *execution* produces a new answer or action from what remains valid. Our measurements map onto these parts, since old-answer rate measures whether the lost support is detected and propagated and revision accuracy measures execution. A recent sentence can quote an obsolete value and an old one can remain valid, so the age of a statement is not a reliable guide to whether it still holds.

2.2 BENCHMARKS AND MEASURES

The controlled benchmark contains 250 problems each in branching arithmetic, Horn logic, and grid navigation, and every problem has eight answer options and a correct canonical derivation. Relevant edits change the exact answer and matched irrelevant edits leave it unchanged; a simulator verifies 3,000 edits of each type, and independent reconstruction reproduces every relevant edited answer. On the same problems we retain the stale derivation with or without an explicit notice, keep only its valid prefix, or supply the correct revised derivation, and an unedited truncation control stops the original derivation at the same depth (Appendices A and H). We also edit GSM8K, FinQA, Wikidata, and CRUXEval items (Cobbe et al., 2021; Chen et al., 2021; Vrandečić & Krötzsch, 2014; Gu et al., 2024) and compare stale-context answers with scratch answers to the edited problem. The immediate-readout study reads the answer letter directly, while in the silent-edit study the problem changes in place with no notice, the stale derivation stays, and the model generates a full response of 512–700 tokens. A separate remedy experiment varies correction notices and conclusion deletion under both of these response procedures, so that each remedy is tested with immediate readout and with generation.

In STALEFLOW, a lookup supplies a value V_{old}, the agent states the value it plans to book, and a mandatory pre-commit lookup returns V_{new} for the same quantity, with the final tool-call argument as the endpoint. We compare an immediate-action condition with a reasoning condition, and a clean condition in which the obsolete value never appears measures the model's ability to act correctly (Appendix E). Throughout, old-answer rate is the frequency of selecting the pre-edit answer after the edit, and revision accuracy the frequency of selecting the new correct answer. We report the two separately because a model can stop giving the old answer without finding the new one. Symbolic intervals use paired problem-level resampling, agent intervals are Wilson intervals, and session intervals use a stratified bootstrap over 1,800 sessions (Appendix F).

3 EXPERIMENTS

The experiments move from the controlled benchmark to benchmark revisions, the reasoning budget, remedies, tool actions, the mechanism inside the model, training, and pruning in multi-turn sessions.

Removing the invalidated steps. For Qwen3-4B-Thinking, the old-answer rate with the stale derivation is 0.969, and keeping only the valid prefix brings it to 0.070, a paired reduction of 0.899 $[0.880, 0.916]$. Each of the 750 problems contributes two items that share a world and a derivation, so all model-level intervals resample problems. Paraphrasing the stale conclusion still gives 0.855,

Table 1: Deletion policies that remove the same number of steps, old-answer rate / revision accuracy. The coherent derivation is a correct derivation of the edited world. Values come from the deletion-control run and agree with Table 4 to within 0.005.

Policy	Qwen3-4B-Thinking	Qwen2.5-7B
Keep everything (stale)	.969 / .011	.816 / .082
Delete only the first invalid step	.825 / .105	.790 / .091
Delete the same number at random	.228 / .231	.274 / .187
Delete the invalidated tail	**.069 / .306**	**.133 / .241**
Coherent derivation	.007 / .950	.004 / .957

so the effect does not depend on its exact wording, and for most models it sets in sharply once three or more downstream steps must be recomputed (Appendix I). The effect also builds up as the stale derivation accumulates, starting from an old-answer rate of 0.067 with no prior reasoning in context and staying near that level for the first two steps. It rises to 0.587 at the third step and reaches 0.969 once the derivation is complete (Figure 4 in Appendix B). The valid-prefix condition stays between 0.066 and 0.071 throughout, even though the edited problem is the same in every condition and only the reasoning in context changes.

Deletion policies and matched controls. Table 1 removes exactly the same number of steps under four policies and reports revision accuracy alongside, and Figure 1b shows three of these policies. Deleting the invalidated tail beats deleting the same number of randomly chosen steps by 0.159 $[0.141, 0.178]$ in old-answer rate for Qwen3-4B-Thinking and 0.141 $[0.124, 0.159]$ for Qwen2.5-7B, and it raises revision accuracy as well. Deleting only the first invalidated step leaves 0.825 and 0.790, and removing the full tail lowers these by a further 0.755 $[0.734, 0.777]$ and 0.657 $[0.633, 0.680]$. Removing only the stated conclusion leaves 0.483–0.601 across four models, compared with 0.071–0.133 when the whole tail is removed. The old answer is supported by the steps that follow from the edited premise, and removing the edited line or the final sentence leaves most of that support in place. A correct derivation of the same kind and length has the opposite effect, raising accuracy to 0.915 across 13 paired model–dataset cells under free generation, above 0.874 with no derivation at all, while the stale derivation lowers it to 0.809. Accuracy is lower under stale reasoning in 11 of the 13 cells, and 8 of these differences survive Holm correction. The effect is also specific to edits that change the answer, because edits to an unused value or an inactive rule arrive in the same format beside the same stale derivation but cannot change the answer. On those, the strongest models are near ceiling (Phi-4 0.996, Llama-3.1-8B 0.979, Qwen3-4B-Thinking 0.979), whereas on edits that do change the answer they fall to 0.164, 0.047, and 0.011. Models therefore answer correctly when the edit leaves the derivation valid and fail when it does not.

Replication across models. Figure 2 runs the same comparison on 12 open models from six families and five developers, including base checkpoints, and on five closed frontier models. All 17 show the effect, all 17 intervals exclude zero, and the 12 open models remain significant after Holm correction. The median effect across the open models is $+0.653$, with an interquartile range of $+0.535$ to $+0.764$. Instruction-tuned models are more affected than base checkpoints (median $+0.713$ against $+0.576$), and the effect is smallest on logic for every model that shows a gap (Appendix B). Closed models do not expose token probabilities, so we instruct them to answer with a single option letter and read the first letter of the reply, with coverage 1.00 in every cell. GPT-5-mini at minimal reasoning ($+0.712$) is more affected than Qwen2.5-7B ($+0.690$), and GPT-4.1 and GPT-4o, which have no reasoning setting, show it clearly.

The effect also holds when models write a full solution: with chain-of-thought generation of up to 2,048 tokens on the controlled benchmark, every model we test still gives the pre-edit answer on part of the stale items. The rate is 0.700 $[0.655, 0.742]$ for Mistral-7B, 0.375 for Llama-3.1-8B, 0.297 for OLMo-2-7B, and 0.106 for Qwen3-4B-Thinking, with navigation the hardest domain (Table 7). It holds as well when the reasoning is the model's own: among originally solved problems, Qwen2.5-7B's old-answer rate is 0.982 with its full generated chain and 0.216 with the invalidated portion removed, and Qwen3-4B's is 0.919 against 0.292. Removing the stale steps does not by itself solve the edited problem, since Qwen3-4B-Thinking's valid-prefix revision accuracy is 0.302, close to

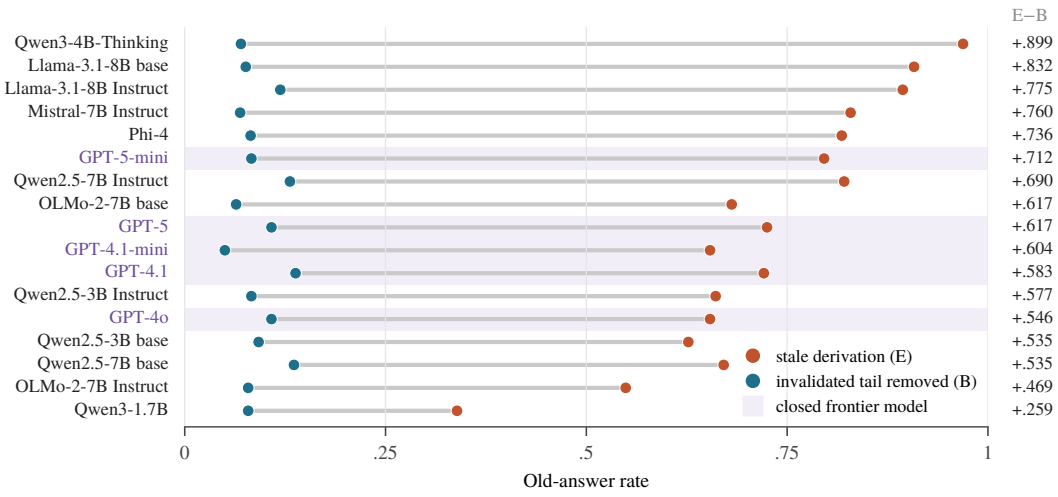


Figure 2: Old-answer rate with the stale derivation (E) and with the invalidated tail removed (B), for 12 open and 5 closed models; the right column gives the paired difference. Every one of the 17 intervals excludes zero (Table 4). Closed models answer with a committed single letter.

Table 2: Benchmark revisions. Old-answer rate under the stale derivation and the range of the paired difference E−B across models; Holm correction within each dataset.

Dataset	Items / model	Models	Old-answer, stale	E−B range
GSM8K	350	11	.983–1.000	+.763 to +.889
Wikidata	150	9	.980–1.000	+.840 to +.933
FinQA	350	7	.969–1.000	+.780 to +.863
CRUXEval	328	9	.811–.988	+.448 to +.595
CRUXEval-CoT	58	4	.931–1.000	+.552 to +.603

the 0.269 it reaches on an unedited problem truncated at the same depth. After the stale steps are removed, the model still has to complete the remaining computation on its own.

Benchmark revisions. We edit GSM8K, FinQA, CRUXEval, and dated Wikidata items, permuting the answer options over three orders and averaging across them. The stale derivation raises the old-answer rate in all 40 model–dataset cells (36 on the four datasets and 4 on CRUXEval-CoT), and all 40 remain significant after Holm correction within each dataset (Table 2). On GSM8K, FinQA, and Wikidata, revision accuracy under the stale derivation is essentially zero, and the largest models are also affected, with Llama-3.1-70B repeating the old answer on every GSM8K item. CRUXEval-CoT uses each model's own chain of thought as the stale derivation, and the effect holds on 4 of 4 models.

Free generation on benchmarks. When the model writes a full solution and the answer is parsed from its own output, the effect is smaller and remains significant in all ten cells we measure, across four datasets and four models. Llama-3.1-8B repeats the old answer at 0.301 on GSM8K against 0.021 with a coherent context ($+0.280$ $[0.215, 0.344]$) and at 0.460 against 0.286 on CRUXEval ($+0.185$ $[0.083, 0.292]$), and the smallest cell, Qwen3-8B on GSM8K, is $+0.069$ $[0.032, 0.112]$. The stale derivation also lowers accuracy, as Llama-3.1-8B's revision accuracy on CRUXEval falls from 0.540 from scratch to 0.379 with the stale derivation present.

Reasoning budget. The amount of computation after the edit determines whether it helps, which we test by stating the correction after the stale derivation and giving Qwen3-4B-Thinking exactly b tokens before forcing an answer (240 items from 120 problems). A filler arm spends the same budget on a fixed unrelated sentence, so the prompt grows by the same number of tokens with no computation in them (Figure 3a). Revision accuracy starts at 0.183, falls to 0.033 at 64 tokens, and rises to 0.775 at 4,096, and both changes are significant: the drop to the trough is 0.150 $[0.117, 0.182]$

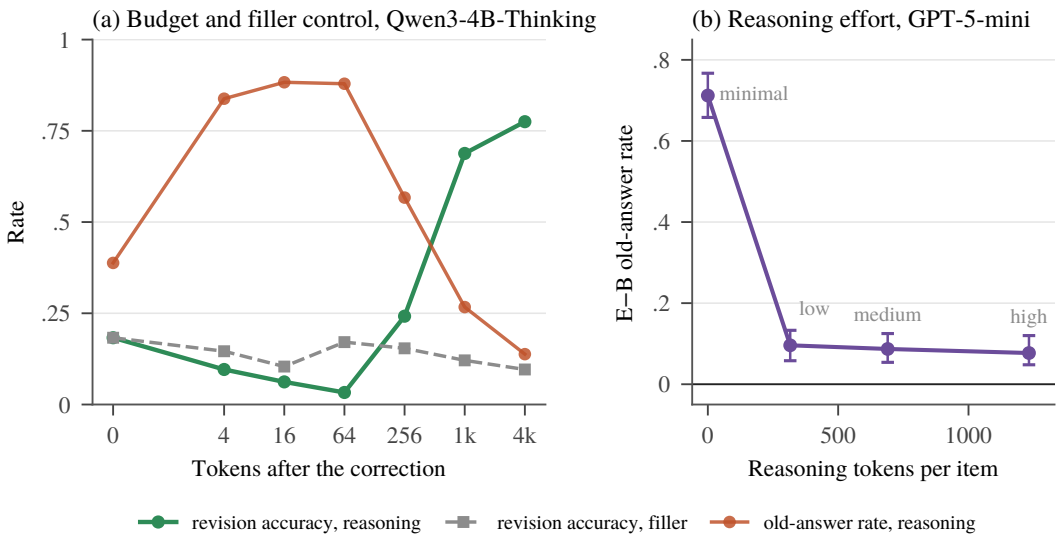


Figure 3: (a) Qwen3-4B-Thinking with a fixed budget of b tokens after the correction, then a forced answer (240 items). The filler arm spends the same budget on an unrelated sentence. (b) The effect for GPT-5-mini at each reasoning effort setting, with 95% intervals.

and the rise from the start is 0.592 $[0.533, 0.647]$. The filler arm stays between 0.096 and 0.183 throughout, whereas between 4 and 64 tokens reasoning gives significantly lower revision accuracy than filler, by 0.042 to 0.138. The drop therefore comes from the content of the partial reasoning and not from the added length. At $1{,}024$ and $4{,}096$ tokens the relation reverses, and reasoning is ahead of filler by 0.567 $[0.500, 0.636]$ and 0.679 $[0.622, 0.738]$. When the model is allowed to stop on its own it uses $2{,}344$ tokens and reaches the same 0.775.

GPT-5-mini shows the long-budget side of the same curve, with an effect of $+0.712$ at minimal effort that low effort, at 316 reasoning tokens per item, cuts to $+0.096$ $[0.058, 0.133]$ (Figure 3b). High effort, at $1{,}232$ tokens per item, leaves $+0.077$ $[0.048, 0.120]$, which is still above zero, so even the largest budget does not remove the effect entirely. A partial rederivation leaves the model more attached to the stale answer than no rederivation at all, and a complete one removes most of the effect.

Correction notices and context-aware decoding. Telling the model that its reasoning is out of date has a small effect when it answers immediately. Appending an explicit notice to the stale derivation removes 0.16–0.32 of the effect and leaves the old answer winning 59%–71% of the time across four models, whereas deleting the invalidated steps takes the same models to 0.071–0.133. A correction block that keeps the original problem and states exactly what changed, placed immediately before the question, still leaves significantly more repetition than deletion in all 12 open models, by up to a factor of nine. Context-aware decoding (Shi et al., 2024), a standard remedy for models that ignore their context, raises the old-answer rate because it increases the weight of the context relative to a run without the prior reasoning. In this setting the prior reasoning is itself the source of the error, so amplifying the context amplifies the stale answer. At every strength we test and on all four models, it raises the old-answer rate above the plain stale readout, by up to $+0.088$ $[0.073, 0.103]$ on Qwen2.5-7B. Its best setting on each model is still worse than deleting the invalidated tail, by 0.724 to 0.905.

Immediate readout and free generation. The response procedure changes which context edit works best: on GSM8K, under immediate readout, deleting the stale conclusion cuts Llama-3.1-8B's old-answer rate from 0.998 to 0.204, while a correction notice leaves it at 0.933. Under free generation the order flips, with an unremedied rate of 0.134, a rate of 0.172 after conclusion deletion, and 0.015 with a notice. Deleting the conclusion removes the answer that an immediate readout extracts, but a model that generates a solution can rederive it from the remaining steps. A notice

Table 3: STALEFLOW stale-action rates under immediate action and reasoning, and correct-action accuracy when no stale value is in context. Intervals are Wilson 95% intervals for the immediate-action cells.

Model	N	Immediate stale	Reason stale	Clean accuracy
Qwen2.5-7B	576	.625 [.585,.664]	.007	.901
Qwen3-32B	288	.573 [.515,.629]	.003	.979
Llama-3.1-8B	576	.352 [.315,.392]	.002	.847
Qwen2.5-72B	240	.021 [.009,.048]	.000	.996

has little effect on an immediate readout and a large effect on a model that recomputes, so context remedies should be evaluated with the response procedure used in practice.

Tool actions in STALEFLOW. In STALEFLOW the update arrives before the action, yet the stale value reaches the committed argument in 0.625 of Qwen2.5-7B's immediate-action episodes, 0.573 of Qwen3-32B's, and 0.352 of Llama-3.1-8B's (Table 3). Allowing the model to reason first reduces these rates to 0.007, 0.003, and 0.002, and every model acts correctly in most episodes (0.847–0.996) when the obsolete value never appears. Parsing succeeds in every non-thinking cell, so the stale values are the arguments the models actually commit. Qwen2.5-72B resists in this setting (0.021) while repeating the stale answer at 1.000 on Wikidata readout, so the size of the effect depends on the interface through which the model answers.

Attention heads and hidden states. Three methods, head ablation, residual patching, and probing, locate the effect in the middle and late layers of the network. First, ablating the attention heads that attend most to the stated conclusion lowers the old-answer rate, while ablating the same number of random heads does not. On 100 items, the top 5, 10, and 20 heads of Qwen2.5-7B take the old-answer rate from 0.880 to 0.540, 0.360, and 0.090, against 0.890, 0.940, and 0.780 for random sets of the same size. On Llama-3.1-8B the top 20 take it from 0.990 to 0.310 against 0.840, and the intervals for targeted and random ablation are disjoint at every size on both models. Second, the old answer is carried in the residual stream, since patching the answer-position residual of a stale run into a run with the invalidated steps redacted transfers the old answer. The redacted run gives it 0.18–0.19 of the time on its own, while the patched run gives it 0.990 by the final layer of Llama-3.1-8B and 0.840 by layer 24 of Qwen2.5-7B. The transfer turns on at layers 12–15 in Llama-3.1-8B and at layers 18–21 in Qwen2.5-7B, so the stale answer is written into the residual stream in the middle of the network (Figure 6 in Appendix C). Third, a linear probe on the residual stream at the answer position predicts, item by item, whether the model will give the old answer. On Qwen2.5-7B it peaks at layer 21 with accuracy 0.844 $[0.817, 0.869]$, against a majority-class floor of 0.514, and its confidence correlates with the model's behavior at $\rho = +0.701$. With shuffled labels the same pipeline gives 0.493, and a probe on the embedding layer sits exactly at the floor. In Qwen2.5-7B the three methods therefore agree on depth, with the heads at layers 16–20, the patching transfer turning on between 18 and 21, and the probe peaking at 21. The same heads are used when the stated conclusion is correct, since in Llama-3.1-8B ablating them lowers accuracy from 0.95 to 0.21 in that case. The heads that produce this behavior are therefore the heads that read a stated answer in general.

Training on edited problems. The behavior depends on the training distribution: a 111M-parameter transformer trained from scratch on 60,000 problems repeats the stale answer at 0.990 when no training problem revises its world. Adding revised problems reduces it monotonically, to 0.981 at 1%, 0.837 at 5%, and 0.296 at 20%, with revision accuracy rising from 0.010 to 0.704. Training has a similar effect on pretrained models, which we show by comparing two rank-16 LoRA objectives with the same problem pool, capacity, and step count. One supervises answers on unedited problems, and the other supervises the correct answer after relevant and irrelevant edits. Answer-only training raises Qwen3-1.7B's old-answer rate under relevant edits from 0.255 to 0.906, whereas revision training raises revision accuracy from 0.152 to 0.513 $[0.482, 0.545]$, reproducible at 0.515 ± 0.002 across three seeds, and stability under irrelevant edits from 0.338 to 0.905. The same recipe helps larger models, as Qwen3-4B improves from 0.178 to 0.575 and Qwen2.5-7B from 0.210 to 0.607. Accuracy alone would reward a model that changes its answer after every edit, so

we also measure whether a model moves its answer only when the edit calls for it. For reasoning prefix k and edit δ_i, we therefore compute the response difference

$$R_k[i, y] = P_k(y \mid \delta_i) - P_k(y) \tag{2}$$

which measures where answer probability moves after the edit. Selectivity is the share of total response magnitude assigned to relevant edits, with 0.5 marking a model that reacts to every edit alike. Revision training raises Qwen3-1.7B's selectivity from 0.558 to 0.883, and to 0.894 ± 0.011 across three seeds, so after training the model changes its answer for relevant edits and keeps it for irrelevant ones.

Dependency-based pruning in multi-turn sessions. The dependency view also applies to context management across a conversation, where facts change from turn to turn and earlier reasoning accumulates in the context. In multi-turn math, finance, and factual sessions, half of the invalidated lines in the math sessions lie two or more dependency links from the changed fact, so checking only statements that mention the fact would miss them. Pruning every line whose support is invalidated, together with its descendants, improves pooled accuracy by 0.273 $[0.251, 0.297]$ over retaining the full session across $1{,}800$ sessions, while using 423 $[415, 431]$ fewer tokens per session.

4 Related work

Truth maintenance systems record justifications and update supported conclusions when assumptions change (Doyle, 1979; de Kleer, 1986), belief revision formalizes rational changes to a belief set (Alchourrón et al., 1985), and model-based diagnosis identifies the components whose failure explains an observed inconsistency (Reiter, 1987). We use this dependency perspective to ask whether language models, which have no explicit justification store, update their answers as if they had one. Knowledge editing changes facts stored in model weights (Meng et al., 2022), and RippleEdits tests whether an edit reaches its consequences (Cohen et al., 2024). Our edits act on the prompt with the weights fixed, and the object of study is an obsolete derivation sitting beside revised evidence. Counterfactual and conflicting-evidence tasks change premises directly (Longpre et al., 2021; Xie et al., 2024; Wu et al., 2024; Zhang & Choi, 2021), and work on knowledge conflicts studies how models weigh evidence in context against what they learned in training (Xu et al., 2024). Our stale-versus-valid-prefix design isolates a different source of conflict, the reasoning history the model has already produced, and asks whether that history prevents the model from using evidence it can read.

Several lines of work show that the content of the context shapes answers beyond its relevance to the question being asked. Models degrade over multi-turn conversations by relying on earlier responses (Laban et al., 2025), long-context evaluations show sensitivity to where evidence appears (Liu et al., 2024), and irrelevant sentences in a problem statement can lower accuracy even when the model could solve the problem without them (Shi et al., 2023). Sycophancy studies find that models revise their answers toward a stated user view rather than the evidence (Sharma et al., 2024). That is the opposite failure to the one studied here, in which the model holds on to a conclusion that the evidence no longer supports. Our deletion controls separate stale content from context length and from the edited line, which these accounts of context sensitivity leave entangled. Self-correction studies show that models rarely fix their own reasoning without new information (Huang et al., 2024), whereas in our setting the new information is already in the prompt and the obstacle is the reasoning built before it arrived.

Chain-of-thought and process supervision concern producing and checking reasoning steps (Wei et al., 2022; Lightman et al., 2024), and faithfulness studies ask whether stated reasoning drives the answer, including experiments that insert mistakes into a chain of thought and measure whether the answer follows (Turpin et al., 2023; Lanham et al., 2023). We study a related case, in which a step was correct when written and loses its support only when the evidence changes. The question is then whether the model tracks the dependency, and not whether the reasoning was faithful when it was produced. Our mechanism analysis builds on causal mediation and activation patching (Vig et al., 2020; Geiger et al., 2021) and on the circuits that copy answers from context (Olsson et al., 2022). The heads responsible for repeating a stale answer are the same heads that read a correct one, which connects the failure to this general copying mechanism.

The setting also matters for agents and for how their context is managed, since an agent acts at each step on whatever information its context holds. Agent frameworks interleave reasoning with tool

calls (Yao et al., 2023), and benchmarks of tool use in realistic domains evaluate whether agents complete multi-step tasks in which later observations routinely supersede earlier ones (Yao et al., 2024). STALEFLOW isolates one step of that process, an update that arrives before an action is committed, and measures whether the committed argument follows the update or the reasoning written before it. Memory systems for long conversations decide what to keep in the context window and what to move out of it (Packer et al., 2023), usually by recency or relevance to the current query. Dependency-based pruning offers a different criterion, removing what has lost its support after a change and keeping what still holds, and our session results show that this criterion improves accuracy while shortening the context.

A similar pattern is well documented in human judgment, where studies of belief perseverance find that people keep conclusions they drew from evidence even after they are told that the evidence was false, because the explanations they built around it remain available (Ross et al., 1975). Anchoring shows a related effect, in which an initial value continues to shape an estimate after it has been shown to be irrelevant (Tversky & Kahneman, 1974). Our deletion results suggest a comparable mechanism in language models, since the old answer survives through the reasoning built on the edited premise and disappears once that reasoning is removed, while a notice that the premise has changed does little when the model answers immediately.

5 Conclusion

When a fact is edited and the earlier reasoning stays in context, language models tend to give the answer that reasoning supports. They do so in all 17 models we test and all 40 benchmark cells, and the effect persists under full chain-of-thought generation. Removing only the invalidated steps removes most of the effect, while deleting the same amount of text at random, removing only the edited line, or removing only the stated conclusion leaves much of it in place. The old answer is therefore carried by the downstream consequences of the edited premise rather than by the line that the edit contradicts. A small set of attention heads carries it, the same heads read a correct stated answer, and the effect reaches committed tool calls as well as answers.

The results also bear on how revision should be handled in practice, both in the context a model is given and in the way it is asked to answer. Correction notices and context-aware decoding do not remove the effect when a model answers immediately, and a short reasoning budget makes it worse. The better context edit depends on whether the model answers immediately or generates a solution, so a remedy has to be evaluated with the response procedure that will use it. Training on edited problems reduces the effect in pretrained models and in a model trained from scratch, and pruning reasoning that depends on a changed fact raises accuracy across multi-turn sessions while using fewer tokens. Taken together, these results suggest treating a model's context as a set of claims with dependencies, so that keeping the context current means deciding after each change which claims have lost their support and which still hold.

The controlled edits change one premise at a time, and extending them to simultaneous and interacting edits would test how far dependency tracking scales. The released Inertia-Wild episodes also make it possible to measure the effect in naturally occurring conversations once they are labelled by people. Dependency-based pruning requires the dependencies between reasoning lines, and inferring them from the transcript alone would turn it into a method that can be applied to any conversation, and any such method should be judged by the final answer or action it produces.

Reproducibility Statement

Code, generators, prompts, and configurations for every experiment are available in an anonymized repository at `https://anonymous.4open.science/r/stale-reasoning`, with one command per table and figure. The appendices describe the controlled studies, the separate cohorts, and the response procedures.

AI Use Statement

Generative AI tools were used to help draft and edit the text of the manuscript.

REFERENCES

Carlos E Alchourrón, Peter Gärdenfors, and David Makinson. On the logic of theory change: Partial meet contraction and revision functions. *The Journal of Symbolic Logic*, 50(2), 1985.

Zhiyu Chen, Wenhu Chen, Charese Smiley, Sameena Shah, Iana Borova, Dylan Langdon, Reema Moussa, Matt Beane, Ting-Hao Huang, Bryan Routledge, and William Yang Wang. FinQA: A dataset of numerical reasoning over financial data. In *Proceedings of EMNLP*, pp. 3697–3711, 2021. doi: 10.18653/v1/2021.emnlp-main.300. URL https://aclanthology.org/2021.emnlp-main.300/.

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. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021. URL https://arxiv.org/abs/2110.14168.

Roi Cohen, Eden Biran, Ori Yoran, Amir Globerson, and Mor Geva. Evaluating the ripple effects of knowledge editing in language models. *Transactions of the Association for Computational Linguistics*, 2024. doi: 10.1162/tacl_a_00644. URL https://aclanthology.org/2024.tacl-1.16/.

Johan de Kleer. An assumption-based TMS. *Artificial Intelligence*, 28(2):127–162, 1986. doi: 10.1016/0004-3702(86)90080-9.

Jon Doyle. A truth maintenance system. *Artificial Intelligence*, 12(3):231–272, 1979. doi: 10.1016/0004-3702(79)90008-0.

Atticus Geiger, Hanson Lu, Thomas Icard, and Christopher Potts. Causal abstractions of neural networks. In *Advances in Neural Information Processing Systems*, 2021.

Alex Gu, Baptiste Rozière, Hugh Leather, Armando Solar-Lezama, Gabriel Synnaeve, and Sida I. Wang. CRUXEval: A benchmark for code reasoning, understanding and execution. *arXiv preprint arXiv:2401.03065*, 2024. URL https://arxiv.org/abs/2401.03065.

Jie Huang, Xinyun Chen, Swaroop Mishra, Huaixiu Steven Zheng, Adams Wei Yu, Xinying Song, and Denny Zhou. Large language models cannot self-correct reasoning yet. In *International Conference on Learning Representations (ICLR)*, 2024.

Philippe Laban, Hiroaki Hayashi, Yingbo Zhou, and Jennifer Neville. LLMs get lost in multi-turn conversation. *arXiv preprint arXiv:2505.06120*, 2025. URL https://arxiv.org/abs/2505.06120.

Tamera Lanham, Anna Chen, Ansh Radhakrishnan, Benoit Steiner, Carson Denison, Danny Hernandez, Dustin Li, Esin Durmus, Evan Hubinger, Jackson Kernion, et al. Measuring faithfulness in chain-of-thought reasoning. *arXiv preprint arXiv:2307.13702*, 2023.

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

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

Shayne Longpre, Kartik Perisetla, Anthony Chen, Nikhil Ramesh, Chris DuBois, and Sameer Singh. Entity-based knowledge conflicts in question answering. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2021.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in GPT. In *Advances in Neural Information Processing Systems*, 2022.

Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova DasSarma, Tom Henighan, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, et al. In-context learning and induction heads. *Transformer Circuits Thread*, 2022.

Charles Packer, Sarah Wooders, Kevin Lin, Vivian Fang, Shishir G. Patil, Ion Stoica, and Joseph E. Gonzalez. MemGPT: Towards LLMs as operating systems. *arXiv preprint arXiv:2310.08560*, 2023.

Raymond Reiter. A theory of diagnosis from first principles. *Artificial Intelligence*, 32(1):57–95, 1987.

Lee Ross, Mark R. Lepper, and Michael Hubbard. Perseverance in self-perception and social perception: Biased attributional processes in the debriefing paradigm. *Journal of Personality and Social Psychology*, 32(5):880–892, 1975.

Mrinank Sharma, Meg Tong, Tomasz Korbak, David Duvenaud, Amanda Askell, Samuel R. Bowman, Newton Cheng, Esin Durmus, Zac Hatfield-Dodds, Scott R. Johnston, et al. Towards understanding sycophancy in language models. In *International Conference on Learning Representations*, 2024.

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

Weijia Shi, Xiaochuang Han, Mike Lewis, Yulia Tsvetkov, Luke Zettlemoyer, and Wen-tau Yih. Trusting your evidence: Hallucinate less with context-aware decoding. In *Proceedings of NAACL*, 2024.

Miles Turpin, Julian Michael, Ethan Perez, and Samuel R Bowman. Language models don't always say what they think: Unfaithful explanations in chain-of-thought prompting. In *Advances in Neural Information Processing Systems*, 2023.

Amos Tversky and Daniel Kahneman. Judgment under uncertainty: Heuristics and biases. *Science*, 185(4157):1124–1131, 1974.

Jesse Vig, Sebastian Gehrmann, Yonatan Belinkov, Sharon Qian, Daniel Nevo, Yaron Singer, and Stuart Shieber. Investigating gender bias in language models using causal mediation analysis. In *Advances in Neural Information Processing Systems*, 2020.

Denny Vrandečić and Markus Krötzsch. Wikidata: A free collaborative knowledgebase. *Communications of the ACM*, 57(10):78–85, 2014. URL https://iccl.inf.tu-dresden.de/w/images/8/89/Wikidata-CACM-2014.pdf.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, 2022.

Zhaofeng Wu, Linlu Qiu, Alexis Ross, Ekin Akyürek, Boyuan Chen, Bailin Wang, Najoung Kim, Jacob Andreas, and Yoon Kim. Reasoning or reciting? exploring the capabilities and limitations of language models through counterfactual tasks. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics*, 2024.

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

Rongwu Xu, Zehan Qi, Zhijiang Guo, Cunxiang Wang, Hongru Wang, Yue Zhang, and Wei Xu. Knowledge conflicts for LLMs: A survey. In *Proceedings of EMNLP*, 2024.

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

Shunyu Yao, Noah Shinn, Pedram Razavi, and Karthik Narasimhan. τ-bench: A benchmark for tool-agent-user interaction in real-world domains. *arXiv preprint arXiv:2406.12045*, 2024.

Michael J.Q. Zhang and Eunsol Choi. SituatedQA: Incorporating extra-linguistic contexts into QA. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2021.

A CONTROLLED-STUDY METHODS

A.1 DOMAINS

We generate 250 problems in each of three domains. Branching arithmetic uses short programs with data-dependent branches, including a value that is never read and a branch that is never taken. Horn logic uses a depth-three implication chain over eight classes. Grid navigation uses a 7×7 corridor containing an agent, key, locked door, and goal, solved by breadth-first search.

Each problem has an eight-option answer set and a canonical sequence of valid reasoning steps. Relevant edits change an input or rule that changes the exact answer. Irrelevant edits change an unused value or an inactive rule and leave the answer fixed. Their text templates and edit types are matched. In the grid task, for example, the same additional-door-opening rule names either a reachable location or a location behind the door.

The simulator verifies 3,000 irrelevant edits with unchanged outcomes and 3,000 relevant edits with changed outcomes. Independently reconstructing each edited world reproduces the generator's answer on all 3,000 relevant edits.

A.2 PROMPT CONDITIONS

We compare five conditions on the same problems:

1. **Stale with notice:** retain the original derivation and explicitly describe the correction.
2. **Stale without notice:** edit the world in place and retain the original derivation.
3. **Valid prefix:** edit the world and remove the derivation from the first invalidated step onward.
4. **Revised derivation:** edit the world and supply its correct canonical derivation.
5. **Unedited control:** retain the original world but truncate its derivation to the same depth as the valid-prefix condition.

The unedited control measures performance with the same amount of valid reasoning available. The revised-derivation condition measures whether the model can report the conclusion of a correct supplied derivation. The valid-prefix condition measures whether it can finish the edited problem itself.

A.3 SELECTION OF SELF-GENERATED CHAINS

To connect the canonical experiment to model-generated text, we sample one derivation per problem and retain derivations that give the correct original answer. We remove the final boxed answer and locate the first statement invalidated by the edit. The stale and valid-prefix conditions are then applied to the model's own words.

Qwen2.5-7B repeats the old answer at rate 0.982 with its full generated chain and 0.216 with the invalidated portion removed. Qwen3-4B gives 0.919 versus 0.292, a paired difference of 0.627 $[0.589, 0.663]$. Qwen3-1.7B gives 0.814 versus 0.138, a difference of 0.676 $[0.649, 0.706]$.

These results are conditional on an initially correct derivation. Retention ranges are 0.94–0.97 in arithmetic, 0.38–0.84 in logic, and 0.12–0.13 in navigation. Among correctly solved original problems, the earlier reasoning continues to influence the answer after the problem changes.

A.4 REGENERATING A SOLUTION SUPPORTS REVISION

A separate free-generation assay allows the model to recompute rather than immediately commit to an option. With the stale derivation still present and no correction notice, Qwen2.5-7B reaches revision accuracy 0.688 and old-answer rate 0.145. Qwen3-1.7B reaches 0.632 and 0.165 over the full problem set.

This comparison makes response regime part of the evaluation. Immediate answer readout measures the influence of the existing context, while regeneration permits a new computation over the

edited premises. Effective revision can therefore be supported both by the training objective and by allocating computation to re-derive the answer.

B ADDITIONAL RESULTS

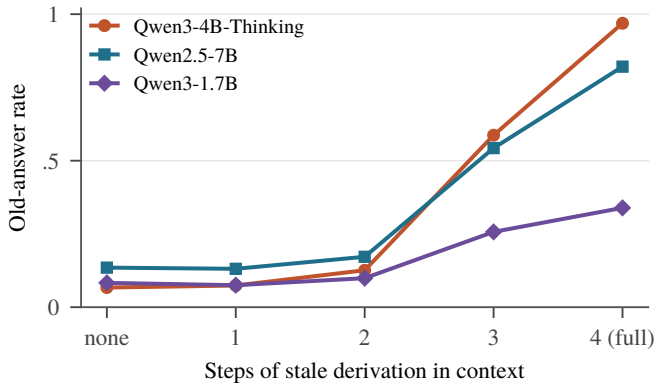


Figure 4: Old-answer rate as the stale derivation accumulates, from no prior reasoning to the full derivation. For every model the first two steps cost little and the rate rises over the final two.

Models. Table 4 gives the values plotted in Figure 2. Open models are scored from restricted token probabilities over 1,500 items; closed models answer with a committed single letter over 240 items (Appendix D).

Table 4: Old-answer rate with the stale derivation (E) and with the invalidated tail removed (B), with the paired difference and its 95% interval. Open models use restricted token probabilities over 1,500 items from 750 problems, with intervals that resample problems; closed models use a committed single-letter answer over 240 items, at minimal reasoning where the setting exists.

Open model	E	B	E−B [CI]	Closed model	E	B	E−B [CI]
Qwen3-4B-Thinking	.969	.070	+.899 [.880,.916]	GPT-5-mini	.796	.083	+.712 [.658,.767]
Llama-3.1-8B base	.908	.076	+.832 [.809,.854]	GPT-5	.725	.108	+.617 [.554,.675]
Llama-3.1-8B Inst.	.894	.119	+.775 [.747,.800]	GPT-4.1-mini	.654	.050	+.604 [.544,.669]
Mistral-7B Inst.	.829	.069	+.760 [.736,.785]	GPT-4.1	.721	.138	+.583 [.517,.646]
Phi-4	.818	.082	+.736 [.712,.759]	GPT-4o	.654	.108	+.546 [.483,.608]
Qwen2.5-7B Inst.	.821	.131	+.690 [.662,.718]				
OLMo-2-7B base	.681	.064	+.617 [.583,.647]				
Qwen2.5-3B Inst.	.661	.083	+.577 [.546,.609]				
Qwen2.5-3B base	.627	.092	+.535 [.503,.567]				
Qwen2.5-7B base	.671	.136	+.535 [.505,.565]				
OLMo-2-7B Inst.	.549	.079	+.469 [.433,.504]				
Qwen3-1.7B	.339	.079	+.259 [.232,.287]				

Table 5: E−B by domain (250 problems per cell).

Model	Arithmetic	Navigation	Logic
Qwen3-4B-Thinking	+.904	+.904	+.888
Llama-3.1-8B-Instruct	+.854	+.844	+.626
Phi-4	+.896	+.858	+.454
Qwen2.5-7B-Instruct	+.838	+.722	+.510

Chain-of-thought generation on the controlled benchmark. Table 7 gives the stale-condition old-answer rate when each model generates up to 2,048 tokens and the answer is parsed from its

Table 6: Distribution of E−B across the open models. The family median uses one representative per family: Qwen3 +.899, Llama-3.1 +.775, Mistral +.760, Phi +.736, Qwen2.5 +.690, OLMo-2 +.469.

Subset	n	Median	Max	Min	IQR
All open models	12	+.653	+.899	+.259	[+.535, +.764]
Base checkpoints	4	+.576	+.832	+.535	–
Instruction-tuned	8	+.713	+.899	+.259	–
One per family	6	+.748	+.899	+.469	–

output. Navigation is the hardest domain under generation, at 0.883 for Mistral-7B, 0.683 for Llama-3.1-8B, and 0.442 for Phi-4.

Table 7: Old-answer rate under the stale derivation with chain-of-thought generation, with problem-level 95% intervals.

Model	Items	Problems	Old-answer rate
Mistral-7B-Instruct	360	180	.700 [.655,.742]
Llama-3.1-8B-Instruct	360	180	.375 [.335,.415]
OLMo-2-7B-Instruct	360	180	.297 [.257,.339]
Qwen3-1.7B	1,500	750	.161 [.147,.176]
Qwen2.5-7B-Instruct	360	180	.158 [.127,.190]
Phi-4	360	180	.147 [.120,.174]
Qwen3-4B-Thinking	360	180	.106 [.079,.131]

Deletion policies. Table 1 gives the deletion policies shown in Figure 1b together with revision accuracy. Removing the conclusion alone, with the derivation body kept, leaves far more repetition than removing the invalidated tail (Table 8).

Table 8: Deleting only the stated conclusion versus deleting the invalidated tail, old-answer rate with paired 95% intervals.

Model	Delete conclusion only	Delete invalidated tail	Difference
Qwen3-4B-Thinking	.587	.071	+.516 [.491,.542]
Llama-3.1-8B-Instruct	.601	.123	+.479 [.451,.506]
Qwen2.5-7B-Instruct	.541	.133	+.408 [.382,.433]
Phi-4	.483	.085	+.397 [.373,.423]

Reasoning budget. Table 9 gives the values in Figure 3a and Table 11 the GPT-5-mini values in Figure 3b. A separate earlier run on 480 items with a finer grid below 64 tokens (Table 10) shows the same fall, with revision accuracy dropping by 0.156 $[0.125, 0.190]$ from 0 to 64 tokens; Qwen2.5-7B falls by 0.062 $[0.033, 0.092]$ over the same range. Its zero point (0.356) matches the correction-block baseline for that rendering (0.375). On items where only two steps need redoing, Qwen3-4B-Thinking starts at 0.45 revision accuracy with no budget and falls to 0.01 at 64 tokens.

Free generation. Table 12 lists every free-generation cell. The model writes a complete solution and the answer is parsed from its own output; the reference arm uses a coherent context.

Prompt-level remedies. Tables 13–15 give the remedy comparisons summarized in the main text and in Figure 5a. An explicit notice appended to the stale derivation reads: *the problem statement has been updated; the reasoning above was written before that update and may be invalid.* The correction block keeps the original world and states exactly what changed immediately before the question. Context-aware decoding scores $(1+\alpha)\log p(\text{full}) - \alpha \log p(\text{no prior reasoning})$, with both arms measured on the same items.

Table 9: Qwen3-4B-Thinking with a fixed budget after the correction (240 items, 120 problems), with reasoning and with a length-matched filler sentence. The last column is the paired difference in revision accuracy.

Tokens	Old-answer reasoning	Revision accuracy reasoning	Revision accuracy filler	Difference
0	.388	.183	.183	–
4	.838	.096	.146	−.050 [−.078,−.020]
16	.883	.062	.104	−.042 [−.068,−.013]
64	.879	.033	.171	−.138 [−.169,−.105]
256	.567	.242	.154	+.087 [.041,.131]
1,024	.267	.688	.121	+.567 [.500,.636]
4,096	.138	.775	.096	+.679 [.622,.738]
Natural stop (2,344)	–	.775	–	–

Table 10: Earlier run: Qwen3-4B-Thinking with a fixed reasoning budget of b tokens after the correction, then a forced answer (480 items).

b	0	1	2	4	8	16	32	64
Old-answer rate	.356	.617	.696	.823	.873	.875	.865	.860
Revision accuracy	.190	.138	.121	.081	.056	.054	.042	.033

Relevant and irrelevant edits. Table 16 gives the values in Figure 5b. Irrelevant edits change an unused value or an inactive rule and arrive in the same format beside the same stale derivation.

Training from scratch. Table 17 gives the 111M results in Figure 5c. Each model is trained on 60,000 problems; only the share of problems whose world is revised mid-derivation varies.

C Mechanism details

Head ablation. Conclusion-reading heads are localized by attention attribution from the answer position to the stated conclusion, then ablated in groups against random head sets of the same size (Table 18). The same targeted ablation lowers Llama-3.1-8B's accuracy from 0.95 to 0.21 when the stated conclusion is correct, so these heads implement general reading of a stated answer.

Residual patching. The answer-position residual stream of a stale run is written into a run whose invalidated steps are redacted, one layer at a time (100 items; Table 19). The redacted run alone gives the old answer 0.18–0.19 of the time. We report only this direction: patching the redacted state into the stale run at the final layer replaces the output representation itself and is uninformative by construction.

Probe. The probe is a logistic regression on the residual stream at the answer position, at the full stale derivation, with triplet-grouped four-fold cross-validation. Its binary target is whether the model gave the pre-edit answer. Accuracy is compared with the majority-class floor, since the classes are unbalanced (Table 20). On Qwen2.5-7B, probe confidence is $+0.594$ $[0.549, 0.635]$ higher on items the model answers with the old answer. The layer-0 probe lands on the majority-class floor for every model, and shuffled labels fall at or below it. Qwen3-1.7B clears its floor by only 0.051 and is the weakest case.

D Closed-model protocol

GPT-5, GPT-5-mini, GPT-4.1, GPT-4.1-mini, and GPT-4o receive the same problems, renderings, and prompt text as the open models. None exposes token log-probabilities, so each is instructed to answer with exactly one option letter, and the first letter from A to H in the reply is taken as the answer. Coverage is 1.00 in every cell. GPT-5 and GPT-5-mini are run at minimal reasoning effort

Table 11: GPT-5-mini by reasoning effort: mean reasoning tokens per item and the paired E−B old-answer difference (240 items).

Effort	Reasoning tokens	E−B	95% CI
Minimal	0	+.712	[.658,.767]
Low	316	+.096	[.058,.133]
Medium	690	+.087	[.054,.125]
High	1,232	+.077	[.048,.120]

Table 12: Free generation, old-answer rate under the stale derivation and under a coherent context, with the paired difference.

Dataset	Model	n	Stale	Reference	E−B [95% CI]
GSM8K	Llama-3.1-8B	186	.301	.021	+.280 [.215,.344]
CRUXEval	Llama-3.1-8B	174	.460	.286	+.185 [.083,.292]
CRUXEval	Qwen2.5-7B	191	.393	.237	+.152 [.068,.236]
FinQA	Qwen2.5-7B	194	.113	.025	+.088 [.041,.139]
FinQA	Llama-3.1-8B	167	.108	.029	+.078 [.030,.132]
GSM8K	Qwen2.5-7B	198	.086	.010	+.076 [.035,.121]
GSM8K	Qwen3-32B	97	.072	.000	+.076 [.033,.130]
CRUXEval	Qwen3-8B	178	.169	.098	+.075 [.006,.144]
Wikidata	Qwen3-8B	100	.070	.000	+.070 [.020,.120]
GSM8K	Qwen3-8B	194	.077	.011	+.069 [.032,.112]

for the main comparison, and GPT-5-mini additionally at low, medium, and high effort; GPT-4.1, GPT-4.1-mini, and GPT-4o have no reasoning setting. Each cell uses 240 items.

E STALEFLOW PROTOCOL

Each episode is a short tool-calling sequence. A lookup tool returns a value V_{old}; the agent states the value it plans to book; a mandatory pre-commit lookup then returns V_{new} for the same quantity; and the agent issues the booking call. The endpoint is the argument of that final call. In the immediate-action condition the call is prefilled so the model commits without intermediate text; in the reasoning condition it may write before committing. A clean condition removes V_{old} from the history entirely and measures the ability to act correctly. Every non-thinking cell parses at rate 1.000 with no forced advances.

F STATISTICAL PROCEDURES

Each of the 750 controlled problems contributes two items that share a world and a derivation. Model-level comparisons (Figure 2, Tables 4, 7, and 9) resample problems, with both items travelling together. The remaining paired contrasts resample items with 4,000 resamples, reusing the same resample indices for both arms. Single rates on small samples, including head ablation and STALEFLOW, use Wilson 95% intervals. Multi-cell claims use Holm correction within the stated family of comparisons. Session-level pruning uses a bootstrap stratified by session domain over 1,800 sessions. Benchmark revisions permute the answer options over three orders and average the renormalized distributions before taking the argmax. Revision training is repeated over three seeds and reported as mean $\pm$ standard deviation.

G INERTIA-WILD

We release 473 naturally occurring episodes mined from WildChat-1M, in which a user revises a fact the assistant had already used and the assistant then answers again. About 250,000 conversations were streamed. The release has a tight split of 128 episodes, with a numeric derivation in the prior reply, a numeric revision, and no hypotheticals, and a broad split of 345. The task mix is other

Table 13: Explicit correction notice appended to the stale derivation.

Model	Stale	With notice	Change	95% CI
Qwen3-4B-Thinking	.969	.646	−.323	[−.347,−.300]
Qwen2.5-7B-Instruct	.822	.589	−.233	[−.255,−.212]
Llama-3.1-8B-Instruct	.893	.705	−.188	[−.208,−.168]
Phi-4	.819	.660	−.159	[−.177,−.141]

Table 14: Original world plus a correction block, against deleting the invalidated tail. All 12 differences remain significant after Holm correction.

Model	Correction block	Tail deleted	Difference	95% CI
Phi-4	.739	.082	+.657	[.632,.680]
Llama-3.1-8B base	.531	.076	+.455	[.428,.481]
Qwen2.5-7B-Instruct	.493	.131	+.362	[.337,.387]
Qwen3-4B-Thinking	.375	.070	+.305	[.281,.329]
Qwen2.5-3B-Instruct	.337	.083	+.253	[.231,.277]
Llama-3.1-8B-Instruct	.349	.119	+.230	[.205,.256]
Qwen3-1.7B	.254	.079	+.175	[.153,.196]
Qwen2.5-7B base	.300	.136	+.164	[.141,.188]
Qwen2.5-3B base	.205	.092	+.113	[.091,.135]
OLMo-2-7B-Instruct	.185	.079	+.105	[.086,.125]
Mistral-7B-Instruct	.163	.069	+.094	[.073,.115]
OLMo-2-7B base	.119	.064	+.055	[.037,.073]

(213), code (132), math (82), planning (26), and finance (20), and the assistant models are gpt-3.5-turbo-0301 (295), gpt-4-0314 (116), and gpt-3.5-turbo-0613 (62). Each episode carries labels from two independent model labellers and an empty human-label field, and a 128-episode adjudication sample orders labeller disagreements first. The release includes a datasheet covering motivation, composition, collection, intended use, and provenance. We report prevalence only after human adjudication, so this appendix describes the resource and not a rate.

H Illustrative renderings

For the invoice example of Figure 1a, with the price edited from 2 to 4, the renderings take the following form. The stale derivation keeps *price* $= 2$*, qty* $= 3$*, so total* $= 6$*; 6 is within budget, so the purchase fits*. Deleting only the first invalid step removes *total* $= 6$ but keeps *the purchase fits*. Deleting the invalidated tail removes both, leaving the premises and the unaffected shipping step. A random deletion removes the same number of steps chosen uniformly. The coherent derivation recomputes *total* $= 12$ and rechecks the budget. The benchmark items follow the same pattern with longer derivations and eight answer options.

I Dependency depth

Each edit has a known number of downstream steps that must be recomputed. Stratifying the 1,500 controlled items by that number (Table 21) shows a sharp threshold: when two steps need redoing, most models recompute, and from three onward the stale answer dominates. Phi-4 moves from $+0.004$ at two steps to $+0.896$ at three, and Mistral from $+0.112$ to $+0.845$. Qwen3-4B-Thinking is affected at every depth, including $+0.916$ at two steps.

J Training configuration

The adapter has rank 16, scaling 32, and dropout 0.05, and is applied to attention and MLP projections. Training uses learning rate 10^{-4} with cosine decay, effective batch size 16, and one epoch over 20,000 rendered prompts derived from 800 newly generated problems. Revision training is re-

Table 15: Context-aware decoding: old-answer rate by contrast strength α ($\alpha = 0$ is the plain stale readout), and the best setting against deleting the invalidated tail.

Model	$\alpha = 0$	0.25	0.5	1	2	Tail deleted	Best CAD − tail
Qwen3-4B-Thinking	.969	.975	.979	.981	.977	.071	+.905 [.890,.919]
Llama-3.1-8B-Instruct	.893	.899	.907	.921	.927	.123	+.777 [.755,.797]
Qwen2.5-7B-Instruct	.822	.857	.872	.895	.910	.133	+.724 [.701,.747]
Phi-4	.819	.825	.829	.829	.835	.085	+.740 [.717,.763]

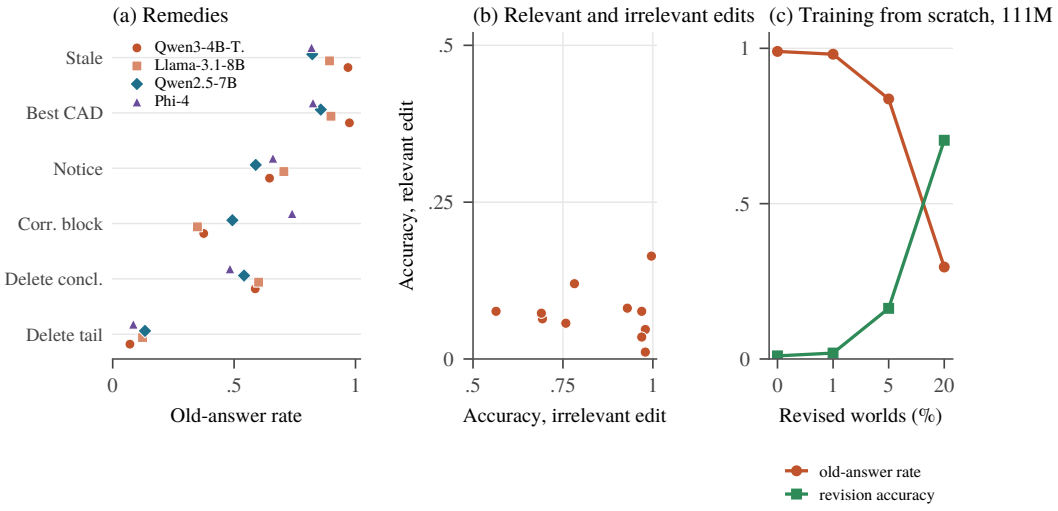


Figure 5: (a) Old-answer rate under each remedy for four models. Removing the invalidated tail gives the lowest rate for every model. (b) Revision accuracy on edits that cannot change the answer against edits that do, for 11 open models. (c) A 111M model trained from scratch, by the share of training problems whose world is revised.

peated over three seeds; the answer-only comparison uses one training run. Evaluation uses different generator seeds. The two-domain training design is kept fixed within each objective comparison.

K Response Normalization

An additive option-logit bias that is fixed across edited and unedited conditions can be removed from a centered log-probability response. Define

$$\Lambda_k[i, y] = \log P_k(y \mid \delta_i) - \log P_k(y) - \frac{1}{|Y|} \sum_{y'} [\log P_k(y' \mid \delta_i) - \log P_k(y')] . \tag{3}$$

The fixed option offsets cancel in the difference, and the shared normalizer cancels after centering. Selectivity excludes states with zero total response, which account for less than 0.2% of measured states. Probability-space and centered-log responses are different summaries and are not interpreted as interchangeable effect sizes.

Table 16: Revision accuracy under the full stale derivation for edits that cannot change the answer and edits that do.

Model	Irrelevant edit	Relevant edit	Gap
Phi-4	.996	.164	+.832
Llama-3.1-8B-Instruct	.979	.047	+.932
Qwen3-4B-Thinking	.979	.011	+.968
Mistral-7B-Instruct	.969	.076	+.893
Llama-3.1-8B base	.969	.035	+.934
Qwen2.5-7B-Instruct	.929	.081	+.848
Qwen2.5-7B base	.782	.120	+.662
Qwen2.5-3B-Instruct	.758	.057	+.701
OLMo-2-7B base	.693	.064	+.629
Qwen2.5-3B base	.690	.073	+.617
OLMo-2-7B-Instruct	.564	.076	+.488

Table 17: A 111M transformer trained from scratch, by the share of revised-world training problems.

Revised worlds	Old-answer rate	Revision accuracy
0%	.990	.010
1%	.981	.019
5%	.837	.163
20%	.296	.704

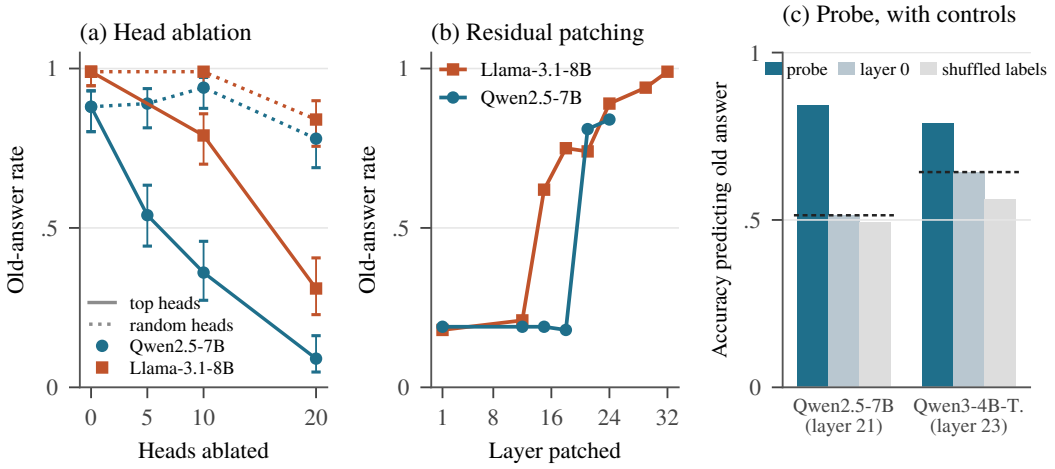


Figure 6: (a) Ablating the heads that read the stated conclusion against size-matched random heads, with Wilson 95% intervals (100 items). (b) Patching the stale state into the redacted run, by layer. (c) Probe accuracy at each model's best layer against layer-0 and shuffled-label controls; dashed lines mark the majority-class floor.

Table 18: Old-answer rate under targeted and size-matched random head ablation, with Wilson 95% intervals (100 items).

Model	Heads ablated	Targeted	Random
Qwen2.5-7B-Instruct	none	.880 [.802,.930]	–
	5	.540 [.443,.634]	.890 [.814,.937]
	10	.360 [.273,.458]	.940 [.875,.972]
	20	.090 [.048,.162]	.780 [.689,.850]
Llama-3.1-8B-Instruct	none	.990 [.946,.998]	–
	10	.790 [.700,.858]	.990 [.946,.998]
	20	.310 [.228,.406]	.840 [.756,.899]

Table 19: Old-answer rate after patching the stale state into the redacted run, by layer.

Layer	1	12	15	18	21	24	29	32
Llama-3.1-8B	.180	.210	.620	.750	.740	.890	.940	.990
Qwen2.5-7B	.190	.190	.190	.180	.810	.840	–	–

Table 20: Probe for old-answer behavior at each model's best layer, with the majority-class floor, the layer-0 and shuffled-label controls, and the correlation between probe confidence and behavior. Qwen2.5-7B has 28 layers and Qwen3-4B-Thinking 36.

Model	Layer	Probe	Floor	Layer 0	Shuffled	ρ
Qwen2.5-7B-Instruct	21	.844 [.817,.869]	.514	.514	.493	+.701
Qwen3-4B-Thinking	23	.790 [.760,.819]	.643	.643	.563	+.593
Qwen3-1.7B	19	.808 [.779,.836]	.757	.757	–	+.450

Table 21: E−B old-answer difference by the number of downstream steps that must be recomputed, with 95% intervals; $n = 250$, 548, and 702 items per column.

Model	2 steps	3 steps	4 steps
Phi-4	+.004 [.000,.012]	+.896 [.870,.920]	+.866 [.840,.890]
Mistral-7B-Instruct	+.112 [.072,.152]	+.845 [.814,.876]	+.925 [.905,.943]
Qwen2.5-7B-Instruct	+.264 [.212,.320]	+.785 [.750,.819]	+.766 [.735,.799]
Llama-3.1-8B-Instruct	+.476 [.416,.536]	+.819 [.786,.852]	+.846 [.819,.873]
Llama-3.1-8B base	+.588 [.528,.648]	+.814 [.781,.847]	+.933 [.914,.952]
Qwen3-4B-Thinking	+.916 [.880,.948]	+.874 [.847,.901]	+.912 [.890,.932]